Phase 5 – Apex Programming (Developer)

1. Introduction

This phase adds custom business logic using Apex. It helps when point-and-click automation (Flows/Validation Rules) is not enough.

2. Apex Trigger – Mark Reservation as "Missed"

Steps:

1. Go to Setup → Developer Console (or VS Code with SFDX).

2. Create a new Apex Trigger: Name = ReservationTrigger, Object = Reservation__c.

3. Example code:

```
trigger ReservationTrigger on Reservation__c (after insert, after
update) {
  for (Reservation__c res : Trigger.new) {
    if (res.Status__c == 'Confirmed' && res.Date__c < Date.today() &&
res.Check_In__c == false) {
      res.Status__c = 'Missed';
    }
  }
}
```

4. Save and test by creating a past reservation record without check-in.

3. Batch Apex – Monthly Revenue Reports

Steps:

1. Go to Setup → Apex Classes → New.

2. Create a class:

```
global class MonthlyRevenueBatch implements
Database.Batchable<SObject> {
  global Database.QueryLocator start(Database.BatchableContext
bc) {
    return Database.getQueryLocator('SELECT Id, Amount__c FROM
```

```
Bill__c WHERE Payment_Status__c = \'Paid\'');
  }
  global void execute(Database.BatchableContext bc, List<Bill__c>
bills) {
    Decimal total = 0;
    for(Bill__c b : bills) {
      total += b.Amount__c;
    }
    System.debug('Monthly Revenue = ' + total);
  }
  global void finish(Database.BatchableContext bc) {
    // send report via email or update custom object
  }
}
```

3. Run batch using: Database.executeBatch(new MonthlyRevenueBatch(), 200);

4. Queueable Apex – Bulk Update Loyalty Points

Steps:

1. Go to Setup → Apex Classes → New.

2. Example code:

```
public class LoyaltyPointsJob implements Queueable {
  public void execute(QueueableContext qc) {
    List<Customer__c> customers = [SELECT Id, Loyalty_Points__c
FROM Customer__c];
    for(Customer__c c : customers) {
      c.Loyalty_Points__c = (c.Loyalty_Points__c == null ? 0 :
c.Loyalty_Points__c) + 10;
    }
    update customers;
  }
}
```

3. Run job with: System.enqueueJob(new LoyaltyPointsJob());

5. Exception Handling

To prevent failures from stopping automation.

```
try {
   update customers;
} catch(Exception e) {
   System.debug('Error: ' + e.getMessage());
}
```

---

6. Test Classes (Deployment Requirement)

Steps:

1. Setup → Apex Classes → New → Name = ReservationTriggerTest.

2. Example code:

```
@isTest
private class ReservationTriggerTest {
   @isTest static void testMissedReservation() {
      Reservation__c r = new Reservation__c(
         Table_Number__c = 1,
         Date__c = Date.today().addDays(-1),
         Status__c = 'Confirmed',
         Check_In__c = false
      );
      insert r;

      Test.startTest();
      update r;
      Test.stopTest();

      Reservation__c res = [SELECT Status__c FROM Reservation__c
WHERE Id = :r.Id];
      System.assertEquals('Missed', res.Status__c);
   }
}
```

---

7. Phase 5 Conclusion

With Apex Programming:
- Reservations are auto-marked as Missed if no check-in.
- Monthly revenue can be calculated using Batch Apex.
- Customers can earn loyalty points via Queueable Apex.
- Test Classes ensure safe deployment.