

# ***PROCEDURES***

## *1. Get Student Profile*

```
SQLQuery1.sql - FR...INGPAN\faraz (59))* X
CREATE PROCEDURE GetStudentProfile
    @StudentID INT
AS
BEGIN
    SELECT s.StudentID, s.FullName, s.Email, s.Gender, s.DateOfBirth,
           d.DepartmentName
    FROM Student s
    JOIN Department d ON s.DepartmentID = d.DepartmentID
    WHERE s.StudentID = @StudentID;
END;
```

### ***Description:***

*Retrieves a student's complete profile along with their department name using their StudentID.*

## *2. Get Student GPA*

```
SQLQuery1.sql - FR...INGPAN\faraz (59))* X
CREATE PROCEDURE GetStudentGPA
    @StudentID INT
AS
BEGIN
    SELECT s.FullName,
           AVG(CASE Grade
                WHEN 'A' THEN 4.0
                WHEN 'B' THEN 3.0
                WHEN 'C' THEN 2.0
                WHEN 'D' THEN 1.0
                WHEN 'F' THEN 0.0
                END) AS GPA
    FROM Grades g
    JOIN Student s ON s.StudentID = g.StudentID
    WHERE g.StudentID = @StudentID
    GROUP BY s.FullName;
END;
```

**Description:**

*Calculates and returns the GPA of a student based on letter grades assigned in the Grades table.*

### 3. Get Total Fee Paid

```
SQLQuery1.sql - FR...INGPAN\faraz (59))*  X
CREATE PROCEDURE GetTotalFeePaid
    @StudentID INT
AS
BEGIN
    SELECT s.FullName, SUM(p.AmountPaid) AS TotalPaid
    FROM Payment p
    JOIN Fee f ON p.FeeID = f.FeeID
    JOIN Student s ON f.StudentID = s.StudentID
    WHERE s.StudentID = @StudentID
    GROUP BY s.FullName;
END;
```

**Description:**

*Returns the total amount of fees paid by a specific student across all semesters.*

### 4. Get Student Books

```
SQLQuery1.sql - FR...INGPAN\faraz (59))*  X
CREATE PROCEDURE GetStudentBooks
    @StudentID INT
AS
BEGIN
    SELECT b.Title, b.Author, bi.IssueDate, bi.ReturnDate
    FROM Book_Issues bi
    JOIN Library b ON bi.BookID = b.BookID
    WHERE bi.StudentID = @StudentID;
END;
```

**Description:**

*Lists all books issued to a student, including issue and return dates, based on their StudentID.*

## 5. Get Attendance Report

```
SQLQuery1.sql - FR...INGPAN\faraz (59))*  X
CREATE PROCEDURE GetAttendanceReport
    @StudentID INT
AS
BEGIN
    SELECT c.CourseName,
           COUNT(*) AS TotalClasses,
           SUM(CASE WHEN a.Status = 'Present' THEN 1 ELSE 0 END) AS DaysPresent,
           SUM(CASE WHEN a.Status = 'Absent' THEN 1 ELSE 0 END) AS DaysAbsent
    FROM Attendance a
    JOIN Course c ON a.CourseID = c.CourseID
    WHERE a.StudentID = @StudentID
    GROUP BY c.CourseName;
END;
```

### **Description:**

*Shows the number of total classes, days present, and days absent for a student in each course.*