integer n = 23;

integer r, q;

//you need to calculate the value of D meain division and R meins Reminder

q = n/2;

r = System.Math.mod(n, 2);

if (n == 0)

System.debug(n +' '+ 'is Even');

else

System.debug(n +' '+ 'is Odd');

=====

integer a=35, b=55, c=45;

if (a > b){

System.debug(a +'is big');

}

else if (b > c)

{

System.debug(b +'is big');

}

else {

System.debug(c +'is big');

}

======

integer a = 23, b = 56, c = -78;

if (a > b){

if (a > c)

System.debug(a + 'is big');

else

System.debug(c + 'is big');

}

else {

if (b > c)

System.debug(b +'is big');

else

System.debug(c + 'is big');

}

======

integer n = 3;

for (integer x = 1; x <=10; x++)

System.debug(n + '\*' +1 + '=' + n \* x);

it will help you to write a Table.

// example -

//3 \* 1 = 3

//3 \* 2 = 6

============

integer [] a = new Integer [3];

a [0] = 10;

a [1] = 20;

a [2] = 30;

System.debug(‘My Integer Number is ' +a[1]);

===============

integer [] a = new integer[] {10,20,30};

string [] s = new string [] {'salesforce', 'visualforce', 'cloudcomputing'};

for(integer i = 0; i<a.size(); i++)

System.debug(a[i] +' '+ s[i]);

==============

double salary = 10000;

double tax = 0.0;

if (salary <= 15000){

tax = salary \* .10;

System.debug('You have to pay '+ tax + ' as a tax amount');

} else if (salary <= 40000){

tax = salary \* .20;

System.debug('You have to pay this much amount '+ tax);

}else {

tax = salary \* .30;

System.debug('You have to pay this much amount ' +tax);

}

==============

string a = 'Afghan Khair';

//System.debug(a);

//System.debug(a.touppercase());

System.debug(a.substring(0,6).touppercase());

===============

trigger opptrigger1 on Opportunity (before insert) {

for (Opportunity a : trigger.new){

if (a.amount < 5000 )

a.adderror ('amount can be less 5000');

}

}

===============

trigger myopptrigger on Opportunity (before insert, before update ) {

for (Opportunity a : trigger.new){

if (trigger.isInsert && a.amount < 5000 )

a.adderror ('Amount can not be less 5000');

else if (trigger.isUpdate && a.amount <3000)

a.adderror ('Updated Amount can not be less than 3000');

}

}

===============

trigger myopptrigger on Opportunity (after insert ) {

contact c = new contact();

for(Opportunity o : trigger.new){

c.accountid = o.accountid;

c.firstname = 'Opportunity';

c.lastname = 'Owner';

insert c;

}

}

===============

trigger mycontact on Contact (before insert, before update) {

for (contact c : trigger.new){

if(c.email == null)

c.adderror ('Please fill email id');

}

}

====================

trigger contact on Contact (before insert, before update) {

for (contact c : trigger.new){

if (c.phone == null)

c.adderror ('please fill phone number');

}

}

==============

trigger contact on Contact (before insert, before update) {

for (contact c : trigger.new) {

if (trigger.isinsert && c.email == null)

c.adderror ('you can not creare without email');

}

}

==============

trigger mytrigger on Account (before insert, before update) {

for (account a : trigger.new){

if (a.industry == 'education')

a.adderror ('we are not working with this industry');

}

}

============

public class trigger\_class {

public static void check\_trigger (trigger.new) {

for (Opportunity op : trigger.new);

total \_amount += op.amount;

if (total\_amount >100000)

op.adderror (‘you have reached your limit’);

}

}

Call the class method.

trigger opp\_trigger on Opportunity (before insert) {

trigger\_class. check\_trigger (trigger.new);

}

========

public class soqlExample2 {

public List <opportunitiy> opps {set;get;}

public soqlExample2 (){

opps=[select name, stagename, amount, closeddate, createddated from opportunitiy];

}

}

<apex:page controller="soqlExample2">

<apex:from>

<apex:pageBlock title = "opportunitiy">

<apex:pageBlockTable value = "{!opps}" var = "a">

<apex:column value = "{!a.name}"/>

<apex:column value = "{!a.stagename}"/>

<apex:column value = "{!a.amount}"/>

<apex:column value = "{!createddate}"/>

<apex:column value = "{!closeddate}"/>

</apex:pageBlockTable>

</apex:pageBlock>

</apex:from>

</apex:page>

================

Integer countnumber = [select count() from account where industry = 'energy'];

system.debug('countnumber :' +countnumber);

aggregateresult result = [select count (phone) from account];

system.debug (result);

system.debug (result.get('expr0'));

aggregateresult result1 = [select sum (amount), min(amount), max(amount) from opportunity];

system.debug ('Total Amount :' +result1.get('expr0'));

system.debug ('Mininum Amount :' +result1.get('expr1'));

system.debug ('Maxmum Amount :' +result1.get(‘expr2'));

================

by this you can insert any record

Lead myLead = new Lead ();

myLead.FirstName = 'John';

myLead.LastName = 'Matthew';

myLead.Company = 'IBM';

insert myLead;

===============

//Defining array

List <string> mylist = new list <string>();

String [] arrayOfProducts = new List<String>();

//Adding elements in Array

arrayOfProducts.add('HCL');

arrayOfProducts.add('H2SO4');

arrayOfProducts.add('NACL');

arrayOfProducts.add('H2O');

arrayOfProducts.add('N2');

arrayOfProducts.add('U296');

for (Integer i = 0; i<arrayOfProducts.size(); i++) {

//This loop will print all the elements in array

system.debug ('Values In Array : ' +arrayOfProducts [i]);

}

================

string [] mp = new list <string>();

mp.add ('hcl');

mp.add ('ibm');

mp.add ('kbm');

for (integer i = 0; i<mp.size(); i++) {

system.debug('my product name :' +mp[i]);

}

=============

**Example of LIST**

List<string> ListOfCities = new List<string>();

System.debug('Value Of ListOfCities'+ListOfCities);

Declaring the initial values of list is optional, but we can do that. Below is the example for the same.

List<string> ListOfStates = new List<string> {'NY', 'LA', 'LV'};

System.debug(' Value ListOfStates'+ListOfStates);

List of Accounts (sObject):

List<account> AccountToDelete = new List<account> ();//This will be null

System.debug(' Value AccountToDelete'+AccountToDelete);

**Nested List**

We can declare the nested List as well. It can go up to five level. This is called as Multidimensional list.

This is the list of set of integers.

List<List<Set<Integer>>> myNestedList = new List<List<Set<Integer>>>();

System.debug('value myNestedList'+myNestedList);

================

Set<string> ProductSet = new Set<string>{'Phenol', 'Benzene', 'H2SO4'};

System.debug('Value of ProductSet’+ProductSet);

Set <string> ProductSet = new Set<string>();

ProductSet.add('Phenol');

ProductSet.add('Benzene');

ProductSet.add('H2SO4');

System.debug('Value of ProductSet' +ProductSet);

Set<string> ProductSet = new Set<string>{'Phenol', 'Benzene', 'H2SO4'};

ProductSet.add('HCL');

System.debug('Set with New Value ‘+ProductSet);

System.debug('Set with New Value '+ProductSet.size());

ProductSet.remove('HCL');

System.debug('Set with removed value ‘+ProductSet);

System.debug('Set with New Value '+ProductSet.size());

ProductSet.contains('HCL');

System.debug('Value of Set with all values ‘+ProductSet);

================

integer a=20, b=30, c=40;

if (a > b) {

if (a > c);

System.debug (a + 'is big');

else

system.debug (c +'is big');

}

esle {

if (b > c);

system.debug(b + 'is big');

else

system.debug(c 'is big';)

}

=================

<apex:page >

<apex:SectionHeader title="My Visualforce Page" subtitle="New Page" Description="this is my 3rd page for visualforce page " help="http://www.google.com"/>

</apex:page>

==============

<apex:page standardController="account">

{!account.name}=={!account.phone}=={!account.industry}

</apex:page>

<https://c.ap2.visual.force.com/apex/standardcontroller?id=0012800000tBa1p>

==============

<apex:page standardController="account">

<apex:form >

<apex:inlineEditSupport >

<apex:outputfield value="{!account.name}"/>

<apex:outputlabel value="{!account.name}"/><br/>

<apex:outputtext value="{!account.name}"/><br/>

</apex:inlineEditSupport>

</apex:form>

</apex:page>

<https://c.ap2.visual.force.com/apex/inlineeditsupport?id=0012800000tBa1p>

===============

<apex:page >

<apex:outputlabel> Enter Name </apex:outputlabel>

<Script>

function show(){

alert("This is a notification")

}

</Script>

<apex:form>

<apex:commandButton value ="click" onclick ="show()"/>

</apex:form>

</apex:page>

===============

<apex:page >

<apex:outputlabel > Enter Name </apex:outputlabel>

<Script>

function show(){

alert("This is your statement")

}

</Script>

<apex:form >

<apex:inputtext onchange="show()"/>

</apex:form>

</apex:page>

=================

Pen\_\_c mypen = [SELECT id,name FROM Pen\_\_c WHERE name = 'afghan'];

system.debug(mypen.id);

system.debug(mypen.name);

===============

string myname = 'afghan';

Pen\_\_c [] ms = [SELECT id,name FROM Pen\_\_c WHERE name =:myname];

system.debug('this is my id'+ms[0].id);

================

**Pen object mein Afghan naam ka kitna record hai.**

string myname = 'afghan';

Pen\_\_c [] ms = [SELECT id,name FROM Pen\_\_c WHERE name =:myname];

system.debug('total number of records with name afghan'+ms.size());

Pen\_\_c[] ms = [SELECT id,name FROM Pen\_\_c WHERE name = 'afghan'];

system.debug('total number of records with name afghan’+ms.size());

=================

string myname = 'afghan';

Pen\_\_c [ ] ms = [SELECT id,name,price\_\_c FROM Pen\_\_c WHERE name =:myname];

system.debug('total number of records with name afghan'+ms.size());

//system.debug(ms);

for(pen\_\_c c : ms){

system.debug(c.id);

system.debug(c.name);

system.debug(c.price\_\_c);

}

===========================**Exceptional Handling**===========================

try{

merchandise\_\_c m = new merchandise\_\_c();

insert m;

}

catch (exception e) {

system.debug('this is an exception handling '+e.getmessage());

}

system.debug('my name is afghan');

system.debug('you name is khair’);

===========================================================================

try{

merchandise\_\_c m = new merchandise\_\_c(price\_\_c=12, item\_\_c=20);

insert m;

}

catch (exception e) {

system.debug('this is an exception handling '+e.getmessage());

}

system.debug('my name is afghan');

system.debug('you name is khair’);

===========================================================================

try{

merchandise\_\_c m = new merchandise\_\_c(name = ‘salesforce' ,price\_\_c=12, item\_\_c=20);

insert m;

}

catch (exception e) {

system.debug('this is an exception handling '+e.getmessage());

}

system.debug('my name is afghan');

system.debug('you name is khair’);

===========================================================================

try{

merchandise\_\_c m = new merchandise\_\_c(name = 'salesforce',price\_\_c=12, item\_\_c=20);

insert m;

list<integer> mylist = new List <integer>();

mylist.add(6);

integer i1 = mylist[0];

integer i2 = mylist[1];

}

catch (DMLexception e) {

system.debug('this is a DML exception block '+e.getmessage());

}

catch (Exception e) {

system.debug('this is a list exception block '+e.getmessage());

}

system.debug('my name is afghan');

system.debug('you name is khair’);

===========================================================================

try{

merchandise\_\_c m = new merchandise\_\_c(name = 'salesforce',price\_\_c=12, item\_\_c=20);

insert m;

list<integer> mylist = new List <integer>();

mylist.add(6);

integer i1 = mylist[0];

//integer i2 = mylist[1];

string s;

boolean b=s.contains('a');

}

catch (DMLexception e) {

system.debug('this is a DML exception block '+e.getmessage());

}

catch (Exception e) {

system.debug('this is a list exception block '+e.getmessage());

}

system.debug('my name is afghan');

system.debug('you name is khair’);

===========================================================================

try{

merchandise\_\_c m = new merchandise\_\_c(name = 'salesforce',price\_\_c=12, item\_\_c=20);

insert m;

list<integer> mylist = new List <integer>();

mylist.add(6);

integer i1 = mylist[0];

//integer i2 = mylist[1];

string s;

boolean b=s.contains('a');

}

catch (DMLexception e) {

system.debug('this is a DML exception block '+e.getmessage());

}

catch (listException e) {

system.debug('this is a list exception block '+e.getmessage());

}

catch (Exception e) {

system.debug('this is a nullpointer exception block '+e.getmessage());

}

system.debug('my name is afghan');

system.debug('you name is khair');

===========================================================================

try{

merchandise\_\_c lm = [SELECT name FROM merchandise\_\_c WHERE name = 'XYZ' LIMIT 1];

}

catch (exception e ){

system.debug('this is a queryexception block');

}

===========================================================================

try{

merchandise\_\_c m = [SELECT name FROM merchandise\_\_c LIMIT 1];

decimal dec=m.price\_\_c;

}

catch (exception e){

system.debug('this is a query exception block');

}

finally {

system.debug ('i am in final block');

}

system.debug('my name is afghan');

=============================**Visual force Pages**============================

<apex:page standardController="account" >

<b>Hello {!$User.FirstName}!</b>

<apex:detail relatedList=“false"/> //you need to use id for any record in url when you write detail

<apex:relatedList list="Opportunities"/>

</apex:page>

**How to override visual force page in setup menu**

setup >> build >> create >> object >> select needed object >> go to Buttons, Links, and Actions >> click on edit of View >> select visual force page >> save

**How to add image in visual force page**

setup >> build >> Develop >> Static Resource >> New >> Save

**How to create Tab for visual force page**

setup >> build >> create >> Tabs >> New >> select you created visual force page >> next >> next >> Save

===========================================================================

<apex:page >

<apex:image value="{!$Resource.Banner}" width="100%" height="50%"/>

<apex:pageBlock title="My World">

</apex:pageBlock>

</apex:page>

===========================================================================

<apex:page controller="clsController">

<apex:image value="{!$Resource.Banner}" width="100%" height="30%"/>

<apex:form >

<apex:pageBlock title="My Contacts List">

<apex:pageBlockButtons >

<apex:commandButton value="Save" action="{!save}"/>

</apex:pageBlockButtons>

<apex:pageBlockTable value="{!contacts}" var="contact">

<apex:column value="{!contact.id}"/>

<apex:column value="{!contact.name}"/>

<apex:column value="{!contact.phone}"/>

<apex:column value="{!contact.accountid}"/>

<apex:column value="{!contact.email}"/>

</apex:pageBlockTable>

</apex:pageBlock>

</apex:form>

</apex:page>

public with sharing class clsController {

public list<contact> contacts {get;set;}

public clscontroller()

{

contacts = [SELECT id,name, accountid, email, phone FROM contact LIMIT 10 ];

}

public void save()

{

update contacts;

}

}

===========================================================================

<apex:page controller="clsController">

<apex:image value="{!$Resource.Banner}" width="100%" height="30%"/>

<apex:form >

<apex:pageBlock title="My Contacts List">

<apex:pageBlockButtons >

<apex:commandButton value="Save" action="{!save}"/>

</apex:pageBlockButtons>

<apex:pageBlockTable value="{!contacts}" var="contact">

<apex:column value="{!contact.name}"/>

<apex:column headerValue="Phone" >

<apex:inputField value="{!contact.phone}"/>

</apex:column>

<apex:column value="{!contact.accountid}"/>

<apex:column headerValue="email" >

<apex:inputField value="{!contact.email}"/>

</apex:column>

</apex:pageBlockTable>

</apex:pageBlock>

</apex:form>

</apex:page>

public with sharing class clsController {

public list<contact> contacts {get;set;}

public clscontroller()

{

contacts = [SELECT id,name, accountid, email, phone FROM contact LIMIT 10 ];

}

public void save()

{

update contacts;

}

}

===========================================================================