



IBM Developing Applications Using IBM Informix ESQL/C

Duration: 4 Days Course Code: IX150G

Overview:

This is the Classroom version of Instructor-Led Online course Developing Applications Using IBM Informix ESQL/C - Instructor Led Online (3X150) and Self-Paced Virtual course Developing Applications Using IBM Informix ESQL/C (SPVC) (2X150). In this course, you will use IBM Informix ESQL/C tools to write applications that contain embedded SQL commands and queries. You will write applications to perform queries that return single and multiple rows; insert, update, and delete rows; create and use forms to display one or more rows of data; manage cursors, and handle various types of data, including large objects. This course replaces US Course Developing Applications Using IBM Informix ESQL/C (L1112).

Target Audience:

This intermediate course is designed for application developers.

Objectives:

Include SQL statements in a C program to add, retrieve, and alter data in an Informix database
Set up a cursor structure for manipulating a set of rows
Work with all SQL data types

Prerequisites:

You should have:

IBM Informix Structured Query Language or equivalent knowledge
 ANSI C programming or C programming experience

Content:

Introduction to IBM Informix ESQL/C

- Identify the components of ESQL/C
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure

IX150G

INSERT and SELECT simple large objects

- Install ESQL/C as part of Client SDK
 Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and

www.globalknowledge.com.sa

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

00 966 92000 9278

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

INTERVAL macros

primary key to select the current row

- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging

Declare a cursor from a prepared

Defer execution of a PREPAREd

statement

statement

a database

stored

training@globalknowledge.com.sa

from a file or from memory

- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing **INSERT Statements**
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of

IX150G

- **INTERVAL** macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory

INSERT and SELECT simple large objects

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables Installing a copy of this database

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

00 966 92000 9278

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

ESQL/C

training@globalknowledge.com.sa

appropriate

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

sqldetach

sqlbreak

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

- sqldetach sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10 Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing **INSERT Statements**
- Unit 23: Working with the Database Server
- Exercise 19
- An Overview of Cursors
- Name the three types of cursors

Install ESQL/C as part of Client SDK

the ESQL/C preprocessor

SQL databases

program

www.globalknowledge.com.sa

Installing a copy of this database

Set environment variables and execute

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

different types

- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C ProgramExercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11

IX150G

Unit 13: Singleton Queries and Lookups

appropriate

- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utilityExplain problems in converting between
- data types
 Use functions to convert variables of
- different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were
- fetched or character data was truncated Ensure referential integrity using
- application logic Use the appropriate cursor for a given
- task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
 Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3

www.globalknowledge.com.sa

- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic

Use a scrolling cursor to browse the

Automatically free a cursor
 Use the OPTOFC feature to reduce

network messaging

selected rows

buffers

statement

statement

a database

stored

sqldetach

salbreak

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

training@globalknowledge.com.sa

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

00 966 92000 9278

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

The Demonstration Database

- The tables in the stores_demo demonstration database
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the

IX150G

- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Using Scroll Cursors

- Use SQL statements to set up a scrolling cursor
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
 Compile a program by using IBM Informix
- ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types

www.globalknowledge.com.sa

Explain the storage needs of character and string data

- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Installing a copy of this database
 Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

00 966 92000 9278

Determine whether SQL NULLs were

Use functions to convert variables of

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

FSQL/C

data types

string data

data type

this structure

statement

information

training@globalknowledge.com.sa

data

different types

library functions

appropriate

- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

Exercise 12
 Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

selected rows

- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

IX150G

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the

- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
 Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9

www.globalknowledge.com.sa

Unit 11: SQL Exception Testing

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections
 Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 19: Time Data Types

Unit 21: Dynamic SQL

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

00 966 92000 9278

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Ensure referential integrity using

Use a scrolling cursor to browse the

Use the OPTOFC feature to reduce

Automatically free a cursor

network messaging

application logic

selected rows

buffers

statement

statement

a database

stored

sqldetach

salbreak

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

training@globalknowledge.com.sa

ESQL/C preprocessor

- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3

IX150G

- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were

www.globalknowledge.com.sa

- fetched or character data was truncated
- Ensure referential integrity using

- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

ESQL/C

data types

string data

data type

this structure

statement

information

application logic

selected rows

buffers

statement

statement

a database

stored

training@globalknowledge.com.sa

data

different types

library functions

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
 Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

00 966 92000 9278

Explain how data is converted as it is

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Determine whether SQL NULLs were

Ensure referential integrity using

Use a scrolling cursor to browse the

Use the OPTOFC feature to reduce

Automatically free a cursor

network messaging

Use functions to convert variables of

- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Embedding SQL Statements

- Embed SQL statements in C
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8,

IX150G

application logic

Use the appropriate cursor for a given task

Use pre-defined DATETIME and

INSERT and SELECT simple large objects

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe how the syntax works with pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

00 966 92000 9278

Describe when optional syntax is

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

training@globalknowledge.com.sa

appropriate

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

INTERVAL macros

sqldetach

sqlbreak

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
 Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
 - Unit 4: Using CONNECT TO
 - Exercise 4
 - Unit 5: Compiling an ESQL/C Program
 - Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Unit 19: Time Data Types

Unit 21: Dynamic SQL

Exercise 16
 Exercise 17

Exercise 18

www.globalknowledge.com.sa

BOOLEAN, and DECIMAL data types

- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application loaic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13 Unit 15: Using Scroll Cursors
- Exercise 14

IX150G

Unit 16: Using Scroll Cursors Effectively Unit 17: Using an Update Cursor

INSERT Statements

- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement

www.globalknowledge.com.sa

Use an INSERT cursor to insert rows into a database

- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

00 966 92000 9278

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Ensure referential integrity using

Use a scrolling cursor to browse the

Automatically free a cursor Use the OPTOFC feature to reduce

network messaging

application logic

selected rows

buffers

statement

statement

a database

stored

sqldetach

sqlbreak

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

training@globalknowledge.com.sa

- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor

- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
 Use an INSERT cursor to insert rows into a

- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
 sqldetach
- sqldetaci
 sqlbreak
- Sqibreak
- Work with multiplexed connections
 Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data TypesExercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
 Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

www.globalknowledge.com.sa

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate

- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

00 966 92000 9278

Use functions to convert variables of

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

ESQL/C

data types

string data

data type

this structure

statement

data

training@globalknowledge.com.sa

different types

library functions

appropriate

- Exercise 10
- Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

database

- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
 Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

IX150G

Using CONNECT TO

- Use the CONNECT TO, DISCONNECT, and SET CONNECTION SQL statements
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C

- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
 Compile a program by using IBM Informix
- ESQL/C
- Explain the basic use of the make utility
 Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
 Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3

www.globalknowledge.com.sa

Unit 4: Using CONNECT TO

- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor

statement

statement

a database

stored

sqldetach sqlbreak

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

training@globalknowledge.com.sa

 Use the OPTOFC feature to reduce network messaging
 Solve the stale data problem by using the

Declare a cursor from a prepared

Defer execution of a PREPAREd

primary key to select the current row

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 19: Time Data Types

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

00 966 92000 9278

INTERVAL macros

program

- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESOI /C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4

IX150G

- Unit 5: Compiling an ESQL/C Program
 Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types

- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Using Scroll Cursors Effectively

- Use a scrolling cursor to select a primary key
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through

www.globalknowledge.com.sa

- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK

Installing a copy of this database

Describe when optional syntax is

SQL databases

program

ESOL/C

data types

string data

data type

this structure

statement

information

application logic

selected rows

buffers

statement

statement

training@globalknowledge.com.sa

data

different types

library functions

appropriate

Set environment variables and execute the ESQL/C preprocessor
 The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

00 966 92000 9278

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Determine whether SQL NULLs were

Ensure referential integrity using

Use a scrolling cursor to browse the

Use the OPTOFC feature to reduce

Automatically free a cursor

network messaging

Use functions to convert variables of

- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information

library functions

- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were
- fetched or character data was truncated Ensure referential integrity using
- application logicUse the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
 Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10

www.globalknowledge.com.sa

- a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

00 966 92000 9278

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

training@globalknowledge.com.sa

appropriate

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

ESQL/C Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18

- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server

- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic

www.globalknowledge.com.sa

- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
 Compile a program by using IBM Informix

Explain the basic use of the make utility
 Explain problems in converting between

Use functions to convert variables of

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

Determine whether SQL NULLs were

Ensure referential integrity using

Use a scrolling cursor to browse the

Automatically free a cursorUse the OPTOFC feature to reduce

network messaging

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

00 966 92000 9278

Unit 4: Using CONNECT TO

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

ESQL/C

data types

string data

data type

this structure

statement

information

application logic

selected rows

buffers

statement

statement

a database

stored

sqldetach

sqlbreak

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

training@globalknowledge.com.sa

data

different types

library functions

Exercise 19

Compiling an ESQL/C Program

- Incorporate other files into your source code
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure

IX150G

INSERT and SELECT simple large objects

- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
 Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
 Exercise 2
- Unit 3: Embedding SQL Statements
 Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
 Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18

www.globalknowledge.com.sa

 Unit 22: Dynamic SQL: Constructing INSERT Statements

- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Declare host variables for INT8, SERIAL8,

00 966 92000 9278

Interface with LVARCHAR data through

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use functions to convert variables of

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

Unit 9: Inserting Rows

ESQL/C preprocessor

SQL databases

program

FSQL/C

data types

string data

training@globalknowledge.com.sa

different types

library functions

appropriate

- Exercise 10
- Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

Dav 2

from a file or from memory

- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of

IX150G

- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute
- the ESQL/C preprocessor The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is

www.globalknowledge.com.sa

data type

buffers

statement

statement

a database

stored

sqldetach

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12
 Exercise 13

training@globalknowledge.com.sa

sqlbreak

- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
 Change the size of FETCH and INSERT

Use the OPTOFC feature to reduce

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

Explain how data is converted as it is

INSERT and SELECT simple large objects

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

00 966 92000 9278

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Automatically free a cursor

network messaging

different types

- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11

IX150G

Unit 13: Singleton Queries and Lookups

stored

- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration DatabaseExercise 2
- Unit 3: Embedding SQL StatementsExercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
 Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications AreaExercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Using an Update Cursor

- Use an update cursor to lock rows that might be updated
- Using a Insert Cursor

www.globalknowledge.com.sa

- Determine if an INSERT cursor is needed for adding rows to a database
- Install ESQL/C as part of Client SDK
- Set environment variables and execute

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

00 966 92000 9278

Determine whether SQL NULLs were

Use a scrolling cursor to browse the

Automatically free a cursorUse the OPTOFC feature to reduce

network messaging

Ensure referential integrity using

Use functions to convert variables of

Installing a copy of this database

Set environment variables and execute the

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

FSQL/C

data types

string data

data type

this structure

statement

information

application logic

selected rows

buffers

training@globalknowledge.com.sa

data

different types

library functions

appropriate

- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor

Use the OPTOFC feature to reduce network

- the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
 sqldetach
- sqluctach sqlbreak

www.globalknowledge.com.sa

- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
 - Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

00 966 92000 9278

The relationships between these tables

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16
Exercise 17

Exercise 18

Exercise 19

training@globalknowledge.com.sa

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

messaging

- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17Unit 21: Dynamic SQL
- Unit 21: Dynam
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

IX150G

ESQL/C Data Types

- Determine which C data type to use as a host variable
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the

Work with multiplexed connections

Installing a copy of this database

SQL databases

program

ESQL/C

data types

string data

data type

this structure

statement

information

application logic

selected rows

buffers

statement

statement

a database

stored

sqldetach sqlbreak

FSQL/C

Exercise 1

training@globalknowledge.com.sa

data

different types

library functions

appropriate

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

00 966 92000 9278

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Determine whether SQL NULLs were

Ensure referential integrity using

Use a scrolling cursor to browse the

Use the OPTOFC feature to reduce

Automatically free a cursor

network messaging

Use functions to convert variables of

- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
 Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
 Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
 Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Time Data Types
- Declare host variables for SQL DATE, DATETIME and INTERVAL data types
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Describe when optional syntax is

appropriate

www.globalknowledge.com.sa

ESQL/C preprocessor

- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3

IX150G

- ESQL/C
- Explain the basic use of the make utility
 Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
 Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5

www.globalknowledge.com.sa

Unit 6: ESQL/C Data Types

- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

00 966 92000 9278

Interface with LVARCHAR data through

Use functions to convert variables of

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

ESQL/C

data types

string data

training@globalknowledge.com.sa

different types

library functions

appropriate

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0
- version syntax and functionality Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESOL /C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type

Use ESQL/C library functions to access data

- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
 Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tablesInstalling a copy of this database
- Identify C variables for use in accessing
- SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in

www.globalknowledge.com.sa

- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
 Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task
Use a scrolling cursor to browse the

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

00 966 92000 9278

INTERVAL macros

Use the OPTOFC feature to reduce

Determine whether SQL NULLs were

Ensure referential integrity using

Automatically free a cursor

network messaging

this structure

statement

information

application logic

selected rows

buffers

statement

statement

a database

stored

sqldetach

sqlbreak

FSQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

training@globalknowledge.com.sa

- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16

this structure

- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

00 966 92000 9278

Determine whether SQL NULLs were

Ensure referential integrity using

Use a scrolling cursor to browse the

Use functions to convert variables of

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

ESQL/C

data types

string data

data type

this structure

statement

information

application logic

selected rows

buffers

training@globalknowledge.com.sa

data

different types

library functions

appropriate

Unit 16: Using Scroll Cursors Effectively

- Determine whether SQL NULLs were
- fetched or character data was truncated Ensure referential integrity using
- application logic Use the appropriate cursor for a given
- task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connectionsUnit 1: Introduction to IBM Informix
- ESQL/C Exercise 1
- Unit 2: The Demonstration Database
 Exercise 2
- Unit 3: Embedding SQL Statements
 Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
 Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14

www.globalknowledge.com.sa

Unit 16: Using Scroll Cursors Effectively
 Unit 17: Using an Update Cursor

- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Character and String Data Types
- Declare host variables for CHAR, CHAR *, VARCHAR, and LVARCHAR data types
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor

- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement

- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Simple Large Objects

- Declare host variables for BYTE and TEXT data types
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
 Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows

www.globalknowledge.com.sa

- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

training@globalknowledge.com.sa

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

00 966 92000 9278

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
 Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality

Change the size of FETCH and INSERT buffers Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Installing a copy of this database

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

BOOLEAN, and DECIMAL data types

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

00 966 92000 9278

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Determine whether SQL NULLs were

Ensure referential integrity using

Use a scrolling cursor to browse the

Use the OPTOFC feature to reduce

Automatically free a cursor

network messaging

Describe the structure for the DECIMAL

Declare host variables for INT8, SERIAL8,

Use functions to convert variables of

ESQL/C preprocessor

SQL databases

program

ESQL/C

data types

string data

data type

this structure

statement

information

application logic

selected rows

buffers

statement

statement

a database

stored

sqldetach

sqlbreak

training@globalknowledge.com.sa

data

different types

library functions

appropriate

- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
 Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Unit 23: Working with the Database

Unit 19: Time Data Types

INSERT Statements

- Exercise 16
- Exercise 17
 Unit 21: Dynamic SQL
 Exercise 18

Server

Exercise 19

www.globalknowledge.com.sa

- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix FSQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- salbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types Exercise 6
- Unit 8: Numeric Data Types Exercise 7

Exercise 8

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8. BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and **INTERVAL** macros
- Describe the locator structure

www.globalknowledge.com.sa

- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
 - Unit 4: Using CONNECT TO
 - Exercise 4
 - Unit 5: Compiling an ESQL/C Program
 - Exercise 5
 - Unit 6: ESQL/C Data Types
 - Unit 7: Character and String Data Types
 - Exercise 6
 - Unit 8: Numeric Data Types
 - Exercise 7
 - Exercise 8
 - Unit 10: The SQL Communications Area
 - Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

00 966 92000 9278

Use functions to convert variables of

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

FSQL/C

data types

training@globalknowledge.com.sa

different types

appropriate

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Numeric Data Types
- Recognize the storage needs for numeric data

Install ESQL/C as part of Client SDK

- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error

IX150G

- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
 Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
 Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data TypesExercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
 Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
 Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Dynamic SQL

www.globalknowledge.com.sa

Use dynamic SQL and the associated data structures and commands

Dynamic SQL: Constructing INSERT Statements

Use dynamic SQL to construct insert statements at runtime

Working with the Database Server

 Explain how to control the database server process with these functions:
 sqlexit string data

- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

primary key to select the current row

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

00 966 92000 9278

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

INTERVAL macros

Declare a cursor from a prepared

Defer execution of a PREPAREd

Ensure referential integrity using application logic

Use a scrolling cursor to browse the

Use the OPTOFC feature to reduce

Automatically free a cursor

network messaging

selected rows

buffers

statement

statement

a database

stored

sqldetach

sqlbreak

ESQL/C Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

training@globalknowledge.com.sa

information

- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18

IX150G

Unit 22: Dynamic SQL: Constructing INSERT Statements

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure

www.globalknowledge.com.sa

INSERT and SELECT simple large

- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14

Exercise 16

Exercise 17

Exercise 18

Exercise 19

- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
 Exercise 15

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task

00 966 92000 9278

Determine whether SQL NULLs were

Ensure referential integrity using

Use a scrolling cursor to browse the

Interface with LVARCHAR data through

Use functions to convert variables of

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

ESQL/C

data types

string data

data type

this structure

statement

information

application logic

training@globalknowledge.com.sa

data

different types

library functions

appropriate

- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak

- objects from a file or from memory sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
 Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

www.globalknowledge.com.sa

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility

selected rows

- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database

INSERT and SELECT simple large objects

- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Unit 23: Working with the Database Server

00 966 92000 9278

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively
 Unit 17: Using an Update Cursor

sqldetach

salbreak

ESOL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

training@globalknowledge.com.sa

- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
 Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types

Explain the storage needs of character and string data

- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connectionsUnit 1: Introduction to IBM Informix
- ESQL/C Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4

www.globalknowledge.com.sa

- Unit 5: Compiling an ESQL/C Program
 Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate

ESQL/C

data types

string data

data type

this structure

statement

information

application logic

selected rows

buffers

statement

statement

a database

stored

training@globalknowledge.com.sa

data

different types

library functions

 Describe how the syntax works with pre-6.0 version syntax and functionality
 Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Declare host variables for INT8, SERIAL8,

Interface with LVARCHAR data through

BOOLEAN, and DECIMAL data types

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

00 966 92000 9278

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Determine whether SQL NULLs were

Ensure referential integrity using

Use a scrolling cursor to browse the

Automatically free a cursor
 Use the OPTOFC feature to reduce

network messaging

Describe the structure for the DECIMAL

Use functions to convert variables of

- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this
- structure
 Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13

Unit 15: Using Scroll Cursors

- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Agenda

- Day 1
- Welcome
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types

www.globalknowledge.com.sa

sqldetach

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

Day 3

- . sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Unit 14: An Overview of Cursors

Install ESQL/C as part of Client SDK

ESQL/C preprocessor

SQL databases

program

training@globalknowledge.com.sa

appropriate

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with pre-6.0 version syntax and functionality

Conditionally preprocess SQL statementsCompile a program by using IBM Informix

00 966 92000 9278

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Inserting Rows

- Code an appropriate embedded SQL statement that inserts a row into a database
- The SQL Communications Area
- Explain the need for and the structure of the SQL Communication Area
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated

IX150G

 Ensure referential integrity using application logic

- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
 Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12

www.globalknowledge.com.sa

ESQL/C

- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic

Automatically free a cursor
 Use the OPTOFC feature to reduce

network messaging

selected rows

buffers

statement

statement

a database

stored

sqldetach

sqlbreak

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

training@globalknowledge.com.sa

Use a scrolling cursor to browse the

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

00 966 92000 9278

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

SQL Exception Testing

- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows

www.globalknowledge.com.sa

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Installing a copy of this database

Describe when optional syntax is

Describe how the syntax works with pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

00 966 92000 9278

Use functions to convert variables of

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

ESQL/C

data types

string data

data type

this structure

statement

data

training@globalknowledge.com.sa

different types

library functions

appropriate

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

Exercise 7

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

- Exercise 8
- Unit 10: The SQL Communications Area

- Describe the kinds of exceptions that might occur during execution of an SQL statement
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak

- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
 Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
 Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

www.globalknowledge.com.sa

information

statement

a database

stored

sqldetach

sqlbreak

FSQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

training@globalknowledge.com.sa

- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 19: Time Data Types

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

00 966 92000 9278

INTERVAL macros

 Declare a cursor from a prepared statement
 Defer execution of a PREPAREd

- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
 Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Using GET DIAGNOSTICS

- Describe the preferred error detection methods
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between

IX150G

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure

www.globalknowledge.com.sa

- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

ESOL/C

data types

string data

data type

this structure

statement

information

application logic

selected rows

buffers

statement

statement

a database

training@globalknowledge.com.sa

data

different types

library functions

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

Simplify exception testing after every SQL

Describe how to obtain warning and error

fetched or character data was truncated

Use the appropriate cursor for a given task

Change the size of FETCH and INSERT

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

00 966 92000 9278

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Determine whether SQL NULLs were

Ensure referential integrity using

Use a scrolling cursor to browse the

Use the OPTOFC feature to reduce

Automatically free a cursor

network messaging

Use functions to convert variables of

Describe when optional syntax is appropriate data types

- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10

IX150G

Unit 12: Using GET DIAGNOSTICS

- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
 Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
 Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
 Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

www.globalknowledge.com.sa

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C

Explain how data is converted as it is stored

INSERT and SELECT simple large objects

- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

Unit 4: Using CONNECT TO

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

Set environment variables and execute the

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Describe when optional syntax is

Describe how the syntax works with

00 966 92000 9278

Installing a copy of this database

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

training@globalknowledge.com.sa

appropriate

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

sqldetach

sqlbreak

ESQL/C

Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Singleton Queries and Lookups

- Use the SELECT statement to retrieve a row from your database
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated

IX150G

 Ensure referential integrity using application logic

- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetachsqlbreak
- Work with multiplexed connections
 Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5

www.globalknowledge.com.sa

- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types

pre-6.0 version syntax and functionality

- Conditionally preprocess SQL statements
 Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
 Change the size of FETCH and INSERT

Use the OPTOFC feature to reduce

Solve the stale data problem by using the

Use an INSERT cursor to insert rows into

INSERT and SELECT simple large objects

Explain how data is converted as it is

Use pre-defined DATETIME and

Describe the locator structure

from a file or from memory

Work with multiplexed connections

Unit 1: Introduction to IBM Informix

Unit 2: The Demonstration Database

Unit 3: Embedding SQL Statements

Unit 5: Compiling an ESQL/C Program

00 966 92000 9278

Unit 4: Using CONNECT TO

INTERVAL macros

primary key to select the current row

Declare a cursor from a prepared

Defer execution of a PREPAREd

Automatically free a cursor

network messaging

buffers

statement

statement

a database

stored

sqldetach

sqlbreak

ESQL/C Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

training@globalknowledge.com.sa

- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
 Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
 Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure

www.globalknowledge.com.sa

Unit 6: ESQL/C Data Types

Unit 8: Numeric Data Types

Unit 11: SQL Exception Testing

Unit 15: Using Scroll Cursors

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

Unit 22: Dynamic SQL: Constructing

Install ESQL/C as part of Client SDK

The relationships between these tables

Identify C variables for use in accessing

Describe the structure of an ESQL/C

Installing a copy of this database

Describe when optional syntax is

Describe how the syntax works with

pre-6.0 version syntax and functionality

Conditionally preprocess SQL statements

Compile a program by using IBM Informix

Explain the basic use of the make utility

Explain problems in converting between

Explain the storage needs of character and

Interface with LVARCHAR data through

Declare host variables for INT8, SERIAL8,

BOOLEAN, and DECIMAL data types

Describe the structure for the DECIMAL

Use ESQL/C library functions to access

Effectively use information contained in

00 966 92000 9278

Use functions to convert variables of

Set environment variables and execute the

Unit 23: Working with the Database Server

Unit 19: Time Data Types

Unit 21: Dynamic SQL

INSERT Statements

ESQL/C preprocessor

SQL databases

program

FSQL/C

data types

string data

data type

this structure

data

training@globalknowledge.com.sa

different types

library functions

appropriate

Unit 12: Using GET DIAGNOSTICS

Unit 13: Singleton Queries and Lookups

Unit 16: Using Scroll Cursors Effectively

Exercise 6

Exercise 7

Exercise 8

Exercise 9

Exercise 10

Exercise 11

Exercise 12

Exercise 13

Exercise 14

Exercise 15

Exercise 16

Exercise 17

Exercise 18

Exercise 19

Unit 7: Character and String Data Types

Unit 10: The SQL Communications Area

- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connectionsUnit 1: Introduction to IBM Informix
- ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15

www.globalknowledge.com.sa

IX150G

- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix
- ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
 - Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14

Exercise 15

training@globalknowledge.com.sa

Unit 16: Using Scroll Cursors Effectively

00 966 92000 9278

Unit 17: Using an Update Cursor

Unit 18: Using an Insert Cursor

- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
 Explain problems in converting between
- data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the

- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd

- primary key to select the current row Declare a cursor from a prepared
- statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

www.globalknowledge.com.sa

IX150G

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables

statement

- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18

Exercise 19

Unit 22: Dynamic SQL: Constructing INSERT Statements

Install ESQL/C as part of Client SDK

ESQL/C preprocessor

SQL databases

program

training@globalknowledge.com.sa

Set environment variables and execute the

The relationships between these tables
 Installing a copy of this database

Identify C variables for use in accessing

00 966 92000 9278

Describe the structure of an ESQL/C

Unit 23: Working with the Database Server

- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fatabased as a share sha
- fetched or character data was truncated Ensure referential integrity using
- application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix

- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO

- ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
 Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access

data

- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and **INTERVAL** macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively

- Exercise 7

- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row

- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix
- ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Day 4

- Unit 20: Simple Large Objects
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the

ESQL/C preprocessor

- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix

ESQL/C

Exercise 1

- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
 Compile a program by using IBM Informix
- ESQL/C Explain the basic use of the make utility
- Explain the basic use of the make utility
 Explain problems in converting between
- data types Use functions to convert variables of
- different types
- Explain the storage needs of character and string data

- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS

- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows

- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and **INTERVAL** macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing **INSERT Statements**
- Unit 23: Working with the Database Server
- Exercise 19

- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach

- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19

Install ESQL/C as part of Client SDK

- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statementsCompile a program by using IBM Informix
- ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of

different types

- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic
- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9

- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server
- Exercise 19
- Install ESQL/C as part of Client SDK
- Set environment variables and execute the ESQL/C preprocessor
- The relationships between these tables
- Installing a copy of this database
- Identify C variables for use in accessing SQL databases
- Describe the structure of an ESQL/C program
- Describe when optional syntax is appropriate
- Describe how the syntax works with pre-6.0 version syntax and functionality
- Conditionally preprocess SQL statements
- Compile a program by using IBM Informix ESQL/C
- Explain the basic use of the make utility
- Explain problems in converting between data types
- Use functions to convert variables of different types
- Explain the storage needs of character and string data
- Interface with LVARCHAR data through library functions
- Declare host variables for INT8, SERIAL8, BOOLEAN, and DECIMAL data types
- Describe the structure for the DECIMAL data type
- Use ESQL/C library functions to access data
- Effectively use information contained in this structure
- Simplify exception testing after every SQL statement
- Describe how to obtain warning and error information
- Determine whether SQL NULLs were fetched or character data was truncated
- Ensure referential integrity using application logic

- Use the appropriate cursor for a given task
- Use a scrolling cursor to browse the selected rows
- Change the size of FETCH and INSERT buffers
- Automatically free a cursor
- Use the OPTOFC feature to reduce network messaging
- Solve the stale data problem by using the primary key to select the current row
- Declare a cursor from a prepared statement
- Defer execution of a PREPAREd statement
- Use an INSERT cursor to insert rows into a database
- Explain how data is converted as it is stored
- Use pre-defined DATETIME and INTERVAL macros
- Describe the locator structure
- INSERT and SELECT simple large objects from a file or from memory
- sqldetach
- sqlbreak
- Work with multiplexed connections
- Unit 1: Introduction to IBM Informix ESQL/C
- Exercise 1
- Unit 2: The Demonstration Database
- Exercise 2
- Unit 3: Embedding SQL Statements
- Exercise 3
- Unit 4: Using CONNECT TO
- Exercise 4
- Unit 5: Compiling an ESQL/C Program
- Exercise 5
- Unit 6: ESQL/C Data Types
- Unit 7: Character and String Data Types
- Exercise 6
- Unit 8: Numeric Data Types
- Exercise 7
- Exercise 8
- Unit 10: The SQL Communications Area
- Exercise 9
- Unit 11: SQL Exception Testing
- Exercise 10
- Unit 12: Using GET DIAGNOSTICS
- Exercise 11
- Unit 13: Singleton Queries and Lookups
- Exercise 12
- Exercise 13
- Unit 15: Using Scroll Cursors
- Exercise 14
- Unit 16: Using Scroll Cursors Effectively
- Unit 17: Using an Update Cursor
- Exercise 15
- Unit 18: Using an Insert Cursor
- Unit 19: Time Data Types
- Exercise 16
- Exercise 17
- Unit 21: Dynamic SQL
- Exercise 18
- Unit 22: Dynamic SQL: Constructing INSERT Statements
- Unit 23: Working with the Database Server

Further Information:

For More information, or to book your course, please call us on 00 966 92000 9278

training@globalknowledge.com.sa

www.globalknowledge.com.sa

Global Knowledge - KSA, 393 Al-Uroubah Road, Al Worood, Riyadh 3140, Saudi Arabia