Business Intelligence Solution Providers

Specialized in creating talent resource pool

Getting Started with SalesForce CRM

Apex Code fundamentals Guide Part I

Description:

BISP is committed to provide BEST learning material to the beginners and advance learners. In the same series, we have prepared a complete end-to end Hands-on Beginner's Guide for SalesForce. The document focuses on writing Apex code. Join our professional training program and learn from experts.

History:

Version	Description Change	Author	Publish Date
0.1	Initial Draft	Chandra Prakash Sharma	4th July 2013
0.1	Review#1	Amit Sharma	4th July 2013

Contents

Apex classes in SalesForce

In SalesForce Force.com Apex Code is an object oriented programming language that allows developers to develop on-demand business applications on the Force.com platform.

Define a class, specify the following :

1. Access modifiers :
    - You can use access modifiers public or global in the declaration of a top-level class.
    - Do not use an access modifier in the declaration of an inner class.
2. Optional definition modifiers (such as virtual, abstract, and so on)
3. Required : The keyword class followed by the name of the class
4. Optional extensions:

Following syntax for defining classes:

```
private | public | global
[virtual | abstract | with sharing | without sharing | (none)]
class ClassName [implements InterfaceNameList | (none)] [extends ClassName | (none)]
{
//Body of the class
}
```

How to Add an Apex Class:

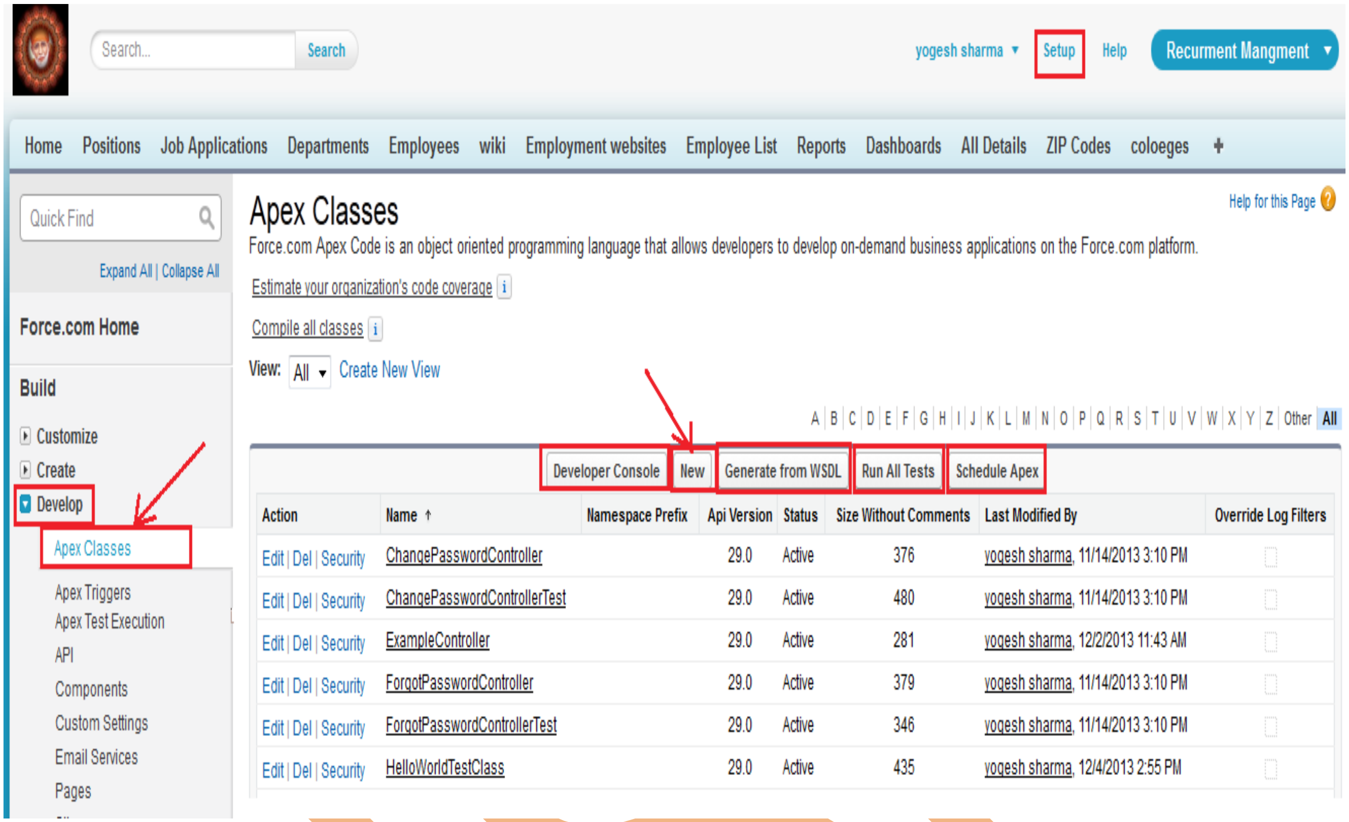
Setup > **Developer** > **Apex classes**, there is many option. you can see below.
Developer Console : In Salesforce.com developer Console is an integrated development environment with a collection of tools you can use to create, debug, and test applications in your SalesForce organization.

New : Click on New button open new page on this page you can write Apex code here.

Generate From WSDL : To access the Force.com Web service, you need a Web Service Description Language (WSDL) file. The WSDL file defines the Web service that is available to you. Your development platform uses this WSDL to generate an API to access the Force.com Web service it defines.

Run All Tests : Click on Run All tests button for run all files.

Schedule Apex : you can schedule Apex code by using **Schedule Apex** button.



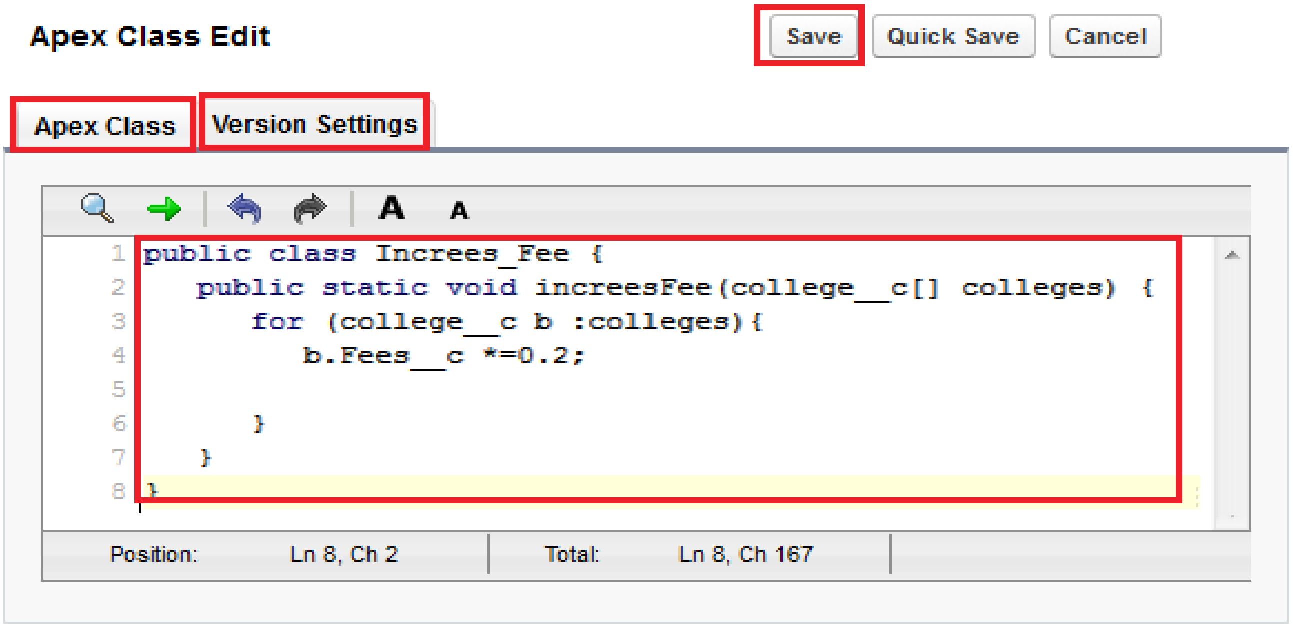
Apex Classes

Force.com Apex Code is an object oriented programming language that allows developers to develop on-demand business applications on the Force.com platform.

Action	Name ↑	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By	Override Log Filters
Edit | Del | Security	ChangePasswordController		29.0	Active	376	yogesh sharma, 11/14/2013 3:10 PM	□
Edit | Del | Security	ChangePasswordControllerTest		29.0	Active	480	yogesh sharma, 11/14/2013 3:10 PM	□
Edit | Del | Security	ExampleController		29.0	Active	281	yogesh sharma, 12/2/2013 11:43 AM	□
Edit | Del | Security	ForgotPasswordController		29.0	Active	379	yogesh sharma, 11/14/2013 3:10 PM	□
Edit | Del | Security	ForgotPasswordControllerTest		29.0	Active	346	yogesh sharma, 11/14/2013 3:10 PM	□
Edit | Del | Security	HelloWorldTestClass		29.0	Active	435	yogesh sharma, 12/4/2013 2:55 PM	□

For create new Apex code click on New button .

Click on **New** button, After that open new web page on this page Apex Code editor is available. you can see below. Write here Apex Class after then click on **Save** button.

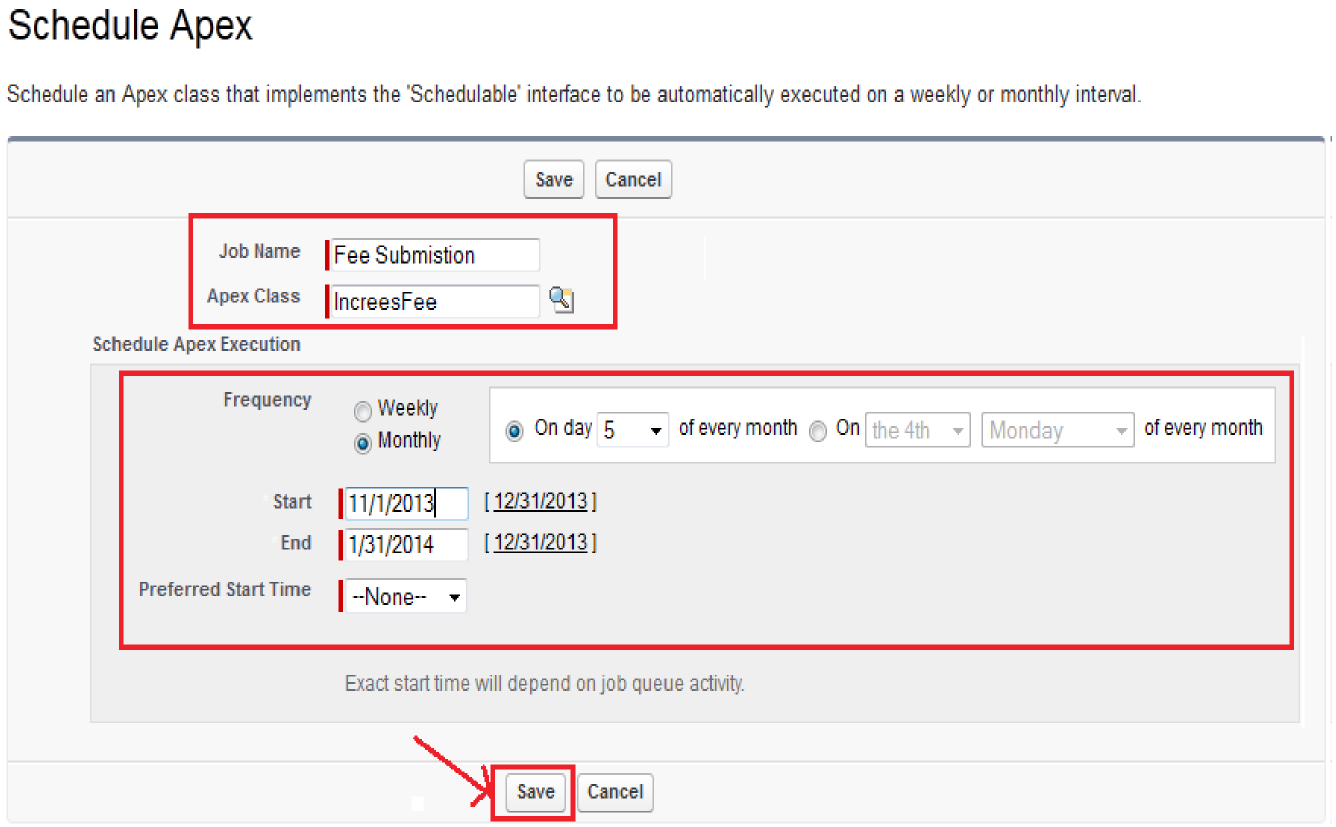


```
public class Increes_Fee {
   public static void increesFee(college__c[] colleges) {
      for (college__c b :colleges){
         b.Fees__c *=0.2;

      }
   }
}
```

Position: Ln 8, Ch 2 | Total: Ln 8, Ch 167

Schedule Apex : Schedule an Apex class that implements the 'Schedulable' interface to be automatically executed on a weekly or monthly interval.
Click on **Schedule Apex** button.



Schedule Apex

Schedule an Apex class that implements the 'Schedulable' interface to be automatically executed on a weekly or monthly interval.

Job Name: Fee Submistion
Apex Class: IncreesFee

Schedule Apex Execution

Frequency: ○ Weekly ● Monthly
● On day 5 of every month ○ On the 4th Monday of every month

Start: 11/1/2013 [12/31/2013]
End: 1/31/2014 [12/31/2013]
Preferred Start Time: --None--

Exact start time will depend on job queue activity.

Apex Triggers in SalesForce

Triggers:

Apex can be invoked through the use of triggers. A trigger is executes before or after the following types of operations:

- insert
- update
- delete
- upsert
- merge
- undelete

Triggers can be divided into two types :

Before triggers can be used to update or validate record values before they are saved to the database.

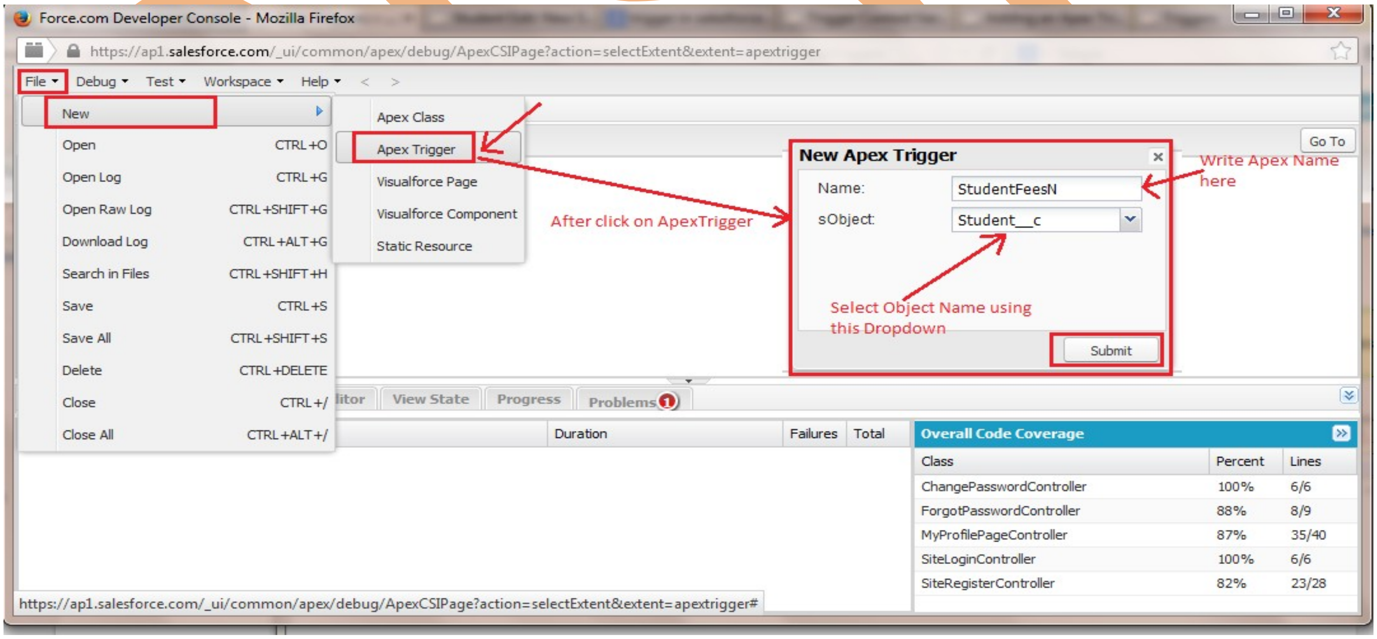
Triggers can be used to access field values that are set by the database, For Example record's Id or lastUpdated field, and to affect changes in other records, such as logging into an audit table or firing asynchronous events with a queue.

Trigger Context Variables :

Variable	Usage
isInsert	Returns true if this trigger was fired due to an insert operation, from the SalesForces user interface, Apex, or the API.
isUpdate	Returns true if this trigger was fired due to an update operation, from the SalesForce user interface, Apex, or the API.
isExecuting	Returns true if the current context for the Apex code is a trigger, not a VisualForce page, a Web service, or an executeanonymous() API call.
isDelete	Returns true if this trigger was fired due to an delete operation, from the SalesForce user interface, Apex, or the API.
isBefore	Returns true if this trigger was fired before any record was saved.
isAfter	Returns true if this trigger was fired after all records were saved.
isUndelete	Returns true if this trigger was fired after a record is recovered from the Recycle Bin (that is, after an undelete operation from the SalesForce user interface, Apex, or the API.)
new	Returns a list of the new versions of the sObject records.
newMap	A map of IDs to the new versions of the sObject records.
old	Returns a list of the old versions of the sObject records.
oldMap	A map of IDs to the old versions of the sObject records.
size	The total number of records in a trigger invocation, both old and new.

How to write Trigger in SalesForce.com :

Setup > **Develop** > **Apex Triggers**, Click on **Developer Console** button. open new web page. You can see below. For create Apex Trigger click on **File** > **New** > **Apex Trigger**.


Force.com Developer Console - Mozilla Firefox
https://ap1.salesforce.com/_ui/common/apex/debug/ApexCSIPage?action=selectExtent&extent=apextrigger
File
Debug
Test
Workspace
Help
New
Open CTRL+O
Open Log CTRL+G
Open Raw Log CTRL+SHIFT+G
Download Log CTRL+ALT+G
Search in Files CTRL+SHIFT+H
Save CTRL+S
Save All CTRL+SHIFT+S
Delete CTRL+DELETE
Close CTRL+/
Close All CTRL+ALT+/
Apex Class
Apex Trigger
Visualforce Page
Visualforce Component
Static Resource
Go To
View State
Progress
Problems 1
Duration
Failures
Total
https://ap1.salesforce.com/_ui/common/apex/debug/ApexCSIPage?action=selectExtent&extent=apextrigger#

New Apex Trigger

Name: StudentFeesN

sObject: Student__c

Submit

After click on ApexTrigger

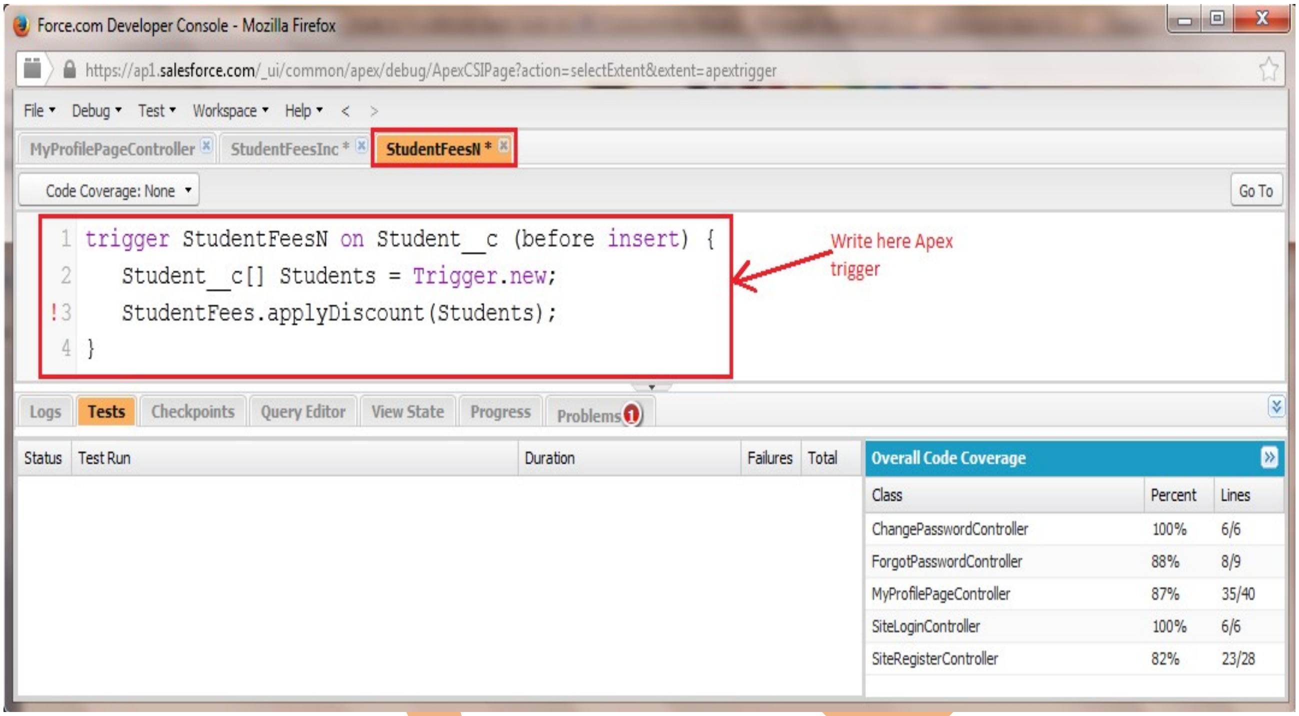
Write Apex Name here

Select Object Name using this Dropdown

Overall Code Coverage

Class	Percent	Lines
ChangePasswordController	100%	6/6
ForgotPasswordController	88%	8/9
MyProfilePageController	87%	35/40
SiteLoginController	100%	6/6
SiteRegisterController	82%	23/28

After that open popup window in this window, Write Trigger name and select Object class (which you need to bind with trigger) then click on submit button. Then you can see new page for write new trigger.



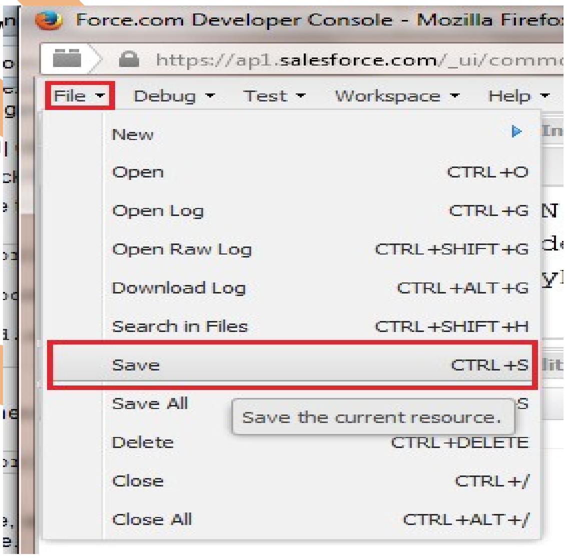
```
trigger StudentFeesN on Student__c (before insert) {
    Student__c[] Students = Trigger.new;
    StudentFees.applyDiscount(Students);
}
```

Write here Apex trigger

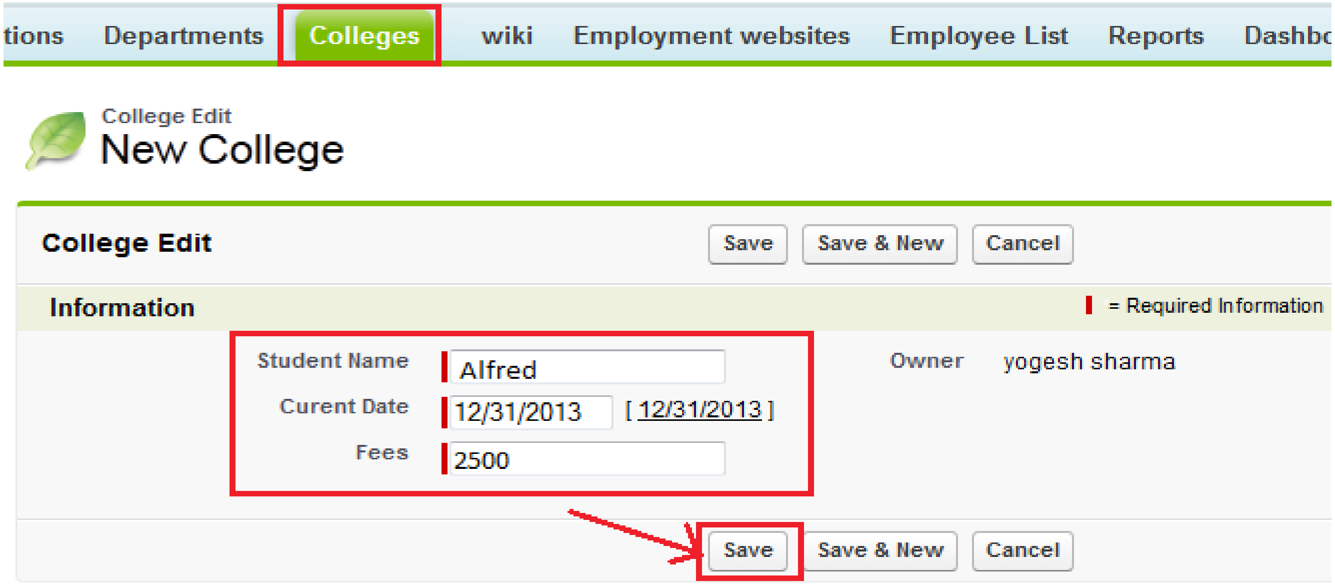
Class	Percent	Lines
ChangePasswordController	100%	6/6
ForgotPasswordController	88%	8/9
MyProfilePageController	87%	35/40
SiteLoginController	100%	6/6
SiteRegisterController	82%	23/28

After Write trigger , go to **File** menu and click on **Save** see below also you use **CTRL + S** by using keyboard.

Note : For understanding Apex Classes or Apex Trigger you can here.
Example , see below .


Force.com Developer Console - Mozilla Firefox
https://ap1.salesforce.com/_ui/comm
File
Debug
Test
Workspace
Help
New
Open CTRL+O
Open Log CTRL+G
Open Raw Log CTRL+SHIFT+G
Download Log CTRL+ALT+G
Search in Files CTRL+SHIFT+H
Save CTRL+S
Save All
Save the current resource.
Delete CTRL+DELETE
Close CTRL+/
Close All CTRL+ALT+/

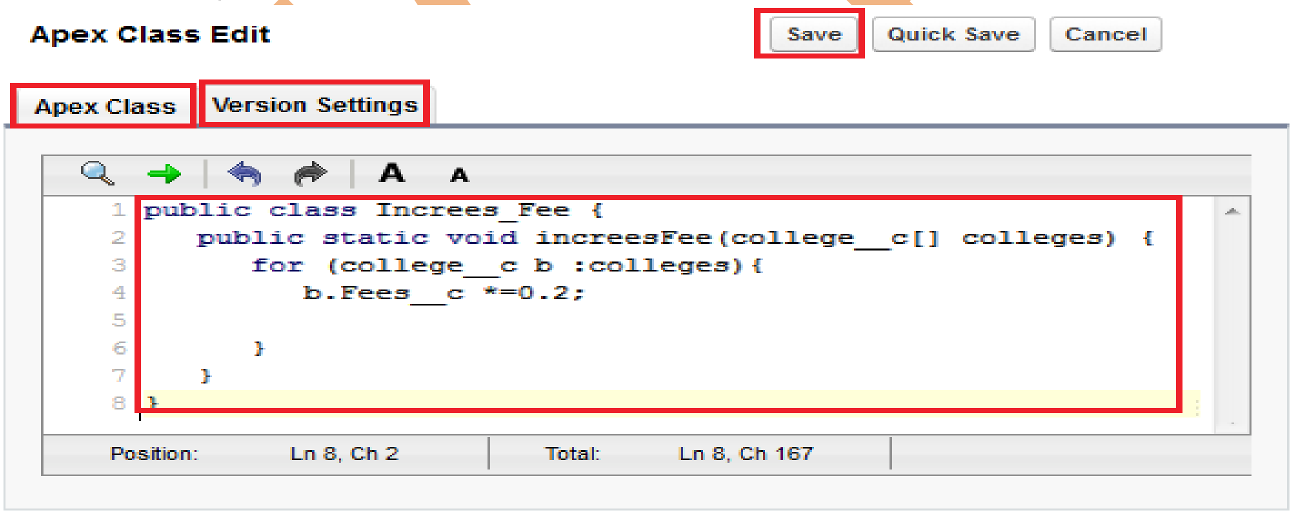
Example : Write apex code for automatically increasing 2% on the submit fees for every student.
Step 1 : Create any custom object tab. For Ex : College you can see below.



Student Name	Alfred	Owner	yogesh sharma
Curent Date	12/31/2013 [12/31/2013]		
Fees	2500		

Step 2 : After then go to adding Apex code, **Setup** > **Developer** > **Apex classes,** click on **New** button.

then write here Apex code.



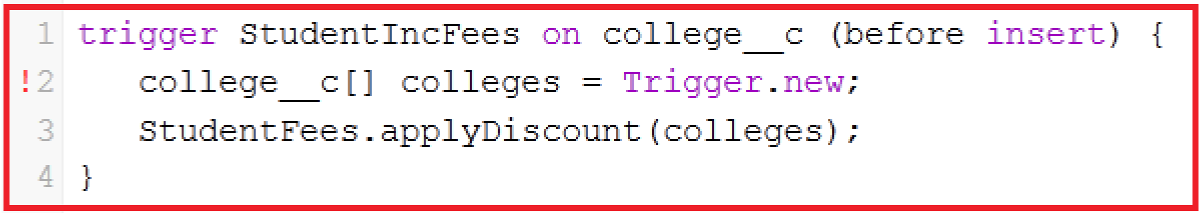
```
public class Increes_Fee {
   public static void increesFee(college__c[] colleges) {
      for (college__c b :colleges){
         b.Fees__c *=0.2;

      }
   }
}
```

Position: Ln 8, Ch 2   Total: Ln 8, Ch 167

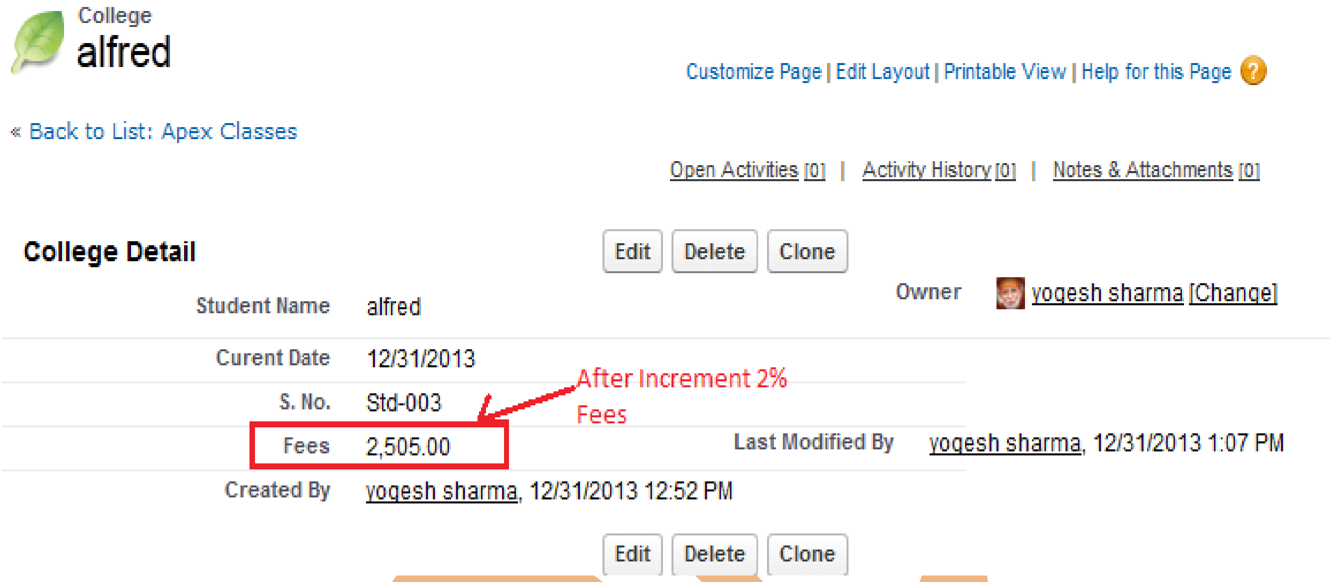
Step 3 :

After write Apex classes , Write Apex Trigger Click on **Apex Trigger** for write Apex Trigger.



```
trigger StudentIncFees on college__c (before insert) {
    college__c[] colleges = Trigger.new;
    StudentFees.applyDiscount(colleges);
}
```

Step 4 : After that Save fees, then see fees increase 2% .



College
alfred

Customize Page | Edit Layout | Printable View | Help for this Page

« Back to List: Apex Classes

Open Activities [0] | Activity History [0] | Notes & Attachments [0]

College Detail

Edit Delete Clone

Student Name	alfred	Owner	yogesh sharma [Change]
Curent Date	12/31/2013		
S. No.	Std-003		
Fees	2,505.00	Last Modified By	yogesh sharma, 12/31/2013 1:07 PM
Created By	yogesh sharma, 12/31/2013 12:52 PM		

After Increment 2% Fees

Edit Delete Clone

How Do VisualForce Pages Compare to S-Controls?

S-Controls, is older Force.com technology that has now been superseded by VisualForce, still function on Force.com. You can continue to use existing S-Controls, but restrictions are being placed on creating new ones.

Creating a VisualForce Component

Salesforce.com provides a library of standard, pre-built components, such as <apex:page renderAs="pdf">, <apex:tabPanel > and <apex:dataTable>, that can be used to develop VisualForce pages. In addition, you can build your own custom components to augment this library.

What are Custom Components :

In custom components you can reuse that method several times in a program, you can encapsulate a common design pattern in a custom component and then reuse that component several times in one or more VisualForce pages.

Example, If you want to create a Employee list using VisualForce pages. Each Employee record in the list has its own border. Rather than repeating the VisualForce markup required for displaying every Employee record in the list. Once defined, standard components such as <apex:relatedList> or <apex:dataTable> .

Using Custom Components in a VisualForce Page:

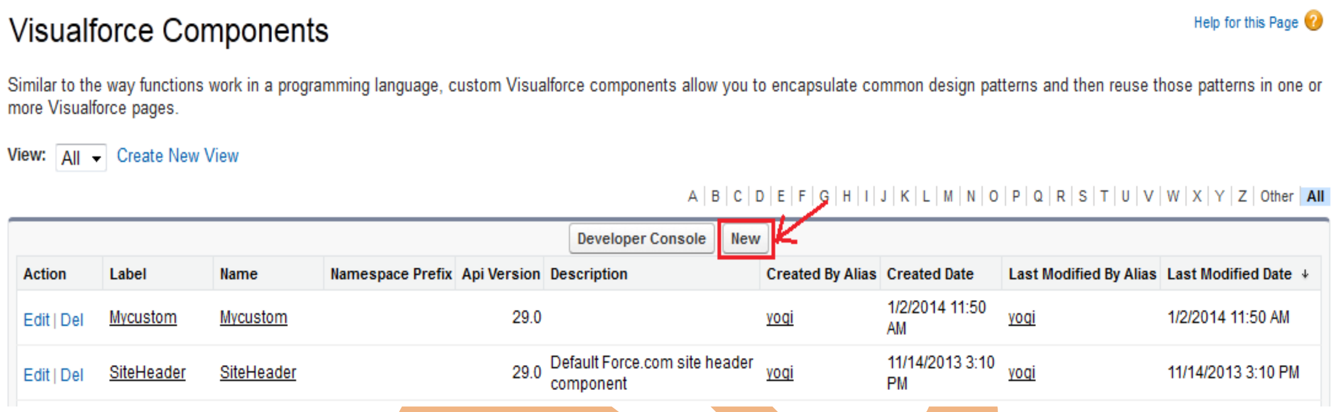
Create a new custom component named recordDisplay.

```
<apex:page >
 <apex:pageBlock title="Employee List" >
 <b> Welcome :</b> {!$User.FirstName}
  <apex:detail />
    </apex:pageBlock>
```

</apex:page>

How to Defining Custom Components :

Setup > **Develop** > **Components**, Click on **New** button.



Help for this Page
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All
Developer Console
New

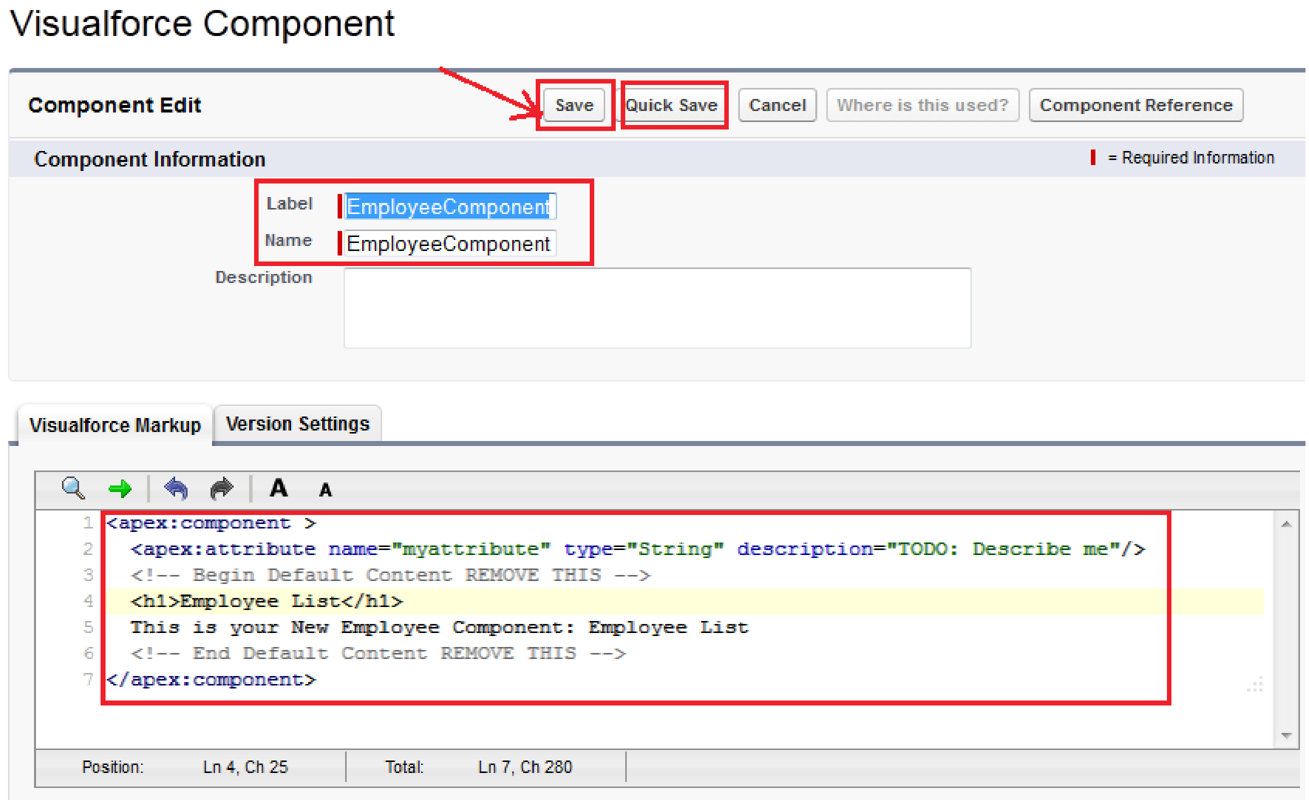
Visualforce Components

Similar to the way functions work in a programming language, custom Visualforce components allow you to encapsulate common design patterns and then reuse those patterns in one or more Visualforce pages.

View: All Create New View

Action	Label	Name	Namespace Prefix	Api Version	Description	Created By Alias	Created Date	Last Modified By Alias	Last Modified Date ↓
Edit | Del	Mycustom	Mycustom		29.0		yogi	1/2/2014 11:50 AM	yogi	1/2/2014 11:50 AM
Edit | Del	SiteHeader	SiteHeader		29.0	Default Force.com site header component	yogi	11/14/2013 3:10 PM	yogi	11/14/2013 3:10 PM

Step 1 : Enter Label Name then write VisualForce Component after then click on **Save** button.



Label: EmployeeComponent

Name: EmployeeComponent

Description:

```
<apex:component >
  <apex:attribute name="myattribute" type="String" description="TODO: Describe me"/>
  <!-- Begin Default Content REMOVE THIS -->
  <h1>Employee List</h1>
  This is your New Employee Component: Employee List
  <!-- End Default Content REMOVE THIS -->
</apex:component>
```

Position: Ln 4, Ch 25 Total: Ln 7, Ch 280

Step 2 : After that create new VisualForce page and write here some cod you can see below.

What is VisualForce

VisualForce is framework that allows developers to build refined, custom user interfaces that can be hosted natively on the Force.com platform. The VisualForce framework includes a tag-based markup language, similar to HTML.

In VisualForce page you can use html code, JavaScript, CSS, Ajax, Flesh.

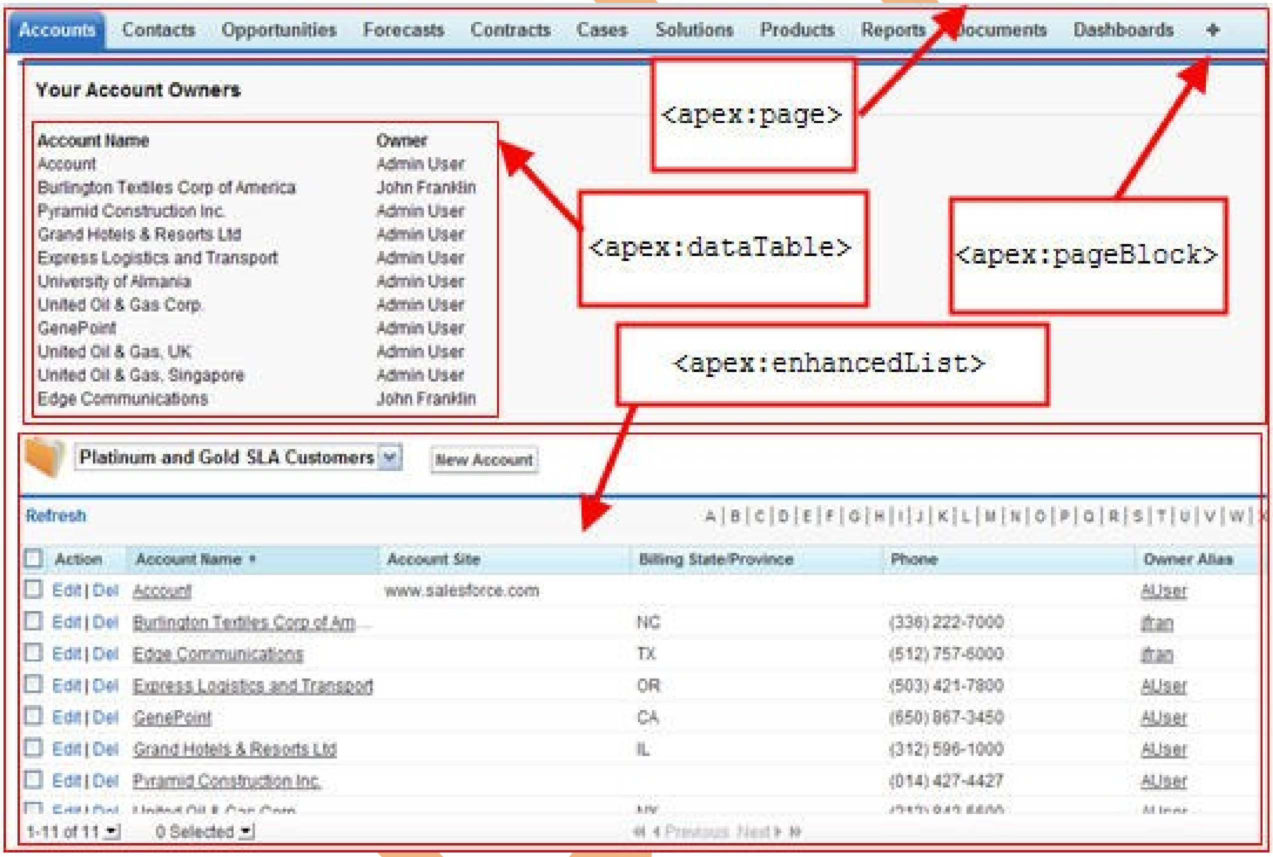
Developers can use VisualForce to create a VisualForce page definition. A page definition consists of two primary elements:
- VisualForce markup.
- A VisualForce controller.

VisualForce Markup :-

VisualForce markup consists of VisualForce tags, HTML, JavaScript, or any other Web-enabled code embedded within a single `<apex: page>` tag. The markup defines the user interface components that should be included on the page, and the way they should appear.

VisualForce Control :-

A VisualForce controller is a set of instructions that specify what happens when a user interacts with the components specified in associated VisualForce markup, such as when a user clicks a button or link.



Your Account Owners

Account Name	Owner
Account	Admin User
Burlington Textiles Corp of America	John Franklin
Pyramid Construction Inc.	Admin User
Grand Hotels & Resorts Ltd	Admin User
Express Logistics and Transport	Admin User
University of Almania	Admin User
United Oil & Gas Corp.	Admin User
GenePoint	Admin User
United Oil & Gas, UK	Admin User
United Oil & Gas, Singapore	Admin User
Edge Communications	John Franklin

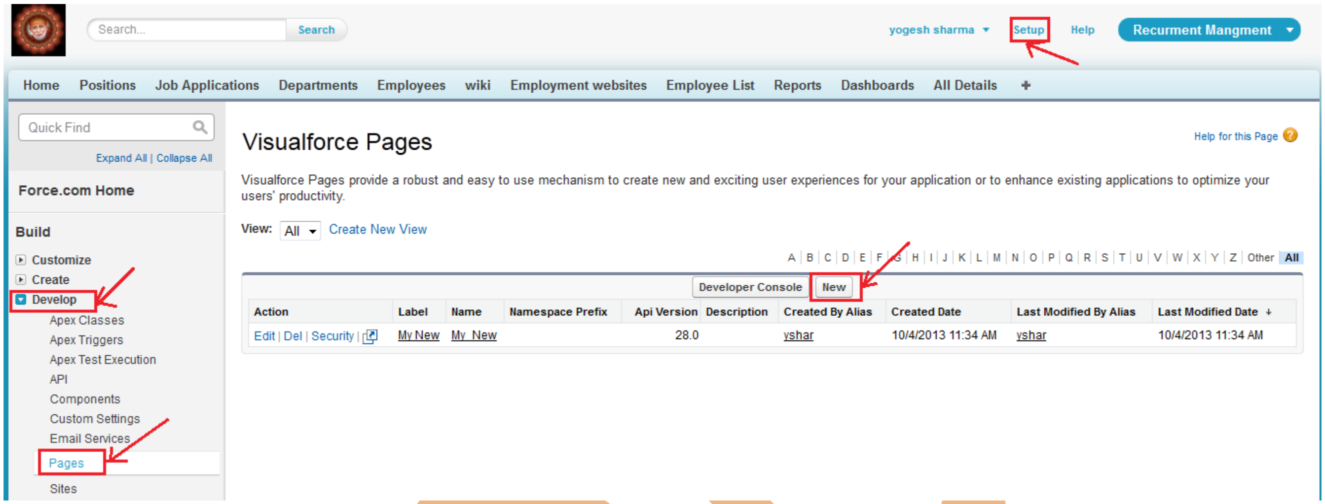
□ Action	Account Name ▾	Account Site	Billing State/Province	Phone	Owner Alias
□ Edit | Del	Account	www.salesforce.com			AUser
□ Edit | Del	Burlington Textiles Corp of Am...		NC	(336) 222-7000	jfran
□ Edit | Del	Edge Communications		TX	(512) 757-6000	jfran
□ Edit | Del	Express Logistics and Transport		OR	(503) 421-7800	AUser
□ Edit | Del	GenePoint		CA	(650) 867-3450	AUser
□ Edit | Del	Grand Hotels & Resorts Ltd		IL	(312) 596-1000	AUser
□ Edit | Del	Pyramid Construction Inc.			(014) 427-4427	AUser
□ Edit | Del	United Oil & Gas Corp.		NY	(212) 842-5500	AUser

Why use VisualForce :

- To create custom interfaces.
- To build wizards.
- To define custom navigation patterns.

How To Create VisualForce Page

Click on **Setup** > **develop** > **Pages**
Then Click on New button.

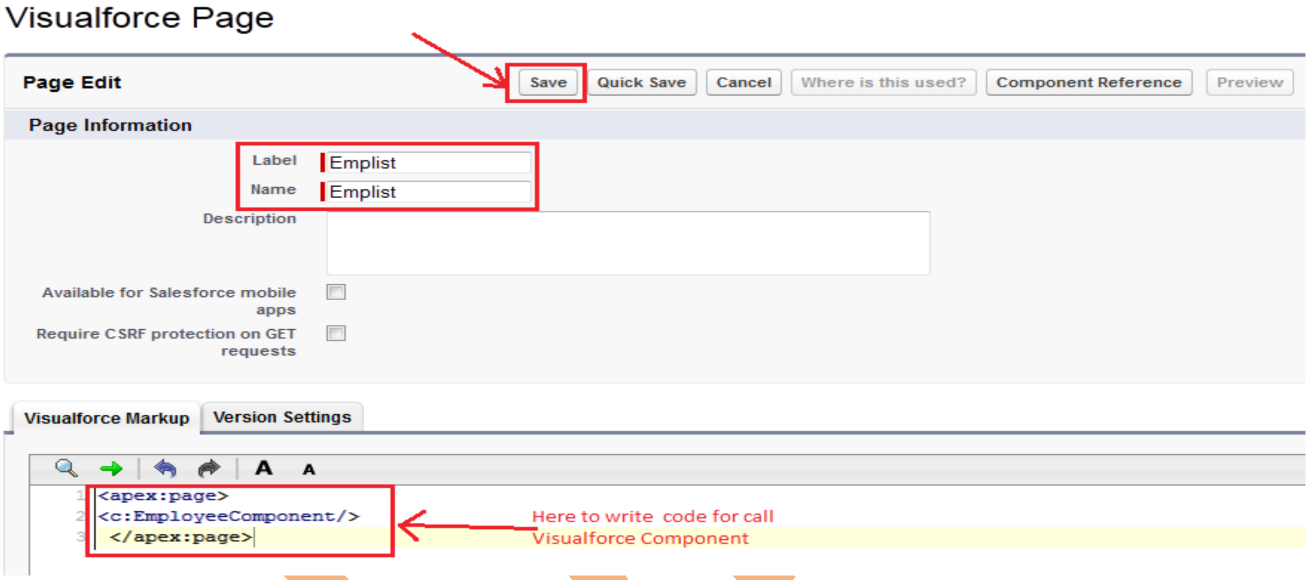


Visualforce Pages

Visualforce Pages provide a robust and easy to use mechanism to create new and exciting user experiences for your application or to enhance existing applications to optimize your users' productivity.

Action	Label	Name	Namespace Prefix	Api Version	Description	Created By Alias	Created Date	Last Modified By Alias	Last Modified Date ↓
Edit | Del | Security |	My New	My_New		28.0		yshar	10/4/2013 11:34 AM	yshar	10/4/2013 11:34 AM

Step 1 :



Visualforce Page

Page Edit

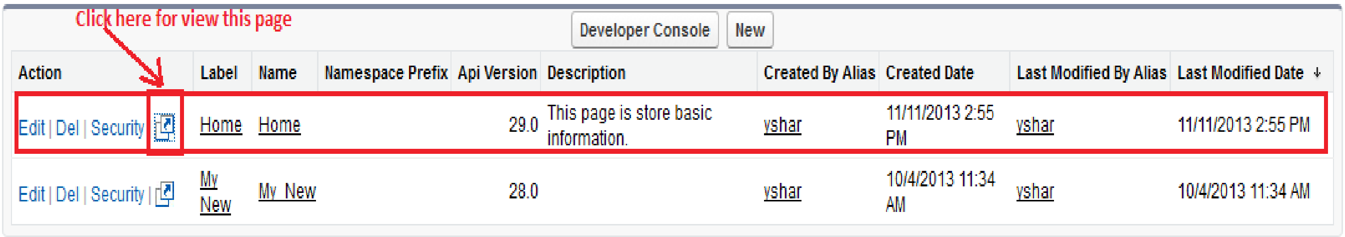
Page Information

Label: Emplist
Name: Emplist
Description:
Available for Salesforce mobile apps ☐
Require CSRF protection on GET requests ☐

```
<apex:page>
<c:EmployeeComponent/>
 </apex:page>
```

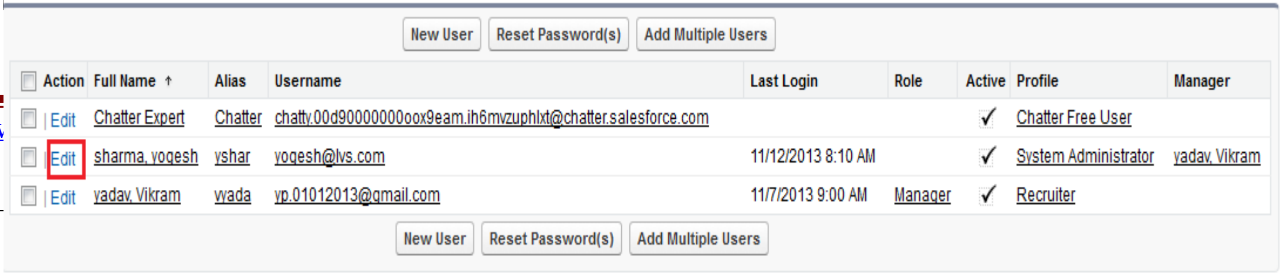
Here to write code for call Visualforce Component

After Saving this page, click on this icon to view Home page.



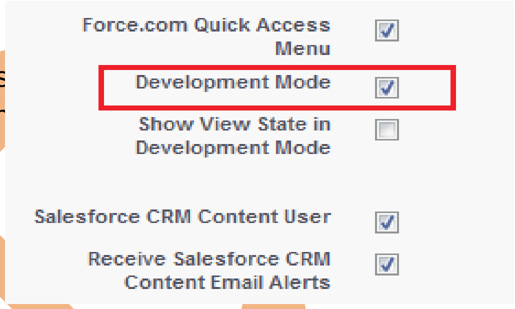
Action	Label	Name	Namespace Prefix	Api Version	Description	Created By Alias	Created Date	Last Modified By Alias	Last Modified Date ↓
Edit | Del | Security |	Home	Home		29.0	This page is store basic information.	yshar	11/11/2013 2:55 PM	yshar	11/11/2013 2:55 PM
Edit | Del | Security |	My New	My_New		28.0		yshar	10/4/2013 11:34 AM	yshar	10/4/2013 11:34 AM

If you are getting error for view page, solve this problem by following some steps.Given below.
Click on **Setup** > **Manage Users** > **Users** then click on Edit link button for edit user settings.

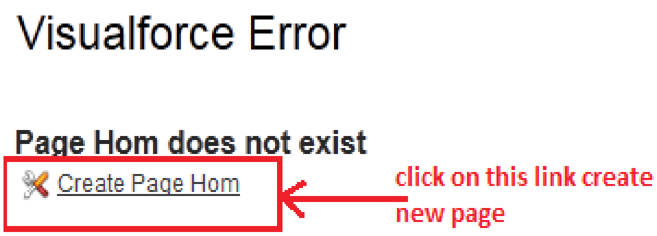


☐	Action	Full Name ↑	Alias	Username	Last Login	Role	Active	Profile	Manager
☐	| Edit	Chatter Expert	Chatter	chatty.00d90000000oox9eam.ih6mvzuphlxt@chatter.salesforce.com			✓	Chatter Free User	
☐	| Edit	sharma, yogesh	yshar	yogesh@lvs.com	11/12/2013 8:10 AM		✓	System Administrator	yadav, Vikram
☐	| Edit	yadav, Vikram	vyada	yp.01012013@gmail.com	11/7/2013 9:00 AM	Manager	✓	Recruiter	

Then you can find Development Mode and checked check box then click on save button.
After then you can click on view icon that time, there is VisualForce to ask you to create new page because th page is open in development mode.



Force.com Quick Access Menu	☑
Development Mode	☑
Show View State in Development Mode	□
Salesforce CRM Content User	☑
Receive Salesforce CRM Content Email Alerts	☑



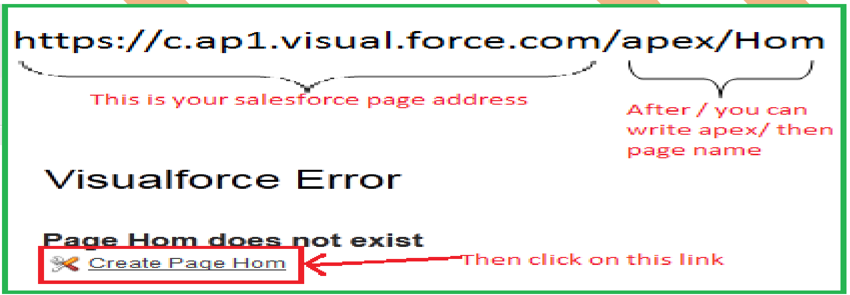
Visualforce Error

Page Hom does not exist

Create Page Hom

click on this link create new page

Second type of create VisualForce page :
go to web browser type UR L, you can see below. Ex : Http:// SalesForce link after login/apex/Page Name



https://c.ap1.visual.force.com/apex/Hom

This is your salesforce page address

After / you can write apex/ then page name

Visualforce Error

Page Hom does not exist

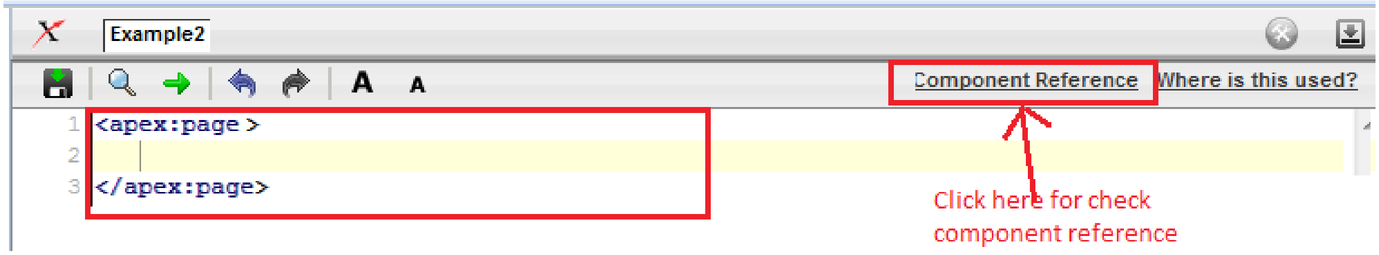
Create Page Hom

Then click on this link

Using the VisualForce Component Library

VisualForce <apex:page></apex:page> tag is use mandatory, tag must be placed at start and end of all VisualForce markup.
and also VisualForce supports all html tag. For example : <img src=" ">, <table><tr><td></td></table>, <marquee></marquee> etc.

How to use VisualForce Component Library :
Just go to create new page. For example I will create Example2 . You can see below.

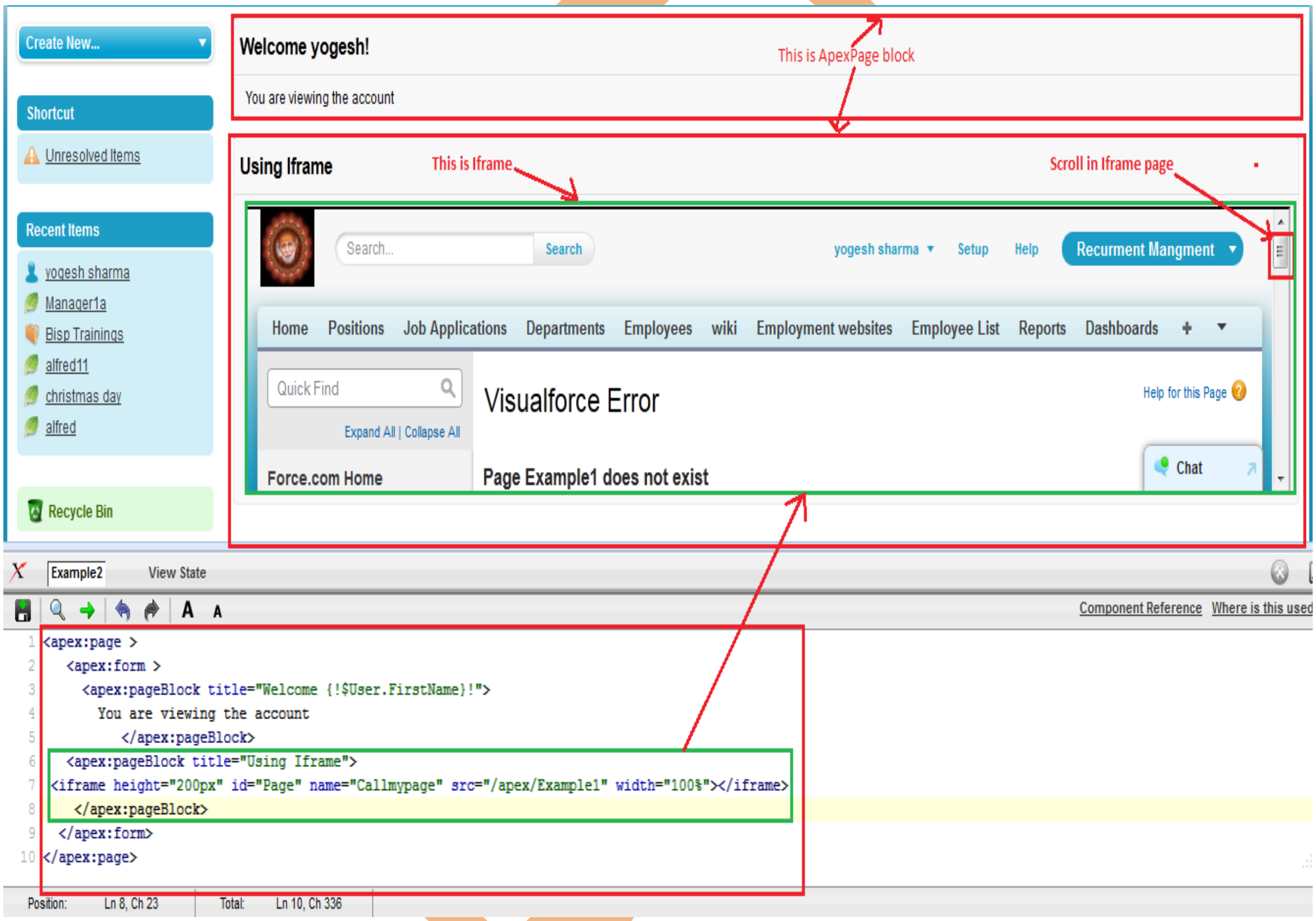


```
<apex:page >

</apex:page>
```

When you click on Component Reference link you can see new web page. On this web page all Component with syntax are available, which Component you want to add Just write in code window. see below.

And also see how to use **Iframe in VisualForce page** .



Welcome yogesh!

You are viewing the account

Using Iframe

Visualforce Error

Page Example1 does not exist

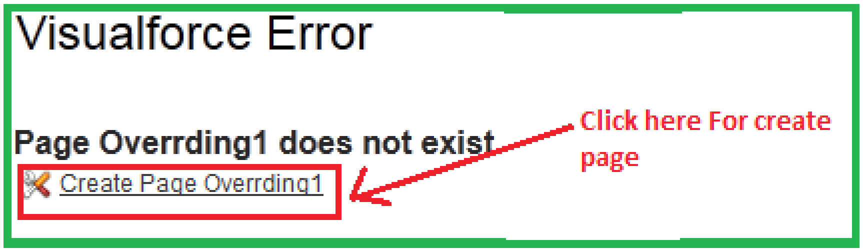
```
<apex:page >
  <apex:form >
    <apex:pageBlock title="Welcome {!$User.FirstName}!">
      You are viewing the account
         </apex:pageBlock>
  <apex:pageBlock title="Using Iframe">
<iframe height="200px" id="Page" name="Callmypage" src="/apex/Example1" width="100%"></iframe>
   </apex:pageBlock>
 </apex:form>
</apex:page>
```

Position: Ln 8, Ch 23 Total: Ln 10, Ch 336

Overriding an Existing Page with a VisualForce Page

If you need to change the format of an existing page, such as the standard account detail page. All the information displays on a single page. If there's a lot of information, you might end up doing a lot of scrolling. Using VisualForce in a page you can make each section(Tab) for an account display in a tab.

1. Firstly, Create a new VisualForce page using the quick fix.
/apex/PageName
For Example We create this page https://c.ap1.visual.force.com/apex/Overrding



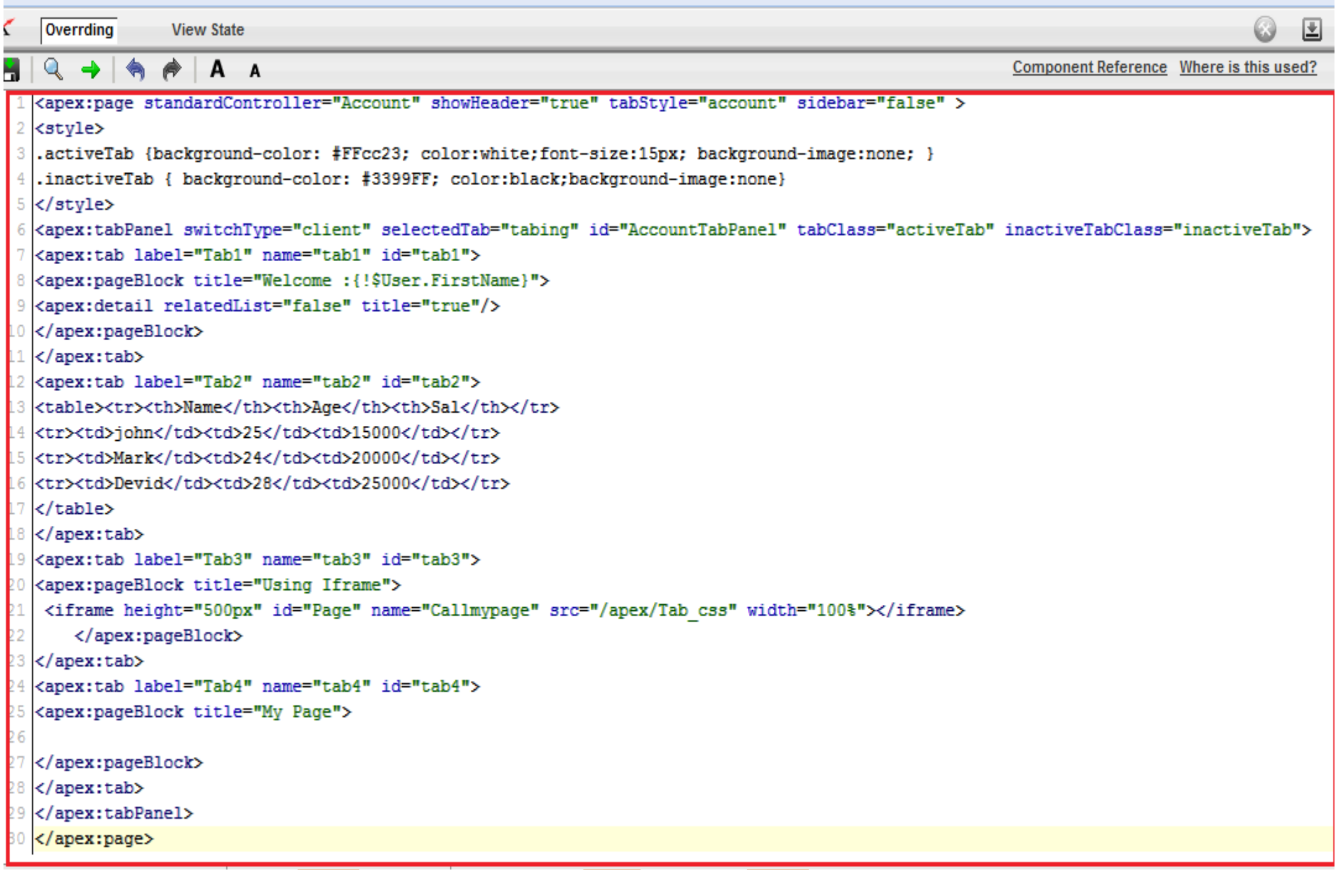
Visualforce Error

Page Overrding1 does not exist

Create Page Overrding1

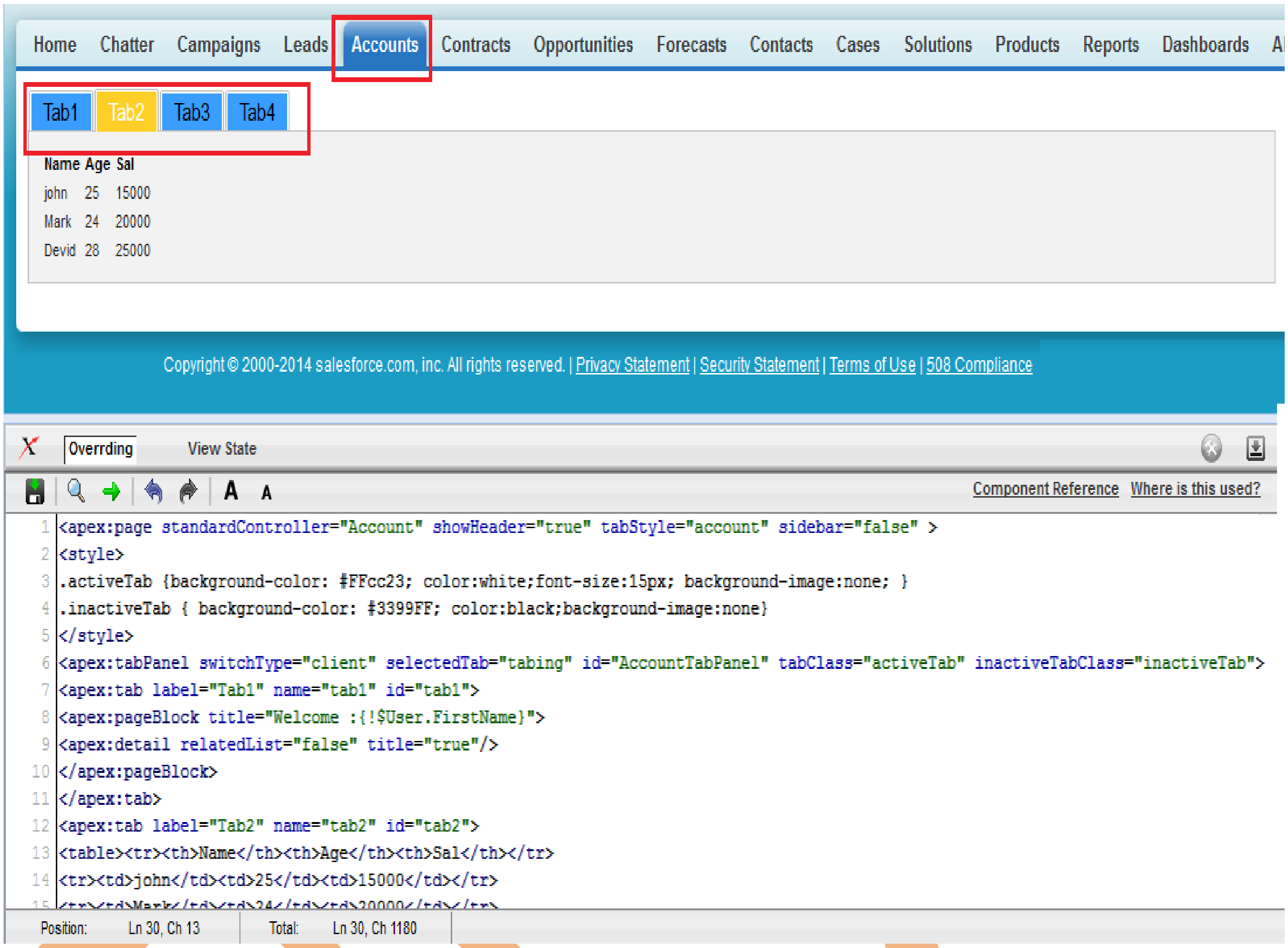
Click here For create page

2. Create Overriding page.
3. Delete Existing code. and paste it below code.
4. See below code.



```
<apex:page standardController="Account" showHeader="true" tabStyle="account" sidebar="false" >
<style>
.activeTab {background-color: #FFcc23; color:white;font-size:15px; background-image:none; }
.inactiveTab { background-color: #3399FF; color:black;background-image:none}
</style>
<apex:tabPanel switchType="client" selectedTab="tabing" id="AccountTabPanel" tabClass="activeTab" inactiveTabClass="inactiveTab">
<apex:tab label="Tab1" name="tab1" id="tab1">
<apex:pageBlock title="Welcome :{!$User.FirstName}">
<apex:detail relatedList="false" title="true"/>
</apex:pageBlock>
</apex:tab>
<apex:tab label="Tab2" name="tab2" id="tab2">
<table><tr><th>Name</th><th>Age</th><th>Sal</th></tr>
<tr><td>john</td><td>25</td><td>15000</td></tr>
<tr><td>Mark</td><td>24</td><td>20000</td></tr>
<tr><td>Devid</td><td>28</td><td>25000</td></tr>
</table>
</apex:tab>
<apex:tab label="Tab3" name="tab3" id="tab3">
<apex:pageBlock title="Using Iframe">
 <iframe height="500px" id="Page" name="Callmypage" src="/apex/Tab_css" width="100%"></iframe>
    </apex:pageBlock>
</apex:tab>
<apex:tab label="Tab4" name="tab4" id="tab4">
<apex:pageBlock title="My Page">

</apex:pageBlock>
</apex:tab>
</apex:tabPanel>
</apex:page>
```



Name	Age	Sal
john	25	15000
Mark	24	20000
Devid	28	25000

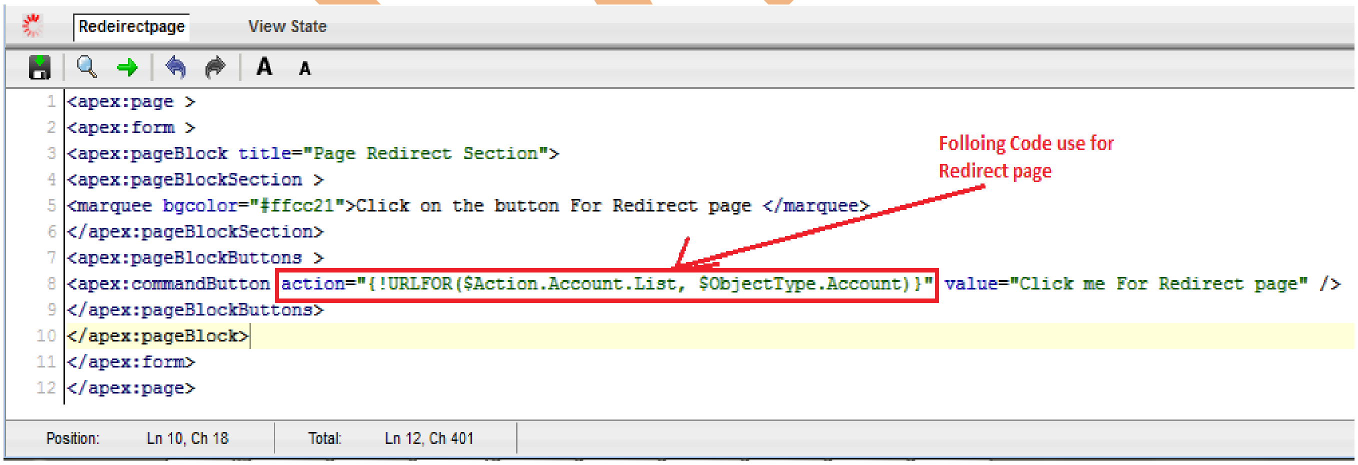
Copyright © 2000-2014 salesforce.com, inc. All rights reserved. | Privacy Statement | Security Statement | Terms of Use | 508 Compliance

```
<apex:page standardController="Account" showHeader="true" tabStyle="account" sidebar="false" >
<style>
.activeTab {background-color: #FFcc23; color:white;font-size:15px; background-image:none; }
.inactiveTab { background-color: #3399FF; color:black;background-image:none}
</style>
<apex:tabPanel switchType="client" selectedTab="tabing" id="AccountTabPanel" tabClass="activeTab" inactiveTabClass="inactiveTab">
<apex:tab label="Tab1" name="tab1" id="tab1">
<apex:pageBlock title="Welcome :{!$User.FirstName}">
<apex:detail relatedList="false" title="true"/>
</apex:pageBlock>
</apex:tab>
<apex:tab label="Tab2" name="tab2" id="tab2">
<table><tr><th>Name</th><th>Age</th><th>Sal</th></tr>
<tr><td>john</td><td>25</td><td>15000</td></tr>
<tr><td>Mark</td><td>24</td><td>20000</td></tr>
```

Position: Ln 30, Ch 13 Total: Ln 30, Ch 1180

Redirecting to a Standard Object List Page

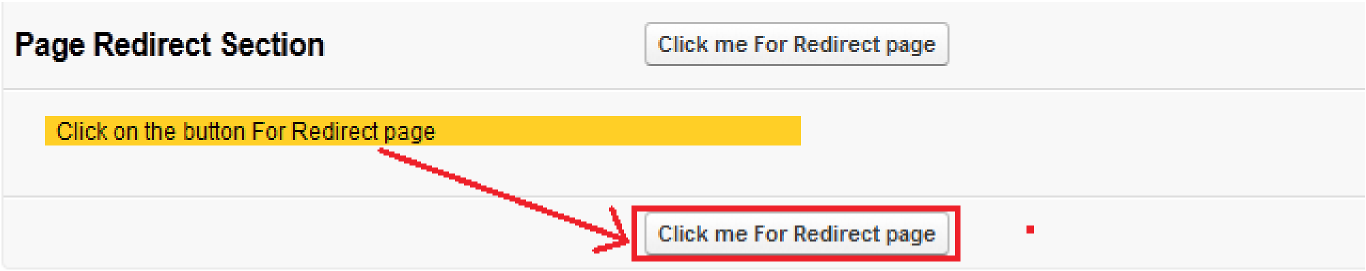
In Salesforce.com, buttons or links that navigate a user to a standard tab, you can redirect the content to present a list of standard objects.
Create a VisualForce page with the following markup :
Following code uses command button, see below.



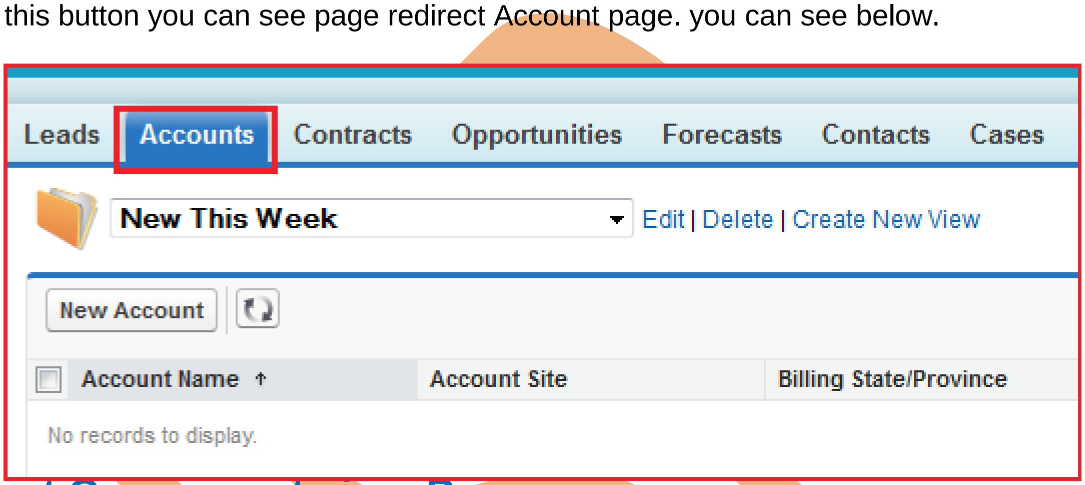
```
<apex:page >
<apex:form >
<apex:pageBlock title="Page Redirect Section">
<apex:pageBlockSection >
<marquee bgcolor="#ffcc21">Click on the button For Redirect page </marquee>
</apex:pageBlockSection>
<apex:pageBlockButtons >
<apex:commandButton action="{!URLFOR($Action.Account.List, $ObjectType.Account)}" value="Click me For Redirect page" />
</apex:pageBlockButtons>
</apex:pageBlock>
</apex:form>
</apex:page>
```

Folloing Code use for Redirect page

Position: Ln 10, Ch 18 Total: Ln 12, Ch 401


Page Redirect Section
Click me For Redirect page
Click on the button For Redirect page
Click me For Redirect page

After Click on this button you can see page redirect Account page. you can see below.

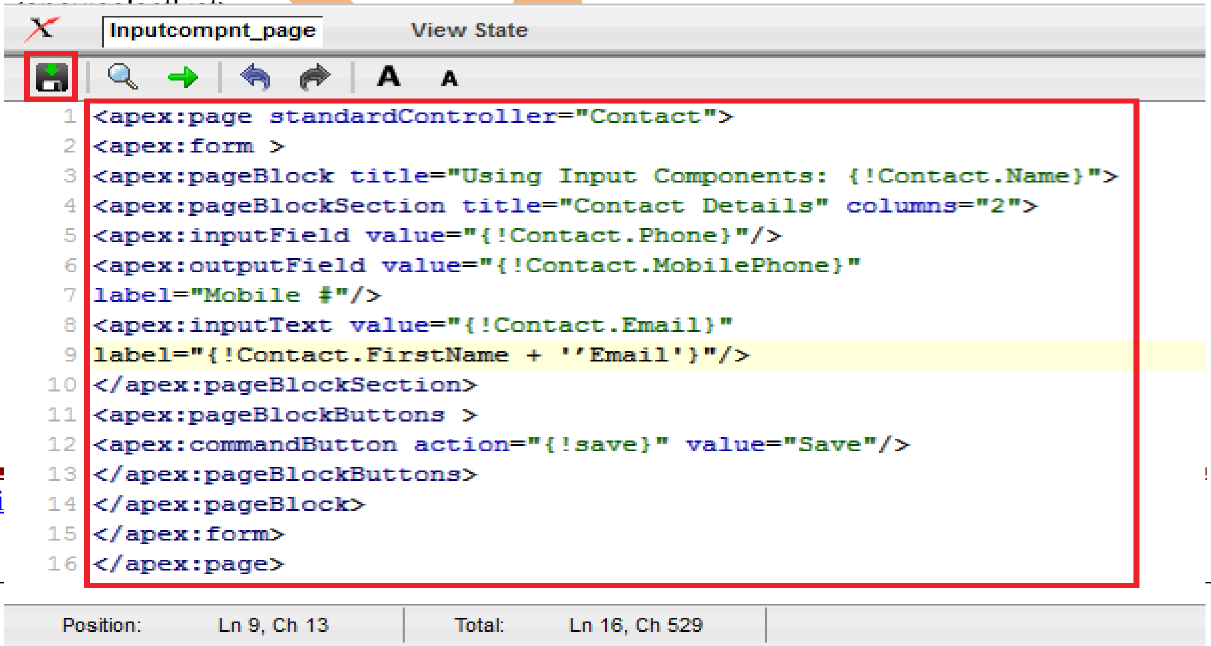


Using Input Components in a Page

In VisualForce you can use the <apex:form> tag with one or more input components and a <apex:commandLink> or <apex:commandButton> tag to submit the form.

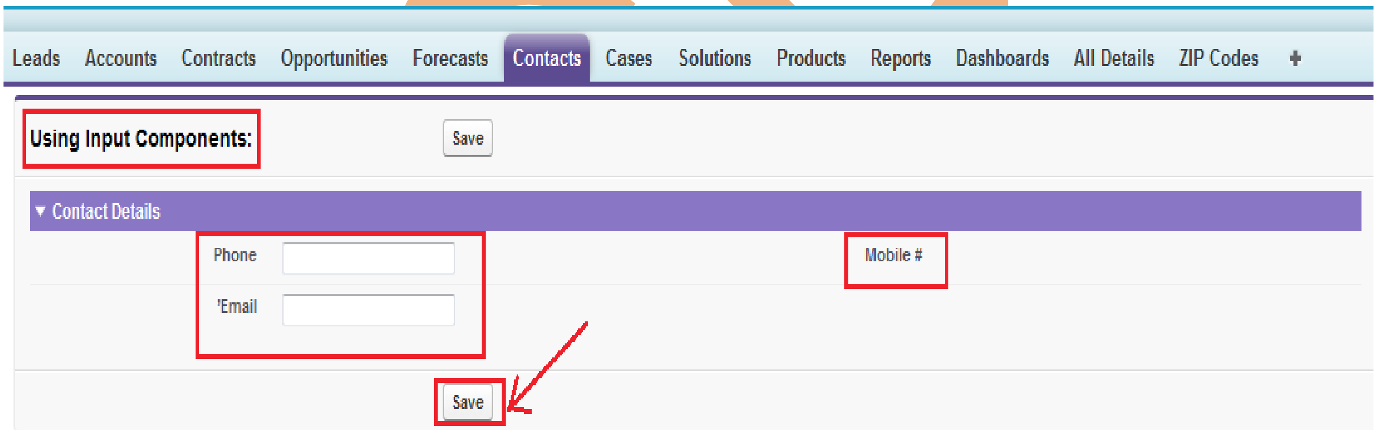
The input component tag :

- <apex:inputCheckbox>
- <apex:inputField>
- <apex:inputSecret>
- <apex:inputText>
- <apex:inputTextarea>
- <apex:outputField>
- <apex:outputText>
- <apex:selectCheckboxes>
- <apex:selectList>



```
<apex:page standardController="Contact">
<apex:form >
<apex:pageBlock title="Using Input Components: {!Contact.Name}">
<apex:pageBlockSection title="Contact Details" columns="2">
<apex:inputField value="{!Contact.Phone}"/>
<apex:outputField value="{!Contact.MobilePhone}"
label="Mobile #"/>
<apex:inputText value="{!Contact.Email}"
label="{!Contact.FirstName + ''Email'}"/>
</apex:pageBlockSection>
<apex:pageBlockButtons >
<apex:commandButton action="{!save}" value="Save"/>
</apex:pageBlockButtons>
</apex:pageBlock>
</apex:form>
</apex:page>
```

Position: Ln 9, Ch 13 | Total: Ln 16, Ch 529


Leads
Accounts
Contracts
Opportunities
Forecasts
Contacts
Cases
Solutions
Products
Reports
Dashboards
All Details
ZIP Codes
+
Using Input Components:
Save
▼ Contact Details
Phone
'Email
Mobile #
Save