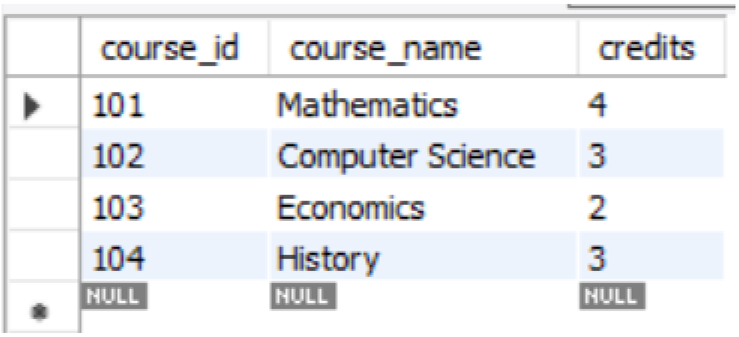
```
CREATE DATABASE UniversityDB;
USE UniversityDB;

-- Students Table
CREATE TABLE Students (
student_id INT PRIMARY KEY,
name VARCHAR(50),
city VARCHAR(50)
);
-- Courses Table
CREATE TABLE Courses (
course_id INT PRIMARY KEY,
course_name VARCHAR(50),
credits INT
);
-- Enrollments Table
CREATE TABLE Enrollments (
enroll_id INT PRIMARY KEY,
student_id INT,
course_id INT,
grade CHAR(2),
FOREIGN KEY (student_id) REFERENCES Students(student_id),
FOREIGN KEY (course_id) REFERENCES Courses(course_id)
);

-- Insert Students
INSERT INTO Students VALUES
(1, 'Rahul', 'Mumbai'),
(2, 'Priya', 'Delhi'),
(3, 'Arjun', 'Bengaluru'),
(4, 'Neha', 'Hyderabad'),
(5, 'Vikram', 'Chennai');
-- Insert Courses
INSERT INTO Courses VALUES
(101, 'Mathematics', 4),
(102, 'Computer Science', 3),
(103, 'Economics', 2),
(104, 'History', 3);
-- Insert Enrollments
INSERT INTO Enrollments VALUES
(1, 1, 101, 'A'),
(2, 1, 102, 'B'),
(3, 2, 103, 'A'),
(4, 3, 101, 'C'),
```

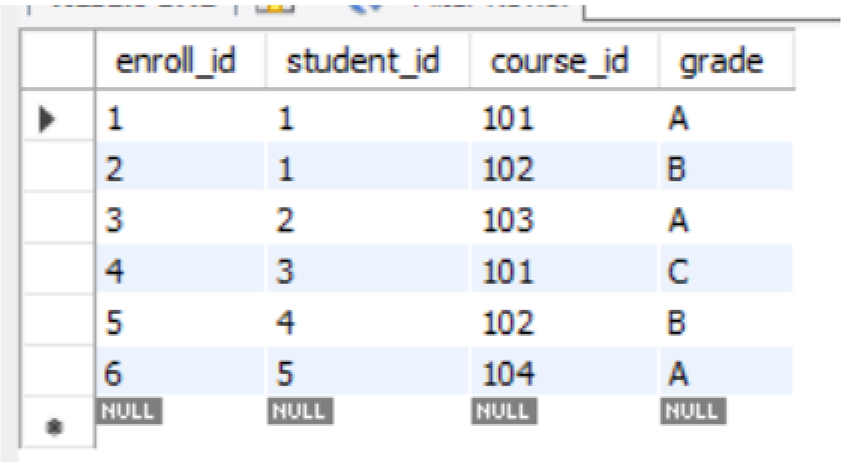
(5, 4, 102, 'B'),
(6, 5, 104, 'A');

COURSES TABLE:



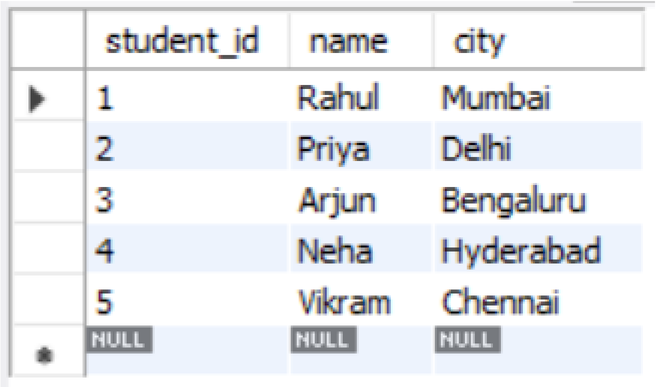
course_id	course_name	credits
101	Mathematics	4
102	Computer Science	3
103	Economics	2
104	History	3
NULL	NULL	NULL

ENROLLMENTS TABLE



enroll_id	student_id	course_id	grade
1	1	101	A
2	1	102	B
3	2	103	A
4	3	101	C
5	4	102	B
6	5	104	A
NULL	NULL	NULL	NULL

STUDENTS TABLE:



student_id	name	city
1	Rahul	Mumbai
2	Priya	Delhi
3	Arjun	Bengaluru
4	Neha	Hyderabad
5	Vikram	Chennai
NULL	NULL	NULL

Level 1: Single Table

1. **List all students**

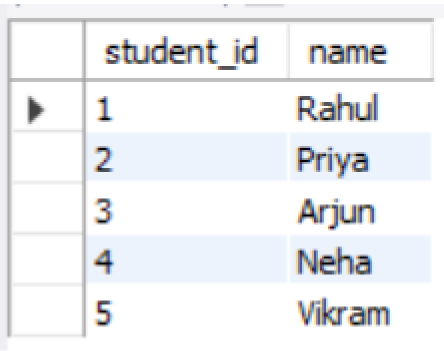
```
DELIMITER $$

CREATE PROCEDURE ListAllStudents()
BEGIN
SELECT student_id, name
FROM Students;
END$$

DELIMITER ;

CALL ListAllStudents();
```

Output:



	student_id	name
▶	1	Rahul
	2	Priya
	3	Arjun
	4	Neha
	5	Vikram

2. **List all course**s

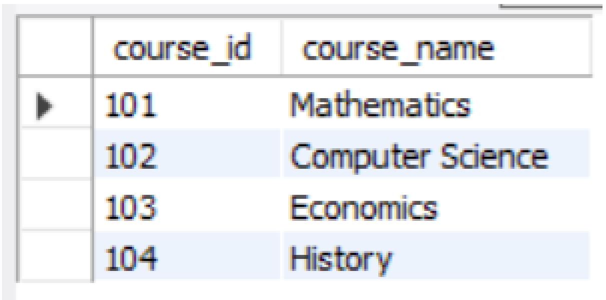
```
DELIMITER $$

CREATE PROCEDURE ListAllCourses()
BEGIN
SELECT course_id, course_name
FROM Courses;
END$$

DELIMITER ;

CALL ListAllCourses();
```

Output:



	course_id	course_name
▶	101	Mathematics
	102	Computer Science
	103	Economics
	104	History

3. **Find all students from a given city**

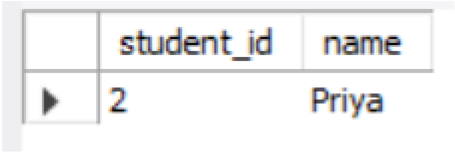
```
DELIMITER $$

CREATE PROCEDURE GetStudent(IN city_name VARCHAR(50))
BEGIN
SELECT student_id, name
FROM Students
WHERE city=city_name;
END$$

DELIMITER ;

CALL GetStudent('Delhi');
```

Output:



	student_id	name
▶	2	Priya

Level 2: Two-Table Joins

4. **List students with their enrolled courses**

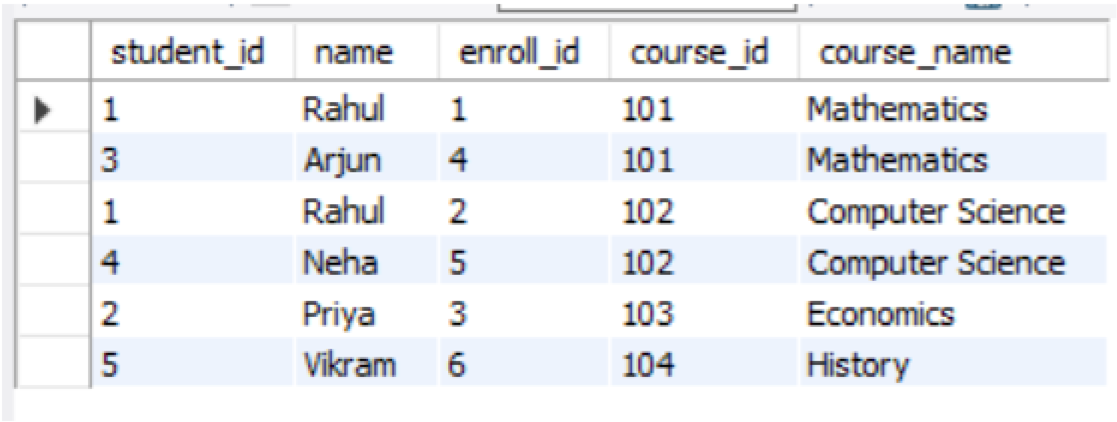
```
DELIMITER $$

CREATE PROCEDURE GetStudentWithCourses()
BEGIN
SELECT s.student_id,s.name,e.enroll_id, c.course_id,c.course_name
FROM Students s
JOIN Enrollments e ON s.student_id=e.student_id
JOIN Courses c ON e.course_id=c.course_id;
END$$
```

```
DELIMITER ;

CALL GetStudentWithCourses();
```

Output:



	student_id	name	enroll_id	course_id	course_name
▶	1	Rahul	1	101	Mathematics
	3	Arjun	4	101	Mathematics
	1	Rahul	2	102	Computer Science
	4	Neha	5	102	Computer Science
	2	Priya	3	103	Economics
	5	Vikram	6	104	History

5. **List all students enrolled in a given course**

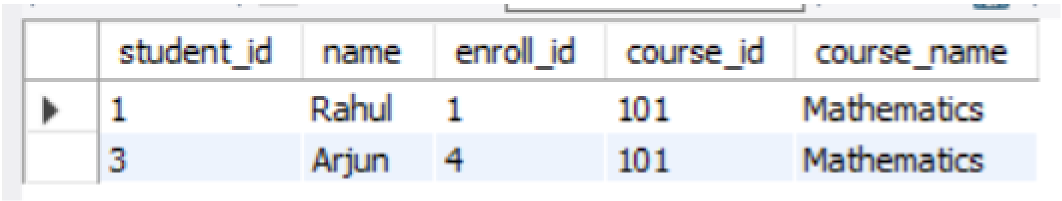
```
DELIMITER $$

CREATE PROCEDURE GetStudentWithCourse(IN course VARCHAR(50))
BEGIN
SELECT s.student_id,s.name,e.enroll_id, c.course_id,c.course_name
FROM Students s
JOIN Enrollments e ON s.student_id=e.student_id
JOIN Courses c ON e.course_id=c.course_id
WHERE c.course_name=course;
END$$

DELIMITER ;

CALL GetStudentWithCourse('Mathematics');
```

Output:



	student_id	name	enroll_id	course_id	course_name
▶	1	Rahul	1	101	Mathematics
	3	Arjun	4	101	Mathematics

6. **Count the number of students in each course**

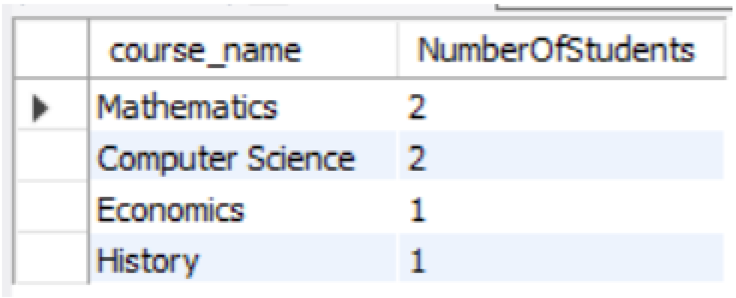
```
DELIMITER $$

CREATE PROCEDURE GetNumberOfStudents()
BEGIN
SELECT c.course_name, COUNT(s.student_id) AS NumberOfStudents
FROM Courses c
JOIN Enrollments e ON c.course_id=e.course_id
JOIN Students AS s ON e.student_id=s.student_id
GROUP BY c.course_name;
END$$

DELIMITER ;

CALL GetNumberOfStudents();
```

Output:



course_name	NumberOfStudents
Mathematics	2
Computer Science	2
Economics	1
History	1

Level 3: Three-Table Joins

7. **List students with course names and grades**

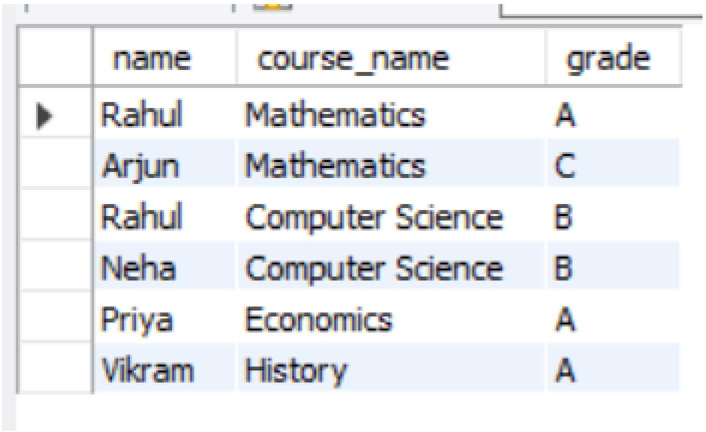
```
DELIMITER $$

CREATE PROCEDURE GetStudentGrade()
BEGIN
SELECT s.name, c.course_name, e.grade
FROM Courses c
JOIN Enrollments e ON c.course_id=e.course_id
JOIN Students AS s ON e.student_id=s.student_id;
END$$

DELIMITER ;

CALL GetStudentGrade();
```

Output:



	name	course_name	grade
▶	Rahul	Mathematics	A
	Arjun	Mathematics	C
	Rahul	Computer Science	B
	Neha	Computer Science	B
	Priya	Economics	A
	Vikram	History	A

8. **Show all courses taken by a given student**

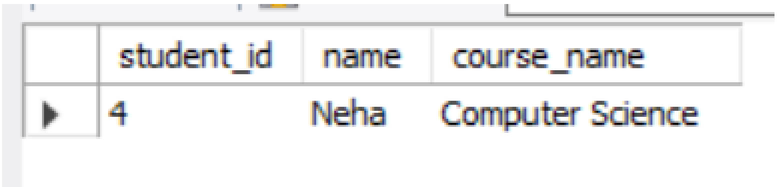
```
DELIMITER $$

CREATE PROCEDURE StudentCourses(IN sid INT)
BEGIN
SELECT s.student_id, s.name, c.course_name
FROM Courses c
JOIN Enrollments e ON c.course_id=e.course_id
JOIN Students AS s ON e.student_id=s.student_id
WHERE s.student_id=sid;
END$$

DELIMITER ;

CALL StudentCourses(4);
```

Output:



	student_id	name	course_name
▶	4	Neha	Computer Science

9. **Show average grade per course**

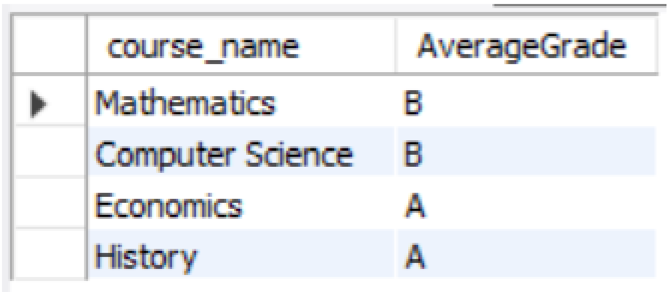
```
DELIMITER $$

CREATE PROCEDURE GetAverageGrades()
BEGIN
SELECT c.course_name, CASE ROUND(AVG(CASE e.grade
          WHEN 'A' THEN 4
          WHEN 'B' THEN 3
          WHEN 'C' THEN 2
          WHEN 'D' THEN 1
          END))
          WHEN 4 THEN'A'
          WHEN 3 THEN'B'
          WHEN 2 THEN'C'
          WHEN 1 THEN'D'
          END AS AverageGrade
FROM Courses c
JOIN Enrollments e ON c.course_id=e.course_id
GROUP BY c.course_name;
END$$

DELIMITER ;

CALL GetAverageGrades();
```

Output:



course_name	AverageGrade
Mathematics	B
Computer Science	B
Economics	A
History	A