```
<apex:page  title="First Vf Page" tabstyle="Account" sidebar="false"
standardController="Account">
   <apex:pageblock>
   <apex:pageblocksection title="SECTION1">
      <H>This is page Block section </H>

      </apex:pageblocksection>

   <apex:pageblocksection TITLE="sECTION2" COLLAPSIBLE="false">

       <H>This is  second page Block section</H>
      </apex:pageblocksection>

   </apex:pageblock>



</apex:page>
```

…….

```
<apex:page  title="First Vf Page" tabstyle="Account" sidebar="false"
standardController="Account">
   <apex:form>
   <apex:pageblock>
   <apex:pageblocksection title="Insert Account Details">
      <apex:inputfield value="{!Account.Name}"></apex:inputfield>
      <apex:inputfield value="{!Account.Type}"></apex:inputfield>
      <apex:inputfield value="{!Account.Industry}"></apex:inputfield>
      <apex:inputfield value="{!Account.Rating}"></apex:inputfield>

      </apex:pageblocksection>
      <apex:pageblockbuttons>
```

```
        <apex:commandbutton value="Save a Record" action="{!save}"/>
      </apex:pageblockbuttons>

   </apex:pageblock>

   </apex:form>
</apex:page>
```

……

**Insert a record through standard controller**

```
<apex:page  title="First Vf Page" tabstyle="Account" sidebar="false"
standardController="Account">
   <apex:form>
   <apex:pageblock>
      <apex:pagemessage  detail="Account name is a required field" severity="warning"
strength="1" summary="warning">
      </apex:pagemessage>
   <apex:pageblocksection title="Insert Account Details">
      <apex:inputfield value="{!Account.Name}"></apex:inputfield>
      <apex:inputfield value="{!Account.Type}"></apex:inputfield>
      <apex:inputfield value="{!Account.Industry}"></apex:inputfield>
      <apex:inputfield value="{!Account.Rating}"></apex:inputfield>

      </apex:pageblocksection>

        <apex:commandbutton value="Save a Record" action="{!save}"/>

   </apex:pageblock>
```

```
   <apex:pagemessages></apex:pagemessages>
    </apex:form>
</apex:page>
```

……

**Insert a record through custom controller**

```
public class Addacc
{
account ac;
    public  Addacc()
    {
        ac=new account();

    }
    public account getaccount()
    {
        return ac;
    }

    public void savearecord()
    {
        insert ac;
    }
}
```

```
<apex:page  title="First Vf Page" tabstyle="Account" sidebar="false" Controller="Addacc">
    <apex:form>
    <apex:pageblock>
        <apex:pagemessage  detail="Account name is a required field" severity="warning"
strength="1" summary="warning">
```

```
        </apex:pagemessage>
    <apex:pageblocksection title="Insert Account Details">
        <apex:inputfield value="{!Account.Name}"></apex:inputfield>
        <apex:inputfield value="{!Account.Type}"></apex:inputfield>
        <apex:inputfield value="{!Account.Industry}"></apex:inputfield>
        <apex:inputfield value="{!Account.Rating}"></apex:inputfield>

        </apex:pageblocksection>


            <apex:commandbutton value="Save a Record" action="{!savearecord}"/>


    </apex:pageblock>
   <apex:pagemessages></apex:pagemessages>
    </apex:form>
</apex:page>
```

…….

```
<apex:page  title="First Vf Page" tabstyle="Account" sidebar="false" Controller="Addacc">
    <apex:form>
    <apex:pageblock>
        <apex:pagemessage  detail="Account name is a required field" severity="warning"
strength="1" summary="warning">
        </apex:pagemessage>
    <apex:pageblocksection title="Insert Account Details">
        <apex:inputfield value="{!Account.Name}"></apex:inputfield>
        <apex:inputfield value="{!Account.Type}"></apex:inputfield>
        <apex:inputfield value="{!Account.Industry}"></apex:inputfield>
        <apex:inputfield value="{!Account.Rating}"></apex:inputfield>
```

```
        </apex:pageblocksection>



            <apex:commandbutton value="Save a Record" action="{!savearecord}"/>
            <apex:commandbutton value="open list" action="{!openlist}"/>


    </apex:pageblock>
   <apex:pagemessages></apex:pagemessages>
    </apex:form>
</apex:page>
public class Addacc
{
account ac;
    public  Addacc()
    {
        ac=new account();


    }
    public account getaccount()
    {
        return ac;
    }


    public pagereference openlist()
    {
        pagereference ps=new pagereference('https://almamate34-dev-
ed.lightning.force.com/lightning/o/Account/list?filterName=Recent');
        return ps;
    }
```

```
    public pagereference  savearecord()
    {

        insert ac;
        pagereference ps=new pagereference('/'+ac.id);
        return ps;
    }
}
```

……

**Practice Scenarios[Discuss in class]**

```
<apex:page title="firstpage" tabStyle="Account" sidebar="false" standardController="Account"
recordsetvar="accounts1">
    <apex:pageblock title="Account information">

 <apex:pageblocksection title=" Show Account details" columns="1">
    <apex:pageblocktable value="{!accounts1}" var="ac">
        <apex:column  headerValue="Account Name" value="{!ac.name}"/>
        <apex:column  headerValue="Account Type" value="{!ac.type}"/>
        <apex:column  headerValue="Account Industry" value="{!ac.industry}"/>



     </apex:pageblocktable>


        </apex:pageblocksection>


    </apex:pageblock>


</apex:page>
```

……

….

Scenario: Create an output link button for view,editing and fleeting a record.

Hint:Use pageblock table with recordsetvar

```
<apex:page title="firstpage" tabStyle="Account" sidebar="false" standardController="Account" recordsetvar="accounts1">
    <apex:pageblock title="Account information">

 <apex:pageblocksection title=" Show Account details" columns="1">
    <apex:pageblocktable value="{!accounts1}" var="ac">
        <apex:column>
        <apex:outputlink  style="color:blue" value="{!urlfor($action.Account.view,ac.id)}"> View Record </apex:outputlink>
        </apex:column>
        <apex:column  headerValue="Account Name" value="{!ac.name}"/>
        <apex:column  headerValue="Account Type" value="{!ac.type}"/>
        <apex:column  headerValue="Account Industry" value="{!ac.industry}"/>


     </apex:pageblocktable>

        </apex:pageblocksection>

    </apex:pageblock>

</apex:page>
```

……..

**Scenario: How to insert an image**

```
<apex:page title="firstpage" tabStyle="Account" sidebar="false"
standardController="Account" recordsetvar="accounts1">
   <apex:pageblock title="Account information">
      <H1>
         welcome to my page


            <center>
       <apex:image  url="{!$resource.book}" height="100"
width="100"></apex:image>
    </center>


      </H1>


 <apex:pageblocksection title=" Show Account details" columns="1">



   <apex:pageblocktable value="{!accounts1}" var="ac">
      <apex:column>
      <apex:outputlink  style="color:blue"
value="{!urlfor($action.Account.view,ac.id)}"> View Record
</apex:outputlink>
      </apex:column>
      <apex:column  headerValue="Account Name" value="{!ac.name}"/>
      <apex:column  headerValue="Account Type" value="{!ac.type}"/>
      <apex:column  headerValue="Account Industry"
value="{!ac.industry}"/>
```

```
    </apex:pageblocktable>

      </apex:pageblocksection>

   </apex:pageblock>

</apex:page>
```

……

**Example for update an account**

```
public class Addacc
{
account ac;
   public  Addacc()
   {
      id acid=apexpages.currentPage().getparameters().get('Id');
       ac=[select name,type,industry,rating  from Account where id=:acid];



   }
   public account getaccount()
   {
      return ac;
   }
```

```
    public pagereference savearecord()
    {
        update ac;
        pagereference ps=new pagereference('https://almamate34-dev-
ed.lightning.force.com/lightning/o/Account/list?filterName=Recent');
        return ps;
    }



}
<apex:page title="First Vf Page" tabstyle="Account" sidebar="false" Controller="Addacc">
    <apex:form >
    <apex:pageblock >
        <apex:pagemessage detail="Account name is a required field" severity="warning"
strength="1" summary="warning">
        </apex:pagemessage>
    <apex:pageblocksection title="Insert Account Details">
        <apex:inputfield value="{!Account.Name}"></apex:inputfield>
        <apex:inputfield value="{!Account.Type}"></apex:inputfield>
        <apex:inputfield value="{!Account.Industry}"></apex:inputfield>
        <apex:inputfield value="{!Account.Rating}"></apex:inputfield>

        </apex:pageblocksection>


            <apex:commandbutton value="Save a Record" action="{!savearecord}"/>



            <apex:commandbutton value="open list" action="{!openlist}"/>
```

```
    </apex:pageblock>
   <apex:pagemessages ></apex:pagemessages>
    </apex:form>
</apex:page>
```

……

**Example to show how Upsert a data through VFpage**

```
public class Addacc
{
account ac;
    public  Addacc()
    {
       id acid=apexpages.currentPage().getparameters().get('Id');
              if(acid==null)
        {
            ac=new account();
        }

        else
        {
        ac=[select name,type,industry,rating  from Account where id=:acid];
        }

    }
    public account getaccount()
    {
        return ac;
    }

    public pagereference savearecord()
```

```
    {
        upsert ac;
        pagereference ps=new pagereference('https://almamate34-dev-
ed.lightning.force.com/lightning/o/Account/list?filterName=Recent');
        return ps;
    }



}
```

……..

**Scenario: How to populate the list of all account names in a list**

```
public class accshow
{
    public id acid{get;set;}
    list<selectoption> acclist=new list<selectoption>();


    public list<selectoption> getaccname()
    {
        acclist.add(new selectoption('-select','-select'));
        account[] arr=[select name from account];
        for(account ac:arr)
        {
            acclist.add(new selectoption(ac.id,ac.name));
        }
        return acclist;
    }




}
```

```
<apex:page controller="accshow" >
   <apex:form>
   <apex:pageblock Title="Account Names according to id">
      <apex:selectlist size="1" value="{!acid}">
      <apex:selectoptions value="{!accname}"></apex:selectoptions>

      </apex:selectlist>




      </apex:pageblock>

   </apex:form>
</apex:page>
```

**Scenario: How to display account name when user click any account name from list**

```
 public list<account> getdispname()
   {
      list<account> listacc=[select name,type,industry from account where id=:acid];
      return listacc;
   }
```

**Add this code on previous class accshow**

**Vfpage**

```
<apex:page controller="accshow" >
   <apex:form>
   <apex:pageblock Title="Account Names according to id">
```

```
        <apex:selectlist size="1" value="{!acid}">
        <apex:selectoptions value="{!accname}"></apex:selectoptions>
            <apex:actionsupport event="onchange" rerender="pbt1"/>
            </apex:selectlist>
        <apex:pageblocksection>

        </apex:pageblocksection>
       <apex:pageblocktable id="pbt1" value="{!dispname}" var="m">
          <apex:column headervalue="Account Name" value="{!m.name}"/>
          <apex:column headervalue="Account Type" value="{!m.Type}"/>
          <apex:column headervalue="Account Industry" value="{!m.Industry}"/>




        </apex:pageblocktable>



        </apex:pageblock>

    </apex:form>
</apex:page>
```

……..

**Scenario: Create a list of all account names and create a list of all associated contacts and display contact details.**

```
public class accshow
{
    public id acid{get;set;}
    public id conid{get;set;}
```

```
list<selectoption> acclist=new list<selectoption>();
list<selectoption> conlist=new list<selectoption>();

public list<selectoption> getaccname()
{
   acclist.add(new selectoption('-select','-select'));
   account[] arr=[select name from account];
   for(account ac:arr)
   {
      acclist.add(new selectoption(ac.id,ac.name));
   }
   return acclist;
}

public list<selectoption> getcon()
{
   conlist.clear();
   conlist.add(new selectoption('-select','-select'));

    for(contact co:[select name from contact where accountid=:acid and
accountid!=null])
   {
      conlist.add(new selectoption(co.id,co.name));
   }
   return conlist;
}

public list<contact> getdispname()
{
```

```
      list<contact> listacc=[select firstname,lastname,email from contact where
id=:conid];
      return listacc;
   }



}
```

```
<apex:page controller="accshow" >
   <apex:form>
   <apex:pageblock Title="Account Names according to id">
      <apex:selectlist size="1" value="{!acid}">
      <apex:selectoptions value="{!accname}"></apex:selectoptions>
         <apex:actionsupport event="onchange" rerender="clist"/>
         </apex:selectlist>

      <b>Select Contact</b>
      <apex:selectlist size="1" id="clist" value="{!conid}">
      <apex:selectoptions value="{!con}"></apex:selectoptions>
      <apex:actionsupport event="onchange" rerender="pbt1"/>
      </apex:selectlist>
      <apex:pageblocksection>

      </apex:pageblocksection>

     <apex:pageblocktable id="pbt1" value="{!dispname}" var="m">
        <apex:column headervalue="Contact FirstName" value="{!m.Firstname}"/>
        <apex:column headervalue="Contact lastName" value="{!m.LastName}"/>
        <apex:column headervalue="Contact Email" value="{!m.Email}"/>
```

```
        </apex:pageblocktable>


        </apex:pageblock>

    </apex:form>
</apex:page>
```

……

Task: Implement same scenarios in Hospital Project

Task:Implement Information about parent and child both.

```
<apex:page controller="actf1">
    <script type="text/javascript">
    function validate()
    {

        if(document.getElementById("{!$Component.form1.pb1.ip1}").value==")
          {
          alert('Please enter firstname');
          }

    else
        {
            callmethod();
          }

     }
```

```
    </script>
   <apex:form id="form1">
   <apex:pageblock title="contact information" id="pb1">

         Enter contact First Name:<apex:Inputtext value="{!fname}" id="ip1"/><br/>
         Enter contact Last Name:<apex:Inputtext value="{!lname}"/><br/>
          Enter contact email:<apex:Inputtext value="{!email}"/><br/>
            <apex:commandbutton value="Save a record" onclick="validate();return false;"/>
         <apex:actionfunction name="callmethod" action="{!save}"/>




      </apex:pageblock>


   </apex:form>
</apex:page>


public class actf1
{
   public string fname{get;set;}
   public string lname{get;set;}
   public string email{get;set;}
   public pagereference save()
   {
      contact con=new contact();
      con.firstname=fname;
```

```
        con.lastname=fname;
        con.email=email;
        insert con;
        pagereference pr=new pagereference('/'+con.id);
        return pr;
    }

}
```

**Wrapper class in Apex Salesforce**

Wrapper Class in Apex Salesforce : A wrapper or container class is a class, a data structure, or an abstract data type which contains different objects or collection of objects as its members.

A wrapper class is a custom object defined by a programmer wherein he defines the wrapper class properties. Consider a custom object in salesforce, what do you have in it? fields right? different fields of different data types. Similarly wrapper class is a custom class which has different data types or properties as per requirement. We can wrap different objects types or any other types in a wrapper class.

A wrapper class is nothing but **a collection of different Salesforce data types**. In Salesforce, you can combine multiple data types and utilize them for various purposes. For example, there is a wrapper class that can access the account records and displays an in-page block table.

In the Visualforce most important use case is to display a table of records with a check box and then process only the records that are selected.

In the example below, we are displaying list of accounts with a checkbox. End user can select account and then click on Show Selected accounts button. Then selected account will be displayed in table.

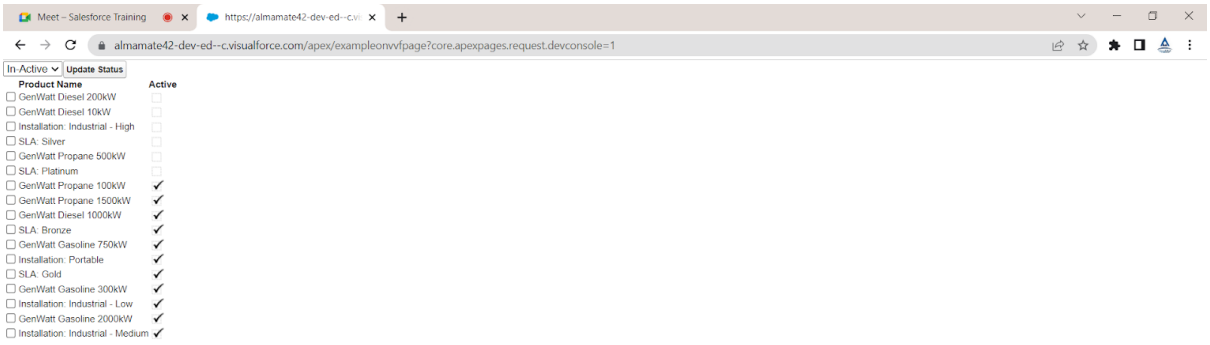
```
public class newclass
{
  public List<cAccount> accList {get; set;}
  public List<Account> selectedAccounts{get; set;}
//Adding the Records to inner class and to get the values for page block table.
  public List<cAccount> getAccounts(){
        if(accList == null){
            accList = new List<cAccount>();
            for(Account acc : [select Id, Name, Phone from Account limit 25]){
                  accList.add(new cAccount(acc));
            }
         }
       return accList;
   }
//on button click it will show the list of records what we have selected.
  public PageReference processSelected(){
        selectedAccounts= new List<Account>();
        for(cAccount cCon : getAccounts()) {
            if(cCon.selected == true){
                selectedAccounts.add(cCon.con);
            }
        }
        return null;
   }
  //  Inner class for capture the records
    public class cAccount {
        public Account con {get; set;}
```

```
        public Boolean selected {get; set;}
        public cAccount(Account c) {
            con = c;
            selected = false;
        }
    }



}
<apex:page controller="newclass">
  <apex:form >
        <apex:pageBlock title="my class">

            <apex:pageBlockButtons >
                <apex:commandButton value="Process Selected"
action="{!processSelected}" rerender="pb1"/>
            </apex:pageBlockButtons>
           <apex:pageBlockSection columns="2">
            <apex:pageBlockTable value="{!Accounts}" var="Acc"
columnsWidth="150px,150px" align="left">
                <apex:column >

                    <apex:inputCheckbox value="{!Acc.selected}"/>
                </apex:column>

                <apex:column value="{!Acc.con.Name}" />
                <apex:column value="{!Acc.con.Phone}" />
            </apex:pageBlockTable>
```

```
            <apex:pageBlockTable value="{!selectedAccounts}" var="Rec" id="pb1"
align="right" title="Selected Accounts">
                <apex:column headerValue="Account Name">
                    <apex:outputField value="{!Rec.name}"/>
                </apex:column>
                <apex:column headerValue="Phone">
                    <apex:outputField value="{!Rec.Phone}"/>
                </apex:column>
            </apex:pageBlockTable>
        </apex:pageBlockSection>
    </apex:pageBlock>
</apex:form>

</apex:page>
```

……

## Create a wrapper class for active and inactive status of Product




```
public class wrappertest
{

    public list<productwrapper> recordlist { get; set;}
    public list<selectoption> status;
    public string selectedstatus { get; set;}

    public wrappertest(){
        list<product2> pro = [select id, name, isactive from product2];
        if(pro.size()!=0){
            recordlist = new list<productwrapper>();
        }
        for(product2 p:pro){
            recordlist.add(new productwrapper(p));
        }

        selectedstatus = 'active';

    }

    public list<selectoption> getStatus(){
        status = new list<selectoption>();
        status.add(new selectoption('active','Active'));
        status.add(new selectoption('inactive','In-Active'));
        return status;
    }
```

```
    public void changestatus(){
        list<product2> prolist = new list<product2>();
        for(productwrapper pw: recordlist){
            if(pw.selected){
                if(selectedstatus == 'active'){
                    pw.record.isactive = true;
                }else if(selectedstatus == 'inactive'){
                    pw.record.isactive = false;
                }
                pw.selected=false;
                prolist.add(pw.record);
            }
        }
        update prolist;
    }

    public class productwrapper{
        public boolean selected { get; set;}
        public product2 record  { get; set;}
        public productwrapper(product2 record){
            this.record = record;
            selected = false;
        }
    }
}


<apex:page controller="wrappertest">
    <apex:form>
```

```
        <apex:selectList value="{!selectedstatus}" size="1">
            <apex:selectOptions value="{!status}"/>
        </apex:selectList>
        <apex:commandButton action="{!changeStatus}" value="Update Status"/>
          <apex:datatable value="{!recordlist}" var="rec">
          <apex:column>
                    <apex:inputCheckbox value="{!rec.selected}"/>
           </apex:column>
           <apex:column value="{!rec.record.name}" headerValue="Product Name"/>
           <apex:column value="{!rec.record.isActive}" headerValue="Active"/>
        </apex:datatable>
    </apex:form>
</apex:page>
```