Phase 5: Apex Programming (Developer)

Goal of this Phase

The goal of Phase 5 was to implement **Apex programming features** in Salesforce to support the Student Management System. This included creating **classes, triggers, SOQL/SOSL queries, collections, asynchronous processing, exception handling, and test classes** to ensure the application is efficient, scalable, and reliable.

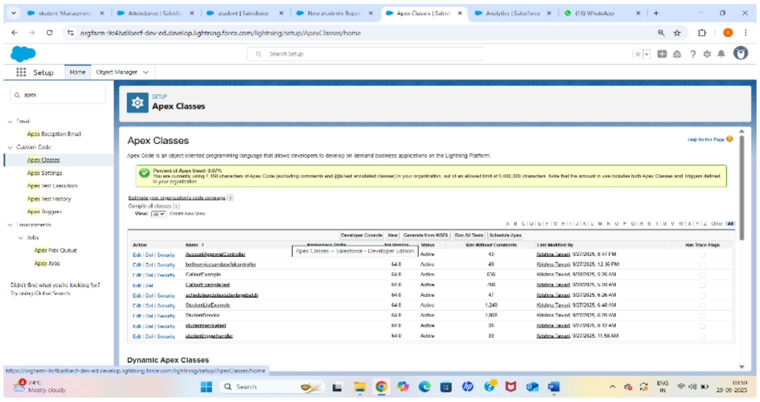
---

1. Classes & Objects

- Created **StudentService** Apex class to handle business logic.
- Added methods for calculating student age based on Date of Birth and updating records.

2. Apex Triggers (Before/After Insert/Update/Delete)

- Implemented a **StudentTrigger** to:
  - Calculate Age before insert/update when Date of Birth is set.
  - Prevent invalid updates using before update triggers.
  - Maintain consistency when student records are deleted.



Apex Classes

Apex Code is an object oriented programming language that allows developers to develop on demand business applications on the Lightning Platform.

Percent of Apex Used: 0.07%
You are currently using 4,150 characters of Apex Code (excluding comments and @isTest annotated classes) in your organization, out of an allowed limit of 6,000,000 characters. Note that the amount in use includes both Apex Classes and Triggers defined in your organization.

Action	Name ↑	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By	Has Trace Flags
Edit | Del | Security	AccountApprovalController			Active	43	Krishna Tiwari, 9/27/2025, 8:47 PM	
Edit | Del | Security	batchservicesandusefulcontroller		64.0	Active	49	Krishna Tiwari, 9/27/2025, 12:36 PM	
Edit | Del | Security	CalloutExample		64.0	Active	636	Krishna Tiwari, 9/28/2025, 6:26 AM	
Edit | Del	CalloutExampleTest		64.0	Active	786	Krishna Tiwari, 9/28/2025, 5:28 AM	
Edit | Del | Security	scheduleupdatestudentagebatch		64.0	Active	47	Krishna Tiwari, 9/27/2025, 6:26 AM	
Edit | Del | Security	StudentListExample		64.0	Active	1,249	Krishna Tiwari, 9/27/2025, 6:48 AM	
Edit | Del | Security	StudentService		64.0	Active	1,068	Krishna Tiwari, 9/27/2025, 6:20 AM	
Edit | Del | Security	studentservicetest		64.0	Active	35	Krishna Tiwari, 9/27/2025, 6:12 AM	
Edit | Del | Security	studenttriggerhandler		64.0	Active	35	Krishna Tiwari, 9/27/2025, 11:58 AM	

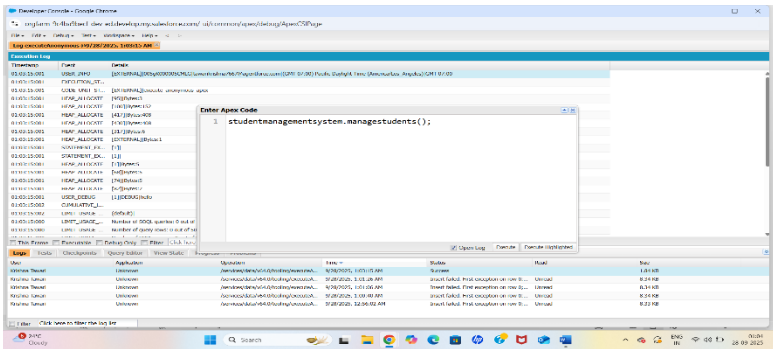
Dynamic Apex Classes

3. Trigger Design Pattern

- Applied **Handler Class Pattern**:
- Trigger delegates logic to StudentTriggerHandler class.
- Ensures clean separation of logic, reusable code, and easier maintenance.

4. SOQL & SOSL

- Used **SOQL queries** to fetch Student records with fields like Id, Date_of_Birth__c, and Age__c.
- Demonstrated **SOSL** for searching students by name across multiple fields.



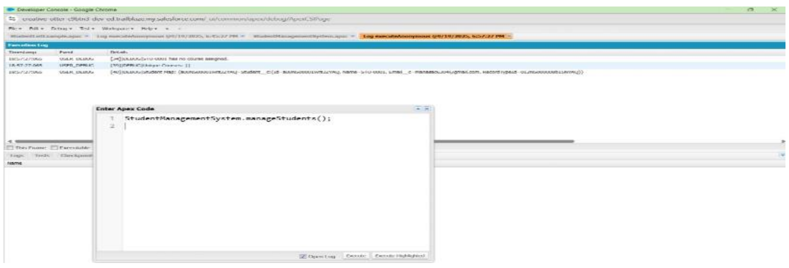
```
Developer Console - Google Chrome
Log executeAnonymous @9/28/2025, 1:03:15 AM

Execution Log
Timestamp      Event               Details
01:03:15:001   USER_INFO           [EXTERNAL]|005gK00000SCHLC|[illegible]|(GMT-07:00) Pacific Daylight Time (America/Los_Angeles)|GMT-07:00
01:03:15:001   EXECUTION_ST...
01:03:15:001   CODE_UNIT_ST...     [EXTERNAL]|execute_anonymous_apex
01:03:15:001   HEAP_ALLOCATE       [95]|Bytes:3
01:03:15:001   HEAP_ALLOCATE       [100]|Bytes:152
01:03:15:001   HEAP_ALLOCATE       [417]|Bytes:408
01:03:15:001   HEAP_ALLOCATE       [430]|Bytes:408
01:03:15:001   HEAP_ALLOCATE       [317]|Bytes:6
01:03:15:001   HEAP_ALLOCATE       [EXTERNAL]|Bytes:1
01:03:15:001   STATEMENT_EX...     [1]
01:03:15:001   STATEMENT_EX...     [1]
01:03:15:001   HEAP_ALLOCATE       [1]|Bytes:5
01:03:15:001   HEAP_ALLOCATE       [68]|Bytes:5
01:03:15:001   HEAP_ALLOCATE       [74]|Bytes:5
01:03:15:001   HEAP_ALLOCATE       [67]|Bytes:7
01:03:15:001   USER_DEBUG          [1]|DEBUG|hello
01:03:15:002   CUMULATIVE_L...
01:03:15:002   LIMIT_USAGE_...     (default)
01:03:15:000   LIMIT_USAGE_...     Number of SOQL queries: 0 out of
01:03:15:000   LIMIT_USAGE_...     Number of query rows: 0 out of 50

Enter Apex Code
studentmanagementsystem.managestudents();

User            Application   Operation                                  Time                     Status                                   Read     Size
Krishna Tiwari  Unknown       /services/data/v64.0/tooling/executeA...   9/28/2025, 1:03:15 AM    Success                                           1.64 KB
Krishna Tiwari  Unknown       /services/data/v64.0/tooling/executeA...   9/28/2025, 1:01:26 AM    Insert failed. First exception on row 0...  Unread   8.34 KB
Krishna Tiwari  Unknown       /services/data/v64.0/tooling/executeA...   9/28/2025, 1:01:06 AM    Insert failed. First exception on row 0...  Unread   8.34 KB
Krishna Tiwari  Unknown       /services/data/v64.0/tooling/executeA...   9/28/2025, 1:00:40 AM    Insert failed. First exception on row 0...  Unread   8.34 KB
Krishna Tiwari  Unknown       /services/data/v64.0/tooling/executeA...   9/28/2025, 12:56:02 AM   Insert failed. First exception on row 0...  Unread   8.33 KB
```

5. Collections (List, Set, Map)

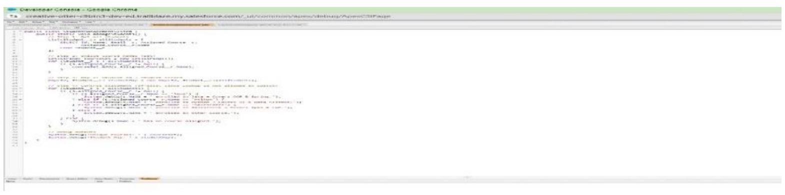
- Used **List<Student__c>** for batch updates.
- Applied **Set<Id>** to handle unique record IDs.
- Implemented **Map<Id, Student__c>** to efficiently compare old and new trigger contexts.

6. Control Statements

- Used **If-Else conditions** for null checks.
- Applied **For loops** to process multiple student records.
- Incorporated **Try-Catch blocks** for exception handling.



```
Developer Console - Google Chrome
creative-otter-c9btn3-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

Execution Log
Timestamp   Event        Details
[illegible] USER_DEBUG   [34]|DEBUG|STU-0001 has no course assigned.
[illegible] USER_DEBUG   [39]|DEBUG|Unique Courses: {}
[illegible] USER_DEBUG   [40]|DEBUG|Student Map: {[illegible]=Student__c:{Id=[illegible], Name=STU-0001, Email__c=[illegible]@gmail.com, RecordTypeId=[illegible]}}

Enter Apex Code
StudentManagementSystem.manageStudents();
```



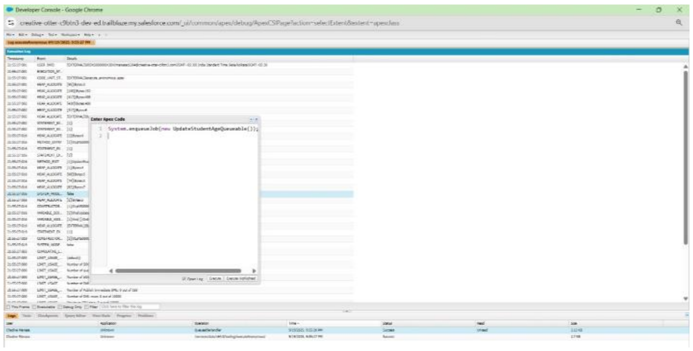
```
Developer Console - Google Chrome
creative-otter-c9btn3-dev-ed.trailblaze.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
[illegible]
```

7. Batch Apex

- Created **UpdateStudentAgeBatch** class implementing Database.Batchable<SObject>.
- Processes students in batches to calculate and update Age.
- Scheduled and monitored using **Apex Jobs**.

8. Queueable Apex

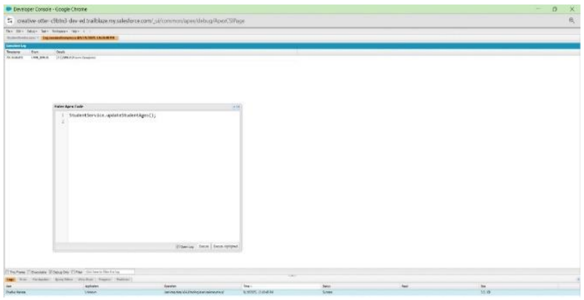
- Developed **UpdateStudentAgeQueueable** class implementing Queueable.
- Allows background execution of student age updates.
- Can be chained for sequential execution.



```
System.enqueueJob(new UpdateStudentAgeQueueable());
```

9. Exception Handling

- Wrapped logic in **try-catch** blocks.
- Handled null Date of Birth scenario to prevent runtime errors.
- Ensured no unhandled exceptions crash the system.

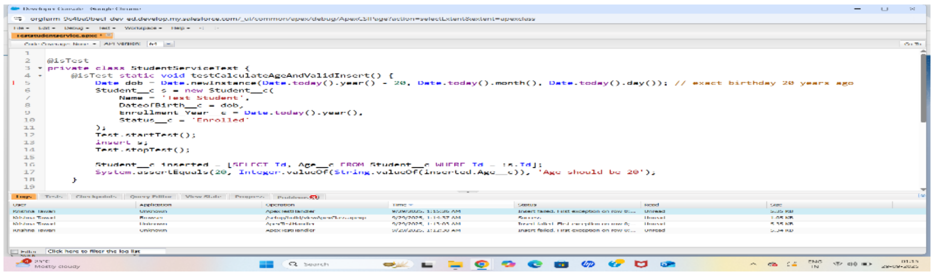


```
StudentService.updateStudentAge();
```

10. Test Classes

- Created **TestStudentService** and other test classes with @isTest.
- Verified Batch, Queueable, Future, and Exception Handling methods.

- Achieved **code coverage** and validated system reliability.

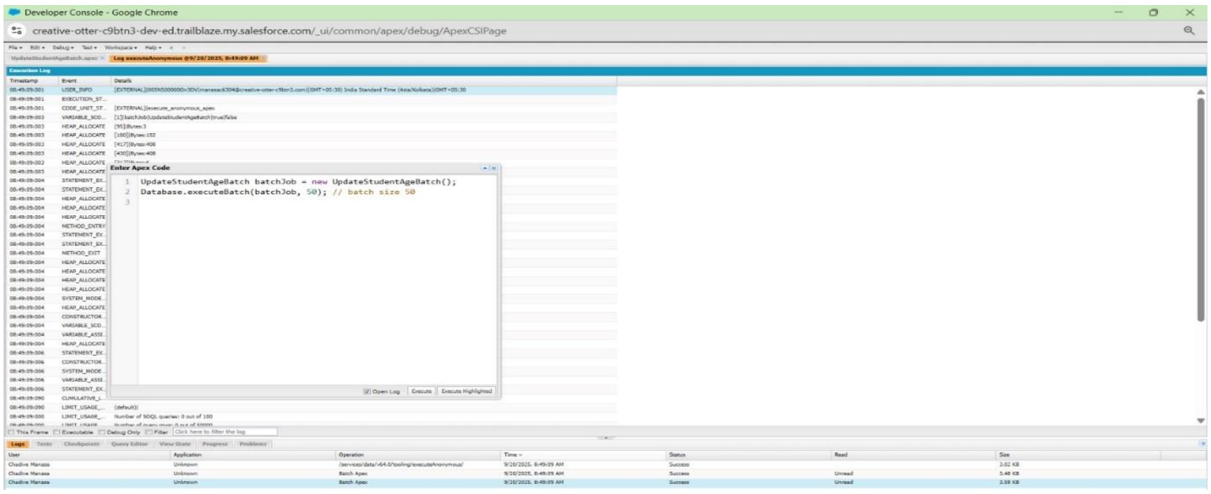


```
@isTest
private class StudentServiceTest {
    @isTest static void testCalculateAgeAndValidInsert() {
        Date dob = Date.newInstance(Date.today().year() - 20, Date.today().month(), Date.today().day()); // exact birthday 20 years ago
        Student__c s = new Student__c(
            Name = 'Test Student',
            DateOfBirth__c = dob,
            Enrollment_Year__c = Date.today().year(),
            Status__c = 'Enrolled'
        );
        Test.startTest();
        insert s;
        Test.stopTest();

        Student__c inserted = [SELECT Id, Age__c FROM Student__c WHERE Id = :s.Id];
        System.assertEquals(20, Integer.valueOf(String.valueOf(inserted.Age__c)), 'Age should be 20');
    }
```

User	Application	Operation	Time	Status	Read	Size
Krishna Tiwari	Unknown	ApexTestHandler	9/20/2025, 1:15:26 AM	Insert failed. First exception on row 0;...	Unread	5.35 KB
Krishna Tiwari	Browser	/setup/build/viewApexClass.apexp	9/20/2025, 1:14:57 AM	Success	Unread	1.05 KB
Krishna Tiwari	Unknown	ApexTestHandler	9/20/2025, 1:15:03 AM	Insert failed. First exception on row 0;...	Unread	5.35 KB
Krishna Tiwari	Unknown	ApexTestHandler	9/20/2025, 1:12:30 AM	Insert failed. First exception on row 0;...	Unread	5.34 KB

11. Asynchronous Processing

- Implemented and tested all four types: **Batch, Queueable, Scheduled, and Future Methods**.
- Verified job execution via **Apex Jobs** in Setup.
- Ensured scalability for handling large volumes of Student records.



Log executeAnonymous @9/20/2025, 8:49:09 AM

```
[EXTERNAL]|005N5000000n3DV|manasac6304@creative-otter-c9bn3.com|(GMT+05:30) India Standard Time (Asia/Kolkata)|GMT+05:30
[EXTERNAL]|execute_anonymous_apex
[1]|batchJob|UpdateStudentAgeBatch|true|false
[95]|Bytes:3
[100]|Bytes:152
[417]|Bytes:408
[430]|Bytes:408
```

Enter Apex Code

```
UpdateStudentAgeBatch batchJob = new UpdateStudentAgeBatch();
Database.executeBatch(batchJob, 50); // batch size 50
```

```
08:49:09:090  LIMIT_USAGE_...  (default)
08:49:09:000  LIMIT_USAGE_...  Number of SOQL queries: 0 out of 100
```

User	Application	Operation	Time	Status	Read	Size
Chadive Manasa	Unknown	/services/data/v64.0/tooling/executeAnonymous/	9/20/2025, 8:49:09 AM	Success		3.02 KB
Chadive Manasa	Unknown	Batch Apex	9/20/2025, 8:49:09 AM	Success	Unread	3.48 KB
Chadive Manasa	Unknown	Batch Apex	9/20/2025, 8:49:09 AM	Success	Unread	3.58 KB