Phase 5: Apex Programming (Developer)

Apex is Salesforce's object-oriented programming language that allows developers to write complex business logic and automate processes that go beyond declarative tools (like Flows or Process Builder). It runs on the Salesforce platform and helps implement customized, scalable, and efficient solutions.

1️⃣ Apex Classes & Utility Classes

Purpose:

- Create reusable code to handle business logic.
- Automate notifications, calculations, or custom logic.

Use Cases:

- Create a utility class to calculate recycled product cost.
- Build classes for automated notifications or custom business logic.

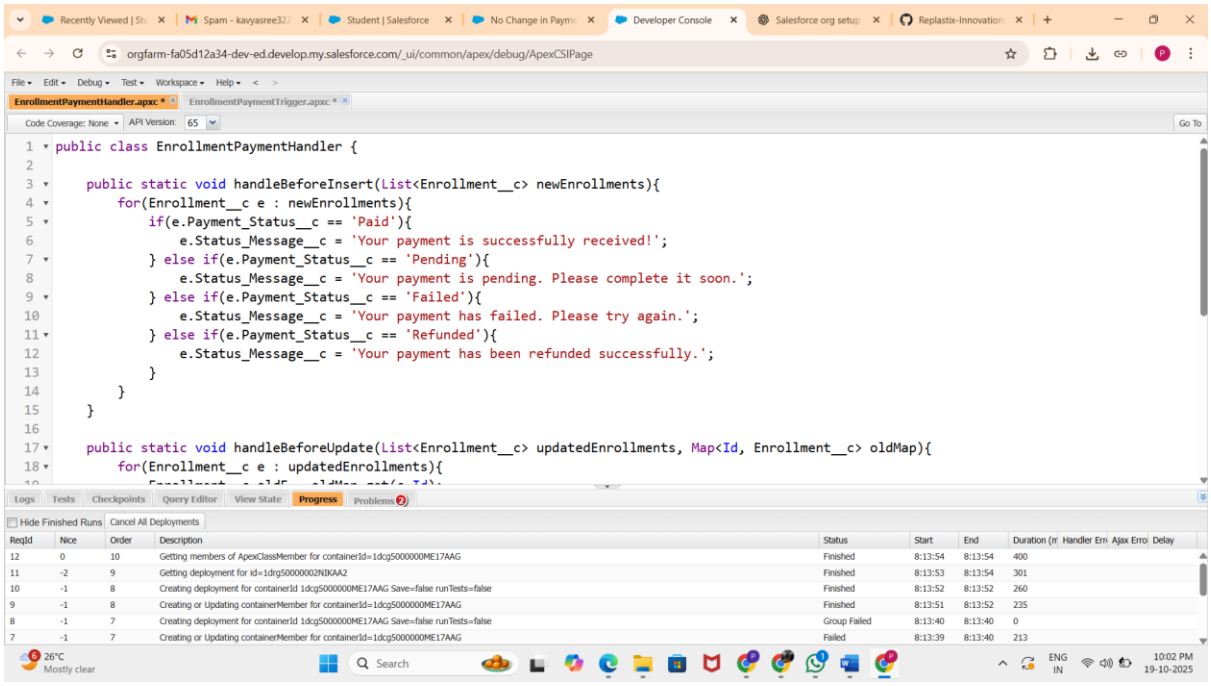
2️⃣ Apex Triggers

Purpose:

- Automatically execute Apex code when records are created, updated, deleted, or undeleted.

Trigger Types:

Type	Purpose	Example Use Case
Before Insert/Update	Validate or modify records before saving	Ensure total waste collected is correct before insert
After Insert/Update/Delete	Execute logic after records are saved	Notify recycling center when order status changes



3 Trigger Design Pattern

Purpose:

- Maintain clean, reusable, and scalable trigger logic.

Best Practices:

- Only one trigger per object.
- Delegate logic to handler classes.
- Ensure triggers are bulkified (handle multiple records efficiently).

Use Case:

- Maintain organized logic for Plastic Waste automation through a central handler class.

4 SOQL & SOSL

Purpose:

- SOQL: Fetch data from one or more Salesforce objects.
- SOSL: Search data across multiple objects and fields.

Use Cases:

- Retrieve all orders placed by a specific customer (SOQL).
- Search for a keyword in multiple objects (SOSL).

5️⃣ Collections

Purpose:

- Store and manage multiple data values efficiently.

Collection	Description	Example Use Case
List	Ordered collection of elements	List of Orders or Waste records
Set	Unordered, unique elements	Unique Waste Types
Map	Key-value pairs	Map of Customer ID → Order List

6️⃣ Control Statements

Purpose:

- Execute code conditionally or repeatedly.

Common Statements:

- Conditional: if, else, switch
- Loops: for, while, do-while
- Others: break, continue

7️⃣ Asynchronous Processing

Purpose:

- Perform background tasks that don't block the main execution thread.

Types & Use Cases:

Type	Purpose	Example Use Case
Batch Apex	Process large datasets asynchronously	Update thousands of recycled product records nightly
Queueable Apex	Run async operations with more complex logic	Send confirmation emails after order approval
Scheduled Apex	Run Apex code at defined times	Generate weekly recycled product reports

Type	Purpose	Example Use Case
Future Methods	Execute code asynchronously, useful for callouts	Notify customers about shipment updates

8️⃣ Exception Handling

Purpose:

- Catch and handle errors gracefully to prevent transaction failures.

Use Cases:

- Handle record insert/update exceptions.
- Log errors for admin review.

9️⃣ Test Classes

Purpose:

- Validate Apex code and ensure proper functionality before deployment.
- Salesforce requires at least 75% code coverage for production deployment.

Use Cases:

- Test triggers, classes, and async logic.
- Simulate user actions and validate outcomes.