1.

public class myFirstAccountController{

public Account acc{get;set;}

public myFirstAccountController(){

acc =[SELECT id, Name FROM Account LIMIT 1];

}

}

<apex:page controller="myFirstAccountController">

<apex:pageBlock title="Hi {!$User.LastName}">

this is your first custom controller data

<br/>

Account Name: <apex:outputField value="{!acc.name}"/>

</apex:pageBlock>

</apex:page>

2.

public getVarName(){

integer varName=2;

return varName;

}

public void setVarName(integer value) { varName = value; }

public integer var1 {

get { return var1; }

set { var1 = value; }

}

public integer var1{get;set;}

3.

public Class customController{

public Account acc{get;set;}

public customController(){

acc = new Account();

}

public void save(){

system.debug(acc);

}

}

<apex:page controller="customController" sidebar="false">

<apex:form>

<apex:pageBlock>

<apex:pageblocksection title="Account Form">

<apex:inputField value="{!acc.Name}" />

<apex:inputField value="{!acc.Industry}"/>

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

</apex:pageblocksection>

</apex:pageBlock>

</apex:form>

</apex:page>

4.

public Class customController{

public Account acc{get;set;}

public customController(){

acc = new Account();

system.debug(acc);

}

public pageReference save(){

system.debug(acc);

insert acc;

return new PageReference('/'+acc.Id);

}

}

5.

<apex:page controller="customController" sidebar="false">

<apex:form>

<apex:pageBlock>

<apex:pageblocksection title="Account Form">

<apex:inputField value="{!acc.Name}" />

<apex:inputField value="{!acc.Industry}"/>

</apex:pageblocksection>

<apex:pageblocksection title="contact Form">

<apex:inputField value="{!cont.lastName}" />

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

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

</apex:pageblocksection>

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

</apex:pageBlock>

</apex:form>

</apex:page>

public Class customController{

public Account acc{get;set;}

public Contact cont{get;set;}

public customController(){

acc = new Account();

cont= new Contact();

}

public pageReference save(){

insert acc;

cont.AccountId= acc.id;

insert cont;

return new PageReference('/'+acc.Id);

}

}

6.

public Class customControllerRecord{

public Account acc{get;set;}

public customControllerRecord(){

Id recId = ApexPages.currentPage().getParameters().get('Id');

acc = [SELECT Name,Industry from Account WHERE ID=:recId];

system.debug(acc);

}

}

<apex:page controller="customControllerRecord" sidebar="false">

<apex:form>

<apex:pageBlock>

<apex:pageblocksection title="Account Detail">

<apex:outputField value="{!acc.Name}" />

<apex:outputField value="{!acc.Industry}"/>

</apex:pageblocksection>

</apex:pageBlock>

</apex:form>

</apex:page>

7.

public Class customControllerRecord{

public Account acc{get;set;}

public customControllerRecord(){

Id recId = ApexPages.currentPage().getParameters().get('Id');

acc = [SELECT Name,Industry,(SELECT Name,Phone from Contacts) from Account WHERE ID=:recId];

system.debug(acc);

}

}

<apex:page controller="customControllerRecord" sidebar="false">

<apex:form>

<apex:pageBlock>

<apex:pageblocksection title="Account Detail">

<apex:outputField value="{!acc.Name}" />

<apex:outputField value="{!acc.Industry}"/>

</apex:pageblocksection>

<apex:pageblocksection title="Related Contact">

<apex:pageBlockTable value="{!acc.Contacts}" var="con">

<apex:column value="{!con.Name}" />

<apex:column value="{!con.Phone}"/>

</apex:pageBlockTable>

</apex:pageblocksection>

</apex:pageBlock>

</apex:form>

</apex:page>

8.

public class accControllerExtension {

private final Account acc;

// The extension constructor initializes the private member

public accControllerExtension(ApexPages.StandardController sc){

this.acc = (Account)sc.getRecord();

}

}

<apex:page standardController="Account" extensions="accControllerExtension">

Hi, <p/>

<apex:form>

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

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

</apex:form>

</apex:page>

<apex:page standardController="Account" extensions="ext1,ext2">

</apex:page>

public class ext1 {

public Integer randomNo{get;set;}

private final Account ac;

public ext1(ApexPages.StandardController sc){

this.ac = (Account)sc.getRecord();

randomNo = Integer.valueOf( Math.random() \* 1000 );

}

}

public class ext2 {

private final Account acc;

public Integer randomNo2{get;set;}

public ext2(ApexPages.StandardController sc){

this.acc = (Account)sc.getRecord();

randomNo2 = Integer.valueOf( Math.random() \* 1000 );

}

}

<apex:page standardController="Account" extensions="ext1,ext2" >

<apex:pageBlock>

First Random no : {!randomNo} <br/>

Second Random no : {!randomNo2}

<apex:form>

<apex:inputField value="{!Account.Name}" />

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

</apex:form>

</apex:pageBlock>

</apex:page>

10.

<apex:page standardController="Account" renderAs="pdf" recordSetVar="acc">

<apex:pageBlock>

<apex:pageblocksection title="Account Form">

<apex:pageblockTable value="{!acc}" var="Account">

<apex:column value="{!Account.Name}" />

<apex:column value="{!Account.Industry}"/>

</apex:pageblockTable>

</apex:pageblocksection>

</apex:pageBlock>

</apex:page>

11.

public class RegisteredStudentList {

// wrapper list

public list<stdWrapper> stdListWrapper{get;set;}

//constructor

public RegisteredStudentList(){

// query for student ( inner query for registration(course) and Fee

list<student\_\_c> stdList=[select id,name,City\_\_c,Date\_Of\_Birth\_\_c,(select id,course\_\_r.Fees\_\_c,course\_\_r.Name from Registrations\_\_r),(select id,Amount\_\_c from Fees\_\_r) from student\_\_c where Total\_Course\_\_c>0];

// wrapper all value

stdListWrapper= new List<stdWrapper>();

double pd=0, payb=0, rem=0;

// courses string

string courseDet='[ ';

for(student\_\_c std:stdList){

stdWrapper stw= new stdWrapper();

for(Registration\_\_c reg: std.Registrations\_\_r){

//payable amount

payb +=reg.course\_\_r.Fees\_\_c;

// courses string

courseDet+=reg.course\_\_r.Name +'--> Rs.'+reg.course\_\_r.Fees\_\_c+', ';

}

courseDet= courseDet.removeEnd(', ');

courseDet+=' ]';

//paid fees

for(Fees\_\_c fe: std.Fees\_\_r)

pd += fe.Amount\_\_c;

// remaining

rem=payb-pd;

stw.stud=std;

stw.payable=payb;

stw.paid=pd;

stw.remain=rem;

stw.crsDet=courseDet;

// adding this wrapper into list

stdListWrapper.add(stw);

}

}

// wrapper class

public class stdWrapper{

public student\_\_c stud{get;set;}

public string crsDet{get;set;}

public double payable{get;set;}

public double paid{get;set;}

public double remain{get;set;}

}

// code s'f

}

--

public class RegStuList {

// wrapper list

public list<stdWrapper> stdListWrapper{get;set;}

//constructor

public RegStuList(){

// query for student ( inner query for registration(course) and Fee

list<student\_\_c> stdList=[select id,name,City\_\_c,Date\_Of\_Birth\_\_c,(select id,course\_\_r.Fees\_\_c,course\_\_r.Name from Registrations\_\_r) from student\_\_c where Total\_Course\_\_c>0];

set<id> stdSet= new set<id>();

Map<id, double> stdPaidMap= new Map<id, double>();

for(student\_\_c std:stdList)

stdSet.add(std.id);

for(AggregateResult agR: [SELECT student\_\_c, sum(Amount\_\_c) totalfee FROM Fees\_\_c where student\_\_c IN:stdSet GROUP BY student\_\_c])

stdPaidMap.put(string.valueOf(agR.get('student\_\_c')), Double.valueOf(agR.get('totalfee')));

// wrapper all value

stdListWrapper= new List<stdWrapper>();

for(student\_\_c std:stdList){

double pd=0, payb=0, rem=0;

stdWrapper stw= new stdWrapper();

// courses string

string courseDet='[ ';

for(Registration\_\_c reg: std.Registrations\_\_r){

//payable amount

payb +=reg.course\_\_r.Fees\_\_c;

// courses string

courseDet+=reg.course\_\_r.Name+' --> '+reg.course\_\_r.Fees\_\_c+', ';

}

courseDet= courseDet.removeEnd(', ');

courseDet+=' ]';

if(stdPaidMap.containsKey(std.id))

pd=stdPaidMap.get(std.id);

// remaining

rem=payb-pd;

stw.stud=std;

stw.payable=payb;

stw.paid=pd;

stw.remain=rem;

stw.crsdt=courseDet;

// adding this wrapper into list

stdListWrapper.add(stw);

}

}

// wrapper class

public class stdWrapper{

public student\_\_c stud{get;set;}

// public string crsDet{get;set;}

public string crsdt {get;set;}

public double payable{get;set;}

public double paid{get;set;}

public double remain{get;set;}

}

}

<apex:page controller="RegisteredStudentList" sidebar="false">

<apex:pageBlock title="Student Panel">

<apex:pageBlockTable value="{!stdListWrapper}" var="stw">

<apex:column headerValue="Id"><apex:outputField value="{!stw.stud.id}"/>

</apex:column>

<apex:column headerValue="Name"><apex:outputField value="{!stw.stud.Name}"/>

</apex:column>

<apex:column headerValue="City"><apex:outputField value="{!stw.stud.City\_\_c}"/>

</apex:column>

<apex:column headerValue="Date of Birth"><apex:outputField value="{!stw.stud.Date\_Of\_Birth\_\_c}"/>

</apex:column>

<apex:column headerValue="Courses"><apex:outputText value="{!stw.crsDet}"/>

</apex:column>

<apex:column headerValue="Payable amount"><apex:outputText value="{!stw.payable}"/>

</apex:column>

<apex:column headerValue="Paid Amount"><apex:outputText value="{!stw.paid}"/>

</apex:column>

<apex:column headerValue="Remainimg Balance"><apex:outputText value="{!stw.remain}"/>

</apex:column>

</apex:pageBlockTable>

</apex:pageBlock>

</apex:page>

11.

//Visualforce page:

<apex:page Controller="StudentVfComponentController" tabStyle="Account" sidebar="false" id="pg">

<apex:form id="frm">

<apex:pageBlock id="pb">

<apex:actionstatus id="status">

<apex:facet name="start"> <div class="waitingSearchDiv" id="el\_loading" style="background-color: #fbfbfb;height: 100%;opacity:0.6;width:100%;"> </div>

<div class="waitingHolder" style="position:center;top:50px;left:50%; opacity:0.9;z-index:50; width: 20px;">

<img class="waitingImage" src="/img/loading.gif" title="Please Wait..." />

<span class="waitingDescription">Loading...</span> </div> </apex:facet>

</apex:actionstatus>

<apex:actionFunction action="{!selectCourse}" name="fun" rerender="frm" status="status">

<apex:param name="s" assignTo="{!selectStudent}" value="" />

</apex:actionFunction>

<apex:pageBlockSection title="Student Panel" id="pbs">

<apex:selectList multiselect="false" size="1" onchange="fun(this.value);">

<apex:selectOptions value="{!Student\_List}" />

</apex:selectList>

<apex:selectList multiselect="false" size="1" rendered="{!Course\_List.size > 0}" >

<apex:selectOptions value="{!Course\_List}" />

</apex:selectList>

<apex:outputPanel rendered="{!flag}" id="">

<span style="color:red;">Not Registered with any Course</span>

</apex:outputPanel>

</apex:pageBlockSection>

</apex:pageBlock>

</apex:form>

</apex:page>

//Controller:

public class StudentVfComponentController {

public List<SelectOption>Student\_List{get; set;}

public List<SelectOption>Course\_List{get; set;}

public String selectStudent{get; set;}

public boolean flag{get;set;}

//Constructor

public StudentVfComponentController(){

Student\_List = new List<SelectOption>();

Course\_List=new List<SelectOption>();

list<Student\_\_c> Stu\_List = [SELECT id,Name FROM Student\_\_c];

Student\_List.add(new SelectOption('','--Select a Student--'));

for( Student\_\_c stu: Stu\_List)

Student\_List.add(new SelectOption(stu.Id,stu.Name));

Course\_List.add(new SelectOption('','--Select course--'));

flag=false;

}

public void selectCourse(){

List<Registration\_\_c> regList=[select id,Course\_\_r.id, Course\_\_r.Name from Registration\_\_c where Student\_\_c=:selectStudent];

Course\_List.clear();

if(regList.size()>0){

Course\_List.add(new SelectOption('','--Select course--'));

for( Registration\_\_c reg: regList)

Course\_List.add(new SelectOption(reg.Course\_\_r.Id,reg.Course\_\_r.Name));

flag=false;

}

else

flag=true;

}

}