APEX - INVOKING

Apex invoking refers to the process of executing the Apex class. Apex class can only be executed when it is invoked via one of the below ways:

- Triggers and Anonymous block
- A trigger invoked for specified events.
- Asynchronous Apex
- Scheduling an Apex class to run at specified intervals, or running a batch job.
- Web Services class
- Apex Email Service class
- Apex Web Services, which allow exposing your methods via SOAP and REST Web services.
- Visualforce Controllers
- Apex Email Service to process inbound email.
- Invoking Apex Using JavaScript
- The Ajax toolkit to invoke Web service methods implemented in Apex.

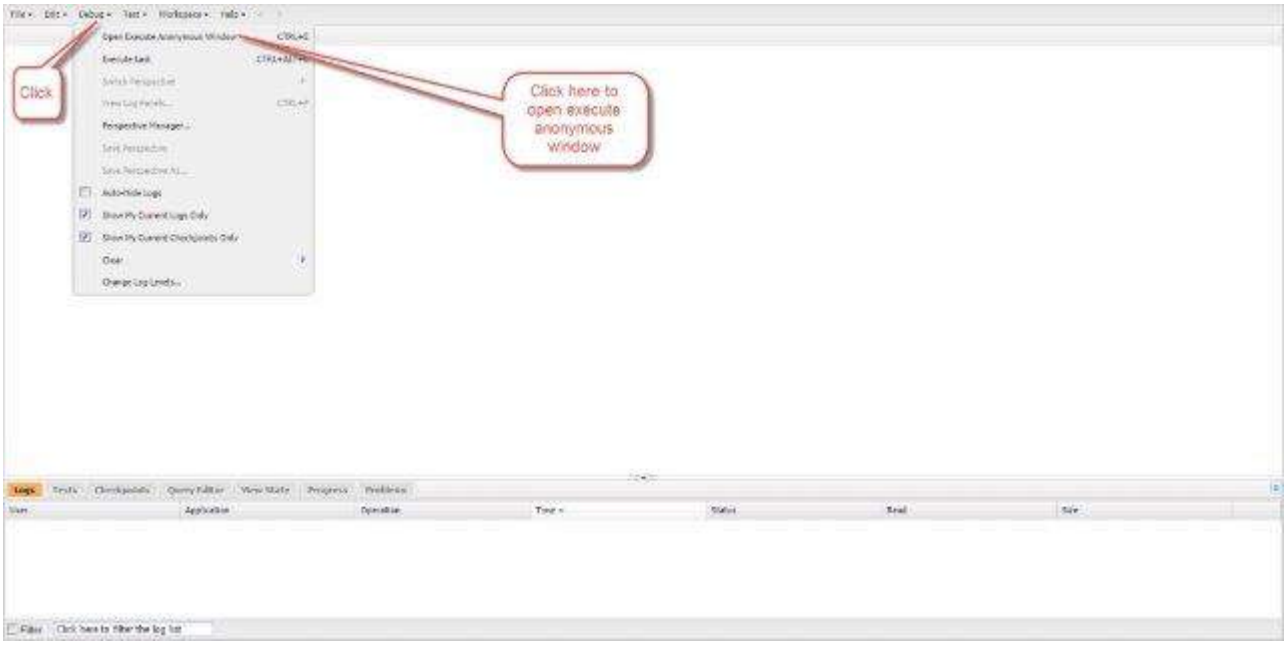
We will have look at some common way to invoke Apex.

From Execute Anonymous Block

You could invoke the Apex class via execute anonymous in Developer console as shown below:

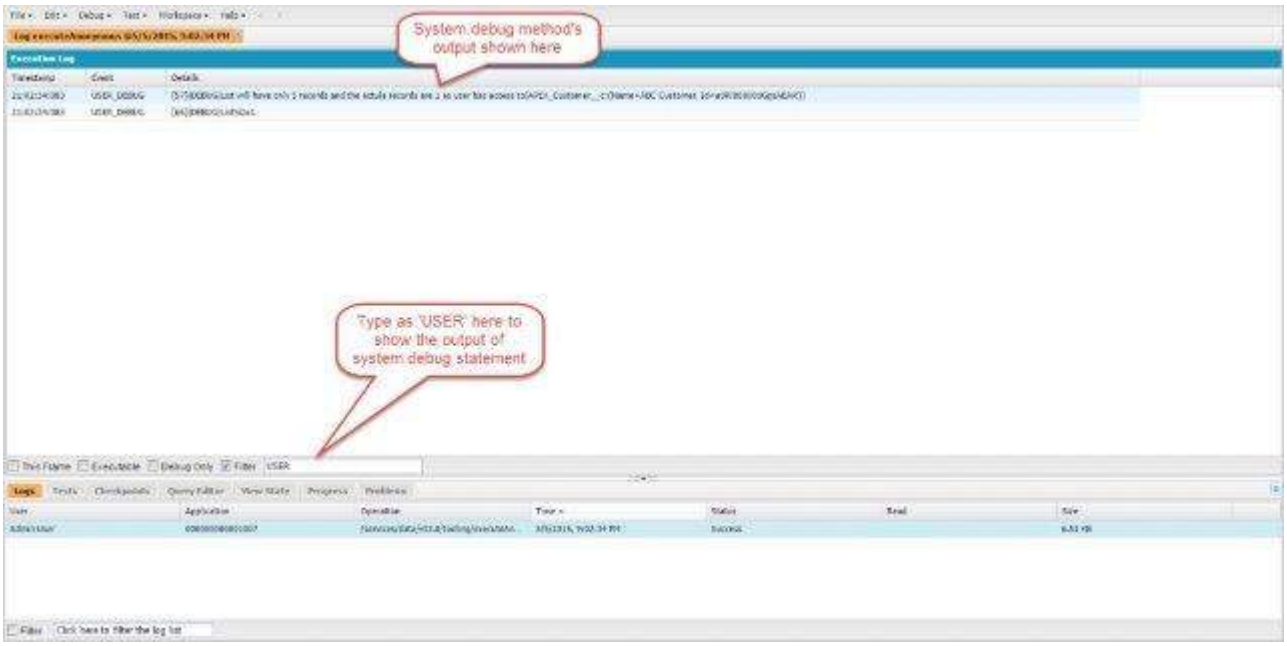
Step 1: Open Developer Console

Step 2: Click on Debug.



Step 3: Execute anonymous window will open as shown below and click on execute button:

Step 4: Open Debug Log when it will appear in Logs pane.



From Trigger

You could call an Apex class from Trigger as well. Triggers are called when a specified event occurs and triggers can call the Apex class when executing.

Below is the sample code that shows how a class gets executed when a Trigger is called.

Example:

```
//Class which will gets called from trigger
public without sharing class MyClassWithSharingTrigger {

    public static Integer executeQuery (List<apex_customer__c> CustomerList) {
        //perform some logic and operations here
        Integer ListSize = CustomerList.size();
        return ListSize;
    }
}

//Trigger Code
trigger Customer_After_Insert_Example on APEX_Customer__c (after insert) {
    System.debug('Trigger is Called and it will call Apex Class');
    MyClassWithSharingTrigger.executeQuery(Trigger.new);//Calling Apex class and method
of an Apex class
}

//This example is for reference, no need to execute and will have detail look on
triggers later chapters.
```

From Visualforce Page Controller Code

Apex class can be called from Visualforce page as well. We can specify the controller or controller extension and the specified Apex class gets called.

Example:

VF Page Code:

```
<apex:page standardController="Event" extensions="EventHistoryControllerExtension">
<apex:pageBlock title="Event Field History Tracking">
    <apex:pageBlockSection columns="1">
        <apex:pageMessages />
            <apex:pageBlockTable value="{!FieldHistoryRecords}" var="objFieldHist">
                <apex:column value="{!objFieldHist.User__c}" width="33%" headerValue="User"/>
                <apex:column value="{!objFieldHist.Date_Time_Changed__c}" headerValue="Date" width="33%"/>
                <apex:column value="{!objFieldHist.Action__c}" headerValue="Action" width="33%"/>
            </apex:pageBlockTable>
    </apex:pageBlockSection>
  </apex:pageBlock>
</apex:page>
```

Apex Class as controller extension

Apex Class Code ***ControllerExtension***

```
Public with sharing class EventHistoryControllerExtension {

    public String strEventId{get;set;}
    public List<Event_Field_History_Tracker__c> lstFieldHistory {get; set;}
    public String message{get;set;}
    public string strEventIdFromDb {get; set;}
    public id strEventId1 {get; set;}

    public EventHistoryControllerExtension(ApexPages.StandardController controller) {
        strEventId = System.currentPageReference().getParameters().get('Id');
        if (strEventId != null) {
        strEventId1 = Id.valueOf(strEventId);
        }
        system.debug('strEventId18Digit'+strEventId );
```

Apex Class which will be called from Page