# LAB ASSIGNMENT-7

## NAME : T.HEMANTH
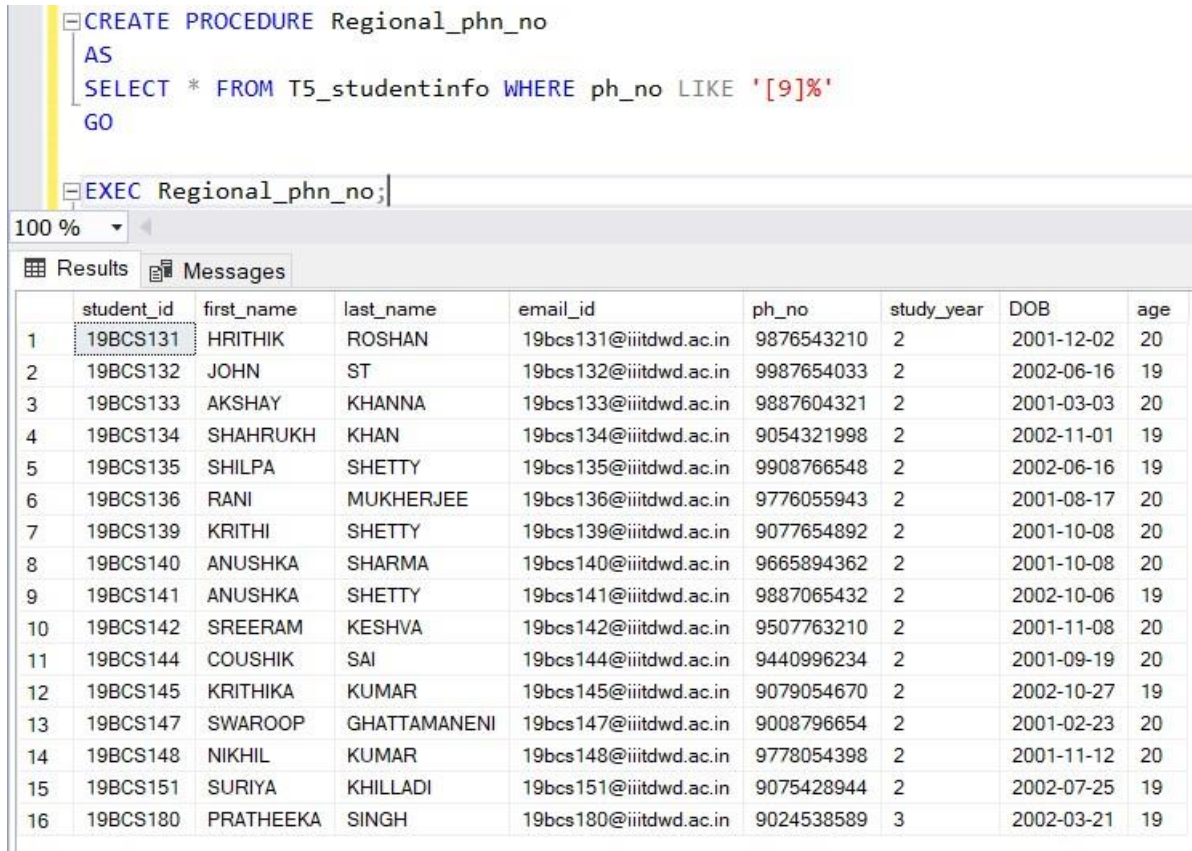## ROLL  : 19BCS107

1)Write two stored Procedures relevant to your database.

**Query1:**

```
CREATE PROCEDURE Regional_phn_no
AS
SELECT * FROM T5_studentinfo WHERE ph_no LIKE '[9]%'
GO

EXEC Regional_phn_no;
```
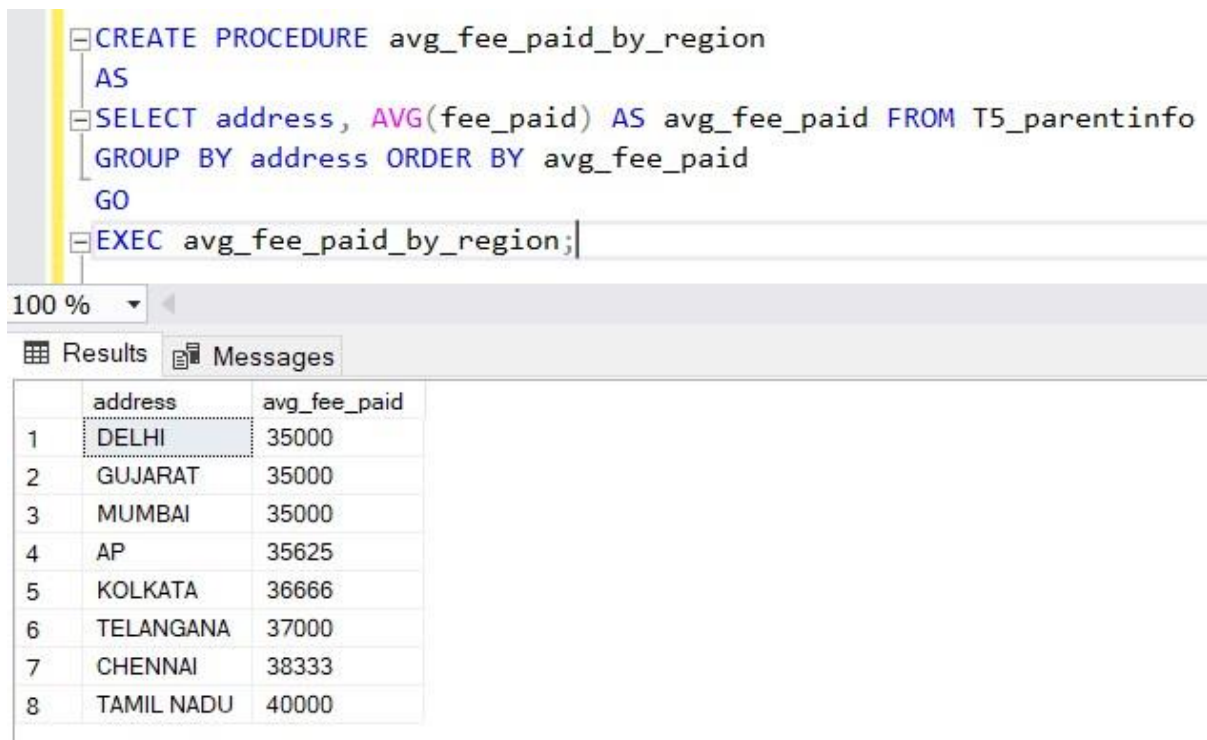
OUTPUT:



| | student_id | first_name | last_name | email_id | ph_no | study_year | DOB | age |
|---|---|---|---|---|---|---|---|---|
| 1 | 19BCS131 | HRITHIK | ROSHAN | 19bcs131@iiitdwd.ac.in | 9876543210 | 2 | 2001-12-02 | 20 |
| 2 | 19BCS132 | JOHN | ST | 19bcs132@iiitdwd.ac.in | 9987654033 | 2 | 2002-06-16 | 19 |
| 3 | 19BCS133 | AKSHAY | KHANNA | 19bcs133@iiitdwd.ac.in | 9887604321 | 2 | 2001-03-03 | 20 |
| 4 | 19BCS134 | SHAHRUKH | KHAN | 19bcs134@iiitdwd.ac.in | 9054321998 | 2 | 2002-11-01 | 19 |
| 5 | 19BCS135 | SHILPA | SHETTY | 19bcs135@iiitdwd.ac.in | 9908766548 | 2 | 2002-06-16 | 19 |
| 6 | 19BCS136 | RANI | MUKHERJEE | 19bcs136@iiitdwd.ac.in | 9776055943 | 2 | 2001-08-17 | 20 |
| 7 | 19BCS139 | KRITHI | SHETTY | 19bcs139@iiitdwd.ac.in | 9077654892 | 2 | 2001-10-08 | 20 |
| 8 | 19BCS140 | ANUSHKA | SHARMA | 19bcs140@iiitdwd.ac.in | 9665894362 | 2 | 2001-10-08 | 20 |
| 9 | 19BCS141 | ANUSHKA | SHETTY | 19bcs141@iiitdwd.ac.in | 9887065432 | 2 | 2002-10-06 | 19 |
| 10 | 19BCS142 | SREERAM | KESHVA | 19bcs142@iiitdwd.ac.in | 9507763210 | 2 | 2001-11-08 | 20 |
| 11 | 19BCS144 | COUSHIK | SAI | 19bcs144@iiitdwd.ac.in | 9440996234 | 2 | 2001-09-19 | 20 |
| 12 | 19BCS145 | KRITHIKA | KUMAR | 19bcs145@iiitdwd.ac.in | 9079054670 | 2 | 2002-10-27 | 19 |
| 13 | 19BCS147 | SWAROOP | GHATTAMANENI | 19bcs147@iiitdwd.ac.in | 9008796654 | 2 | 2001-02-23 | 20 |
| 14 | 19BCS148 | NIKHIL | KUMAR | 19bcs148@iiitdwd.ac.in | 9778054398 | 2 | 2001-11-12 | 20 |
| 15 | 19BCS151 | SURIYA | KHILLADI | 19bcs151@iiitdwd.ac.in | 9075428944 | 2 | 2002-07-25 | 19 |
| 16 | 19BCS180 | PRATHEEKA | SINGH | 19bcs180@iiitdwd.ac.in | 9024538589 | 3 | 2002-03-21 | 19 |

## Query2:

```
CREATE PROCEDURE avg_fee_paid_by_region
AS
SELECT address, AVG(fee_paid) AS avg_fee_paid FROM T5_parentinfo
GROUP BY address ORDER BY avg_fee_paid
GO
EXEC avg_fee_paid_by_region;
```

## OUTPUT:

```
CREATE PROCEDURE avg_fee_paid_by_region
AS
SELECT address, AVG(fee_paid) AS avg_fee_paid FROM T5_parentinfo
GROUP BY address ORDER BY avg_fee_paid
GO
EXEC avg_fee_paid_by_region;
```

100 %

Results | Messages

|   | address | avg_fee_paid |
|---|---------|--------------|
| 1 | DELHI | 35000 |
| 2 | GUJARAT | 35000 |
| 3 | MUMBAI | 35000 |
| 4 | AP | 35625 |
| 5 | KOLKATA | 36666 |
| 6 | TELANGANA | 37000 |
| 7 | CHENNAI | 38333 |
| 8 | TAMIL NADU | 40000 |

## 2) Write a transaction to illustrate atomicity (related to your database).

## Query:

```
BEGIN TRAN Transaction_grades
UPDATE T5_grades SET marks='92' where course_id='cs201' and student_id='19BCS133'
INSERT INTO T5_grades VALUES ('90','A','CS201','19BCS190')
COMMIT
SELECT * FROM T5_grades
```

OUTPUT:

```sql
BEGIN TRAN Transaction_grades
UPDATE T5_grades SET marks='92' where course_id='cs201' and student_id='19BCS133'
INSERT INTO T5_grades VALUES ('90','A','CS201','19BCS190')
COMMIT
SELECT * FROM T5_grades
```

110 %

Results | Messages

| | marks | grade | course_id | student_id |
|---|---|---|---|---|
| 1 | 92 | A- | CS201 | 19BCS133 |
| 2 | 85 | A- | CS201 | 19BCS138 |
| 3 | 89 | A | CS201 | 19BCS140 |
| 4 | 91 | A | CS201 | 19BCS141 |
| 5 | 84 | A- | CS201 | 19BCS142 |
| 6 | 90 | A | CS201 | 19BCS190 |
| 7 | 90 | A | CS210 | 19BCS123 |
| 8 | 69 | B | CS210 | 19BCS130 |
| 9 | 59 | C | CS210 | 19BCS132 |
| 10 | 93 | A | CS210 | 19BCS134 |
| 11 | 78 | B- | CS210 | 19BCS135 |
| 12 | 88 | A | CS210 | 19BCS139 |
| 13 | 92 | A | CS210 | 19BCS140 |
| 14 | 55 | C | CS210 | 19BCS142 |
| 15 | 67 | B | CS210 | 19BCS143 |
| 16 | 65 | B | CS210 | 19BCS145 |
| 17 | 90 | A | CS210 | 19BCS190 |
| 18 | 89 | A | CS211 | 19BCS123 |
| 19 | 77 | B- | CS211 | 19BCS144 |

Now, let us we will insert wrong information in the T5_grades table to fail the insertion deliberately.

```sql
BEGIN TRAN Transaction_grades
UPDATE T5_grades SET marks='70' where course_id='cs201' and student_id='19BCS133'
INSERT INTO T5_grades VALUES ('A','CS201','19BCS190')
COMMIT
SELECT * FROM T5_grades
```

110 %

Messages

Msg 213, Level 16, State 1, Line 20
Column name or number of supplied values does not match table definition.

Completion time: 2021-04-30T19:41:29.9091305+05:30

```
BEGIN TRAN Transaction_grades
UPDATE T5_grades SET marks='70' where course_id='cs201' and student_id='19BCS133'
INSERT INTO T5_grades VALUES ('A','CS201','19BCS190')
COMMIT
SELECT * FROM T5_grades
```

110 %

Results | Messages

| | marks | grade | course_id | student_id |
|----|-------|-------|-----------|------------|
| 1 | 92 | A- | CS201 | 19BCS133 |
| 2 | 85 | A- | CS201 | 19BCS138 |
| 3 | 89 | A | CS201 | 19BCS140 |
| 4 | 91 | A | CS201 | 19BCS141 |
| 5 | 84 | A- | CS201 | 19BCS142 |
| 6 | 90 | A | CS201 | 19BCS190 |
| 7 | 90 | A | CS210 | 19BCS123 |
| 8 | 69 | B | CS210 | 19BCS130 |
| 9 | 59 | C | CS210 | 19BCS132 |
| 10 | 85 | A | CS210 | 19BCS134 |
| 11 | 78 | B- | CS210 | 19BCS135 |
| 12 | 88 | A | CS210 | 19BCS139 |
| 13 | 92 | A | CS210 | 19BCS140 |
| 14 | 55 | C | CS210 | 19BCS142 |
| 15 | 67 | B | CS210 | 19BCS143 |
| 16 | 65 | B | CS210 | 19BCS145 |
| 17 | 90 | A | CS210 | 19BCS190 |
| 18 | 89 | A | CS211 | 19BCS123 |
| 19 | 77 | B- | CS211 | 19BCS144 |

Here, we can clearly see that the transaction got rolled back as error have been occurred in insert operation. And thus, update have not been worked due to atomic property and the previous values of the table are displayed.

**3) Write a transaction to illustrate isolation level. It can be on commit or uncommit read (related to your database).**

**Query:**

**Window1:**

```
USE school;
GO
BEGIN TRAN Trans_Isolation
UPDATE T5_course_staff
SET teacher_id = 'EC02'
WHERE course_id = 'cs211'
```

**Window2:**
```
USE school;
```

```
GO
SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED
GO
BEGIN TRAN Trans_Isolation1
SELECT * FROM T5_course_staff
WHERE course_id='cs211'
```

## OUTPUT:

```
USE school;
GO
BEGIN TRAN Trans_Isolation
UPDATE T5_course_staff
SET teacher_id = 'EC02'
WHERE course_id = 'cs211'
```

110 %

Messages

(2 rows affected)

Completion time: 2021-04-30T19:58:42.3546205+05:30

```
USE school;
GO
SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED
GO
BEGIN TRAN Trans_Isolation1
SELECT * FROM T5_course_staff
WHERE course_id='cs211'
```

110 %

Results | Messages

| | grade | student_id | course_id | teacher_id | marks |
|---|---|---|---|---|---|
| 1 | A | 19BCS123 | CS211 | EC02 | 87 |
| 2 | B- | 19BCS144 | CS211 | EC02 | 79 |

• When we set the isolation level to read uncommitted, we will be able to see the teacher_id set to 'EC02', called Dirty Read.