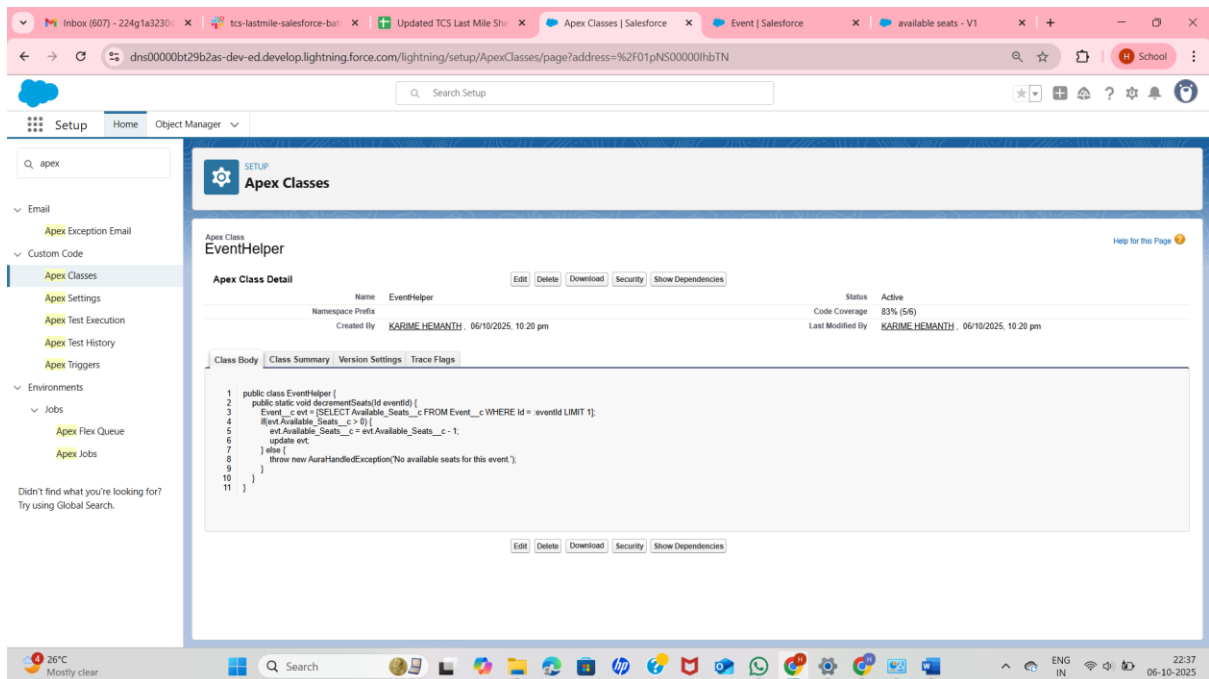


Phase 5: Apex Programming Developer – MAX FIT

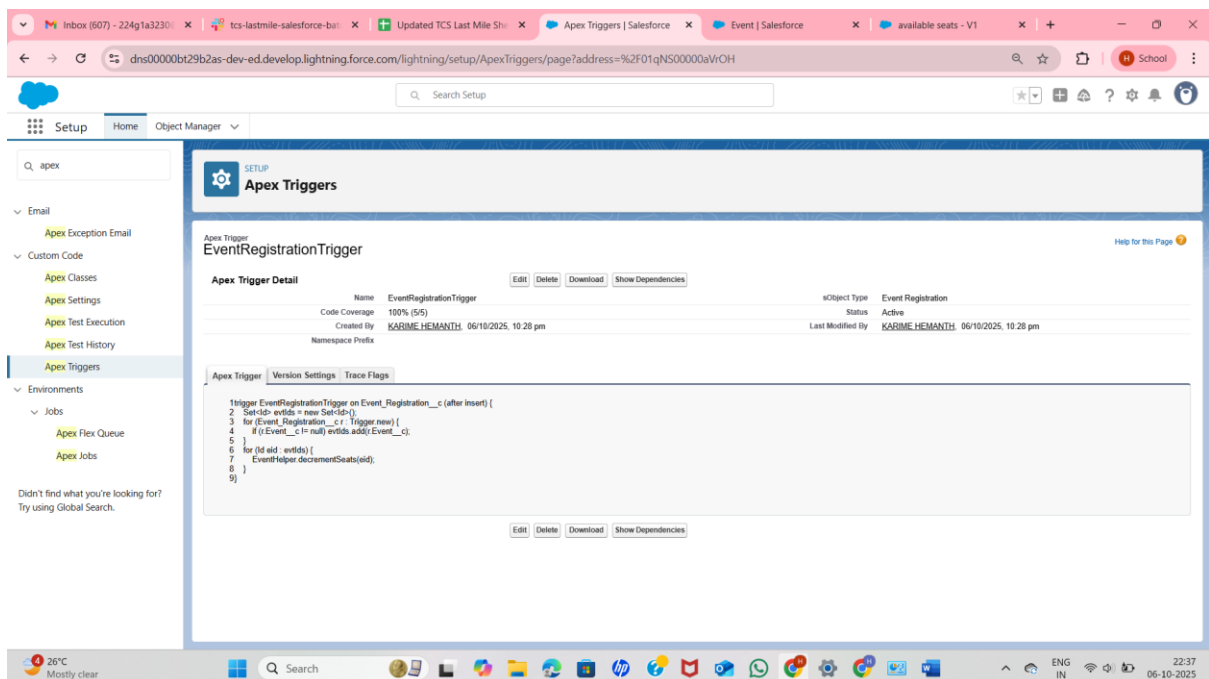
App Apex Classes: Custom classes for MAX FIT help calculate real-time attendance, generate participant statistics after each fitness session, and automate sending reminders or notifications to organizers and attendees. Business logic is modular and easy to extend for new fitness features.



The screenshot shows the Salesforce Setup interface for Apex Classes. The left sidebar contains navigation links for Email, Custom Code, Apex Classes, Apex Settings, Apex Test Execution, Apex Test History, Apex Triggers, Environments, Jobs, Apex Flex Queue, and Apex Jobs. The main content area displays the details for the 'EventHelper' Apex Class. The class is active, created by KARIME HEMANTH on 06/10/2025, 10:20 pm, and has a code coverage of 83% (5/6). The class body is shown in a code editor with the following code:

```
1 public class EventHelper {
2     public static void decrementSeats(Id eventId) {
3         Event__c evt = (SELECT Available_Seats__c FROM Event__c WHERE Id = :eventId LIMIT 1);
4         if (evt.Available_Seats__c > 0) {
5             evt.Available_Seats__c = evt.Available_Seats__c - 1;
6             update evt;
7         }
8         throw new AuraHandledException('No available seats for this event. ');
9     }
10 }
11 }
```

Apex Triggers: Registration triggers check for duplicate entries and instantly update seat counts, keeping classes at safe capacity. Triggers also enforce attendance limits and send confirmation emails to new registrants, providing a seamless experience for users.



The screenshot shows the Salesforce Setup interface for Apex Triggers. The left sidebar contains navigation links for Email, Custom Code, Apex Classes, Apex Settings, Apex Test Execution, Apex Test History, Apex Triggers, Environments, Jobs, Apex Flex Queue, and Apex Jobs. The main content area displays the details for the 'EventRegistrationTrigger' Apex Trigger. The trigger is active, created by KARIME HEMANTH on 06/10/2025, 10:28 pm, and has a code coverage of 100% (5/5). The trigger body is shown in a code editor with the following code:

```
1 trigger EventRegistrationTrigger on Event_Registration__c (after insert) {
2     Set<Id> evIds = new Set<Id>();
3     for (Event_Registration__c r : Trigger.new) {
4         if (r.Event__c != null) evIds.add(r.Event__c);
5     }
6     for (Id id : evIds) {
7         EventHelper decrementSeats(id);
8     }
9 }
```

Batch/Scheduled Jobs: Scheduled batch jobs automatically refresh event attendance metrics and dispatch daily messages to organizers about upcoming fitness sessions. This keeps app data current and engagement high for every workout event.

Testing: Every Apex logic and automation is strongly covered by test classes with positive/negative scenarios to guarantee stable releases and predictable behavior. Exception handling in both triggers and classes maintains robust, error-free operation even as data grows.

