

 무료 전자 책

배우기

Salesforce

Free unaffiliated eBook created from
Stack Overflow contributors.

#salesforce

SOQL	5
SOQL	6
SOQL	6
Apex SOQL	7
Apex SOQL	7
Apex SOQL	7
.....	7
SOQL.....	7
5: Visualforce	9
Examples.....	9
.....	9
.....	9
6: Visualforce	10
Examples.....	10
\$	10
\$	10
\$	10
7:	11
Examples.....	11
.....	11
.....	11
.....	11
Salesforce ETL	12
.....	12
8:	13
Examples.....	13
.....	13
9:	14
Examples.....	14
.....	14
10:	15
.....	15

.....	15
.....	15
Examples.....	15
.....	15
.....	15
.....	15
/	16
.....	16
.....	16
.....	16
.....	17
.....	18
.....	18
Apex	18
.....	18
.....	18
.....	19
Apex	20
11:	25
.....	25
Examples.....	25
.....	25
ProcessNode.....	26
ProcessInstance.....	26
ProcessInstanceStep ProcessInstanceWorkitem.....	26
ProcessInstanceHistory *	26
12:	28
.....	28
.....	28
Examples.....	28
.....	28
.....	28
.....	29

13:	30
Examples	30
.....	30
.....	30
testSetup	31
.....	31
.....	31
14:	33
Examples	33
.....	33
15:	34
.....	34
Examples	34
.....	34
16:	35
.....	35
Examples	35
.....	35
.....	38

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [salesforce](#)

It is an unofficial and free Salesforce ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official Salesforce.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

1: Salesforce

salesforce .

Salesforce . salesforce .

Examples

Salesforce .

Developer Edition ("DE org") . Developer Edition Salesforce.com . Developer Edition , .

Salesforce

Salesforce .

, [Salesforce](#)

, [Salesforce](#) , [Trailhead](#)

, [Salesforce](#)

,

, [Salesforce](#) , [Trailhead](#)

IoT

, [Salesforce](#)

, [Salesforce](#)

Salesforce : <https://riptutorial.com/ko/salesforce/topic/3768/salesforce->

2: Salesforce CI

Jenkins Sonar CI

Examples

Jenkins ?

Salesforce ? ? CI

SalesForce Jenkins CI

1. [Jenkins](#) : Jenkins , .
2. [Sonar Qube](#) : SonarQube .
3. [Apache Ant](#) : Apache Ant Java , .
4. [Apache Maven](#) : Apache Maven . Maven POM (Project Object Model) , .
5. [SfApexDoc](#) : JavaDoc . Ant / Jenkins .
6. [JUnit APEX](#) : Force.com com.salesforce.ant.DeployTask JUnitReport XML junitreportdir . Jenkins JUnitReport Ant .
7. : **GIT** , **SVN** .
8. [PMD Apex](#) : Apex PMD .
9. [Apex \(enforce-sonarqube-plugin\)](#) : Apex . . SFDC .

[SalesForce CI](https://riptutorial.com/ko/salesforce/topic/10044/salesforce-ci-) : <https://riptutorial.com/ko/salesforce/topic/10044/salesforce-ci->

3: Salesforce REST API

Force.com REST API API .

Examples

OAuth2

OAuth2

```
curl https://login.salesforce.com/services/oauth2/token -d "grant_type=password" -d "client_id=myclientid" -d "client_secret=myclientsecret" -d "username=mylogin@salesforce.com" -d "password=myspassword123456"
```

```
{
  "access_token":
  "00D6F0xxx001g1qs!ARsAQL7BRiQQ01gTW7zXu3kILJBxxxxxHvDnChF2ETBFJpX0T2LsBsm8MVABhAvINAYzqgDIAHhJDp6QjuF62

  "instance_url": "https://ap4.salesforce.com",
  "id": "https://login.salesforce.com/id/00D6F000001xxxxAA/0056F000006DMcxxxx",
  "token_type": "Bearer",
  "issued_at": "14878401xxxx",
  "signature": "Ra5Sdm6gq4xxxeZYk3H2yBIVpZ6hBUDgkQ4Tj9Q="
}
```

```
{
  "tooling": "/services/data/v20.0/tooling",
  "eclair": "/services/data/v20.0/eclair",
  "prechatForms": "/services/data/v20.0/prechatForms",
  "async-queries": "/services/data/v20.0/async-queries",
  "query": "/services/data/v20.0/query",
  "chatter": "/services/data/v20.0/chatter",
  "wave": "/services/data/v20.0/wave",
  "search": "/services/data/v20.0/search",
  "identity": "https://login.salesforce.com/id/00D6F000001g1qsUAA/0056F000006DMcMQAW",
  "subjects": "/services/data/v20.0/subjects",
  "serviceTemplates": "/services/data/v20.0/serviceTemplates",
  "recent": "/services/data/v20.0/recent",
  "connect": "/services/data/v20.0/connect",
  "licensing": "/services/data/v20.0/licensing"
}
```

Salesforce REST API : <https://riptutorial.com/ko/salesforce/topic/9210/salesforce-rest-api>

4: Salesforce (SOQL)

- SELECT ID
- SELECT ID, FROM
- SELECT ID From Account WHERE Name = 'SomeAccountName'
- SELECT ID, (SELECT ID, FROM) FROM Account
- SELECT ID, FROM ID Id :: apexVariableName

Examples

SOQL

```
SELECT Id, Name FROM Account
```

ID . .

SOQL

```
SELECT Name FROM User WHERE IsActive = true
```

.

```
SELECT Name, Phone FROM Contact WHERE CreatedDate >= 2016-01-01T00:00:00.000Z
```

2016 1 1 .

```
SELECT Id, Name FROM Account LIMIT 100
```

100 .

```
SELECT Id, Name, Phone FROM Lead WHERE Phone LIKE '(%) %-%'
```

. "%" .

LIKE '% %' () CONTAINS () .

```
SELECT Email FROM Lead WHERE LeadSource LIKE '%Google%'
```

Google (: 'Google ' 'Google ') .

SOQL

```
SELECT Id, Name FROM User ORDER BY LastName
```

```
SELECT Id, Name FROM Contact ORDER BY LastModifiedDate DESC
```

```
SELECT Name, Title FROM User ORDER BY Title ASC NULLS FIRST
```

```
SELECT Id FROM Contact ORDER BY LastName ASC NULLS LAST, FirstName ASC NULLS FIRST
```

SOQL

SOQL Map .

```
Map<Id, Account> accounts = new Map<Id, Account>([SELECT Id, Name FROM Account]);  
System.debug(accounts);
```

accounts ID . .

```
11:15:10:025 USER_DEBUG [13]|DEBUG|{  
  XXXXXXXXXXXXXXXXXXXX=Account:{Id=XXXXXXXXXXXXXXXXXXXX, Name=Account 1},  
  YYYYYYYYYYYYYYYYYY=Account:{Id=YYYYYYYYYYYYYYYYYYY, Name=Account 2},  
  ZZZZZZZZZZZZZZZZZZ=Account:{Id=ZZZZZZZZZZZZZZZZZZZ, Name=Account 3},  
  ...  
}
```

ID Account . ID Map keySet () .

```
System.debug(accounts.keySet ());
```

:

```
11:23:21:010 USER_DEBUG [15]|DEBUG|{XXXXXXXXXXXXXXXXXXXX, YYYYYYYYYYYYYYYYYY,  
ZZZZZZZZZZZZZZZZZZZ, ...}
```

.

SOQL

- " . .

```
SELECT FirstName, Account.Name, Account.Category__c FROM Contact
```

5 .

```
SELECT Account.Owner.Profile.CreatedBy.Name FROM Contact
```

Primary_Influencer__c __c __r .

```
SELECT Primary_Influencer__r.Nickname__c FROM Contact
```

SOQL

```
SELECT Id, Name, (SELECT Id, FirstName, LastName FROM Contacts) FROM Account
```

Apex SOQL

Apex . . .

```
List<Account> allAccounts = [SELECT Id, Name FROM Account];
Account oldestAccount = [SELECT Id, Name FROM Account ORDER BY CreatedDate LIMIT 1];
```

Apex SOQL

(:).

```
Datetime targetDate = Datetime.now().addDays(-7);
List<Lead> recentLeads = [SELECT Id FROM Lead WHERE CreatedDate > :targetDate];

string targetName = 'Unknown';
List<Contact> incompleteContacts = [SELECT Id FROM Contact WHERE FirstName = :targetName];
```

Apex SOQL

QueryException .

```
try {
    Account a = [SELECT Id FROM Account WHERE Name = 'Non-existent Account'];
} catch (QueryException e) {
    // List has no rows for assignment to SObject
}

try {
    Account a = [SELECT Id FROM Account];
} catch (QueryException e) {
    // List has more than 1 row for assignment to SObject
}
```

SObjectException

```
Account a = [SELECT Id FROM Account LIMIT 1];
try {
    System.debug( a.Name );
} catch (SObjectException e) {
    // SObject row was retrieved via SOQL without querying the requested field: Name
}
```

```
SELECT Id, Name FROM Account WHERE AccountId IN
(SELECT Id FROM Opportunity WHERE IsClosed = false)
```

SOQL

SOQL String .

```
String tableName = 'Account';  
String queryString = 'SELECT Id FROM ' + tableName + ' WHERE CreatedDate >= YESTERDAY';  
List<SObject> objects = Database.query(queryString);
```

SOQL :variable Apex .

Salesforce (SOQL) : <https://riptutorial.com/ko/salesforce/topic/4217/salesforce-----soql->

5: Visualforce

Examples

VisualForce :

```
<apex:page>
  <h1>Hello, world!</h1>
</apex:page>
```

.

?id=SALESFORCE_ID .

```
<apex:page standardController="Account" >apex:page> .
```

```
<apex:page standardController="Account">
  This is a page for {!Account.Name}
</apex:page>
```

.

- cancel() - PageReference ().
- delete() - PageReference .
- edit() - PageReference .
- save() - PageReference .
- view() - PageReference .

.

```
<apex:page standardController="Account">
  Name: <apex:inputField value="{!Account.Name}" />
  <apex:commandButton value="Update record" action="{!save}" />
</apex:page>
```

Visualforce : <https://riptutorial.com/ko/salesforce/topic/6372/visualforce-->

6: Visualforce

Examples

\$

\$Resource .

```

```

URLFOR .

```
<apex:includeScript value="{!URLFOR($Resource.myResources, 'js/app.js')}" />
```

\$

\$Label .

```
<apex:outputText value="{!$Label.Welcome_Message}" />
```

.Welcome_Message Welcome_Message .

```
Welcome to our site, {0}!
```

.

```
<apex:outputText value="{!$Label.Welcome_Message}"
rendered="{!NOT(ISBLANK($User.ContactId))}">
  <apex:param value="{!$User.Contact.FirstName}" />
</apex:outputText>
```

\$

\$User User .

```
You are logged in as a user from {!$User.CompanyName}.
```

Visualforce : <https://riptutorial.com/ko/salesforce/topic/5184/visualforce--->

7:

Examples

, , Visualforce / Lightning / IDE .

- [Force.com IDE](#) - Eclipse
- [JedIDE](#) - IntelliJ IDEA [Force.com IDE](#)
- [MavensMate](#) - [Sublime Text](#) [Atom](#) [VS](#)
- [FuselT SFDC Explorer](#) - .
- [Welkin Suite](#) - .
- [Illuminated Cloud](#) - IntelliJ IDE
- [Aside.io](#) - IDE
- [9](#) - IDE
- [VimAwesome](#) - Force.com VIM
- [HaoIDE](#) - Force.com
- [Metaforce](#) - Salesforce Chrome

- Salesforce Navigator ([Google](#))
- Force.com ([Google](#) , [Firefox](#))
- Salesforce Developer Tool Suite ([Google](#))
- Salesforce Lightning Components Inspector ([Google](#))
- Salesforce Developer Tool Suite ([Google](#))
- Salesforce ([Google](#))
- Salesforce Boost ([Google](#))
- Salesforce API ([GoogleChrome](#))
- ([Google](#))
- Salesforce Inspector ([Google](#) , [Firefox](#))
- Salesforce ([Google](#))
- ([Google](#))

- [Apex](#)
 -
 - Force.com IDE
 - Salesforce .

- -
 - IDE
 -

- :
 -
 - JetBrains
 -

- [Salesforce Apex](#)

-
- VS
-
-

Salesforce ETL

- [Salesforce DataLoader](#)
- [DataLoader.io](#)
-
- [SFXOrgData](#)
- [DreamFactory Monarch](#)
-
-

- [CodeClimate](#) :
- [CodeScan](#) :
- [Clayton.io](#) :
- [VSCode Apex PMD](#) : VS
- [Apex PMD](#) :

: [https://riptutorial.com/ko/salesforce/topic/4095/-](https://riptutorial.com/ko/salesforce/topic/4095/)

8:

Examples

DateTime .

2021 2 . .

```
Integer month = 2;
Integer day = null;
Integer year = 2021;

// Create a new DateTime object for the first day of the month following
// the date you're looking for.
DateTime dtTarget = DateTime.newInstance(year, month, 1);
//In this case we would be sure that month would not be out of bound
dtTarget = dtTarget.addMonths(1);
// Then use the .addDays() method to add negative 1 days. This gives you
// the last day of the target month.
DateTime lastDayOfMonth = dtTarget.addDays(-1);
day = lastDayOfMonth.day();

System.debug(lastDayOfMonth);
System.debug(day);
```

```
18:19:57:005 USER_DEBUG [15]|DEBUG|2021-02-28 08:00:00
18:21:10:003 USER_DEBUG [16]|DEBUG|28
```

add DateTimes .

: <https://riptutorial.com/ko/salesforce/topic/4928/-->

9:

Examples

Salesforce

: Salesforce .

Apex . Apex Apex .

:

```
trigger accountTestTrggr on Account (before insert, before update)
{

    //This only handles the first record in the Trigger.new collection
    //But if more than 1 Account initiated this trigger, those additional records
    //will not be processed
    Account acct = Trigger.new[0];
    List<Contact> contacts = [select id, salutation, firstname, lastname, email
                            from Contact where accountId =&nbsp;;:acct.Id];

}
```

Bulkified :

```
trigger accountTestTrggr on Account (before insert, before update)
{
    List<String> accountNames = new List<String>{};
    //Loop through all records in the Trigger.new collection
    for(Account a: Trigger.new){
        //Concatenate the Name and billingState into the Description field
        a.Description = a.Name + ':' + a.BillingState
    }
}
```

: [https://riptutorial.com/ko/salesforce/topic/4272/-](https://riptutorial.com/ko/salesforce/topic/4272/)

10:

., / (/).).

- .
- .

- .

(Apex).).

., , , Apex SOAP API ([Salesforce](#)).

List Custom Settings . 2 , . Salesforce .

:
Metadata API .

. CMD " ."

Examples

.

>>>

> > >

(). .

.

.

. .

() .

.

Validation Rule Controller

If the custom setting is a list, click **New** to add a new set of data. For example, if your and dialing code.

If the custom setting is a hierarchy, you can add data for the user, profile, or organization whether a specific user is running the app, a specific profile, or just a general user.



▼ Default Organization Level Value

View: [Edit](#) | [Create New View](#)

A | B

Action	Name ↑	Setup Owner
View Edit Del	Validation Rule Controller (Profile)	System Administrator

/

Validation Rule Controller

Create the fields for your custom setting. The data in these fields are cached with the application.

Custom Setting Definition Detail



Label	Validation Rule Controller
API Name	Val_Rule_Cntrlr__c
Visibility	Public
Namespace Prefix	
Last Modified Date	28/07/2016 14:00

All Opportunity Disabled

[Back to Validation Rule Controller](#)

Custom Field Definition Detail

Edit

Field Information

Field Label	All Opportunity Disabled
Field Name	All_Oppportunity_Disabled
API Name	All_Oppportunity_Disabled__c
Description	When checked, all validation rules on the Opportunity object will be disabled.
Help Text	
Created By	Alex Sherwood , 28/07/2016 14:01

General Options

Default Value Unchecked


Validation Rule Controller Edit

Provide values for the fields you created. This data is cached with the application.

Edit Validation Rule Controller

Save Cancel

Validation Rule Controller Information

Location Profile ▼ System Administrator 

All Opportunity Disabled

Validation Rule Controller Edit

Provide values for the fields you created. This data is cached with the application.

Edit Validation Rule Controller

Validation Rule Controller Information

Location
All Opportunity Disabled <input checked="" type="checkbox"/>

```
AND (  
  /* the below is the reference to the Val_Rule_Cntrlr__c custom setting's checkbox field  
  All_Opportunity_Disabled__c  
  */  
  $Setup.Val_Rule_Cntrlr__c.All_Opportunity_Disabled__c = FALSE,  
  
  /* the below is the remainder of the validation rule's formula  
  */  
  CloseDate < TODAY()  
)
```

TRUE .

All_Opportunity_Disabled__c TRUE FALSE .

Apex

Trigger 10 .

Apex // .

```
trigger CloseDateUpdate on Opportunity (before insert, before update) {  
  
  Id userId;  
  Apx_Cntrlr__c userApexController;  
  Boolean userSetting;  
  
  userId = userinfo.getUserId();  
  userApexController = Apx_Cntrlr__c.getInstance(userId);  
  userSetting = userApexController.Close_Date_Update_Disabled__c;  
  
  if (userSetting == false) {  
    for(Opportunity opp : Trigger.new) {  
      opp.CloseDate = date.today().addDays(10);  
    }  
  }  
}
```

```

    }
}
}

```

```

@isTest
public class CloseDateUpdateTest {

    @testSetup
    static void dataSetup() {

        Profile p = [SELECT Id FROM Profile WHERE Name = 'System Administrator' LIMIT 1];

        User u = new User(LastName = 'Test', Alias = 't1', Email = 'example@gmail.com', Username
= 'sotest@gmail.com', ProfileId = p.Id, TimeZoneSidKey = 'America/Denver', LocaleSidKey =
'en_US', EmailEncodingKey = 'UTF-8', LanguageLocaleKey = 'en_US');
        insert u;
    }

    static testMethod void testCloseDateUpdateEnabled() {

        User u = [SELECT Id FROM User WHERE Username = 'sotest@gmail.com'];
        // set the custom setting field to FALSE so that the trigger is not deactivated
        Apx_Cntrlr__c apexController = new Apx_Cntrlr__c(SetupOwnerId =
u.Id, Close_Date_Update_Disabled__c = false);
        upsert apexController;

        Opportunity[] opportunities1 = new Opportunity[]{};

        test.startTest();
        system.runAs(u) {

            for(integer i = 0; i < 200; i++) {
                opportunities1.add(new Opportunity(
                    Name           = 'Test Opp ' + i,
                    OwnerId        = u.Id,
                    StageName      = 'Prospecting',
                    CloseDate      = date.today().addDays(1),
                    Amount         = 100));
            }
            insert opportunities1;
        }
        test.stopTest();

        List<Opportunity> opportunities2 = [SELECT CloseDate FROM Opportunity];

        for(Opportunity o : opportunities2) {
            system.assertEquals(date.today().addDays(10), o.closeDate, 'CloseDateUpdate
trigger should have changed the Opportunity close date as it was not disabled by the
apexController custom setting');
        }
    }

    static testMethod void testCloseDateUpdateDisabled() {

        User u = [SELECT Id FROM User WHERE Username = 'sotest@gmail.com'];
        // set the custom setting field to TRUE to deactivate the trigger
        Apx_Cntrlr__c apexController = new Apx_Cntrlr__c(SetupOwnerId =
u.Id, Close_Date_Update_Disabled__c = true);
        upsert apexController;
    }
}

```



```

Opportunity[] opportunities1 = new Opportunity[]{};

test.startTest();
system.runAs(u){

    for(integer i = 0; i < 200; i++) {
        opportunities1.add(new Opportunity(
            Name           = 'Test Opp ' + i,
            OwnerId        = u.Id,
            StageName      = 'Prospecting',
            CloseDate      = date.today().addDays(1),
            Amount         = 100));
    }
    insert opportunities1;
}
test.stopTest();

List<Opportunity> opportunities2 = [SELECT CloseDate FROM Opportunity];

for(Opportunity o : opportunities2){
    system.assertEquals(date.today().addDays(1), o.closeDate, 'CloseDateUpdate trigger
should not have changed the Opportunity close date as it was disabled by the apexController
custom setting');
}
}
}

```

Apex

Apex . 6 20 .

Val_Rule_Cntrlr__c .

```

global class Scheduled_OppCloseDateUpdate implements Schedulable {

    global void execute(SchedulableContext SC) {
        updOpportunityCloseDates();
    }

    global void updOpportunityCloseDates() {

        Id userId;
        Val_Rule_Cntrlr__c setting;
        Boolean validationRulesAlreadyDisabled;
        List<Opportunity> processedOpps = new List<Opportunity>();
        Date d;

        // get running user's Id
        userId = userinfo.getUserId();
        // retrieve Custom Setting status, for running user
        setting = Val_Rule_Cntrlr__c.getInstance(userId);

        // if the setting field is false, update it to disable validation rules
        if (setting.All_Opportunity_Disabled__c == false) {

```

```

        setting.All_Opportunity_Disabled__c = true;
        upsert setting;
    }
    // if the setting field was already true, there's no need to disable it
    // but it shouldn't be switched to false by this class once the process has been
completed
    else {
        validationRulesAlreadyDisabled = true;
    }

    // execute code to manage business process

    d = system.today().addDays(6);

    for(Opportunity o : [SELECT Id, CloseDate
                        FROM Opportunity
                        WHERE CloseDate <= :d
                        // class only updates open Opportunities
                        AND Probability > 0 AND Probability < 100])
    {
        o.CloseDate = System.today().addDays(20);
        processedOpps.add(o);
    }

    if (processedOpps.size() > 0) {
        update processedOpps;
    }

    // reactivate validation rules
    if (validationRulesAlreadyDisabled == false) {
        setting.All_Opportunity_Disabled__c = false;
        upsert setting;
    }

}
}

```

Trigger_Validation_Rule__c () .

```

AND(
    $Setup.Val_Rule_Cntrlr__c.All_Opportunity_Disabled__c = FALSE,
    Trigger_Validation_Rule__c = TRUE,

    /* allow the above criteria to be met while inserting the Opportunities, without triggering
the rule, in the @testSetup portion of the test */
    NOT(ISNEW())
)

```

true .

```

@isTest
private class WE_ScheduledCloseDateUpdateTest {

    @testSetup
    static void dataSetup() {

        Profile p = [SELECT Id FROM Profile WHERE Name = 'System Administrator' LIMIT 1];
    }
}

```

```

    User u = new User(LastName = 'Test',Alias = 't1',Email = 'example@gmail.com',Username
= 'sotest@gmail.com',ProfileId = p.Id,TimeZoneSidKey = 'America/Denver',LocaleSidKey =
'en_US',EmailEncodingKey = 'UTF-8',LanguageLocaleKey = 'en_US');
    insert u;

    Val_Rule_Cntrlr__c valRuleCntrlr = new Val_Rule_Cntrlr__c(SetupOwnerId =
u.Id,All_Opportunity_Disabled__c = false);
    upsert valRuleCntrlr;

    List<Opportunity> testOpps = new List<Opportunity>();

    // create the Opportunities that will be updated by the class
    for(integer i = 0; i < 200; i++) {
        testOpps.add(new Opportunity(
            Name           = 'Test Opp Update' + i,
            OwnerId       = u.Id,
            StageName     = 'Prospecting',
            CloseDate     = date.today().addDays(1),
            Amount        = 100,
            // set checkbox field to true, to trigger validation rules if they've not been
deactivated by class
            Trigger_Validation_Rule__c = true));
    }
    // create the Opportunities that won't be updated by the class
    for(integer i = 0; i < 200; i++) {
        testOpps.add(new Opportunity(
            Name           = 'Test Opp Skip' + i,
            OwnerId       = u.Id,
            StageName     = 'Prospecting',
            CloseDate     = date.today().addDays(15),
            Amount        = 100,
            Trigger_Validation_Rule__c = true));
    }
    insert testOpps;

}

// code required to test a scheduled class, see
https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/apex\_scheduler.htm
for more details
public static String CRON_EXP = '0 0 0 15 3 ? 2022';

static testmethod void testCloseDateUpdates() {

    // execute scheduled class

    Test.startTest();

    String jobId = System.schedule('ScheduleApexClassTest',
        CRON_EXP,
        new Scheduled_OppCloseDateUpdate());

    CronTrigger ct = [SELECT Id, CronExpression, TimesTriggered, NextFireTime
        FROM CronTrigger
        WHERE id = :jobId];

    System.assertEquals(CRON_EXP, ct.CronExpression);
    System.assertEquals(0, ct.TimesTriggered);
    System.assertEquals('2022-03-15 00:00:00', String.valueOf(ct.NextFireTime));

    Test.stopTest();
}

```

```

// test results

Integer updateCount = 0;
Integer skipCount   = 0;

List <Opportunity> opportunitys = [SELECT Id, Name, CloseDate FROM Opportunity];

for(Opportunity o : opportunitys) {
    if (o.Name.contains('Update') &&
        updateCount == 0)
    {
        System.assertEquals(date.today().addDays(20), o.CloseDate, 'Opportunity\'s
Close Date should have been updated as it was less than 7 days away');
        updateCount = 1;
    }
    if (o.Name.contains('Skip') &&
        skipCount == 0)
    {
        System.assertEquals(date.today().addDays(15), o.CloseDate, 'Opportunity should
not have been updated as it\'s Close Date is more than 7 days away');
        skipCount = 1;
    }
}
// check that both lists of Opportunities have been tested
System.assertEquals(2, updateCount + skipCount, 'Count should be 2 once all assertions
have been executed');

}

// check that the class does not change the custom setting's field to false, if it was
true before class was executed
static testmethod void testSettingUpdates() {

    User u = [SELECT Id FROM User WHERE UserName = 'sotest@gmail.com'];

    // switch the custom setting field to true before the scheduled job executes
    Val_Rule_Cntrlr__c setting;
    setting = Val_Rule_Cntrlr__c.getInstance(u.Id);
    setting.All_Opportunity_Disabled__c = true;
    upsert setting;

    System.runAs(u) {

        Test.startTest();

        String jobId = System.schedule('ScheduleApexClassTest',
                                      CRON_EXP,
                                      new Scheduled_OppCloseDateUpdate());

        CronTrigger ct = [SELECT Id, CronExpression, TimesTriggered, NextFireTime
                          FROM CronTrigger
                          WHERE id = :jobId];

        System.assertEquals(CRON_EXP, ct.CronExpression);
        System.assertEquals(0, ct.TimesTriggered);
        System.assertEquals('2022-03-15 00:00:00', String.valueOf(ct.NextFireTime));

        Test.stopTest();
    }
    setting = Val_Rule_Cntrlr__c.getInstance(u.Id);
}

```

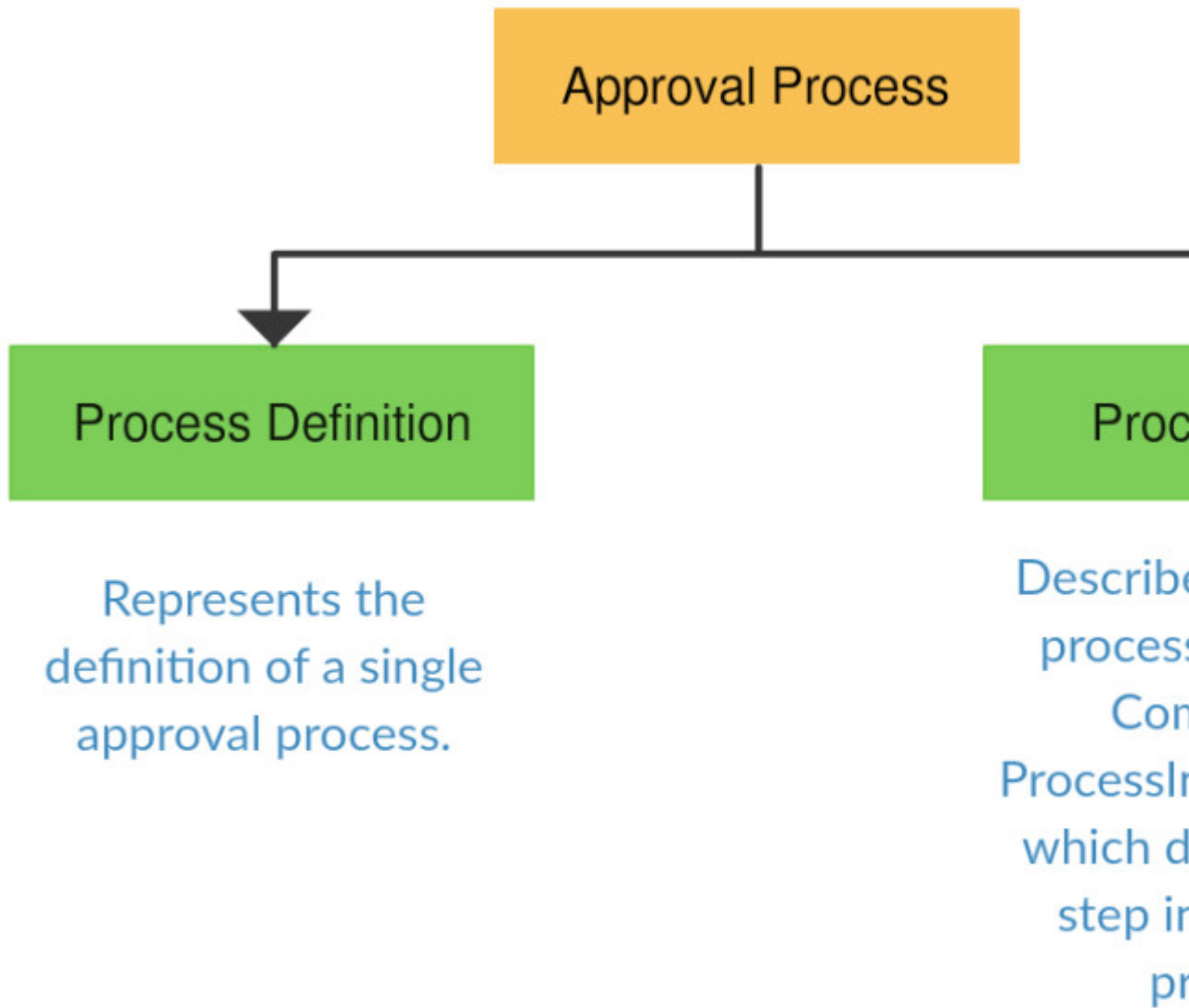
```
        // check that the class did not change the All_Opportunity_Disabled__c field to false
        System.assertEquals(true, setting.All_Opportunity_Disabled__c);
    }
}
```

: [https://riptutorial.com/ko/salesforce/topic/4927/-](https://riptutorial.com/ko/salesforce/topic/4927/)

11:

Salesforce

ProcessDefinition ProcessNode



Above design is High Level Design of Approval Process used to create the Approval Process

Examples

ProcessDefinition Object

~ ~

```
SELECT CreatedById, CreatedDate, Description, DeveloperName, LastModifiedById,
LastModifiedDate, LockType, Name, State, SystemModstamp, TableEnumOrId, Type, Id
FROM ProcessDefinition
```

Salesforce .

ProcessNode

(ProcessDefinition) . . ProcessNode . , , .

~ ~

```
SELECT Description, DeveloperName, Name, ProcessDefinitionId, SystemModstamp
, Id, FROM ProcessNode
```

ProcessDefinitionId ProcessDefinition Object Table . ProcessDefinition .

ProcessInstance

, . ProcessInstance . . .

~ ~

```
SELECT CompletedDate, CreatedById, CreatedDate, ElapsedTimeInDays,
ElapsedTimeInHours, ElapsedTimeInMinutes, Id, IsDeleted, LastActorId,
LastModifiedById, LastModifiedDate, ProcessDefinitionId, Status,
SubmittedById, SystemModstamp, TargetObjectId FROM ProcessInstance
```

ProcessInstance CompletedDate LastActorId . ProcessDefinitionId ProcessDefinition ID.

ProcessInstanceStep ProcessInstanceWorkitem

ProcessInstanceStep ProcessInstanceWorkItem ProcessInstance . ProcessInstanceStep
(ProcessInstance) ProcessInstanceWorkItem (ProcessInstance) . .

~ ~

```
SELECT CreatedById, CreatedDate, ElapsedTimeInDays, ElapsedTimeInHours,
ElapsedTimeInMinutes, Id, IsDeleted, OriginalActorId, ProcessInstanceId,
ActorId, SystemModstamp FROM ProcessInstanceWorkitem
```

```
SELECT ActorId, Comments, CreatedById, CreatedDate, ElapsedTimeInDays, Id,
ElapsedTimeInHours, ElapsedTimeInMinutes, OriginalActorId, ProcessInstanceId
, StepNodeId, StepStatus, SystemModstamp FROM ProcessInstanceStep
```

ProcessInstanceHistory *

ProcessInstanceHistory (ProcessInstance) . Salesforce .
ProcessInstanceStep ProcessInstanceWorkitem ProcessInstanceHistory .

ProcessInstanceHistory ProcessInstance soql . soql ProcessInstance
ProcessInstanceHistory **StepsAndWorkitems** . .

~ ~

```
SELECT CompletedDate, CreatedById, CreatedDate, Id, IsDeleted, LastActorId,  
LastModifiedById, LastModifiedDate, ProcessDefinitionId, Status, SubmittedById  
, SystemModstamp, TargetObjectId, (SELECT ID, ProcessNodeId, StepStatus,  
Comments, TargetObjectId, ActorId, CreatedById, IsDeleted, IsPending,  
OriginalActorId, ProcessInstanceId, RemindersSent, CreatedDate  
FROM StepsAndWorkitems ) FROM ProcessInstance
```

: <https://riptutorial.com/ko/salesforce/topic/6387/-->

12:

- `<> <API> (<>) {/ }`

- API	.
	. insert / update / delete before / after . before after undelete .

Examples

```
trigger AccountTrigger on Account (before insert) {
    System.debug('Account(s) are about to be inserted');
}
```

```
trigger ContactTrigger on Contact (before insert, after insert,
    before update, after update,
    before delete, after delete,
    after undelete) {
    /** Before or After trigger execution**/
    //Returns true if trigger is before
    System.debug('Trigger:Time:Before : ' + Trigger.isBefore);
    //Returns true if trigger is after
    System.debug('Trigger:Time:After : ' + Trigger.isAfter);

    /**DML Operation trigger execution **/
    //Returns true if trigger is insert
    System.debug('Trigger:DML:Insert : ' + Trigger.isInsert);
    //Returns true if trigger is update
    System.debug('Trigger:DML:Update : ' + Trigger.isUpdate);
    //Returns true if trigger is delete
    System.debug('Trigger:DML:Delete : ' + Trigger.isDelete);
    //Returns true if trigger is undelete
    System.debug('Trigger:DML:Undelete: ' + Trigger.isUndelete);

    /** Records on Trigger execution **/
    //Returns data in state before DML. Records are read only
    //Not available for Insert Operation
    //Format: List<sObject>
    List<Contact> old_contacts = Trigger.old;
    System.debug('Trigger:Data:Old : ' + old_contacts);
    //Returns data in state before DML. Records are read only
    //Not available for Insert Operation
    //Format: Map<Id, sObject>
    Map<Id, Contact> old_contacts_map = Trigger.oldMap;
    System.debug('Trigger:Data:OldMap : ' + old_contacts_map);
    //Returns data in state after DML.
    //Allowed for modifications in before context only
    //Not available for Delete Operation
    //Format: List<sObject>
    List<Contact> new_contacts = Trigger.new;
    System.debug('Trigger:Data:New : ' + new_contacts);
    //Returns data in after before DML.
```

```

//Allowed for modifications in before context only
//Not available for InsertOperation
//Format: Map<Id, sObject>
Map<Id, Contact> new_contacts_map = Trigger.newMap;
System.debug('Trigger:Data:NewMap : ' + new_contacts_map);

/** Another context variables */
//Returns amount of record in DML for trigger execution
System.debug('Trigger:Size      : ' + Trigger.size);
//Returns true if the current context for the Apex code
//is a trigger, not VF, web service or anonymous apex
System.debug('Trigger:isExecuting : ' + Trigger.isExecuting);

//Simple example how to use above context variables
//for different scenarios in combination
if (Trigger.isBefore && Trigger.isUpdate) {
    // actions for before update
} else if (Trigger.isAfter) {
    if (Trigger.isUpdate) {
        // actions for after update
    } else if (Trigger.isInsert) {
        // actions for after insert
    }
}
}

```

```

trigger MyTrigger on SomeObject__c (after insert, after update) {
    if (Trigger.isAfter && Trigger.isInsert) {
        System.debug('The following records were inserted: ');
        for (SomeObject__c o : Trigger.new) {
            System.debug(o.Name);
        }
    } else if (Trigger.isAfter && Trigger.isUpdate) {
        for (Id key : Trigger.newMap) {
            SomeObject__c theOldOne = Trigger.newMap.get(key);
            SomeObject__c theNewOne = Trigger.oldMap.get(key);
            if (theNewOne.Name != theOldOne.Name) {
                System.debug('The name of ' + key + ' has been changed');
            }
        }
    }
}
}

```

[: https://riptutorial.com/ko/salesforce/topic/4864/-](https://riptutorial.com/ko/salesforce/topic/4864/-)

13:

Examples

```
static TestMethod void DmlTest() {
    List<Contact> allContacts = [SELECT Id FROM Contact];
    System.assert(allContacts.isEmpty());

    Contact testContact = new Contact(FirstName = 'John', LastName = 'Doe');
    insert testContact;
    allContacts = [SELECT Id FROM Contact];
    System.assertNotEquals(0, allContacts.size(), 'Optional message in case of failure');

    delete allContacts;
    allContacts = [SELECT Id FROM Contact];
    System.assertEquals(0, allContacts.size());
}
```

```
SomeClass IsBlank(...) . SomeClass . static .
```

```
public class SomeClass {

    public static Boolean IsBlank(String someData) {
        if (someData == null) {
            return true;
        } else if (someData == '') {
            return true;
        } else {
            return false;
        }
    }
}
```

```
if. System.assertEquals(...) IsBlank(...) .
```

```
@isTest
public class SomeClass_test {

    @isTest
    public static void SomeClass_IsBlank_test() {

        String testData;

        // SomeClass.IsBlank() returns true for Null values
        System.assertEquals(true, SomeClass.IsBlank(testData));

        testData = '';

        // SomeClass.IsBlank() returns true for empty strings
        System.assertEquals(true, SomeClass.IsBlank(testData));

        testData = 'someData';

        // SomeClass.IsBlank() returns false when testData is neither
```

```

        // an empty string nor Null
        System.assertEquals(false, SomeClass.IsBlank(testData));
    }
}

```

testSetup

@testSetup .

```

public class AccountService {
    public static Account fetchAccount() {
        return [ SELECT Id, Name FROM Account LIMIT 1 ];
    }
}

```

```

@isTest
public class AccountServiceTest {
    private static final String TEST_ACCOUNT_NAME = 'My Test Account';

    @testSetup
    public static void setUpAccountData() {
        Account a = new Account(Name = TEST_ACCOUNT_NAME);
    }

    @isTest
    public static void testFetchAccount() {
        Account a = AccountService.fetchAccount();
        System.assertNotEquals(null, a, 'Account should not be null');
        System.assertEquals(TEST_ACCOUNT_NAME, a.Name, 'Account name should be correct');
    }
}

```

@testSetup . static .

```

@isTest
public class MyTest {
    static {
        // code here will be run before each test is executed
    }
}

```

true System.assert System.assert .

```

System.assert(Service.isActive());
System.assert(!Service.getItems().isEmpty(), 'items should not be empty');

```

System.assertEquals System.assertNotEquals . .

```

System.assertEquals(4, Service.getItems().size());
System.assertNotEquals(null, Service.getItems());

// failure messages are optional:

```

```
System.assertEquals(true, Service.doWork(), 'doWork should return true');  
System.assertNotEquals(null, Service.doWork(), 'doWork should not be null');
```

: [https://riptutorial.com/ko/salesforce/topic/4930/-](https://riptutorial.com/ko/salesforce/topic/4930/)

14:

Examples

salesforce . data.gov REST .

HTTP .

```
string url= 'http://search.ams.usda.gov/farmersmarkets/v1/data.svc/zipSearch?zip=10017';
Http h = new Http();
HttpRequest req = new HttpRequest();
HttpResponse res = new HttpResponse();
req.setEndpoint(url);
req.setMethod('GET');
res = h.send(req);
System.Debug('response body '+res.getBody());
```

: <https://riptutorial.com/ko/salesforce/topic/4926/-->

15:

Apex . [UserInfo Class] [1]. / . / . [1] :
https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/apex_methods_system_userinfo.htm

Examples

[getFirstName \(\)](#) - .
[getLastName \(\)](#) - .

: <https://riptutorial.com/ko/salesforce/topic/8174/-->

16:

Slasforce StandardSetController sObject List

Examples

:

```
public with sharing class Pagination {
}
,
```

```
public class contactWrapper{
    public Contact cont {get;set;}
    public Boolean isSelected{get;set;}
    public contactWrapper(Contact c, Boolean s)
    {
        cont=c;
        isSelected=s;
    }
}
```

Apex StandardSetController (Paginate) . StandardSetController Visualforce
StandardSetController SOQL

```
Public Integer noOfRecords{get; set;} // Future reference in Visual force Page
Public Integer size{get;set;} // Future reference in Visual force Page
public final Integer Page_Size=10; // Number records in a Page should be displayed

public ApexPages.StandardSetController setCon {
    get{
        if(setCon == null){
            size=Page_Size;
            string queryString = 'Select Id,Name, Email, Birthdate, Phone, MobilePhone
from Contact order by Name';
            setCon = new
ApexPages.StandardSetController(Database.getQueryLocator(queryString));
            setCon.setPageSize(size);
            noOfRecords = setCon.getResultSize();
        }
        return setCon;
    }set;
}
```

setCon.getRecords () setCon 10 Variable setCon .
ApexPages.StandardSetController sObject List Visual Force

```
public list<contactWrapper> contWpr{get;set;}
public set<id> selectedContactIds{ get,private set;} // to maintain state of the selected
contact
```



```

// through out paginating

public Pagination() {
    selectedContactIds=new Set<id>();
}

public List<ContactWrapper> getContacts(){
    contWpr =new List<ContactWrapper>();
    for(Contact c: (List<Contact>)setCon.getRecords())
        if(selectedContactIds.contains(c.id))
            contWpr.add(new ContactWrapper(c,true));
        else
            contWpr.add(new ContactWrapper(c,false));
    return contWpr;
}

```

? ApexPages.StandardSetController.Look StandardSetController , , .
StandardSetController . .

```

public Boolean hasNext {
    get {
        return setCon.hasNext();
    }
    set;
}

public Boolean hasPrevious {
    get {
        return setCon.hasPrevious();
    }
    set;
}

public Integer pageNumber {
    get {
        return setCon.getPageNumber();
    }
    set;
}

public void first() {
    setCon.first();
    // do you operation here
}

public void last() {
    setCon.last();
    // do you operation here
}

public void previous() {
    setCon.previous();
    // do you operation here
}

public void next() {
    setCon.next();
    // do you operation here
}

```

```

public void contactSelection()
{
    Id id=(Id)ApexPages.currentPage().getParameters().get('cId');
    if(selectedContactIds.contains(id))
        selectedContactIds.remove(id);
    else
        selectedContactIds.add(id);
}

public void deleteContacts()
{
    List<contact> contactToDelete=[select id from contact where id in
:selectedContactIds];
    if(contactToDelete.size()!=0) // if(!contactToDelete.isEmpty()) // Best Practice
    {
        try { delete contactToDelete; } // You may get Exception if you try to
delete the // related contact ,include try block to avoid
error.
        catch(exception ex){ System.debug(ex); }
        refresh();
    }
}

public pageReference refresh() {
    setCon = null;
    selectedContactIds=new set<id>();
    getContacts();
    setCon.setPageNumber(1);
    return null;
}

```

.. : <https://riptutorial.com/ko/salesforce/topic/10744/----->

S. No		Contributors
1	Salesforce	abhi , Alex S , Andrii Muzychuk , Ashwani , Community , Reshma
2	SalesForce CI	Sanjay Kharwar
3	Salesforce REST API	radbrawler
4	Salesforce (SOQL)	abhi , Alex S , Andrii Muzychuk , battery.cord , Ben , ca_peterson , Doug B , Eric Dobbs , LDP , Ratan Paul
5	Visualforce	Ben
6	Visualforce	Ben
7		abhi , Andrii Muzychuk , Daniel Ballinger , Force2b , Gres , hlebot bread , itzmukeshy7 , Mahmood , NSjonas , Pavel Slepiankou , pchittum , Ratan Paul , sorenkrabbe , wintermute
8		kurunve , LDP , RamenChef
9		abhi , hillary.fraley , LDP
10		Alex S
11		Ajay Gupta
12		kurunve , Pedro Otero
13		Andree Wille , battery.cord , Ben , Eric Dobbs
14		abhi , mnoronha
15		Andrii Muzychuk
16	.	NITESH K