

1. Final Resultant Table if the student has been allotted to a subject:

```
INSERT INTO Allotments (SubjectId, StudentId)
VALUES (cur_subject, cur_student);
```

2. Final Resultant Table if the student is unallotted:

```
INSERT INTO UnallotedStudents (StudentId)
VALUES (cur_student);
```

3. Stored Procedure to assign all the students to a respective subject according to the workfl

```
DELIMITER $$
CREATE PROCEDURE AllocateSubjects()
BEGIN
DELETE FROM Allotments;
DELETE FROM UnallotedStudents;

DECLARE done INT DEFAULT 0;
DECLARE cur_student INT;
DECLARE cur_preference INT;
DECLARE cur_subject VARCHAR(10);
DECLARE cur_remaining INT;
DECLARE allocated BOOL;

DECLARE student_cursor CURSOR FOR
SELECT StudentId FROM StudentDetails
ORDER BY GPA DESC;

DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;

OPEN student_cursor;
student_loop: LOOP
FETCH student_cursor INTO cur_student;
IF done THEN
LEAVE student_loop;
END IF;

SET allocated = FALSE;
SET cur_preference = 1;

preference_loop: WHILE cur_preference <= 5 DO
SELECT SubjectId INTO cur_subject
FROM StudentPreference
WHERE StudentId = cur_student
AND Preference = cur_preference;

IF cur_subject IS NOT NULL THEN
SELECT RemainingSeats INTO cur_remaining
FROM SubjectDetails
WHERE SubjectId = cur_subject;
```

```

IF cur_remaining > 0 THEN
UPDATE SubjectDetails
SET RemainingSeats = cur_remaining - 1
WHERE SubjectId = cur_subject;

INSERT INTO Allotments (SubjectId, StudentId)
VALUES (cur_subject, cur_student);

SET allocated = TRUE;
LEAVE preference_loop;
END IF;
END IF;

SET cur_preference = cur_preference + 1;
END WHILE preference_loop;

IF NOT allocated THEN
INSERT INTO UnallotedStudents (StudentId)
VALUES (cur_student);
END IF;
END LOOP student_loop;

CLOSE student_cursor;
END $$
DELIMITER ;

```