[salesforce.stackexchange.com](http://salesforce.stackexchange.com/)  
•      salesforce subreddit (<https://www.reddit.com/r/salesforce/>)  
•      salesforce discord (<https://discordapp.com/invite/6eQFVUv>)  
•      <https://dramatic-technician-f4b.notion.site/Salesforce-Developer-Module-4b6a7600e1cd4c3d8da83a86fc248b0c>  
•      [https://www.apexsandbox.io](https://www.apexsandbox.io/)

reddit**reddit**

[**Salesforce • r/salesforce**](https://www.reddit.com/r/salesforce/)

This is an unofficial (no salesforce sponsorship) forum to discuss getting certified and building on salesforce platforms ([force.com](http://force.com/), heroku,...

**Discord**

[**Join the SFXD - Salesforce X Discord Discord Server!**](https://discordapp.com/invite/6eQFVUv)

SFXD is a community-first Salesforce discussion group founded in 2016, with a very active core membership. | 12,859 members (4 kB)

<https://discordapp.com/invite/6eQFVUv>

**SELCUK GOKTAS's Notion on Notion**

[**Salesforce Developer  (Module)**](https://dramatic-technician-f4b.notion.site/Salesforce-Developer-Module-4b6a7600e1cd4c3d8da83a86fc248b0c)

Salesforce Developer (1 MB)

<https://dramatic-technician-f4b.notion.site/Salesforce-Developer-Module-4b6a7600e1cd4c3d8da83a86fc248b0c>

**apexsandbox.io**

[**Apex Practice Problems - ApexSandbox.io**](https://www.apexsandbox.io/)

Become an expert Apex developer with a growing collection of Apex programming problems

public class MyOutterClass {

private String outterText;

private Integer outterNumber;

public void myOutterMethod(){

System.debug('This is outter class');

}

public class MyInnerClass1{

public void myInnerMethod1(){

System.debug('This is inner class1');

}

}

public class MyInnerClass2{

public void myInnerMethod2(){

System.debug('This is inner class2');

}

}

}

MyOutterClass outterTest = new MyOutterClass();

outterTest.myOutterMethod();

MyOutterClass.MyInnerClass1 innerTest1 = new MyOutterClass.MyInnerClass1();

innerTest1.myInnerMethod1();

Dinner.Meal myMeal = new Dinner.Meal();

MyMeal.starterMethod();

// Try Catch Finally

Try {

// the code that needs to be checked for error

}Catch (){

// the code that will be run when above code hits any error

}Finally{

// the code will always run

}

Account myAccount = new Account();

myAccount.name = 'Salesforce';

insert myAccount;

Try{

Account myAccount = new Account();

insert myAccount;

} Catch(Exception e){

System.debug('Error occured ' + e.getMessage());

}Final

Account myAccount = new Account();

myAccount.name = 'Salesforce';

insert myAccount;

Try{

Account myAccount = new Account();

insert myAccount;

} Catch(Exception e){

System.debug('Error occured ' + e.getMessage());

} Finally{ //optional

System.debug('This is finally block');

}

1. Data Types  
A. Primitive Data types  
    //If you will not assign any value return will be null  
String  
    Syron  
Integer  
Long  
Decimal  
Double  
Date  
DateTime  
Time  
Boolean  
Id  
Blob  
Collections  
List  
Set  
Map  
Constants  
Final  
EnumControl Flow Statements  
If Else  
Switch  
Loops  
Do While  
While  
For  
Traditional For  
List/Set For  
SOQL For  
Classes  
Class  
Method  
Encapsulation  
Abstraction  
Inheritance  
Polymorphism  
Inner Class  
Try Catch Finally Error Handling

Account myAccount = new Account();

myAccount.name = 'Salesforce';

insert myAccount;

Try{

Account myAccount = new Account();

insert myAccount;

} Catch(Exception e){

System.debug('Error occured ' + e.getMessage());

} Finally{ //optional

System.debug('This is finally block');

}

1. Data Types

A. Primitive Data types

//If you will not assign any value return will be null

String

String myText = 'I can create a text on Apex'; //single quotes make anything string

myText.capitalize(); // available functions with (.) notation

Integer

Integer myNumber = 4; // whole numbers up until ~2 billion

Long

Long myLongNumber = 3000000000L; // bigger numbers than ~ 2 billion, use L at the end

Decimal

Decimal myDecimalNumber = 3.55; // decimal numbers up to ~2 billion

Double

Double myDoubleNumber = 3000000000.56; // decimal numbers bigger than ~2 billion

Date

Date myDate = Date.newInstance(2022, 12, 21);

Date currentDate = Date.today();

currentDate.format();

DateTime // Date, DateTime, and Time logics are the same

Time// Date, DateTime, and Time logics are the same

Boolean

Boolean myChoice = true; // True, False or null

Id

Id myId = '00B3t00000DrPXpEAN'; // 15 or 18 alphanumeric characters

Blob

Blob myBlob = Blob.valueOf(Text); // (eg.: 101001101) - attachment, web service or files

B. Collections

List

List<String> myList = new List<String> ();

myList.add('item1');

myList.add('item2');

myList.remove(0); // indexed ordered list starting from 0

Set // values are unique

Set<Integer> mySet = new Set<Integer> ();

mySet.add(5);

mySet.add(4);

mySet.add(5); // this will not add since all values are unique in the Set

mySet.remove(4); // there is no index

Map // key value pairs: keys should be unique value can be the same ( eg. sudent id (key), student number)

Map<String, String> myMap = new Map<String, String>();

myMap.put('S0001', 'Michael');

myMap.put('S0002', 'Michael'); // key is unique, so value can be the same

myMap.remove('S0001'); // put key value to remove

Constants

Final

final Decimal PI\_NUMBER = 3.14; // SNAKE\_CASE

final Decimal DISCOUNT\_RATE = .05 // 5% discount

Enum

Control Flow Statements

If Else

Switch

Loops

Do While

While

For

Traditional For

List/Set For

SOQL For

Classes

Class

Method

Encapsulation

Abstraction

Inheritance

Polymorphism

Inner Class

Try Catch Finally Error Handling

Account myAccount = new Account();

myAccount.name = 'Salesforce';

insert myAccount;

Try{

Account myAccount = new Account();

insert myAccount;

} Catch(Exception e){

System.debug('Error occured ' + e.getMessage());

} Finally{ //optional

System.debug('This is finally block');

}

1. Data Types

A. Primitive Data types

//If you will not assign any value return will be null

String

String myText = 'I can create a text on Apex'; //single quotes make anything string

myText.capitalize(); // available functions with (.) notation

Integer

Integer myNumber = 4; // whole numbers up until ~2 billion

Long

Long myLongNumber = 3000000000L; // bigger numbers than ~ 2 billion, use L at the end

Decimal

Decimal myDecimalNumber = 3.55; // decimal numbers up to ~2 billion

Double

Double myDoubleNumber = 3000000000.56; // decimal numbers bigger than ~2 billion

Date

Date myDate = Date.newInstance(2022, 12, 21);

Date currentDate = Date.today();

currentDate.format();

DateTime // Date, DateTime, and Time logics are the same

Time// Date, DateTime, and Time logics are the same

Boolean

Boolean myChoice = true; // True, False or null

Id

Id myId = '00B3t00000DrPXpEAN'; // 15 or 18 alphanumeric characters

Blob

Blob myBlob = Blob.valueOf(Text); // (eg.: 101001101) - attachment, web service or files

B. Collections

List

List<String> myList = new List<String> ();

myList.add('item1');

myList.add('item2');

myList.remove(0); // indexed ordered list starting from 0

Set // values are unique

Set<Integer> mySet = new Set<Integer> ();

mySet.add(5);

mySet.add(4);

mySet.add(5); // this will not add since all values are unique in the Set

mySet.remove(4); // there is no index

Map // key value pairs: keys should be unique value can be the same ( eg. sudent id (key), student number)

Map<String, String> myMap = new Map<String, String>();

myMap.put('S0001', 'Michael');

myMap.put('S0002', 'Michael'); // key is unique, so value can be the same

myMap.remove('S0001'); // put key value to remove

Constants

Final

final Decimal PI\_NUMBER = 3.14; // SNAKE\_CASE

final Decimal DISCOUNT\_RATE = .05 // 5% discount

Enum

Enum String SEASON {'WINTER', 'SUMMER'}

Season.Winter

Control Flow Statements

If Else

If(x<10){ //fist logic - true or false

// code block to execute if logic is true

} else if(x==6){

// code block to execute if logic is true

} else if (x==){

// code block to execute if logic is true

}else{

// code block to execute if none of the logics above true

}

Switch

Switch on x {

when 1{

// code block to execute

} when 2,4{

// code block to execute

} } when 9, 10{

// code block to execute

}

when else{

// code block to execute

}

}

Loops

Do While

Do{

// the code will run at least one time

//do not forget to add exit logic like increase or decreas otherwise governor limitrs will stop the loop

x++;

}while (condition)

While

While (condition){

// it may not be run if condition will not met

//do not forget to add exit logic like increase or decreas otherwise governor limitrs will stop the loop

}

For

Traditional For

for(Integer myInteger = 1; myInteger<5; myInteger++){// (Initial staement; exit condition; increase or decrease)

// code block to execute

}

for (Integer i=1; i<10; i++){

// I will use i variable in the code block and it will be executed

}

List/Set For

// List<String> myColors = new List<String>(); // String list

List<Account> myAccountList = List<Account>();

for(Account newAccount : myAccountList){ // for (Return type itemlabel : listName)

// code block to execute

}

SOQL For

for( Account newAccount : [SELECT Name FROM Account]){ // for (returnType itemLabel : SOQL Query)

// code block to execute

}

Classes

Class

public class ClassName{ // class declaration and ClassName is required

}

Method

public void MethodName{ // return type and MethodName is required

}

Encapsulation

Abstraction

Inheritance

Polymorphism

Inner Class

Try Catch Finally Error Handling