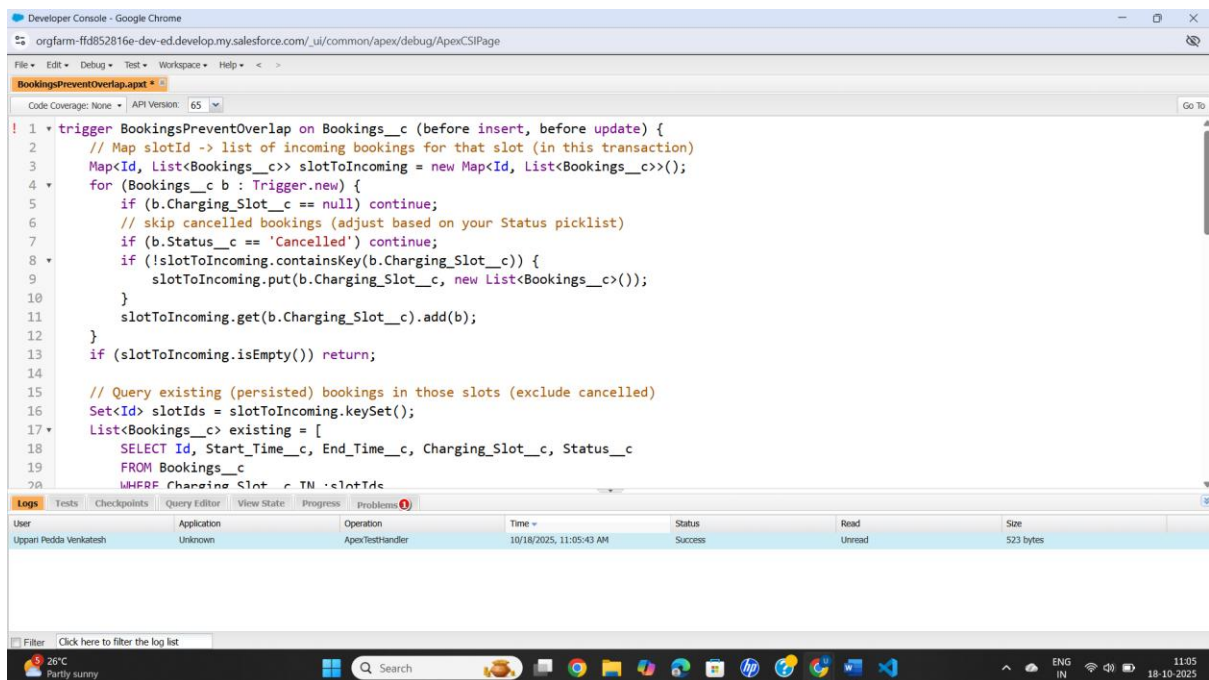


Phase 5: Apex Programming (Developer)

Apex code is used for implementing custom business logic that exceeds the capabilities of declarative tools like Flow or requires more complex operations, such as handling bulk records efficiently or preparing for future integrations.

Apex Trigger: BookingTrigger

- **Object:** Booking__c
- **Events:** after insert, after update
- **Purpose:** To listen for the creation or modification of Booking records and delegate the processing logic to a dedicated handler class (BookingTriggerHandler). This follows the best practice of keeping triggers logic-less.
- **Logic:**
 - On after insert, calls BookingTriggerHandler.handleAfterInsert(trigger.new).
 - On after update, calls BookingTriggerHandler.handleAfterUpdate(trigger.new, trigger.oldMap).



```
1 trigger BookingPreventOverlap on Booking__c (before insert, before update) {
2     // Map slotId -> list of incoming bookings for that slot (in this transaction)
3     Map<Id, List<Booking__c>> slotToIncoming = new Map<Id, List<Booking__c>>();
4     for (Booking__c b : Trigger.new) {
5         if (b.Charging_Slot__c == null) continue;
6         // skip cancelled bookings (adjust based on your Status picklist)
7         if (b.Status__c == 'Cancelled') continue;
8         if (!slotToIncoming.containsKey(b.Charging_Slot__c)) {
9             slotToIncoming.put(b.Charging_Slot__c, new List<Booking__c>());
10        }
11        slotToIncoming.get(b.Charging_Slot__c).add(b);
12    }
13    if (slotToIncoming.isEmpty()) return;
14
15    // Query existing (persisted) bookings in those slots (exclude cancelled)
16    Set<Id> slotIds = slotToIncoming.keySet();
17    List<Booking__c> existing = [
18        SELECT Id, Start_Time__c, End_Time__c, Charging_Slot__c, Status__c
19        FROM Booking__c
20        WHERE Charging_Slot__c IN :slotIds
21    ];
```

User	Application	Operation	Time	Status	Read	Size
Uppari Pedda Venkatesh	Unknown	ApexTestHandler	10/18/2025, 11:05:43 AM	Success	Unread	523 bytes

Apex Class: BookingTriggerHandler

This class contains the core logic executed by the BookingTrigger.

- **handleAfterInsert(List<Booking__c> newBookings) Method:**
 - **Purpose:** Updates the related Slot__c record's Availability__c to 'Unavailable' when a new Booking is confirmed.
 - **Logic:**
 1. Collects the Slot__c IDs from the newly inserted Booking__c records where Status__c is 'Confirmed' into a Set<Id> (to handle bulk records efficiently and avoid duplicates).
 2. Uses a single **SOQL query** *outside* any loops to retrieve all necessary Slot__c records based on the collected IDs.
 3. Iterates through the retrieved Slot__c records (**Collections**) and sets their Availability__c field to 'Unavailable'.
 4. Performs a single DML update operation on the list of modified Slot__c records *outside* the loop.
 5. Includes a **try-catch block (Exception Handling)** around the DML operation to log errors and potentially add errors back to the triggering Booking__c records if the slot update fails.
- **handleAfterUpdate(List<Booking__c> newBookings, Map<Id, Booking__c> oldMap) Method:**
 - **Purpose:** Handles updates to Booking records, specifically:
 - Updates the related Slot__c record's Availability__c to 'Unavailable' if the Booking Status__c changes to 'Confirmed'.
 - Updates the related Slot__c record's Availability__c to 'Available' if the Booking Status__c changes to 'Cancelled'.

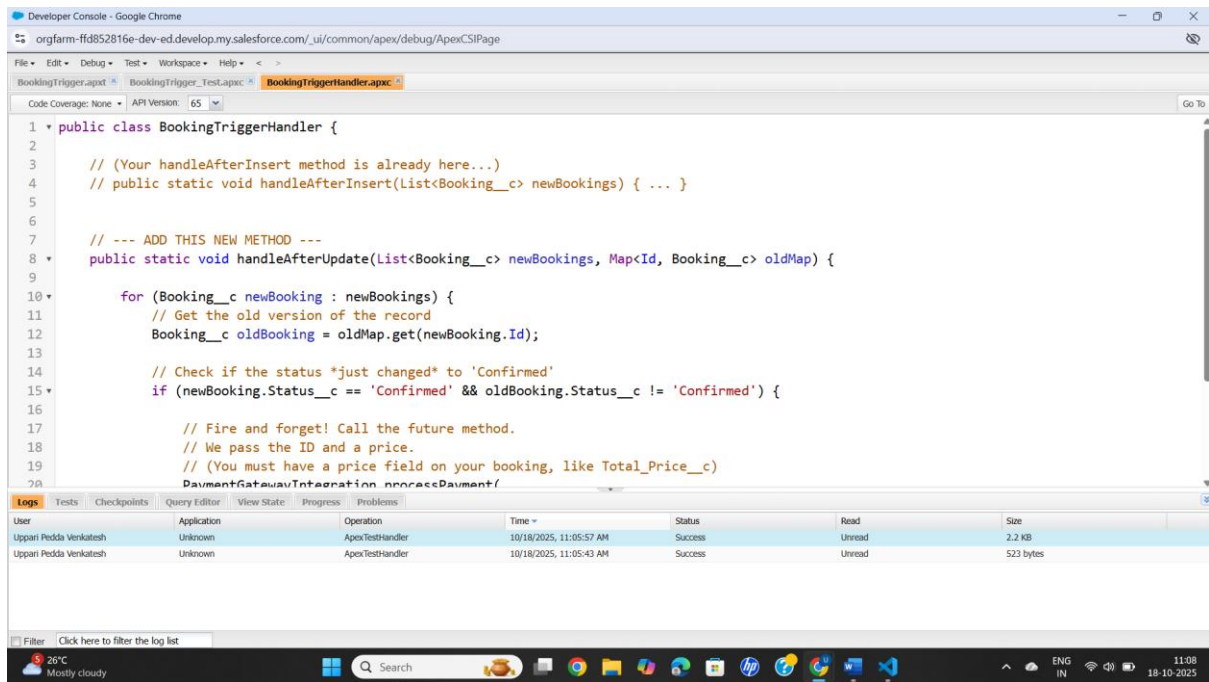
- Initiates the payment processing callout (Phase 7) if the Booking Status__c changes to 'Confirmed'.

- **Logic:**

1. Iterates through the newBookings list.
2. Compares the Status__c in the newBooking record with the corresponding record in oldMap to detect relevant status changes.
3. Collects Slot__c IDs that need to be marked as 'Unavailable' or 'Available' into separate Set<Id> collections.
4. If a status changed to 'Confirmed', calls the future method
PaymentGatewayIntegration.processPayment(...),
passing the Booking ID and Total_Price__c.
5. After the loop, performs SOQL queries (if necessary) and bulk DML update operations on the collected Slot__c records to update their availability. Uses try-catch blocks for DML.

- **calculatePayment(Decimal hours, Decimal rate) Utility Method:**

- **Purpose:** Provides a reusable static method to calculate payment amounts based on duration and rate. Can be called from other Apex classes or potentially Lightning Components.
- **Logic:** Takes hours and rate as input, performs basic validation (checks for null or negative values), calculates hours * rate, sets the scale to 2 decimal places, and returns the result.



Test Classes: BookingTrigger_Test

- **Purpose:** To verify the functionality of the Apex trigger and handler class, ensuring they meet business requirements and achieve the minimum 75% code coverage required for deployment.
- **Key Components:**
 - **@isTest Annotation:** Marks the class and methods as test code, isolated from production data.
 - **@testSetup Method:** Creates common test data (e.g., Accounts, Contacts, Stations, Slots) once, making it available for all test methods efficiently.
 - **Test Methods (e.g., testAfterInsertTrigger_UpdatesSlot, testAfterUpdate_ConfirmUpdatesSlot, testAfterUpdate_CancelFreesSlot, testPaymentCalculator):** Each method tests a specific scenario or unit of logic.
 - **Arrange:** Sets up specific data for the test scenario (e.g., create a Booking with 'Pending' status).
 - **Act:** Performs the action that invokes the code being tested (e.g., inserting a new confirmed Booking, updating a Booking's status). Uses `Test.startTest()` and

Test.stopTest() around the action to reset governor limits.

- **Assert:** Uses System.assertEquals(expected, actual) or System.assertNotEquals() etc., to verify that the outcome matches the expected result.

User	Application	Operation	Time	Status	Read	Size
Uppari Pedda Venkatesh	Unknown	ApexTestHandler	10/18/2025, 11:10:29 AM	Success	Unread	2.19 KB
Uppari Pedda Venkatesh	Unknown	ApexTestHandler	10/18/2025, 11:10:29 AM	Success	Unread	523 bytes
Uppari Pedda Venkatesh	Unknown	ApexTestHandler	10/18/2025, 11:05:57 AM	Success	Unread	2.2 KB
Uppari Pedda Venkatesh	Unknown	ApexTestHandler	10/18/2025, 11:05:43 AM	Success	Unread	523 bytes

Filter [Click here to filter the log list](#)