

## Week 4 Answer

```
CREATE TABLE StudentDetails (
```

```
    StudentId INT PRIMARY KEY,
```

```
    StudentName VARCHAR(100),
```

```
    GPA DECIMAL(3, 2),
```

```
    Branch VARCHAR(50),
```

```
    Section VARCHAR(10)
```

```
);
```

```
CREATE TABLE SubjectDetails (
```

```
    SubjectId VARCHAR(10) PRIMARY KEY,
```

```
    SubjectName VARCHAR(100),
```

```
    MaxSeats INT,
```

```
    RemainingSeats INT
```

```
);
```

```
CREATE TABLE StudentPreference (
```

```
    StudentId INT,
```

```
    SubjectId VARCHAR(10),
```

```
    Preference INT,
```

```
    FOREIGN KEY (StudentId) REFERENCES StudentDetails(StudentId),
```

```
    FOREIGN KEY (SubjectId) REFERENCES SubjectDetails(SubjectId)
```

```
);
```

```
CREATE TABLE Allotments (
```

```
    SubjectId VARCHAR(10),
```

```
    StudentId INT,
```

```
    FOREIGN KEY (SubjectId) REFERENCES SubjectDetails(SubjectId),
```

```
    FOREIGN KEY (StudentId) REFERENCES StudentDetails(StudentId)
```

```
);
```

```
CREATE TABLE UnallotedStudents (
```

```
    StudentId INT,
```

```
    FOREIGN KEY (StudentId) REFERENCES StudentDetails(StudentId)
```

```

);
DELIMITER //
CREATE PROCEDURE AllocateSubjects()
BEGIN
    DECLARE done INT DEFAULT FALSE;
    DECLARE studentId INT;
    DECLARE highestGPA DECIMAL(3, 2);
    DECLARE cur CURSOR FOR
        SELECT StudentId
        FROM StudentDetails
        ORDER BY GPA DESC;

    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;

    OPEN cur;

allocation_loop: LOOP
    FETCH cur INTO studentId;
    IF done THEN
        LEAVE allocation_loop;
    END IF;

    DECLARE preference INT DEFAULT 1;
    DECLARE subjectId VARCHAR(10);

preference_loop: LOOP
    IF preference > 5 THEN
        INSERT INTO UnallotedStudents (StudentId) VALUES (studentId);
        LEAVE preference_loop;
    END IF;

```

```

SELECT SubjectId INTO subjectId
FROM StudentPreference
WHERE StudentId = studentId AND Preference = preference;

IF subjectId IS NOT NULL THEN
    DECLARE remainingSeats INT;
    SELECT RemainingSeats INTO remainingSeats
    FROM SubjectDetails
    WHERE SubjectId = subjectId;

    IF remainingSeats > 0 THEN
        INSERT INTO Allotments (SubjectId, StudentId) VALUES (subjectId, studentId);
        UPDATE SubjectDetails
        SET RemainingSeats = RemainingSeats - 1
        WHERE SubjectId = subjectId;
        LEAVE preference_loop;
    END IF;
END IF;

SET preference = preference + 1;
END LOOP preference_loop;
END LOOP allocation_loop;

CLOSE cur;
END //

DELIMITER ;

```