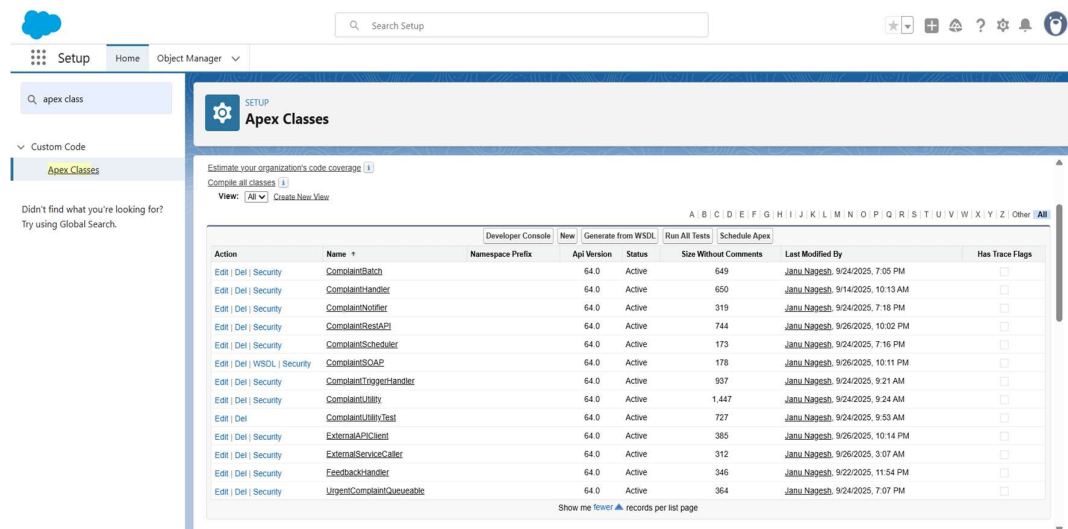# Phase 5: Apex Programming (Developer)

1. Apex Classes & Objects

- **Where**: Setup → Apex Classes → New, or create in VS Code under force-app/main/default/classes.

- **Created Classes**:
  - ComplaintUtility.cls: contains complaint logic (status, SLA calculation).
  - ComplaintNotifier.cls: future method for async notifications.
  - UrgentComplaintQueueable.cls: queueable job for escalation.
  - ComplaintScheduler.cls: runs daily via cron expression.
  - FeedbackHandler.cls: handles feedback insert.

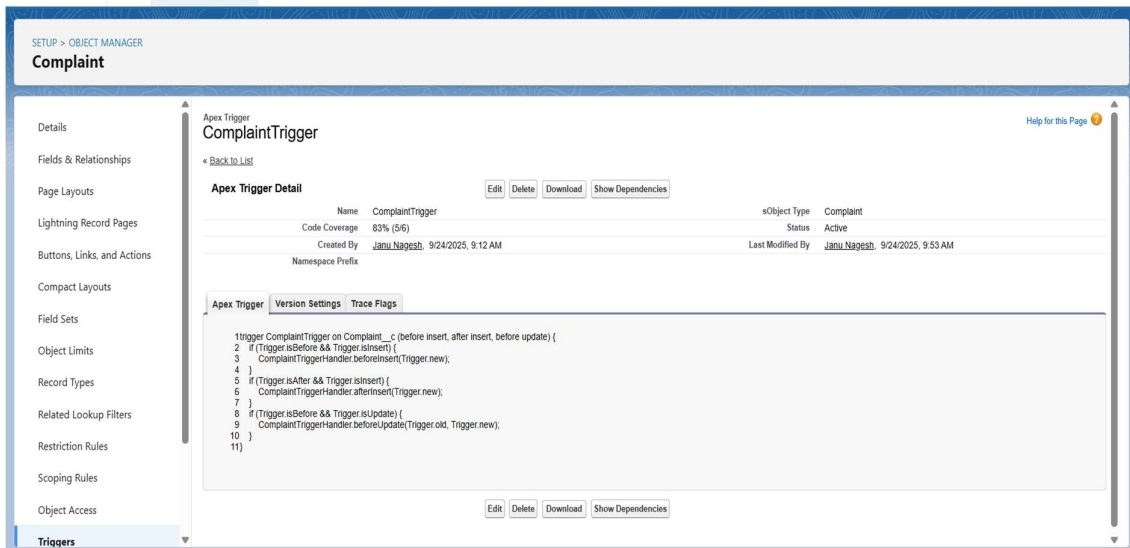- **Values**: Add public with sharing to enforce sharing rules.

2. Apex Triggers

- **Where**: Setup → Object Manager → Complaint__c → Triggers → New.

- **Trigger**: ComplaintTrigger → Before Insert, After Insert.

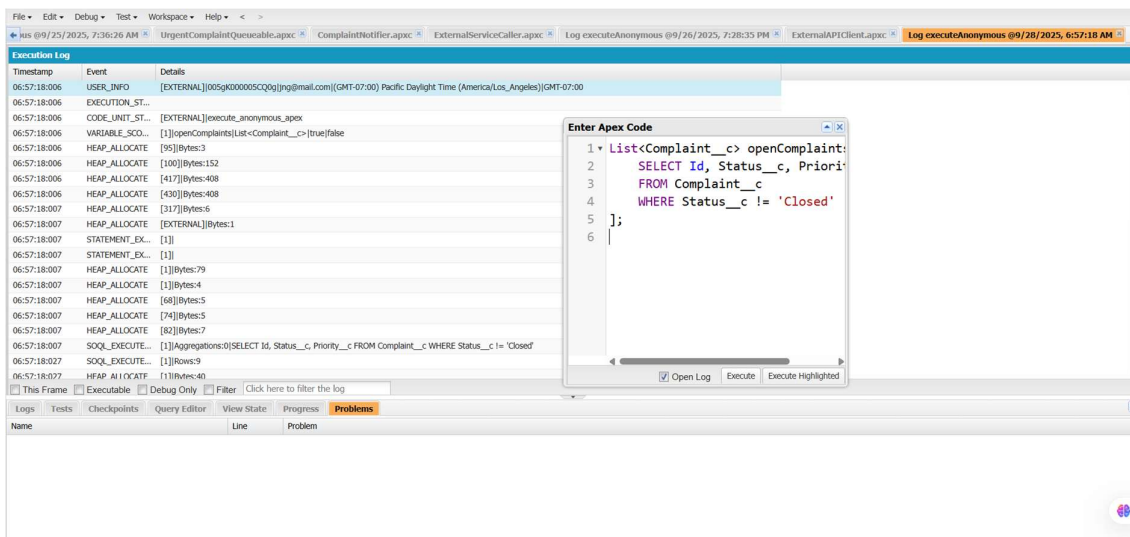- **Logic**: Calls ComplaintUtility instead of writing inline code.

3. SOQL & Collections

- **SOQL Example**:

List<Complaint__c> openComplaints = [

SELECT Id, Status__c, Priority__c

FROM Complaint__c

WHERE Status__c != 'Closed'

];

- **Collections**: Used Lists (List<Complaint__c>), Maps (Map<Id, Complaint__c>), Sets (Set<String>).
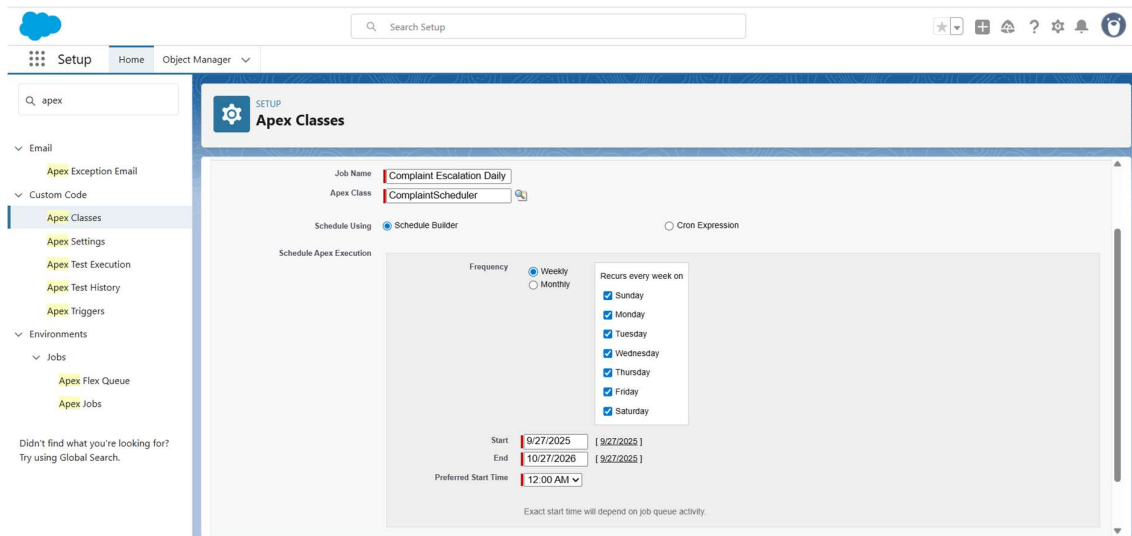
4. Asynchronous Apex

- **Queueable Apex**:
    - Class: UrgentComplaintQueueable.
    - Run via: System.enqueueJob(new UrgentComplaintQueueable());.

- **Scheduled Apex**:
    - Class: ComplaintScheduler.
    - Setup → Apex Classes → Schedule Apex → Name = "Complaint Escalation Daily".
    - CRON: Run every day at 12:00 AM.

5. Exception Handling

```
try {
    insert comp;
} catch (DmlException e) {
    throw new AuraHandledException('Error creating complaint: ' + e.getMessage());
}
```

6. Test Classes

- **Where**: force-app/main/default/classes/

- **Command**:

sf apex run test --test-class-name ComplaintUtilityTest --target-org SmartCityMain