Phase 8: Data Management & Deployment – (Online Laptop Booking)

1. Data Import Wizard / Data Loader

Both tools are used for **bulk importing and updating records** in Salesforce.

a. Data Import Wizard

- **Purpose:** Simple, user-friendly tool for importing standard and custom objects.
- **Project Use Case:**
    - Bulk import **Laptop records** with fields like Model, Brand, Price, Stock.
    - Import **Laptop Dealers** with Name, Contact, Location.
    - Import **Customers** with Name, Email, Contact, Address.
- **Features:**
    - Supports mapping CSV columns to Salesforce fields
    - Can prevent duplicates if configured

b. Data Loader

- **Purpose:** Advanced tool for **large-volume data operations** (>50,000 records) and updates.
- **Project Use Case:**
    - Update **Laptop Stock** after testing or service
    - Mass updates to **order statuses** or dealer information
- **Features:**
    - Can perform insert, update, upsert, delete, and export operations
    - Handles large datasets efficiently

Impact:

- Ensures **fast onboarding of existing data** into Salesforce
- Reduces manual entry and **errors from typing or copy-paste**

2. Duplicate Rules

- **Purpose:** Prevents creation of duplicate records in Salesforce.
- **Project Use Case:**
    - Prevent duplicate **Customer records** (matching Email or Contact Number)
    - Prevent duplicate **Orders** (matching Order Number or Customer + Order Date)

- **Implementation:**
    - Defined **matching rules** for fields like Email, Phone, or Order Number
    - Configured duplicate rules to **block or alert users** during record creation
- **Impact:**
    - Maintains **clean, accurate data**
    - Prevents confusion or errors in **order fulfillment, reporting, and customer communication**

3. Data Export & Backup

- **Purpose:** Regular backup of Salesforce data to prevent data loss.
- **Project Implementation:**
    - Scheduled **weekly exports** of all custom objects: Laptop, Dealer, Customer, Order, Testing, Service Testing
    - Exported in CSV format for **offline storage and recovery**
- **Impact:**
    - Provides **disaster recovery** in case of accidental deletion or system failure
    - Supports **auditing and compliance** by maintaining historical snapshots

4. Deployment: Change Sets / SFDX / VS Code

Deploying from Sandbox (development environment) to Production ensures **all tested configurations and customizations are moved safely**.

a. Change Sets

- **Purpose:** Salesforce's native tool for deploying metadata.
- **Use Case:**
    - Move custom objects, fields, flows, triggers, LWCs, and page layouts from **Sandbox to Production**
- **Process:**

1. Create **Outbound Change Set** in Sandbox
2. Add components (objects, triggers, flows, LWCs)
3. Upload to Production and deploy

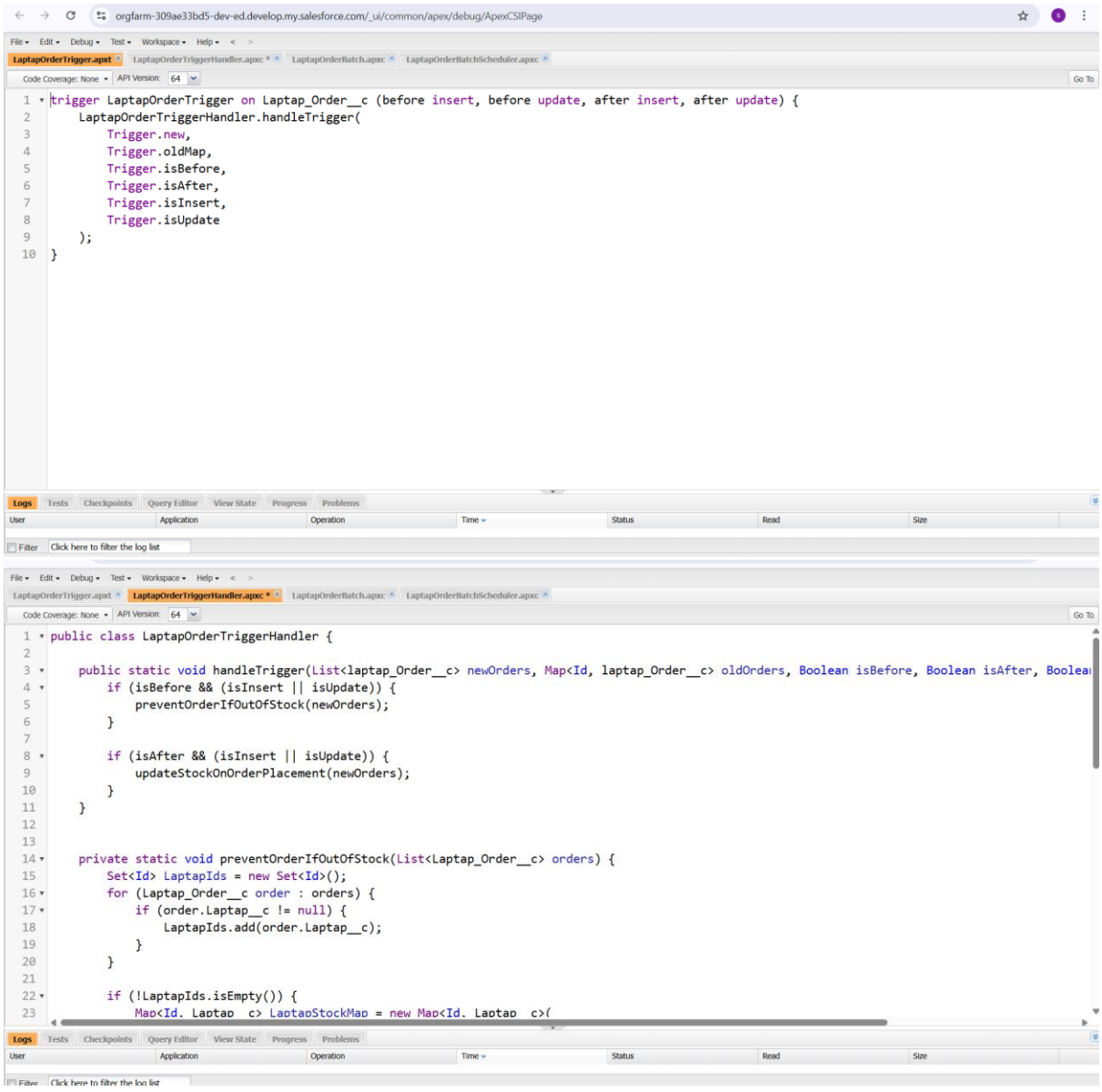
b. SFDX / VS Code

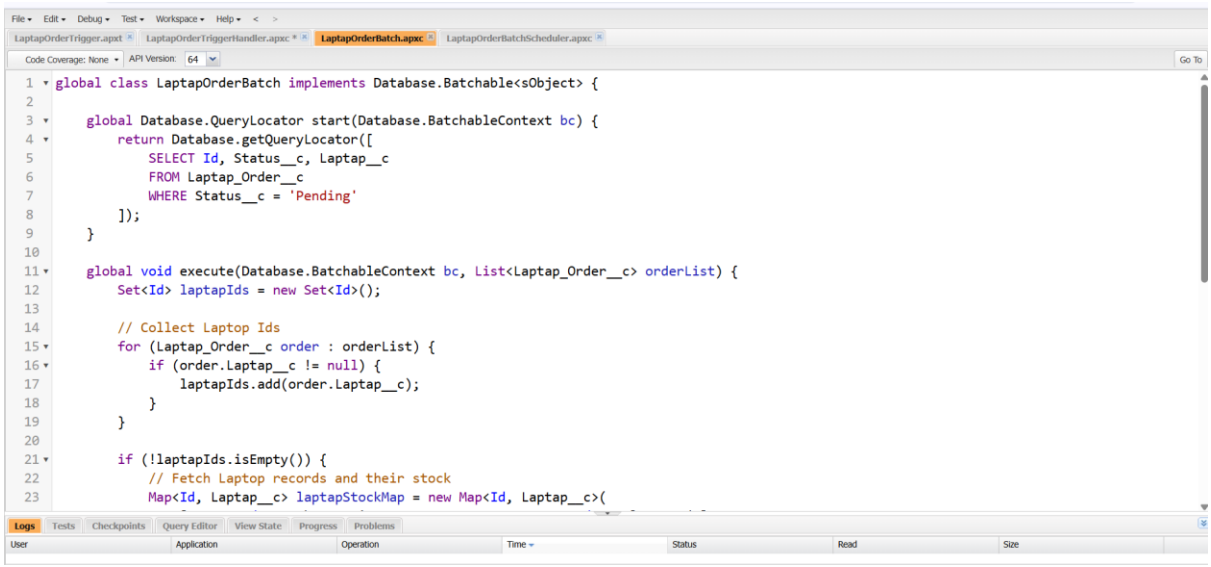
- **Purpose:** Modern, developer-friendly deployment with **version control**.

- **Use Case:**
    - Track changes to Apex classes, triggers, LWCs, and metadata
    - Deploy using **source control (Git)** for team collaboration
- **Advantages:**
    - Supports **continuous integration/continuous deployment (CI/CD)**
    - Enables **rollback** and better management of complex deployments

Impact:

- Ensures **consistent, error-free deployment**
- Maintains **version control** and historical record of changes
- Reduces **risk of production downtime**





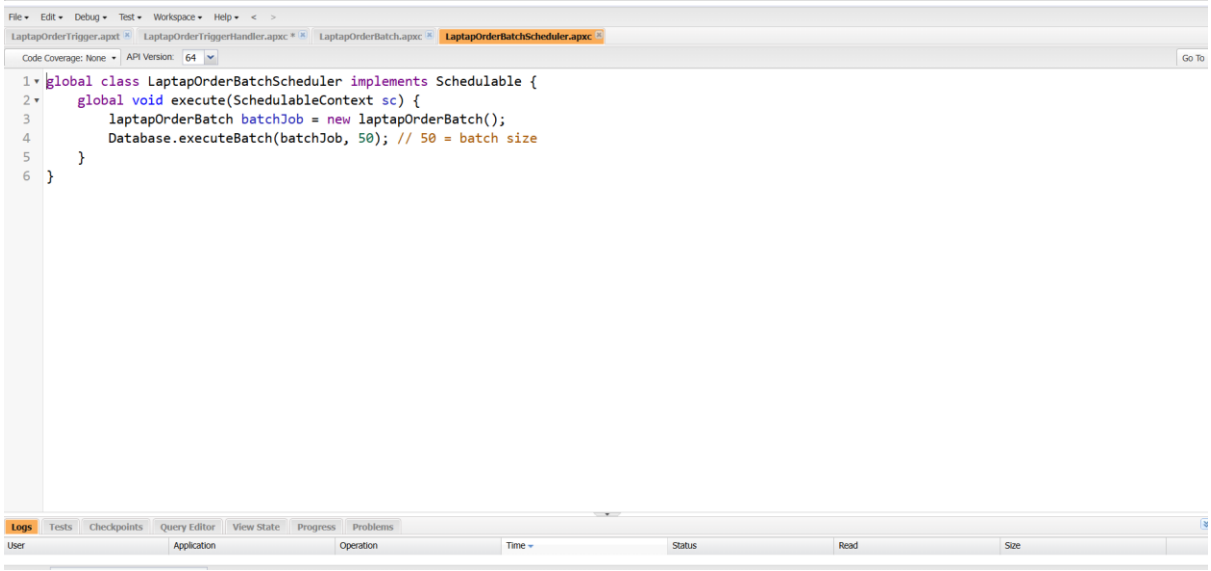
```
global class LaptapOrderBatch implements Database.Batchable<sObject> {

    global Database.QueryLocator start(Database.BatchableContext bc) {
        return Database.getQueryLocator([
            SELECT Id, Status__c, Laptap__c
            FROM Laptap_Order__c
            WHERE Status__c = 'Pending'
        ]);
    }

    global void execute(Database.BatchableContext bc, List<Laptap_Order__c> orderList) {
        Set<Id> laptapIds = new Set<Id>();

        // Collect Laptop Ids
        for (Laptap_Order__c order : orderList) {
            if (order.Laptap__c != null) {
                laptapIds.add(order.Laptap__c);
            }
        }

        if (!laptapIds.isEmpty()) {
            // Fetch Laptop records and their stock
            Map<Id, Laptap__c> laptapStockMap = new Map<Id, Laptap__c>(
```



```
global class LaptapOrderBatchScheduler implements Schedulable {
    global void execute(SchedulableContext sc) {
        laptapOrderBatch batchJob = new laptapOrderBatch();
        Database.executeBatch(batchJob, 50); // 50 = batch size
    }
}
```

Summary of Phase 8

1. **Data Import Wizard / Data Loader:** Bulk import and update laptops, dealers, and customers efficiently.
2. **Duplicate Rules:** Prevent duplicate customer and order records to maintain data integrity.
3. **Data Export & Backup:** Scheduled exports ensure disaster recovery and compliance.
4. **Change Sets / SFDX / VS Code:** Deploy changes safely from Sandbox to Production, supporting version control and team collaboration.