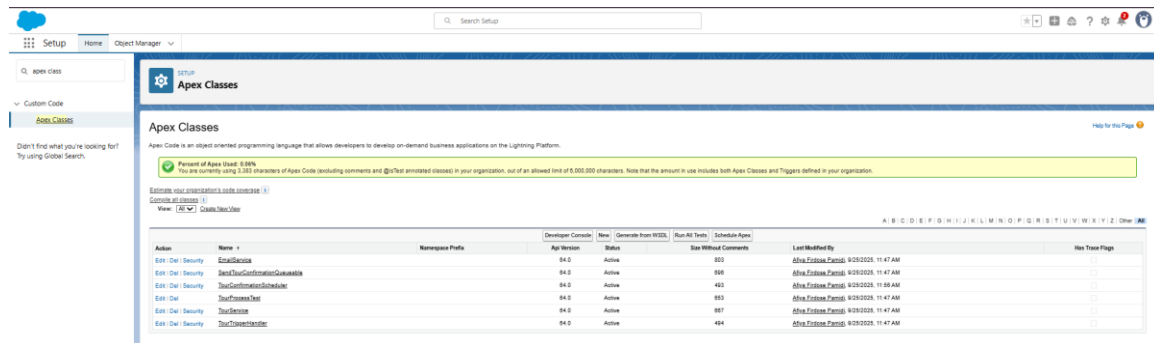


TourFlow: Real Estate Tour and Feedback Automator

Phase 5: Apex Programming (Developer)

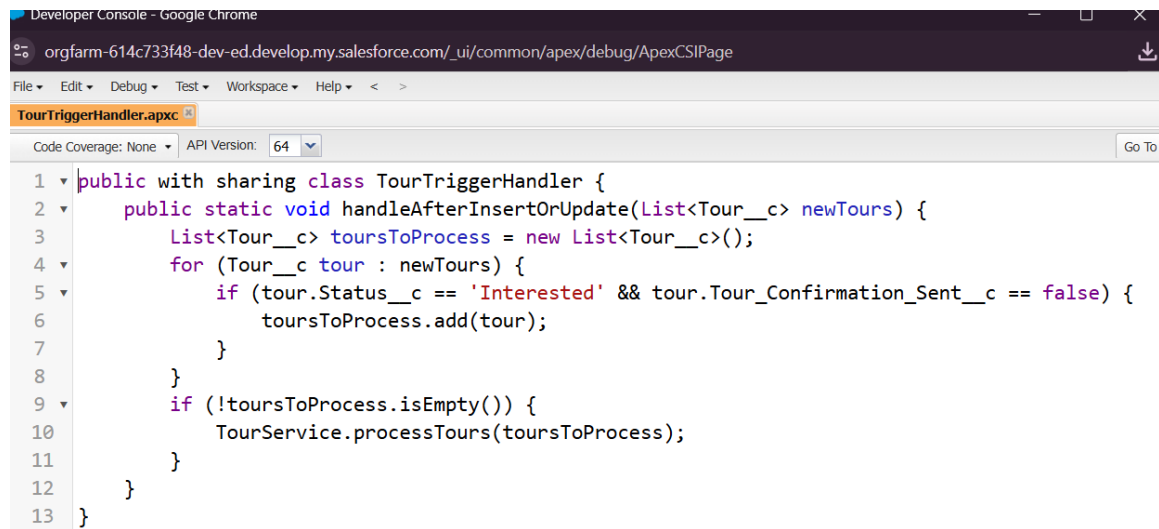
Classes & Objects

We created Apex Classes to organize business logic in a modular way:



Name	Namespace Prefix	Developer Console	API Version	Status	Size	Comments	Last Modified By
ApexClass	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg
ApexClass2	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg
ApexClass3	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg
ApexClass4	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg
ApexClass5	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg
ApexClass6	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg
ApexClass7	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg
ApexClass8	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg
ApexClass9	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg
ApexClass10	ApexClass	Active	64.0	Active	103		Alina.Froese.Pamg

- TourTriggerHandler: Handles the logic when a Tour record is inserted/updated.



```
1 public with sharing class TourTriggerHandler {
2     public static void handleAfterInsertOrUpdate(List<Tour__c> newTours) {
3         List<Tour__c> toursToProcess = new List<Tour__c>();
4         for (Tour__c tour : newTours) {
5             if (tour.Status__c == 'Interested' && tour.Tour_Confirmation_Sent__c == false) {
6                 toursToProcess.add(tour);
7             }
8         }
9         if (!toursToProcess.isEmpty()) {
10             TourService.processTours(toursToProcess);
11         }
12     }
13 }
```

- TourService: Processes Tours and enqueues jobs for sending confirmation emails.

```
1 public with sharing class TourService {
2     public static void processTours(List<Tour__c> tours) {
3         Set<Id> clientIds = new Set<Id>();
4         for (Tour__c t : tours) {
5             if (t.Client__c != null) clientIds.add(t.Client__c);
6         }
7         Map<Id, Client__c> clientMap = new Map<Id, Client__c>(
8             [SELECT Id, Email__c FROM Client__c WHERE Id IN :clientIds]
9         );
10        for (Tour__c t : tours) {
11            Client__c c = clientMap.get(t.Client__c);
12            if (c != null && String.isNotBlank(c.Email__c)) {
13                System.enqueueJob(new SendTourConfirmationQueueable(t.Id, c.Email__c));
14            }
15        }
16    }
17 }
```

- EmailService : Encapsulates the email-sending logic using Salesforce Email functionality.

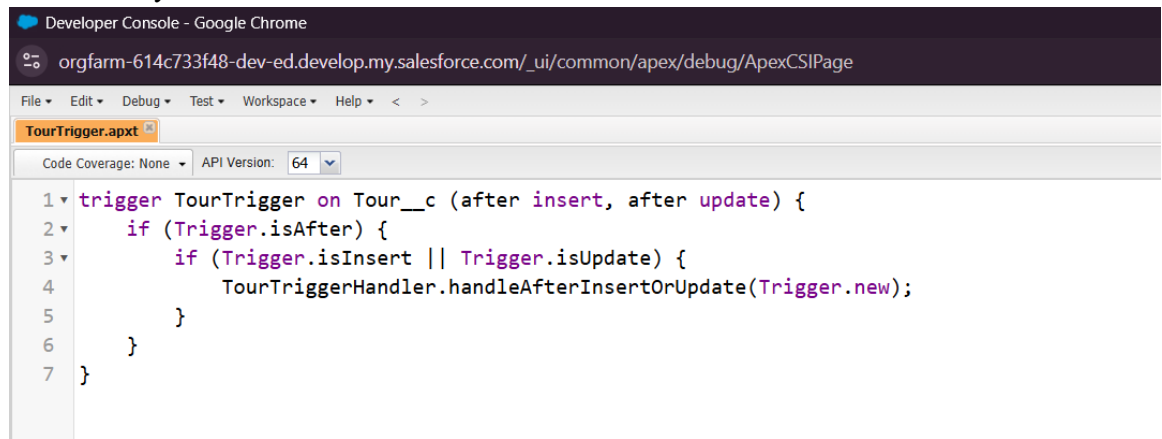
```
1 public with sharing class EmailService {
2     public static void sendTourConfirmation(String toAddress, Id tourId) {
3         Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
4         mail.setToAddresses(new String[] { toAddress });
5         try {
6             EmailTemplate template = [SELECT Id FROM EmailTemplate WHERE DeveloperName='Tour_Confirmation_Template' LIMIT 1];
7             mail.setTemplateId(template.Id);
8             mail.setTargetObjectId(UserInfo.getUserId());
9             mail.setWhatId(tourId);
10        } catch (Exception e) {
11            mail.setSubject('Tour Confirmation');
12            mail.setPlainTextBody('Thank you for your interest! Your tour has been confirmed.');
```

Apex Triggers (after insert/update)

Implemented a Trigger on Tour__c:

- Works on after insert and after update.
- Checks if the Status__c = Interested and Tour_Confirmation_Sent__c = false.

- Delegates the logic to TourTriggerHandler.
This ensures that whenever a Tour is marked as “Interested”, an email confirmation is automatically sent.



```

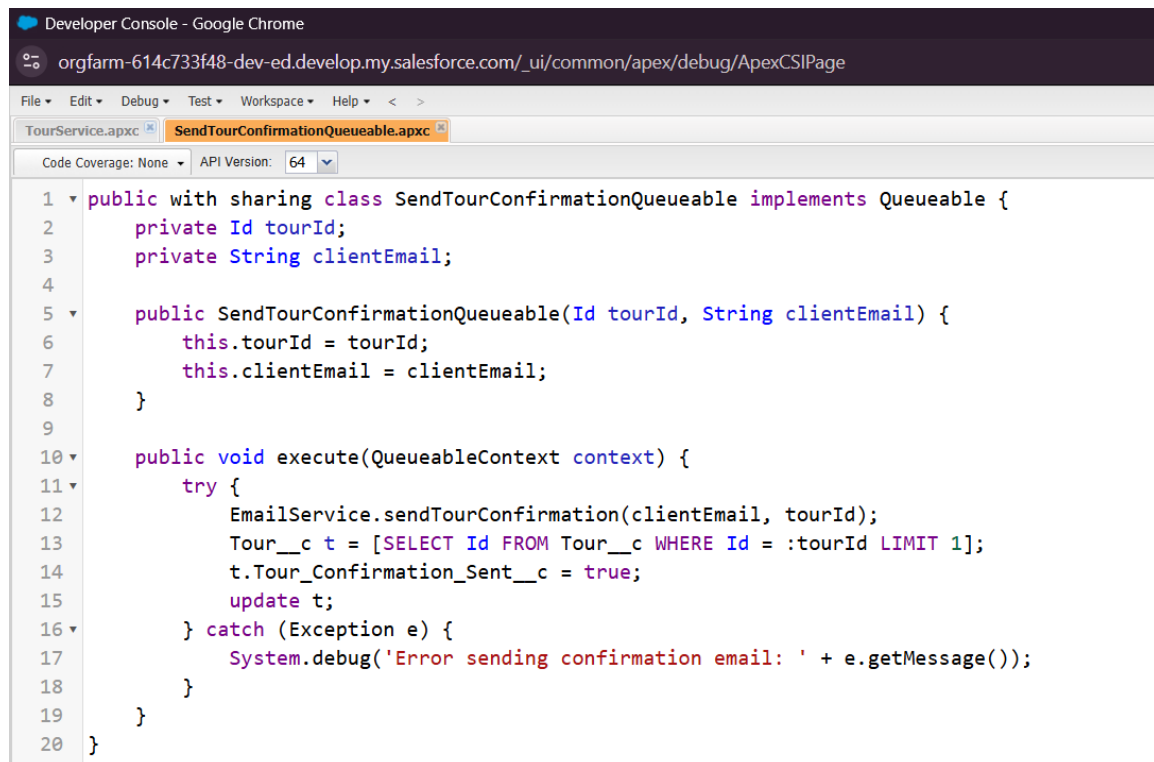
1 trigger TourTrigger on Tour__c (after insert, after update) {
2     if (Trigger.isAfter) {
3         if (Trigger.isInsert || Trigger.isUpdate) {
4             TourTriggerHandler.handleAfterInsertOrUpdate(Trigger.new);
5         }
6     }
7 }

```

Queueable Apex

Created SendTourConfirmationQueueable:

- Runs asynchronously using System.enqueueJob().
- Sends the email and updates the Tour_Confirmation_Sent__c field.
- Useful for long-running or bulk operations.



```

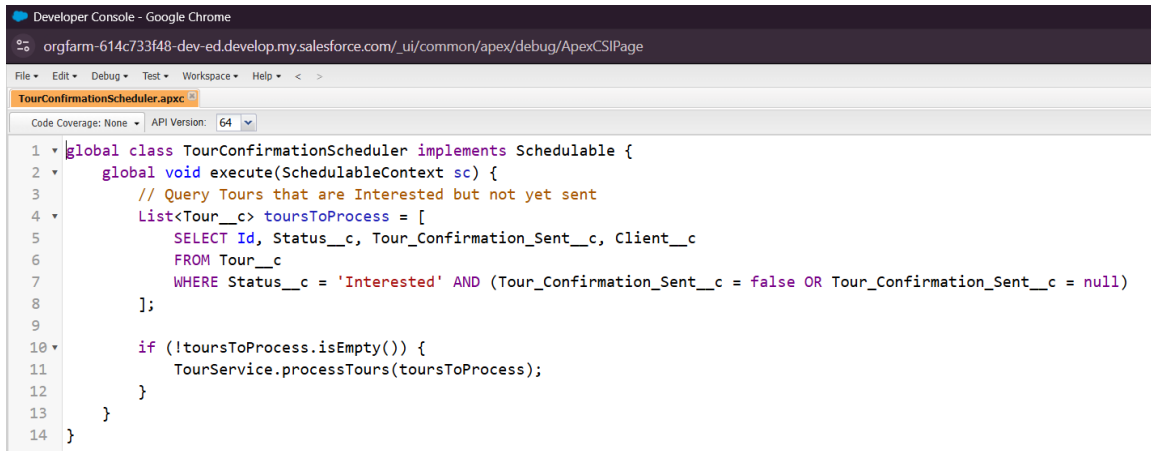
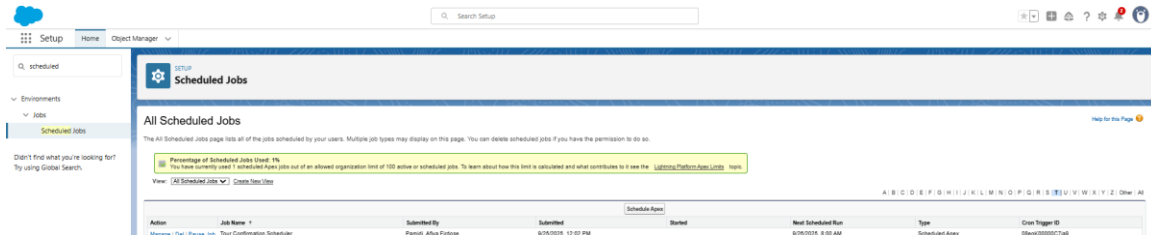
1 public with sharing class SendTourConfirmationQueueable implements Queueable {
2     private Id tourId;
3     private String clientEmail;
4
5     public SendTourConfirmationQueueable(Id tourId, String clientEmail) {
6         this.tourId = tourId;
7         this.clientEmail = clientEmail;
8     }
9
10    public void execute(QueueableContext context) {
11        try {
12            EmailService.sendTourConfirmation(clientEmail, tourId);
13            Tour__c t = [SELECT Id FROM Tour__c WHERE Id = :tourId LIMIT 1];
14            t.Tour_Confirmation_Sent__c = true;
15            update t;
16        } catch (Exception e) {
17            System.debug('Error sending confirmation email: ' + e.getMessage());
18        }
19    }
20 }

```

Scheduled Apex

Created TourConfirmationScheduler:

- Runs at a scheduled time (daily/weekly).
 - Finds all Tours where Status__c = Interested and Tour_Confirmation_Sent__c = false.
 - Calls TourService to process them automatically.
- This ensures no interested Tours are missed even if triggers fail.



Test Classes

Test classes were written to:

- Insert sample Tour__c and Client__c records.
- Simulate status updates (Not Interested → Interested).
- Verify that the trigger fires, emails are enqueued, and Tour_Confirmation_Sent__c is updated.
- Ensure code coverage is above Salesforce requirements (75%).

Developer Console - Google Chrome

orgfarm-614c733f48-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File • Edit • Debug • Test • Workspace • Help • < • >

TourProcesTest.apex

Code Coverage: None • API Version: 64

Run Test • Go To

```
1  @isTest
2  • public class TourProcesTest {
3
4      @isTest
5      static void testTourConfirmationFlow() {
6          client__c = new Client__c(Name='Test Client', Email__c='test@example.com');
7          insert c;
8
9          Tour__c t = new Tour__c(
10              Name='Test Tour',
11              Status__c='Interested',
12              Client__c=c.Id,
13              Tour_Confirmation_Sent__c=false
14          );
15          insert t;
16
17          Test.startTest();
18          t.Status__c = 'Interested';
19          update t;
20          Test.stopTest();
21
22          t = [SELECT Tour_Confirmation_Sent__c FROM Tour__c WHERE Id=:t.Id];
23          System.assertEquals(true, t.Tour_Confirmation_Sent__c);
24      }
25  }
```

Logs • Tests • Checkpoints • Query Editor • View State • Progress • Problems

User	Application	Operation	Time	Status	Read	Size
Allyia Firdose Permadi	Unknown	ApexTestHandler	9/26/2025, 2:58:24 PM	Success	Unread	46.95 KB