# DBAS 2010 Design II
# Assignment 2
# Designing Database Tables with SQL

February 21, 2017
(final version with SQL code included)

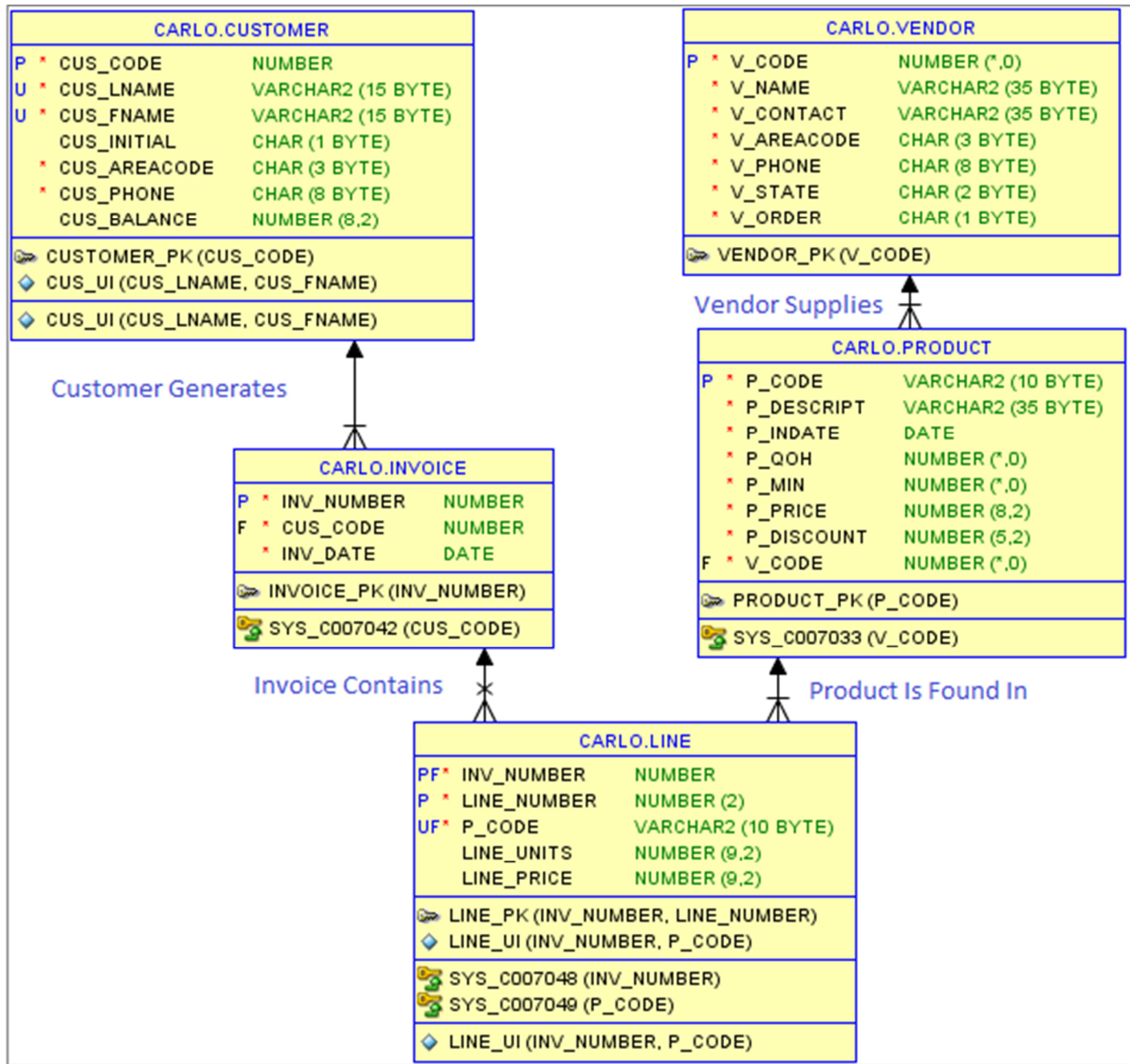Carlo Carandang, w0408173

# Table of Contents

# 1. Objectives

**This assignment has three learning Objectives. They are as follows:**

1.1 To further your understanding of database design by transforming business rules and ERD's into database tables that are implemented in a DBMS using SQL.

1.2 To introduce ANSI Standard SQL. You will receive more instruction in SQL beginning in the next class, however the instruction you have received to this point will be sufficient to complete this assignment.

1.3 To introduce the non-standard and diverse implementation of ANSI Standard SQL across various DBMS platforms and how that implementation may affect your choice of DBMS for the physical implementation of your database.

## 2. Business Rules for Database:

2.1  A customer may generate many invoices.

2.2 Each invoice is generated by one customer.

2.3 Each invoice contains one or more lines.

2.4 Each invoice line is associated with one invoice.

2.5 Each invoice line references one product.

2.6 A product may be found in many invoice lines (on different invoices).

2.7 A vendor may supply many products.

2.8 A product is supplied by only one vendor.

2.9 A vendor may exist that does not supply any products (yet).

## 3. ER Diagram

**CARLO.CUSTOMER**

| | | | |
|---|---|---|---|
| P | * | CUS_CODE | NUMBER |
| U | * | CUS_LNAME | VARCHAR2 (15 BYTE) |
| U | * | CUS_FNAME | VARCHAR2 (15 BYTE) |
| | | CUS_INITIAL | CHAR (1 BYTE) |
| | * | CUS_AREACODE | CHAR (3 BYTE) |
| | * | CUS_PHONE | CHAR (8 BYTE) |
| | | CUS_BALANCE | NUMBER (8,2) |

- CUSTOMER_PK (CUS_CODE)
- CUS_UI (CUS_LNAME, CUS_FNAME)
- CUS_UI (CUS_LNAME, CUS_FNAME)

**CARLO.VENDOR**

| | | | |
|---|---|---|---|
| P | * | V_CODE | NUMBER (*,0) |
| | * | V_NAME | VARCHAR2 (35 BYTE) |
| | * | V_CONTACT | VARCHAR2 (35 BYTE) |
| | * | V_AREACODE | CHAR (3 BYTE) |
| | * | V_PHONE | CHAR (8 BYTE) |
| | * | V_STATE | CHAR (2 BYTE) |
| | * | V_ORDER | CHAR (1 BYTE) |

- VENDOR_PK (V_CODE)

**Vendor Supplies**

**Customer Generates**

**CARLO.INVOICE**

| | | | |
|---|---|---|---|
| P | * | INV_NUMBER | NUMBER |
| F | * | CUS_CODE | NUMBER |
| | * | INV_DATE | DATE |

- INVOICE_PK (INV_NUMBER)
- SYS_C007042 (CUS_CODE)

**CARLO.PRODUCT**

| | | | |
|---|---|---|---|
| P | * | P_CODE | VARCHAR2 (10 BYTE) |
| | * | P_DESCRIPT | VARCHAR2 (35 BYTE) |
| | * | P_INDATE | DATE |
| | * | P_QOH | NUMBER (*,0) |
| | * | P_MIN | NUMBER (*,0) |
| | * | P_PRICE | NUMBER (8,2) |
| | * | P_DISCOUNT | NUMBER (5,2) |
| F | * | V_CODE | NUMBER (*,0) |

- PRODUCT_PK (P_CODE)
- SYS_C007033 (V_CODE)

**Invoice Contains**

**Product Is Found In**

**CARLO.LINE**

| | | | |
|---|---|---|---|
| PF | * | INV_NUMBER | NUMBER |
| P | * | LINE_NUMBER | NUMBER (2) |
| UF | * | P_CODE | VARCHAR2 (10 BYTE) |
| | | LINE_UNITS | NUMBER (9,2) |
| | | LINE_PRICE | NUMBER (9,2) |

- LINE_PK (INV_NUMBER, LINE_NUMBER)
- LINE_UI (INV_NUMBER, P_CODE)
- SYS_C007048 (INV_NUMBER)
- SYS_C007049 (P_CODE)
- LINE_UI (INV_NUMBER, P_CODE)

## 4. High Level Table and Column Description

| Table Name | Column Names | Data Type | Notes (Constraints) |
|---|---|---|---|
| CUSTOMER | CUS_CODE | NUMBER | Primary Key. |
| | CUS_LNAME | VARCHAR2(15) | Not null. |
| | CUS_FNAME | VARCHAR2(15) | Not null. |
| | CUS_INITIAL | CHAR(1) | |
| | CUS_AREACODE | CHAR(3) | Not null. Default value is 902 for Nova Scotia but check that it is one of 902 or 816. |
| | CUS_PHONE | CHAR(8) | Not null. |
| | CUS_BALANCE | NUMBER(8,2) | Default value should be 0.00. |
| | | | ADDED CUS_UI CONSTRAINT TO CHECK THAT THE PAIRING OF CUS_LNAME AND CUS_FNAME IS UNIQUE IN THE TABLE. |
| | | | |
| INVOICE | INV_NUMBER | NUMBER | Primary Key. |
| | CUS_CODE | NUMBER | Not null and references the customer table. |
| | INV_DATE | DATE | Not null. Use system date as the default value. |
| | | | WAS NOT ABLE TO ADD NAMED CONSTRAINT TO CHECK THAT INV_DATE IS GREATER THAN OR EQUAL TO JANUARY 1, 2017. |
| | | | |
| LINE | INV_NUMBER | NUMBER | Part of primary key. Not null. Foreign key that references the invoice table. If the invoice is deleted, then this entry in the line table should be automatically deleted. |
| | LINE_NUMBER | NUMBER(2,0) | Part of primary key. Not null. |
| | P_CODE | VARCHAR2(10) | Foreign key that references the product table. Not null. |
| | LINE_UNITS | NUMBER(9,2) | Default value should be 0.00. |
| | LINE_PRICE | NUMBER(9,2) | Default value should be 0.00. |
| | | | Added line_ui constraint to ensure that the pairing of INV_NUMBER and P_CODE is unique in this table. |
| | | | |
| PRODUCT | P_CODE | VARCHAR2(10) | Primary Key. |
| | P_DESCRIPT | VARCHAR2(35) | Not null. |
| | P_INDATE | DATE | Not null. |
| | P_QOH | SMALLINT | Not null. |
| | P_MIN | SMALLINT | Not null. |
| | P_PRICE | NUMBER(8,2) | Not null. |
| | P_DISCOUNT | NUMBER(5,2) | Not null. |
| | V_CODE | INTEGER | Not null. |

| | | | Added a named constraint that indicates that V_CODE is a foreign key and that it references the vendor table. However, was not able to add the following: If the vendor's V_CODE is updated in the vendor table then this V_CODE entry should be automatically updated. |
|---|---|---|---|
| | | | |
| VENDOR | V_CODE | INTEGER | Primary Key. |
| | V_NAME | VARCHAR2(35) | Not null. |
| | V_CONTACT | VARCHAR2(35) | Not null. |
| | V_AREACODE | CHAR(3) | Not null. |
| | V_PHONE | CHAR(8) | Not null. |
| | V_STATE | CHAR(2) | Not null. |
| | V_ORDER | CHAR(1) | Not null. |

# 5. Questions

1. For your chosen DBMS, what observations can you make with respect to the support that exists or does not exist within that DBMS for the required data types and constraints?
   a. I used Oracle- virtually no support existed when making an error in Oracle SQL command line.
2. If your chosen DBMS cannot support the required constraints as part of a CREATE TABLE command then, for each constraint that cannot be implemented (if any), include a statement that you are deviating from the required constraint as described in the design. Specifically describe the weakness in the chosen DBMS that requires you to abandon this constraint (i.e. *CASCADE ON UPDATE is not supported by ……*).
   a. I was not able to add named constraint to check that inv_date is greater than or equal to January 1, 2017:

   ```
   SQL> ALTER TABLE
     2  invoice
     3  ADD CONSTRAINT
     4  DATE
     5  CHECK(inv_date)>TO_DATE('01-JAN-2017','DD-MON-YYYY');
   CHECK(inv_date)>TO_DATE('01-JAN-2017','DD-MON-YYYY')
                  *
   ERROR at line 5:
   ORA-00920: invalid relational operator
   ```

   b. Added a named constraint that indicates that V_CODE is a foreign key and that it references the vendor table. However, was not able to add the following: If the vendor's V_CODE is updated in the vendor table then this V_CODE entry should be automatically updated.

   ```
   SQL> ALTER TABLE
     2  product
     3  ADD CONSTRAINT
     4  FOREIGN KEY (v_code) REFERENCES vendor,
     5  ON UPDATE CASCADE;
   FOREIGN KEY (v_code) REFERENCES vendor,
                *
   ERROR at line 4:
   ORA-00902: invalid datatype
   ```

3. Continuing from question 2 above, are you aware of a DBMS that supports the required constraint directly within its implementation of SQL? If so, then name the other DBMS and explain why you did not choose that DBMS instead. For example, a reason could be that you did not have ready access to that other DBMS for this assignment or there were more constraints supported in your chosen DBMS, just not the constraint in question or the constraint is implemented automatically by the DBMS because (for example) it is enforced by referential integrity.
   a. As a novice to DBMS systems in general, I only know of the Oracle database. I have Oracle Express Edition and SQL Developer installed on my laptop.
   b. I also have MS Access installed on my laptop, but I am not familiar with it. I hope to gain familiarity with other DBMS's, especially those used in industry.

## 6. Tables in Oracle Database

Includes table names, attributes, and datatypes:

```
Run SQL Command Line
SQL> select tablespace_name, table_name from user_tables;

TABLESPACE_NAME                  TABLE_NAME
------------------------------   -----------------------------
SYSTEM                           CUSTOMER
SYSTEM                           VENDOR
SYSTEM                           PRODUCT
SYSTEM                           INVOICE
SYSTEM                           LINE

SQL> describe customer;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 CUS_CODE                                  NOT NULL NUMBER
 CUS_LNAME                                 NOT NULL VARCHAR2(15)
 CUS_FNAME                                 NOT NULL VARCHAR2(15)
 CUS_INITIAL                                        CHAR(1)
 CUS_AREACODE                              NOT NULL CHAR(3)
 CUS_PHONE                                 NOT NULL CHAR(8)
 CUS_BALANCE                                        NUMBER(8,2)

SQL> describe vendor;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 V_CODE                                    NOT NULL NUMBER(38)
 V_NAME                                    NOT NULL VARCHAR2(35)
 V_CONTACT                                 NOT NULL VARCHAR2(35)
 V_AREACODE                                NOT NULL CHAR(3)
 V_PHONE                                   NOT NULL CHAR(8)
 V_STATE                                   NOT NULL CHAR(2)
 V_ORDER                                   NOT NULL CHAR(1)

SQL> describe product;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 P_CODE                                    NOT NULL VARCHAR2(10)
 P_DESCRIPT                                NOT NULL VARCHAR2(35)
 P_INDATE                                  NOT NULL DATE
 P_QOH                                     NOT NULL NUMBER(38)
 P_MIN                                     NOT NULL NUMBER(38)
 P_PRICE                                   NOT NULL NUMBER(8,2)
 P_DISCOUNT                                NOT NULL NUMBER(5,2)
 V_CODE                                    NOT NULL NUMBER(38)

SQL> describe invoice;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 INV_NUMBER                                NOT NULL NUMBER
 CUS_CODE                                  NOT NULL NUMBER
 INV_DATE                                  NOT NULL DATE

SQL> describe line;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 INV_NUMBER                                NOT NULL NUMBER
 LINE_NUMBER                               NOT NULL NUMBER(2)
 P_CODE                                    NOT NULL VARCHAR2(10)
 LINE_UNITS                                         NUMBER(9,2)
 LINE_PRICE                                         NUMBER(9,2)
```

## 7. SQL Code

```
CREATE TABLE vendor(
v_code INTEGER NOT NULL PRIMARY KEY,
v_name VARCHAR(35) NOT NULL,
v_contact VARCHAR(35) NOT NULL,
v_areacode CHAR(3) NOT NULL,
v_phone CHAR(8) NOT NULL,
v_state CHAR(2) NOT NULL,
v_order CHAR(1) NOT NULL
);
CREATE TABLE product(
p_code VARCHAR(10) NOT NULL PRIMARY KEY,
p_descript VARCHAR(35) NOT NULL,
p_indate DATE NOT NULL,
p_qoh SMALLINT NOT NULL,
p_min SMALLINT NOT NULL,
p_price NUMBER(8,2) NOT NULL,
p_discount NUMBER(5,2) NOT NULL,
v_code INTEGER NOT NULL,
FOREIGN KEY (v_code) REFERENCES vendor);
CREATE TABLE customer (
cus_code NUMBER PRIMARY KEY,
cus_lname VARCHAR(15) NOT NULL,
cus_fname VARCHAR(15) NOT NULL,
cus_initial CHAR(1),
cus_areacode CHAR(3)
DEFAULT '902' NOT NULL
CHECK(cus_areacode IN('902','816')),
cus_phone CHAR(8) NOT NULL,
cus_balance NUMBER(8,2) DEFAULT 0.00,
CONSTRAINT cus_ui UNIQUE(cus_lname,cus_fname));
CREATE TABLE invoice (
inv_number NUMBER PRIMARY KEY,
cus_code NUMBER NOT NULL,
inv_date DATE DEFAULT SYSDATE NOT NULL,
```

1.1 / 2                                                        2017.02.10 23:39:51

```
FOREIGN KEY(cus_code) REFERENCES customer);
CREATE TABLE line (
inv_number NUMBER NOT NULL,
line_number NUMBER(2,0) NOT NULL,
p_code VARCHAR(10) NOT NULL,
line_units NUMBER(9,2) DEFAULT 0.00,
line_price NUMBER(9,2) DEFAULT 0.00,
PRIMARY KEY(inv_number,line_number),
FOREIGN KEY(inv_number) REFERENCES
invoice ON DELETE CASCADE,
FOREIGN KEY(p_code) REFERENCES product,
CONSTRAINT line_ui UNIQUE(inv_number,p_code)
);
```