

Phase 5 Documentation – Apex Programming (Developer)

Project: Customer Complaint CRM

Objective

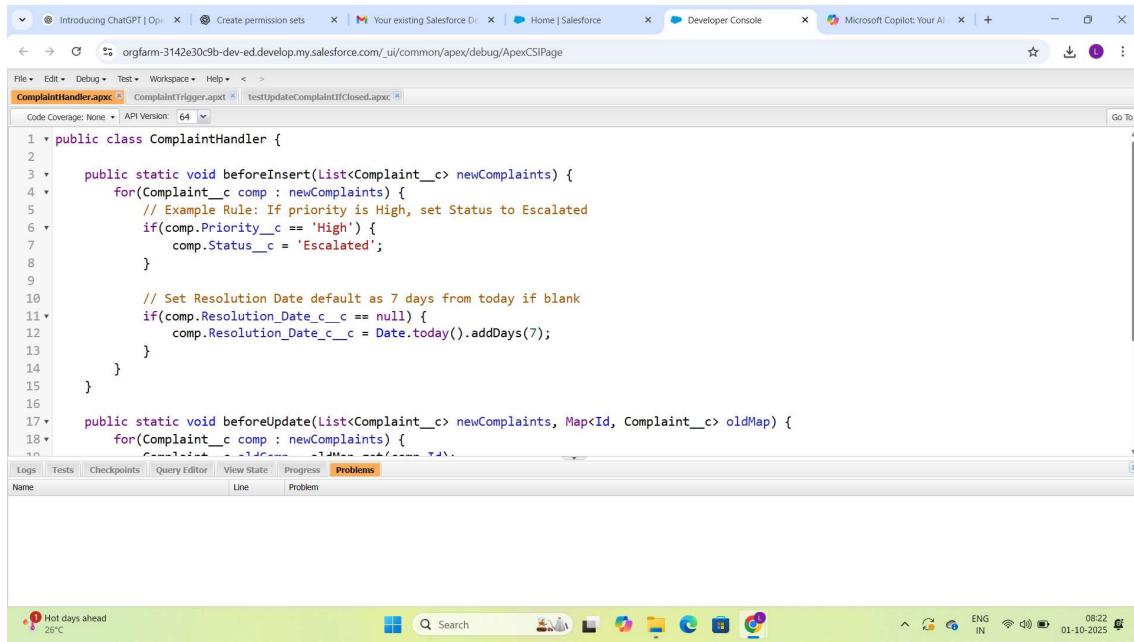
Enable automation and data handling in Salesforce using **Apex Classes, Triggers, SOQL, Collections, Control Statements, Exception Handling, and Test Classes**.

Steps

Step 1: Create an Apex Class (Handler)

1. Go to **Setup** → **Developer Console**.
2. Click **File** → **New** → **Apex Class**.
3. Name: **ComplaintHandler**.
4. Paste the following code:

```
public class ComplaintHandler {  
    public static void handleBeforeInsert(List<Complaint__c> complaints) {  
        for(Complaint__c comp : complaints) {  
            // Example rule: auto-set Resolution Date to 7 days after  
            // Complaint Date  
            if(comp.Complaint_Date__c != null && comp.Resolution_Date__c ==  
                null) {  
                comp.Resolution_Date__c =  
                    comp.Complaint_Date__c.addDays(7);  
            }  
  
            // Escalate if Priority is High  
            if(comp.Priority__c == 'High') {  
                comp.Status__c = 'Escalated';  
            }  
        }  
    }  
  
    public static void handleBeforeUpdate(List<Complaint__c> complaints) {  
        for(Complaint__c comp : complaints) {  
            // Example update rule  
            if(comp.Priority__c == 'Low' && comp.Status__c == 'Escalated') {  
                comp.Status__c = 'In Progress';  
            }  
        }  
    }  
}
```



```
1 public class ComplaintHandler {
2
3     public static void beforeInsert(List<Complaint__c> newComplaints) {
4         for(Complaint__c comp : newComplaints) {
5             // Example Rule: If priority is High, set Status to Escalated
6             if(comp.Priority__c == 'High') {
7                 comp.Status__c = 'Escalated';
8             }
9
10            // Set Resolution Date default as 7 days from today if blank
11            if(comp.Resolution_Date__c == null) {
12                comp.Resolution_Date__c = Date.today().addDays(7);
13            }
14        }
15    }
16
17    public static void beforeUpdate(List<Complaint__c> newComplaints, Map<Id, Complaint__c> oldMap) {
18        for(Complaint__c comp : newComplaints) {
19            // Example Rule: If priority is High, set Status to Escalated
20            if(comp.Priority__c == 'High') {
21                comp.Status__c = 'Escalated';
22            }
23        }
24    }
25}
```

Step 2: Create an Apex Trigger

1. In Developer Console → File → New → Apex Trigger.
2. Name: ComplaintTrigger.
3. Select sObject: Complaint__c.
4. Paste this code:

```
trigger ComplaintTrigger on Complaint__c (before insert, before update) {
    if(Trigger.isBefore && Trigger.isInsert) {
        ComplaintHandler.handleBeforeInsert(Trigger.new);
    }
    if(Trigger.isBefore && Trigger.isUpdate) {
        ComplaintHandler.handleBeforeUpdate(Trigger.new);
    }
}
```

```

1 trigger ComplaintTrigger on Complaint__c (before insert, before update) {
2     if(Trigger.isBefore && Trigger.isInsert) {
3         ComplaintHandler.beforeInsert(Trigger.new);
4     }
5     if(Trigger.isBefore && Trigger.isUpdate) {
6         ComplaintHandler.beforeUpdate(Trigger.new, Trigger.oldMap);
7     }
8 }

```

Step 3: SOQL Example

- Run in Execute Anonymous (Ctrl+E) for testing:

```

List<User> users = [SELECT Id, Name FROM User WHERE IsActive = TRUE LIMIT 5];
System.debug('Active Users: ' + users);

```

```

1 @isTest
2 private class testUpdateComplaintIfClosed {
3     @isTest
4     static void testClosedComplaintSetsResolutionDate() {
5         // Create a new Complaint record without setting the Name field
6         Complaint__c comp = new Complaint__c();
7         comp.Priority__c = 'High';
8         comp.Status__c = 'Open';
9         insert comp;
10
11         // Update the status to Closed
12         comp.Status__c = 'Closed';
13         update comp;
14
15         // Query the updated record
16         Complaint__c updatedComp = [
17             SELECT Status__c, Resolution_Date__c
18             FROM Complaint__c
19         ];
20     }
21 }

```

Step 4: Collections (Practice)

□ Run in Execute Anonymous:

```
List<String> complaintTypes = new List<String>{'Billing','Service','Product'};  
Set<String> statuses = new Set<String>{'New','In Progress','Closed'};  
Map<Id, Complaint__c> compMap = new Map<Id, Complaint__c>(  
    [SELECT Id, Name FROM Complaint__c LIMIT 10]  
);  
  
System.debug('Types: ' + complaintTypes);  
System.debug('Statuses: ' + statuses);  
System.debug('Complaints Map: ' + compMap);
```

Step 5: Control Statements

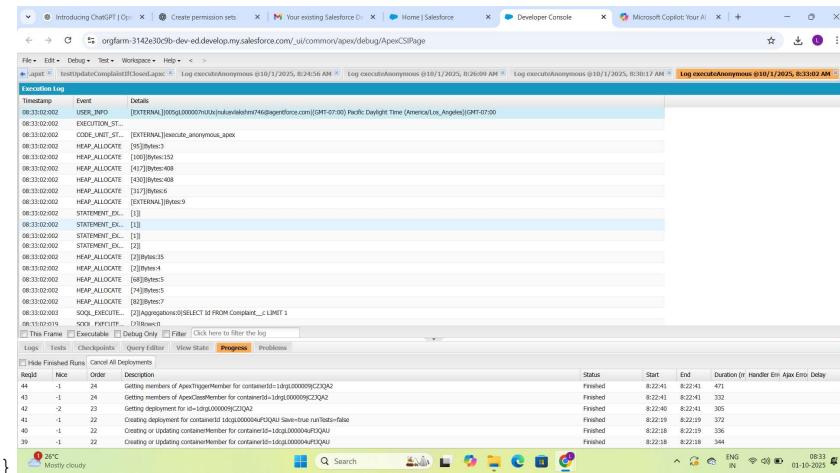
□ Already used in handler. Example:

```
for(Complaint__c comp : Trigger.new) {  
    if(comp.Priority__c == 'High') {  
        comp.Status__c = 'Escalated';  
    }  
}
```

Step 6: Exception Handling

□ Practice in Execute Anonymous:

```
try {  
    List<Complaint__c> comps = [SELECT Id FROM Complaint__c LIMIT 1];  
    System.debug('Found complaint: ' + comps[0].Id);  
} catch(Exception e) {  
    System.debug('Error: ' + e.getMessage());
```



Step 7: Test Class (Mandatory for Deployment)

1. Developer Console → File → New → Apex Class → Name:

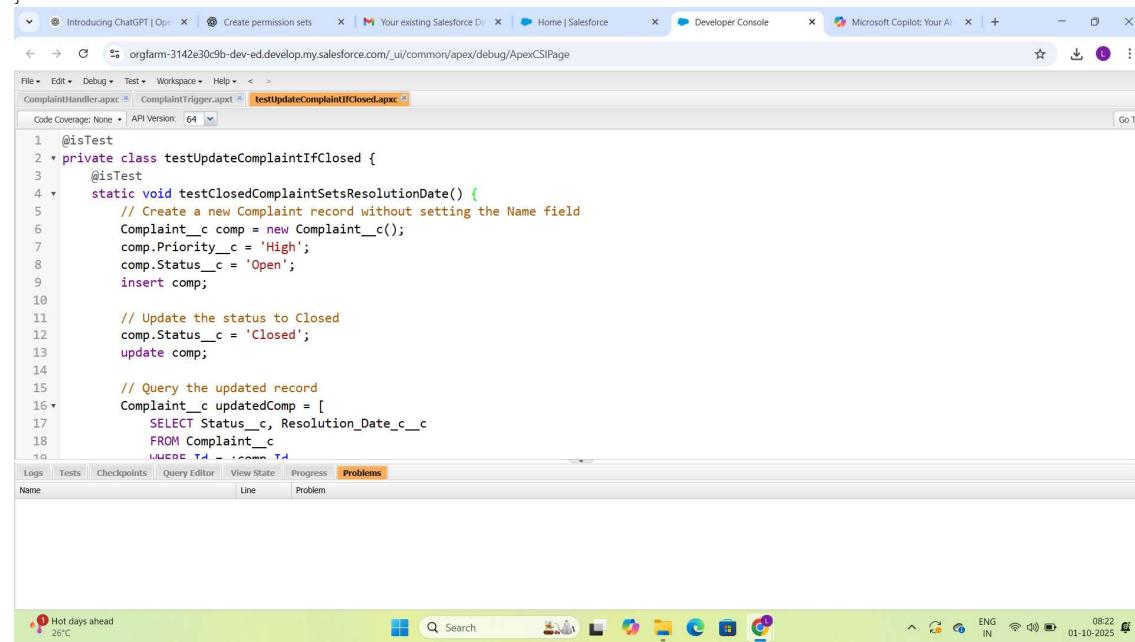
ComplaintTriggerTest.

2. Paste this:

```
@isTest
public class ComplaintTriggerTest {
    @isTest
    static void testInsertComplaint() {
        Complaint__c comp = new Complaint__c(
            Name='Test Complaint',
            Priority__c='High',
            Complaint__Date__c=Date.today()
        );
        insert comp;

        Complaint__c insertedComp = [SELECT Status__c, Resolution__Date__c
                                      FROM Complaint__c
                                      WHERE Id = :comp.Id];

        System.assertEquals('Escalated', insertedComp.Status__c);
        System.assertNotEquals(null, insertedComp.Resolution__Date__c);
    }
}
```



The screenshot shows the Salesforce Developer Console interface. The code editor contains the test class code. The status bar at the bottom of the browser window displays a weather forecast for 'Hot days ahead 26°C' and the date '01-10-2025'.

