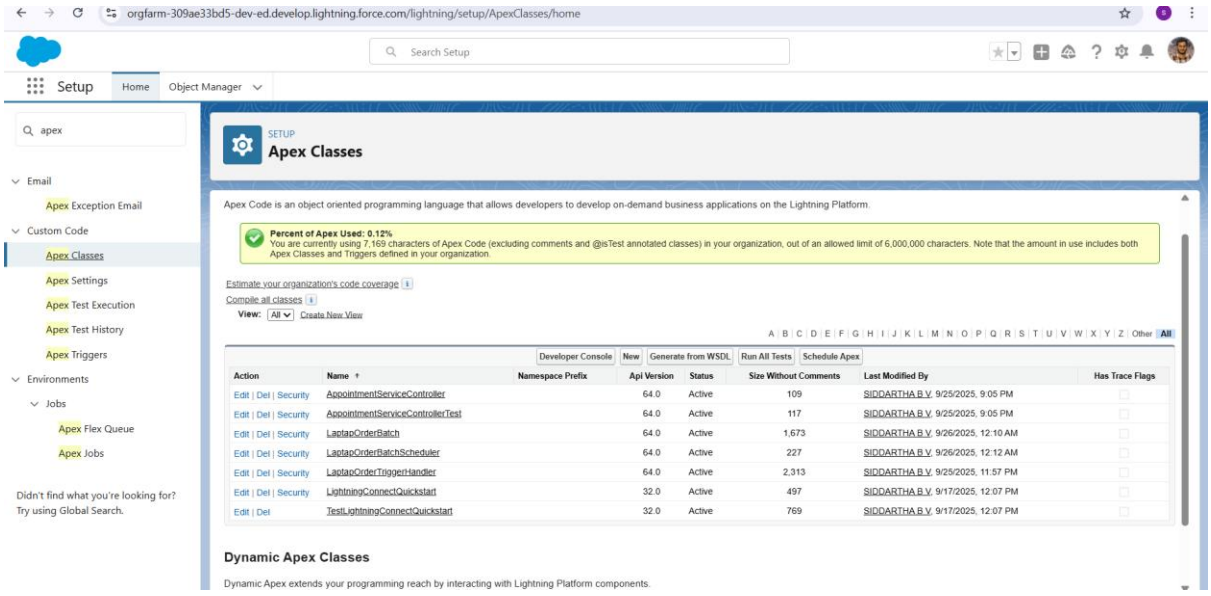
Phase 5: Apex Programming (Online Laptop Booking)

Apex is Salesforce's **proprietary programming language**, used when **declarative tools (Flows, Workflow Rules) are not sufficient**. For your laptop ordering system, Apex is used to handle **complex business logic, automation, and asynchronous processing**.

1. Classes & Objects

- **Example:** LaptopOrderHandler.cls
- **Purpose:** Encapsulates the logic for laptop orders, such as:
    - Validating order data
    - Updating laptop stock
    - Triggering related service/testing processes
- **Why it matters:**
    - Promotes **reusability** – the same logic can be called from **triggers, batch jobs, or flows**.
    - Keeps code **organized and maintainable**.



2. Triggers

Triggers are **pieces of Apex code that execute automatically before or after DML operations (Insert, Update, Delete, Undelete)**.

Project Examples:

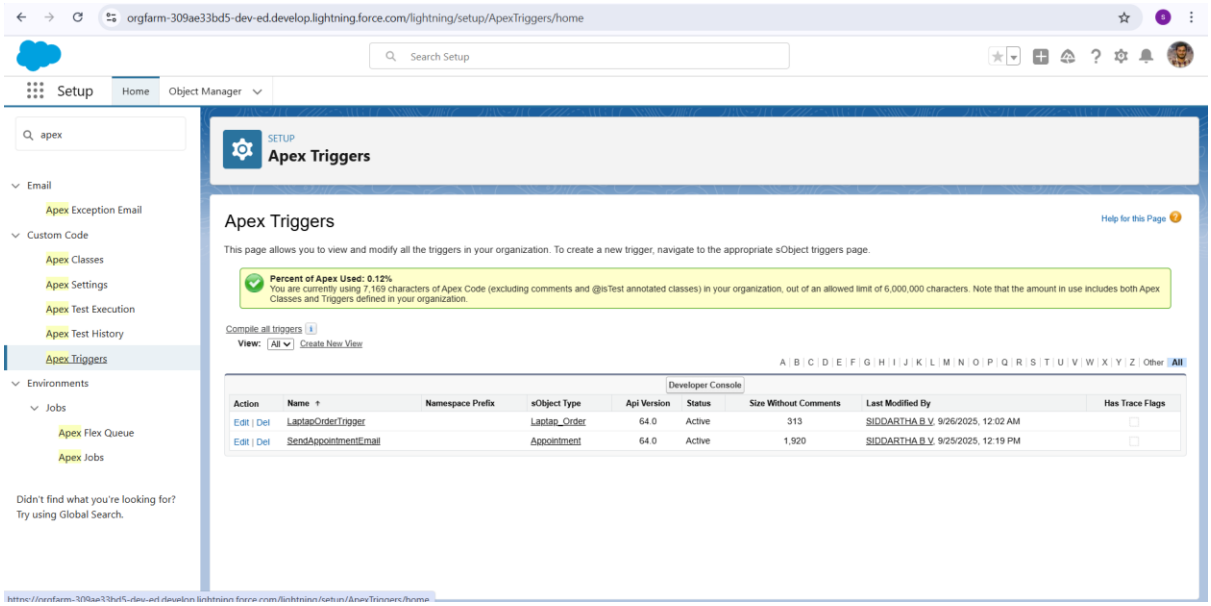
1. **LaptopOrderTrigger**

- **Purpose:** Runs before insert/update of orders
- **Logic:**
  - Validate order quantity
  - Update available laptop stock
  - Prevent orders if stock is insufficient
- **Type:** before insert / before update

2. **LaptopTestingTrigger**
   - **Purpose:** Runs after insert/update of testing or service records
   - **Logic:**
     - Log testing results
     - Update related order status
   - **Type:** after insert / after update

Impact:

- Ensures **data integrity** and **automates business rules** without manual intervention.
- Triggers enable **real-time updates** in your system.



3. Collections: Lists, Sets, and Maps

- **Purpose:** Handle multiple records efficiently.
- **Usage Examples:**
  - **List:** Store multiple orders to update in bulk

    - **Set:** Track unique customer IDs to avoid duplicates
    - **Map:** Map Order ID → Laptop for easy lookup during processing
- **Why it matters:**
    - Supports **bulk processing**, essential to avoid Salesforce governor limits.
    - Improves **performance and code maintainability**.

4. Batch Apex / Queueable / Scheduled Apex

Used for asynchronous or large-scale processing.

1. **Batch Apex:**
    - Example: Update all pending laptop testing records nightly
    - Splits records into manageable batches to process large volumes
2. **Queueable Apex:**
    - Example: Process complex order logic asynchronously after a trigger
    - More flexible than future methods; supports chaining
3. **Scheduled Apex:**
    - Example: Nightly batch job to check pending orders or testing records
    - Can schedule jobs at specific times or intervals

Impact:

- Handles **large datasets without hitting governor limits**
- Automates **recurring processes** (like updating stock or checking testing status)

Summary of Phase 5

- **Classes & Objects:** Centralize business logic for reusability
- **Triggers:** Automate real-time actions during record insert/update
- **SOQL & SOSL:** Efficiently retrieve Salesforce data
- **Collections:** Handle bulk operations while avoiding governor limits
- **Batch / Queueable / Scheduled Apex:** Automate large-volume or recurring processes