Title: Creating an Application in SalesForce.com using Apex programming Language

Step No 1:

Create new org:

https://developer.salesforce.com/signup

Step No 2:

After signup, loging using following URl

https://login.salesforce.com/

Step No 3:

Login Page

Graphical user interface, application, website

Description automatically generated

Step No:4

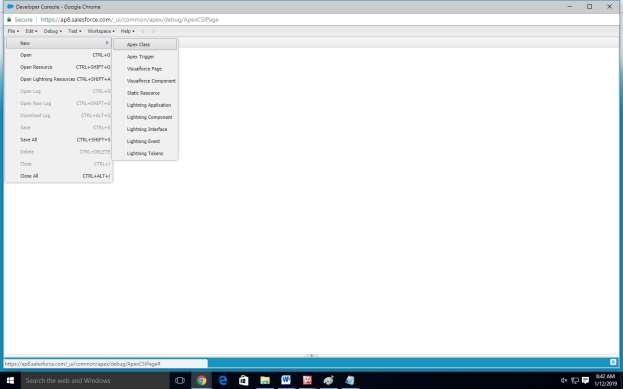
Go to Developer Console for writing Program

Graphical user interface, text, application, email

Description automatically generated

Step No 5:

Click on file then Select Apex class and write a code



Step No 6:

Select tabs section (for creating GUI)

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Step No 7:

Open a pages

Graphical user interface, application, table, Word

Description automatically generated

Step No 8:

Program Windows

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Step No 9:

Program Screenshots

Graphical user interface, text, application, email

Description automatically generated

**Output**

Graphical user interface, text, application

Description automatically generated

**Code –**

CreateAccount.apxc

public class CreateAccount {

public String name {get; set;}

public String phoneNumber {get; set;}

public String selectedname {get; set;}

public String websiteURL {get; set;}

public List<Selectoption> lstnamesel {get; set;}

public CreateAccount(ApexPages.StandardController controller) {

lstnamesel = new List<selectoption>(); lstnamesel.add(new selectOption('','- None -')); lstnamesel.add(new selectOption('IT','IT')); lstnamesel.add(new selectOption('MECH','MECH'));

lstnamesel.add(newselectOption('CHEM','CHEM')); lstnamesel.add(new selectOption('PHARMA','PHARMA'));

}

public PageReference createAccount() {

System.debug('teset create');

if(!String.isEmpty(name)) {

Account accountRecord = new Account(Name = name,

Phone = phoneNumber,

Industry = selectedname,

Website = websiteURL);

INSERT accountRecord;

PageReference pg = new PageReference('/'+accountRecord.Id);

pg.setRedirect(true);

return pg;

}

else {

ApexPages.addmessage(new ApexPages.message(ApexPages.severity.CONFIRM,'Please enter Account Name'));

}

return NULL;

}

public PageReference cancelAccount() {

PageReference pg = new PageReference('/'+Schema.SObjectType.Account.getKeyPrefix()+'/o');

return pg;

}

}

CreateAccount.vfp

<apex:page standardController="Account" extensions="CreateAccount"> <apex:form id="apexFrom" >

<apex:pageBlock title="Create Account:" id="pageBlockId">

<apex:pageMessages id="showmsg"></apex:pageMessages> <apex:pageBlockSection columns="2" >

<!--<div class = "requiredInput">

<div class = "requiredBlock"></div>

<apex:inputText value="{!name}" label="Account Name" required="true"/> </div>-->

<apex:inputText value="{!name}" label="Account Name" required="true"/> <apex:inputText value="{!phoneNumber}" label="Phone" /> <apex:selectList size="1" value="{!selectedname}" label="Industry">

<apex:selectOptions value="{!lstnamesel}"/>

</apex:selectList>

<apex:inputText value="{!websiteURL}" label="Website"/>

</apex:pageBlockSection>

<apex:pageBlockButtons >

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

</apex:pageBlockButtons>

</apex:pageBlock>

<!-- ACTION FUNCTIONS-->

<apex:actionFunction name="createFunction" action="{!createAccount}"/> <apex:actionFunction name="cancelFunction" action="{!cancelAccount}"/>

</apex:form>

</apex:page>