

Assignment 8

- A. Create a variable avg_cred to store the average of total credits of students. Show names of the students with total credit more than twice the average of total credits using variable avg_cred.

```
MariaDB [university]> SET @avg_cred = (SELECT AVG(tot_cred) FROM student);  
Query OK, 0 rows affected (0.006 sec)
```

```
MariaDB [university]> SELECT @avg_cred;
```

```
+-----+  
| @avg_cred |  
+-----+  
| 65.692307692 |  
+-----+  
1 row in set (0.001 sec)
```

```
MariaDB [university]> SELECT *  
    -> FROM student  
    -> WHERE tot_cred > 2 * @avg_cred;  
Empty set (0.008 sec)
```

Justification: Setting a variable to store the average credits value, and reusing the value when comparing with the average credits value.

- B. Show the details of the departments which have budgets more than the average budget across all departments. First show it without defining any function, then show it by defining a function avg_budget that returns the average budget across all departments.

```
MariaDB [university]> SELECT *  
    -> FROM department  
    -> WHERE budget > (SELECT AVG(T.budget) FROM department AS T);
```

dept_name	building	budget
Biology	Watson	90000.00
Comp. Sci.	Taylor	100000.00
Finance	Painter	120000.00

```
3 rows in set (0.006 sec)
```

Justification: Nested query is required to compare budget with the average budget over all departments.

```

MariaDB [university]> DELIMITER #
MariaDB [university]> CREATE FUNCTION avg_budget()
  -> RETURNS FLOAT DETERMINISTIC
  -> BEGIN
  -> DECLARE value FLOAT;
  -> SELECT AVG(budget) FROM department INTO value;
  -> RETURN value;
  -> END; #
Query OK, 0 rows affected, 1 warning (0.090 sec)

```

```

MariaDB [university]> SELECT *
  -> FROM department
  -> WHERE budget > avg_budget();

```

dept_name	building	budget
Biology	Watson	90000.00
Comp. Sci.	Taylor	100000.00
Finance	Painter	120000.00

```

3 rows in set (0.001 sec)

```

Justification: writing a function which reuters the average budget of all departments. And using it when comparing.

- C. Create a procedure top_budg that when called will accept a number N and return budgets of top N departments. Use N=3 and N=5 to verify your procedure.

```

MariaDB [university]> DELIMITER #
MariaDB [university]> CREATE PROCEDURE top_budg(in N int)
  -> BEGIN
  -> SELECT * FROM department ORDER BY budget DESC LIMIT N;
  -> END; #
Query OK, 0 rows affected (0.066 sec)

MariaDB [university]> DELIMITER ;
MariaDB [university]> CALL top_budg(3);
+-----+-----+-----+
| dept_name | building | budget |
+-----+-----+-----+
| Finance   | Painter  | 120000.00 |
| Comp. Sci. | Taylor   | 100000.00 |
| Biology    | Watson   | 90000.00  |
+-----+-----+-----+
3 rows in set (0.001 sec)

Query OK, 0 rows affected (0.001 sec)

MariaDB [university]> CALL top_budg(5);
+-----+-----+-----+
| dept_name | building | budget |
+-----+-----+-----+
| Finance   | Painter  | 120000.00 |
| Comp. Sci. | Taylor   | 100000.00 |
| Biology    | Watson   | 90000.00  |
| Elec. Eng. | Taylor   | 85000.00  |
| Music      | Packard  | 80000.00  |
+-----+-----+-----+
5 rows in set (0.001 sec)

Query OK, 0 rows affected (0.001 sec)

```

D.

Justification: Creating the required procedure, to create view to allow newly joined students to partially view the student info.

```

MariaDB [University]> DELIMITER #
MariaDB [University]> CREATE PROCEDURE stud_access (IN stud_name VARCHAR(20))
-> BEGIN
-> SET @CreateViewStatement =
-> CONCAT("CREATE OR REPLACE VIEW stud_strict AS
-> (SELECT S.ID, S.name, S.dept_name, T.tot_cred
-> FROM student AS S LEFT JOIN (SELECT L.ID, L.name, L.dept_name, L.tot_cred
-> FROM student AS L WHERE L.name = '', stud_name, '') AS T ON S.ID = T.ID)");
-> PREPARE QUERY FROM @CreateViewStatement;
-> EXECUTE QUERY;
-> END; #
Query OK, 0 rows affected (0.066 sec)

```

```

MariaDB [University]> DELIMITER ;
MariaDB [University]> SELECT * FROM student;

```

ID	name	dept_name	tot_cred
00128	Zhang	Comp. Sci.	102
12345	Shankar	Comp. Sci.	32
19991	Brandt	History	80
23121	Chavez	Finance	110
44553	Peltier	Physics	56
45678	Levy	Physics	46
54321	Williams	Comp. Sci.	54
55739	Sanchez	Music	38
70557	Snow	Physics	0
76543	Brown	Comp. Sci.	58
76653	Aoi	Elec. Eng.	60
98765	Bourikas	Elec. Eng.	98
98988	Tanaka	Biology	120

```

13 rows in set (0.000 sec)

```

Justification: Inserting two new entries into table, as newly joined students.

```

MariaDB [University]> INSERT INTO student VALUES('00000', 'Vipul', 'Comp. Sci.', 100);
Query OK, 1 row affected (0.021 sec)

MariaDB [University]> INSERT INTO student VALUES('00001', 'Mohan', 'Comp. Sci.', 0);
Query OK, 1 row affected (0.005 sec)

```

```
MariaDB [University]> SELECT * FROM student;
```

ID	name	dept_name	tot_cred
00000	Vipul	Comp. Sci.	100
00001	Mohan	Comp. Sci.	0
00128	Zhang	Comp. Sci.	102
12345	Shankar	Comp. Sci.	32
19991	Brandt	History	80
23121	Chavez	Finance	110
44553	Peltier	Physics	56
45678	Levy	Physics	46
54321	Williams	Comp. Sci.	54
55739	Sanchez	Music	38
70557	Snow	Physics	0
76543	Brown	Comp. Sci.	58
76653	Aoi	Elec. Eng.	60
98765	Bourikas	Elec. Eng.	98
98988	Tanaka	Biology	120

```
15 rows in set (0.001 sec)
```

Justification: Creating the user, as the newly joined student. And giving him access to the view.

```
MariaDB [University]> CREATE USER Vipul IDENTIFIED BY 'pass@vipul';
Query OK, 0 rows affected (0.015 sec)

MariaDB [University]> call stud_access('Vipul');
Query OK, 0 rows affected (0.022 sec)

MariaDB [University]> GRANT SELECT ON stud_strict TO Vipul;
Query OK, 0 rows affected (0.015 sec)
```

Justification: Logging in as a newly created user and viewing the partial students table. Only full info corresponding to the newly created student is shown, all other is hidden into NULL.

```

MariaDB [(none)]> SELECT CURRENT_USER();
+-----+
| CURRENT_USER() |
+-----+
| Vipul@%s       |
+-----+
1 row in set (0.000 sec)

MariaDB [(none)]> SELECT * FROM University.stud_strict;
+----+-----+-----+-----+
| ID  | name  | dept_name | tot_cred |
+----+-----+-----+-----+
| 00000 | Vipul | Comp. Sci. | 100      |
| 00001 | Mohan | Comp. Sci. | NULL     |
| 00128 | Zhang | Comp. Sci. | NULL     |
| 12345 | Shankar | Comp. Sci. | NULL     |
| 19991 | Brandt | History   | NULL     |
| 23121 | Chavez | Finance   | NULL     |
| 44553 | Peltier | Physics   | NULL     |
| 45678 | Levy   | Physics   | NULL     |
| 54321 | Williams | Comp. Sci. | NULL     |
| 55739 | Sanchez | Music     | NULL     |
| 70557 | Snow   | Physics   | NULL     |
| 76543 | Brown  | Comp. Sci. | NULL     |
| 76653 | Aoi    | Elec. Eng. | NULL     |
| 98765 | Bourikas | Elec. Eng. | NULL     |
| 98988 | Tanaka | Biology   | NULL     |
+----+-----+-----+-----+
15 rows in set (0.003 sec)

```

Justification: Creating another new user as a newly joined student. And granting him the access to the view.

```

MariaDB [University]> CREATE USER Mohan IDENTIFIED BY 'pass@mohan';
Query OK, 0 rows affected (0.013 sec)

MariaDB [University]> call stud_access('Mohan');
Query OK, 0 rows affected (0.027 sec)

MariaDB [University]> GRANT SELECT ON stud_strict TO Mohan;
Query OK, 0 rows affected (0.014 sec)

```

Justification: Logging in as a newly created user, and viewing the info. Only partial is visible.

```
MariaDB [(none)]> SELECT CURRENT_USER();
```

CURRENT_USER()
Mohan@%

1 row in set (0.000 sec)

```
MariaDB [(none)]> SELECT * FROM University.stud_strict;
```

ID	name	dept_name	tot_cred
00000	Vipul	Comp. Sci.	NULL
00001	Mohan	Comp. Sci.	0
00128	Zhang	Comp. Sci.	NULL
12345	Shankar	Comp. Sci.	NULL
19991	Brandt	History	NULL
23121	Chavez	Finance	NULL
44553	Peltier	Physics	NULL
45678	Levy	Physics	NULL
54321	Williams	Comp. Sci.	NULL
55739	Sanchez	Music	NULL
70557	Snow	Physics	NULL
76543	Brown	Comp. Sci.	NULL
76653	Aoi	Elec. Eng.	NULL
98765	Bourikas	Elec. Eng.	NULL
98988	Tanaka	Biology	NULL

15 rows in set (0.006 sec)