

# SafeCity Salesforce Project:

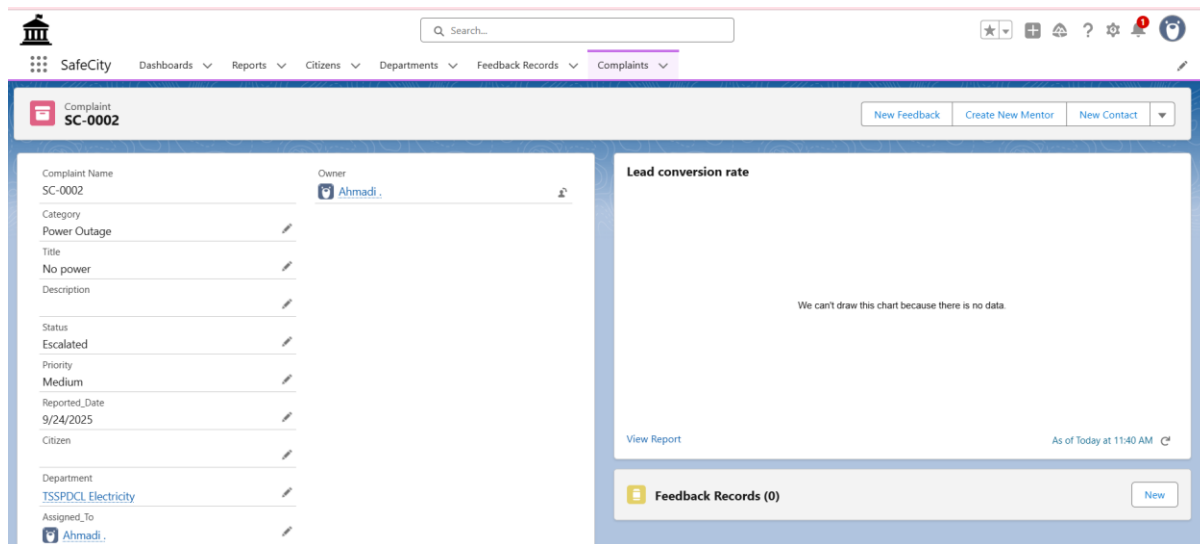
## Phase 5: Apex Programming

### 1. Classes & Objects

**Purpose:** To encapsulate complaint-related logic into reusable components, improving maintainability.

**Implementation:** Created a class `ComplaintManager` with fields for owner and reported date, along with a method to check if the SLA is overdue.

**Benefit:** Ensures code reusability and separates business logic from triggers/flows.

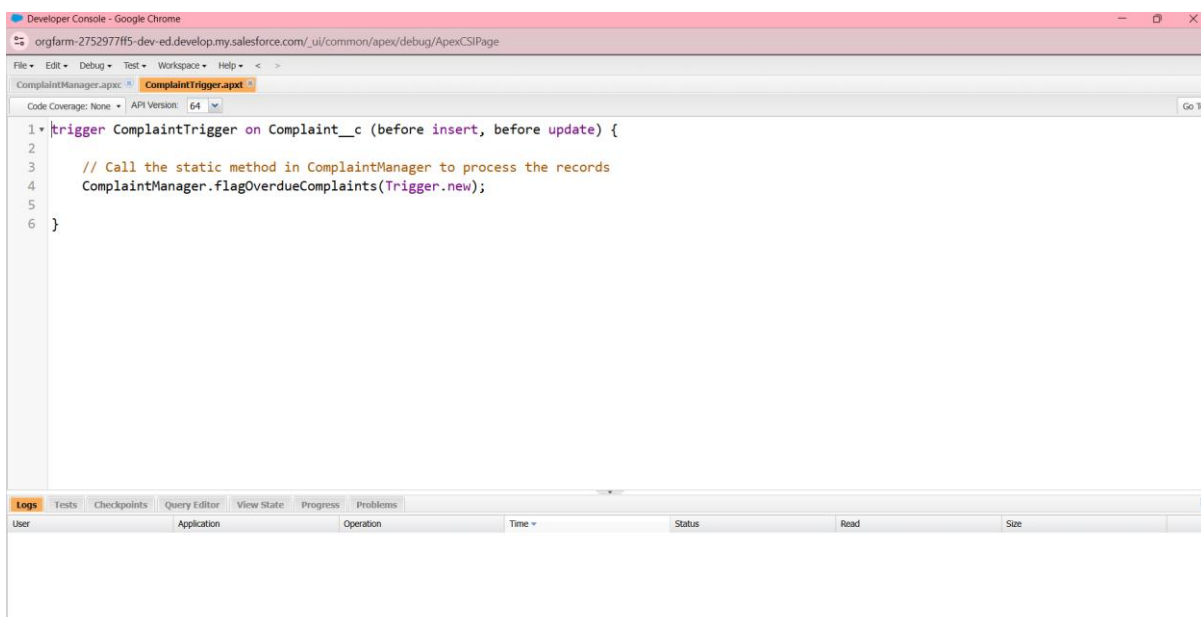


### 2. Apex Triggers

**Purpose:** To automate actions when Complaint records are inserted or updated.

**Implementation:** Designed a trigger on `Complaint__c` that checks SLA dates and escalates complaints if they are overdue. Logic is handled in a separate handler class.

**Benefit:** Automatically enforces SLA rules without manual monitoring.

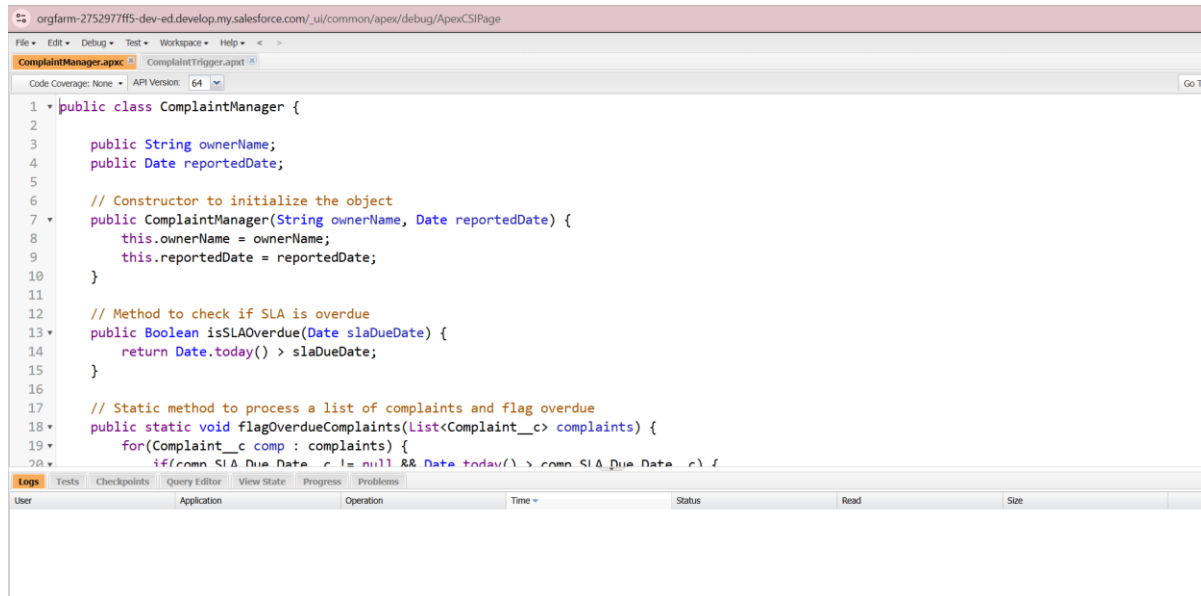


### 3. Trigger Design Pattern

**Purpose:** To follow best practices by keeping triggers lightweight.

**Implementation:** Created a ComplaintHandler class where the main escalation logic resides, while the trigger only delegates work.

**Benefit:** Increases scalability, readability, and makes testing/debugging easier.



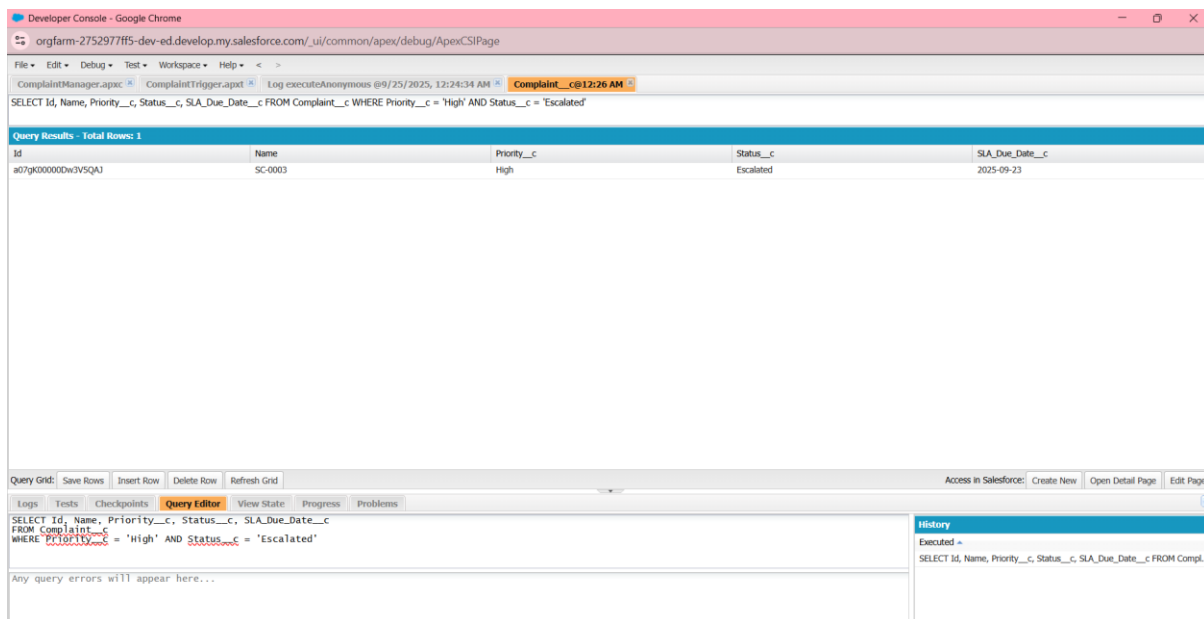
```
1 public class ComplaintManager {
2
3     public String ownerName;
4     public Date reportedDate;
5
6     // Constructor to initialize the object
7     public ComplaintManager(String ownerName, Date reportedDate) {
8         this.ownerName = ownerName;
9         this.reportedDate = reportedDate;
10    }
11
12    // Method to check if SLA is overdue
13    public Boolean isSLAOverdue(Date slaDueDate) {
14        return Date.today() > slaDueDate;
15    }
16
17    // Static method to process a list of complaints and flag overdue
18    public static void flagOverdueComplaints(List<Complaint__c> complaints) {
19        for(Complaint__c comp : complaints) {
20            if(comp.SLA_Due_Date__c != null && Date.today() > comp.SLA_Due_Date__c) {
```

### 5. SOSL (Salesforce Object Search Language)

**Purpose:** To search complaints across multiple fields using keywords.

**Implementation:** Used SOSL to search for complaints by description keywords like “pothole.”

**Benefit:** Helps staff quickly find records even when exact details are not known.



Query Results - Total Rows: 1

Id	Name	Priority__c	Status__c	SLA_Due_Date__c
a07gk00000dw3V5QAJ	SC-0003	High	Escalated	2025-09-23

Query Editor: SELECT Id, Name, Priority\_\_c, Status\_\_c, SLA\_Due\_Date\_\_c FROM Complaint\_\_c WHERE Priority\_\_c = 'High' AND Status\_\_c = 'Escalated'

History: Executed = SELECT Id, Name, Priority\_\_c, Status\_\_c, SLA\_Due\_Date\_\_c FROM Compl...

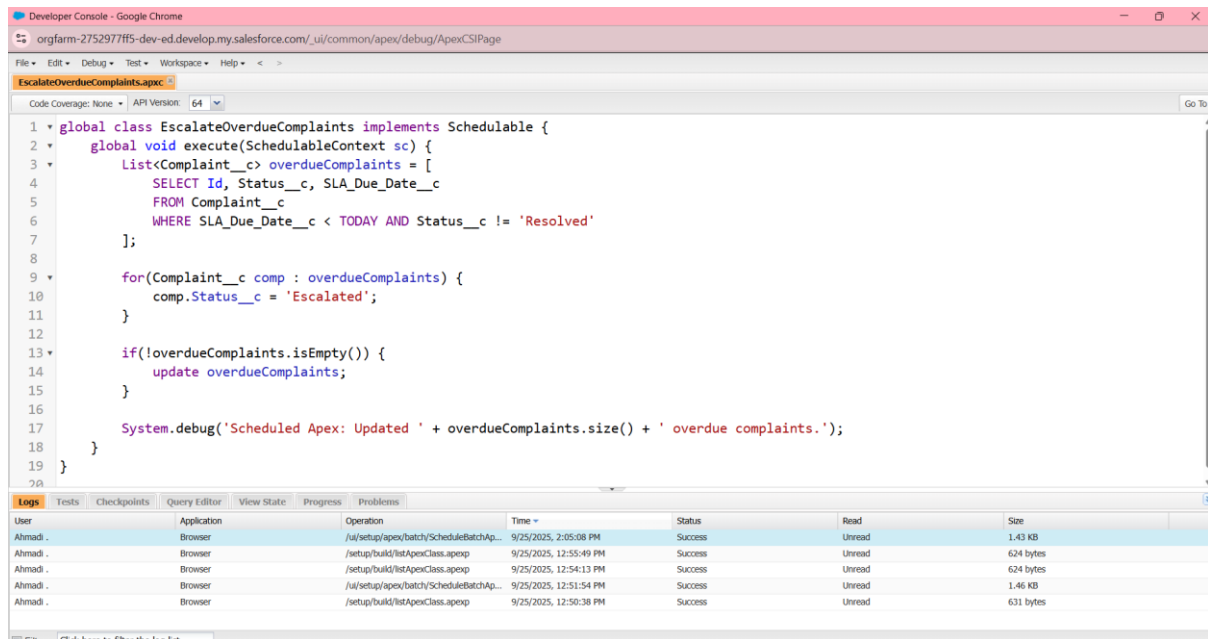
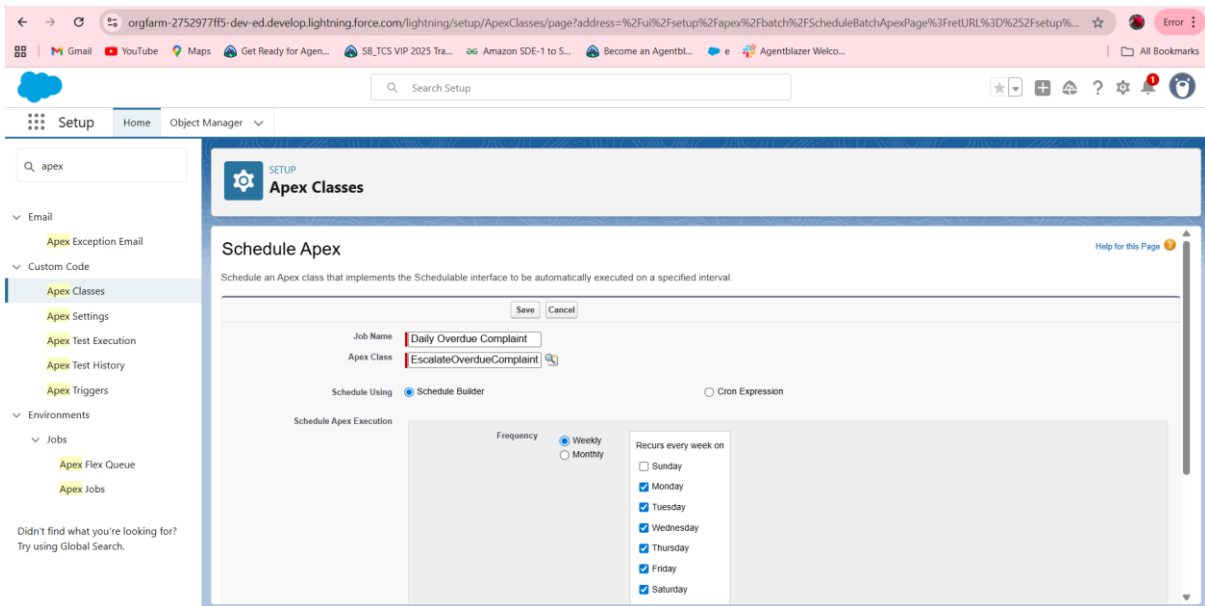
### 6. Scheduled Apex

**Purpose:** To automate complaint monitoring at fixed time intervals.

**Implementation:** Created a scheduled job that runs daily to escalate overdue complaints automatically.

Scheduled via Setup/Execute Anonymous.

**Benefit:** Reduces manual oversight and ensures SLA compliance consistently.



## 7. Batch Apex

**Purpose:** To process large complaint datasets asynchronously in manageable chunks.

**Implementation:** Developed a Batch Apex job that identifies all complaints marked as “Resolved” and closes them in bulk. Executed through Developer Console.

**Benefit:** Ensures system can handle bulk updates without hitting governor limits, maintaining performance.

Developer Console - Google Chrome

orgfarm-2752977f5-dev-ed.develop.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage

EscalateOverdueComplaints.apex CloseResolvedComplaints.apex Log executeAnonymous @9/25/2025, 2:09:57 PM

Code Coverage: None API Version: 64

```
6 SELECT Id, Status__c
7 FROM Complaint__c
8 WHERE Status__c = 'Resolved'
9
10 });
11
12 // Step 2: Process each batch of records
13 * global void execute(Database.BatchableContext BC, List<Complaint__c> scope) {
14     for (Complaint__c comp : scope) {
15         comp.Status__c = 'Closed';
16     }
17     update scope;
18 }
19
20 // Step 3: Run final logic after all batches complete
21 * global void finish(Database.BatchableContext BC) {
22     System.debug('Batch Job Finished: All resolved complaints closed.');
```

23 }
24
25 }

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
Ahmad	Unknown	/services/data/v64.0/tooling/executeA...	9/25/2025, 2:09:57 PM	Success	Unread	3.07 KB
Ahmad	Unknown	Batch Apex	9/25/2025, 2:09:57 PM	Success	Unread	3.64 KB
Ahmad	Unknown	Batch Apex	9/25/2025, 2:09:57 PM	Success	Unread	3.5 KB
Ahmad	Browser	/ui/setup/apex/batch/ScheduleBatchA...	9/25/2025, 2:05:08 PM	Success	Unread	1.43 KB
Ahmad	Browser	/setup/build/listApexClass.apex	9/25/2025, 12:55:49 PM	Success	Unread	624 bytes
Ahmad	Browser	/setup/build/listApexClass.apex	9/25/2025, 12:54:13 PM	Success	Unread	624 bytes

Filter Click here to filter the log list

orgfarm-2752977f5-dev-ed.develop.my.salesforce.com/\_ui/common/apex/debug/ApexCSIPage

EscalateOverdueComplaints.apex CloseResolvedComplaints.apex Log executeAnonymous @9/25/2025, 2:09:57 PM

Execution Log

Timestamp	Event	Details
14:09:57:005	VARIABLE_SCO...	[11][this]Database.BatchableContextImpl[true/false]
14:09:57:005	VARIABLE_ASS...	[11][this]{}0x8ab6967
14:09:57:005	VARIABLE_SCO...	[11][jobId]{}false
14:09:57:005	VARIABLE_ASS...	[11][jobId]"707gK00000Dzpl3QA"
14:09:57:005	VARIABLE_SCO...	[11][childJobId]{}false
14:09:57:005	VARIABLE_ASS...	[11][childJobId]true
14:09:57:024	HEAP_ALLOCATE	[EXTERNAL]]bytes:57
14:09:57:024	HEAP_ALLOCATE	[EXTERNAL]]bytes:7
14:09:57:024	HEAP_ALLOCATE	[EXTERNAL]]bytes:2
14:09:57:024	METHOD_ENTRY	[1]01pgK0000050sc2X[CloseResolvedComplaints.CloseResolvedComplaints]
14:09:57:024	STATEMENT_EX...	[1]
14:09:57:024	STATEMENT_EX...	[1]
14:09:57:024	METHOD_EXIT	[1]CloseResolvedComplaints
14:09:57:024	HEAP_ALLOCATE	[EXTERNAL]]bytes:4
14:09:57:024	HEAP_ALLOCATE	[EXTERNAL]]bytes:12
14:09:57:024	VARIABLE_SCO...	[21][this]CloseResolvedComplaints[true/false]
14:09:57:024	VARIABLE_ASS...	[21][this]{}0x28eb29a9
14:09:57:024	VARIABLE_SCO...	[21][BC]Database.BatchableContextImpl[true/false]
14:09:57:024	VARIABLE_ASS...	[21][BC]{"jobId":"707gK00000Dzpl3QA"}0x8ab6967
14:09:57:024	STATEMENT_EX...	[21]
14:09:57:024	STATEMENT_EX...	[22]

This Frame Executable Debug Only Filter Click here to filter the log

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
Ahmad	Unknown	/services/data/v64.0/tooling/executeA...	9/25/2025, 2:09:57 PM	Success	Unread	3.07 KB
Ahmad	Unknown	Batch Apex	9/25/2025, 2:09:57 PM	Success	Unread	3.64 KB
Ahmad	Unknown	Batch Apex	9/25/2025, 2:09:57 PM	Success	Unread	3.5 KB
Ahmad	Browser	/ui/setup/apex/batch/ScheduleBatchA...	9/25/2025, 2:05:08 PM	Success	Unread	1.43 KB
Ahmad	Browser	/setup/build/listApexClass.apex	9/25/2025, 12:55:49 PM	Success	Unread	624 bytes
Ahmad	Browser	/setup/build/listApexClass.apex	9/25/2025, 12:54:13 PM	Success	Unread	624 bytes

Setup Home Object Manager

Search Setup

Apex Jobs

Click here to go to the new batch jobs page

Apex Jobs

Monitor the status of all Apex jobs, and optionally, abort jobs that are in progress.

Percent of Asynchronous Apex Used: 0%

You have currently used 0 asynchronous Apex operations out of an allowed 24-hour organization limit of 250,000. To learn about how this limit is calculated and what contributes to it, see the Lightning Platform Apex Limits topic.

View: All Create New View

Action	Submitted Date	Job Type	Status	Status Detail	Total Batches	Batches Processed	Failures	Submitted By	Completion Date	Apex Class	Apex Method	Apex Job ID
	9/25/2025, 1:39 AM	Batch Apex	Completed		0	0	0	Ahmad	9/25/2025, 1:39 AM	CloseResolvedComplaints		707gK00000Dzpl3
	9/25/2025, 1:35 AM	Scheduled Apex	Queued		0	0	0	Ahmad		EscalateOverdueComplaints		707gK00000Dzcyv
	9/25/2025, 12:21 AM	Scheduled Apex	Queued		0	0	0	Ahmad		EscalateOverdueComplaints		707gK00000Dz14y

Didn't find what you're looking for? Try using Global Search.