CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY

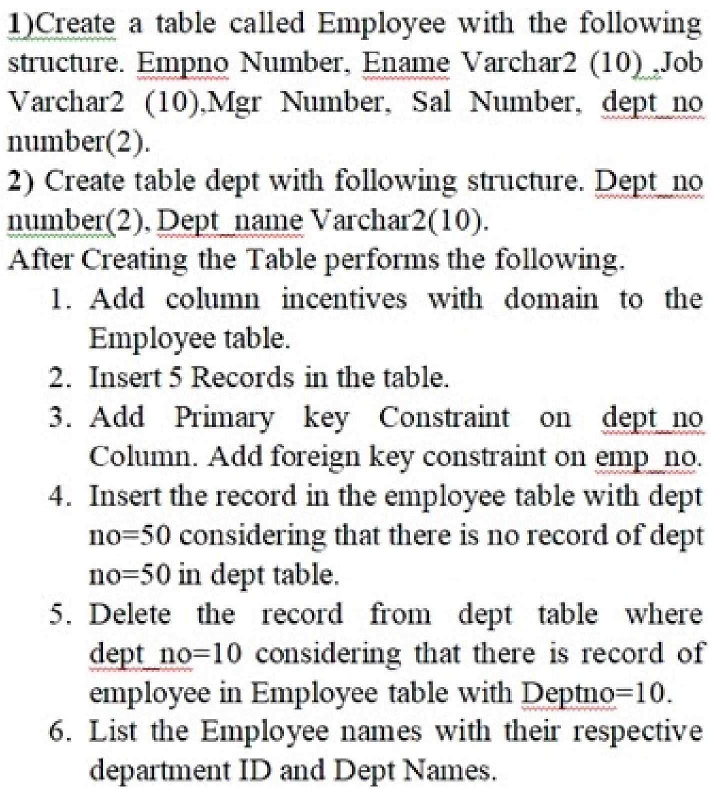
U. & P. U. PATEL DEPARTMENT OF COMPUTER ENGINEERING

DEVANG PATEL INSTITUE OF ADVANCE TECHNOLOGY AND RESEARCH

ID:	19DCE056	**Name:**	Yash Kavaiya
Subject Code:	CE246	**Subject Name:**	Database Management System
Date of Exam:	1st May, 2021	**Batch:**	C
Contact:	9537601879	**Question Set:**	

PRACTICAL – 1

AIM: Perform following queries.



1)Create a table called Employee with the following structure. Empno Number, Ename Varchar2 (10) ,Job Varchar2 (10),Mgr Number, Sal Number, dept_no number(2).

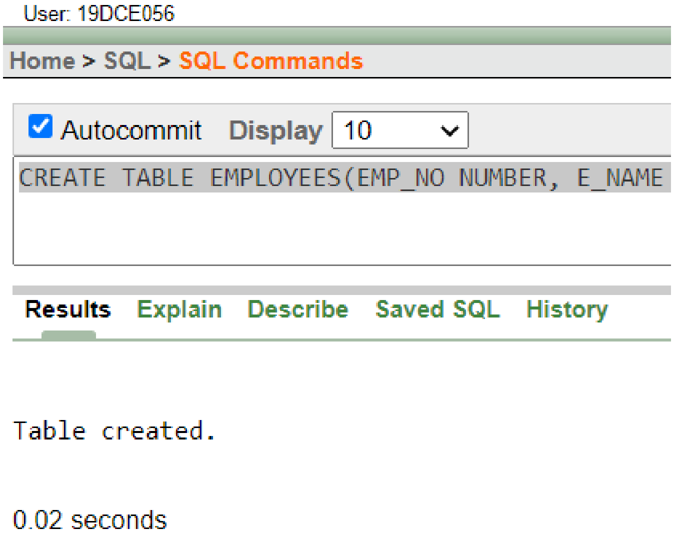
2) Create table dept with following structure. Dept_no number(2), Dept_name Varchar2(10).

After Creating the Table performs the following.

1. Add column incentives with domain to the Employee table.
2. Insert 5 Records in the table.
3. Add Primary key Constraint on dept_no Column. Add foreign key constraint on emp_no.
4. Insert the record in the employee table with dept no=50 considering that there is no record of dept no=50 in dept table.
5. Delete the record from dept table where dept_no=10 considering that there is record of employee in Employee table with Deptno=10.
6. List the Employee names with their respective department ID and Dept Names.

COMMANDS:

CREATE TABLE EMPLOYEES(EMP_NO NUMBER, E_NAME VARCHAR2(10), JOB VARCHAR2(10), MGR NUMBER, SALARY NUMBER, DEPT_NO NUMBER(2));

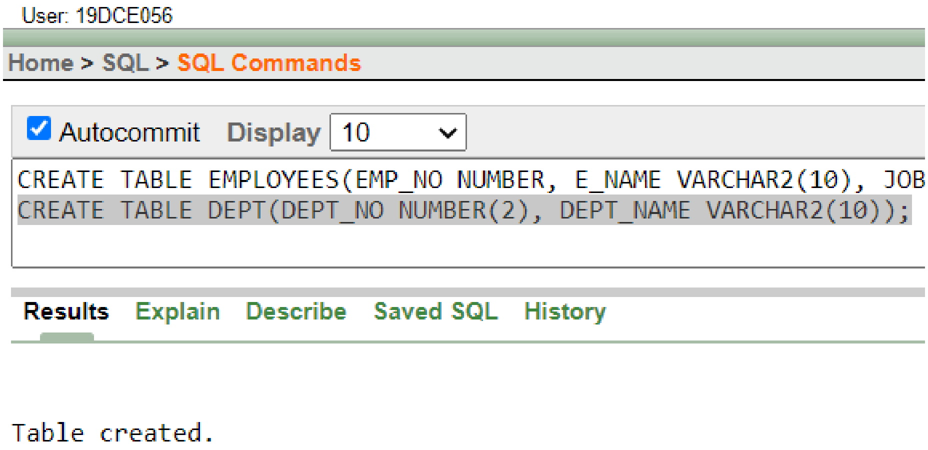

User: 19DCE056
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History

```
CREATE TABLE EMPLOYEES(EMP_NO NUMBER, E_NAME
```

```
Table created.
```

0.02 seconds

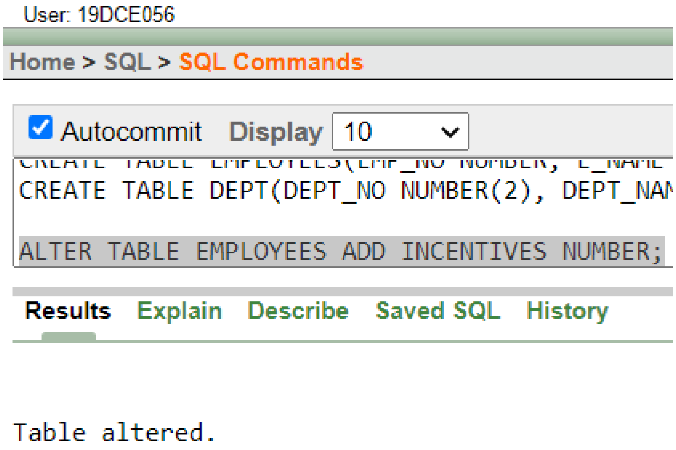
CREATE TABLE DEPT(DEPT_NO NUMBER(2), DEPT_NAME VARCHAR2(10));


User: 19DCE056
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History

```
CREATE TABLE EMPLOYEES(EMP_NO NUMBER, E_NAME VARCHAR2(10), JOB
CREATE TABLE DEPT(DEPT_NO NUMBER(2), DEPT_NAME VARCHAR2(10));
```

```
Table created.
```

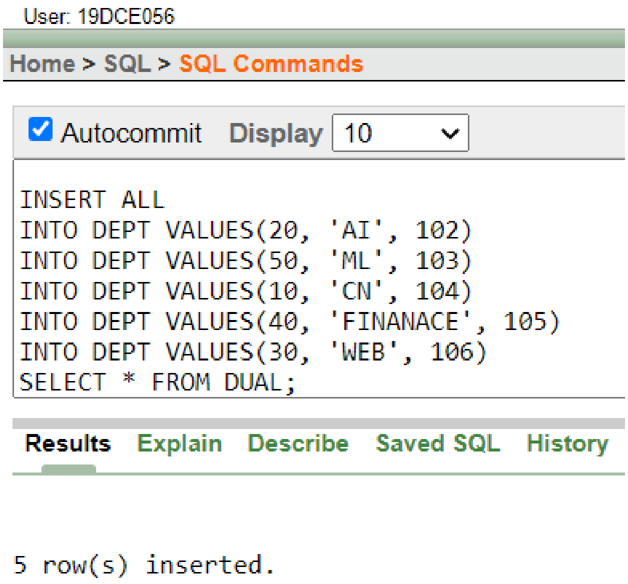
ALTER TABLE EMPLOYEES ADD INCENTIVES NUMBER;


User: 19DCE056
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History

```
CREATE TABLE EMPLOYEES(EMP_NO NUMBER, E_NAME
CREATE TABLE DEPT(DEPT_NO NUMBER(2), DEPT_NAN

ALTER TABLE EMPLOYEES ADD INCENTIVES NUMBER;
```

```
Table altered.
```

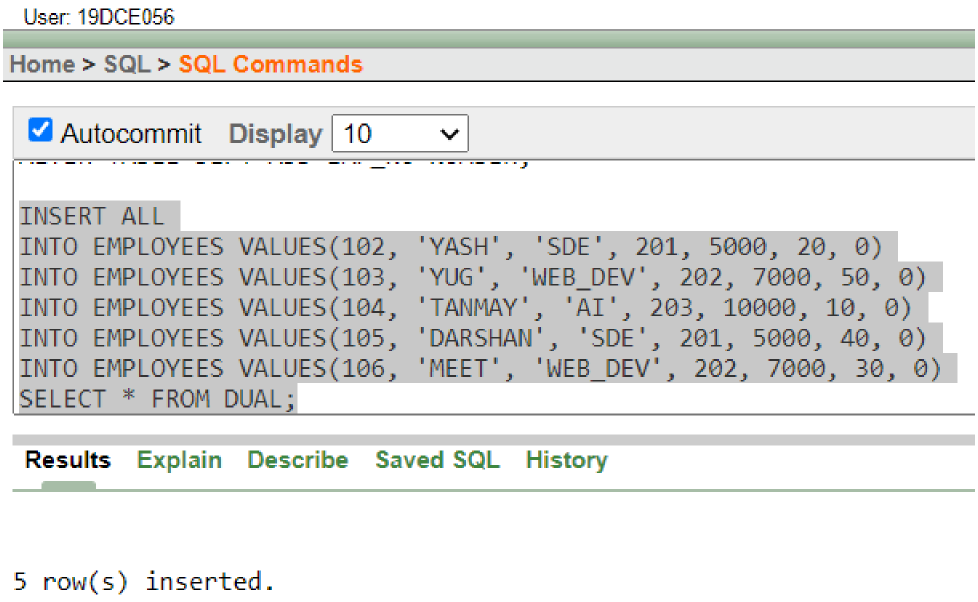
```
INSERT ALL
INTO DEPT VALUES(20, 'AI', 102)
INTO DEPT VALUES(50, 'ML', 103)
INTO DEPT VALUES(10, 'CN', 104)
INTO DEPT VALUES(40, 'FINANACE', 105)
INTO DEPT VALUES(30, 'WEB', 106)
SELECT * FROM DUAL;
```



```
INSERT ALL
INTO DEPT VALUES(20, 'AI', 102)
INTO DEPT VALUES(50, 'ML', 103)
INTO DEPT VALUES(10, 'CN', 104)
INTO DEPT VALUES(40, 'FINANACE', 105)
INTO DEPT VALUES(30, 'WEB', 106)
SELECT * FROM DUAL;
```

```
5 row(s) inserted.
```

```
INSERT ALL
INTO EMPLOYEES VALUES(102, 'YASH', 'SDE', 201, 5000, 20, 0)
INTO EMPLOYEES VALUES(103, 'YUG', 'WEB_DEV', 202, 7000, 50, 0)
INTO EMPLOYEES VALUES(104, 'TANMAY', 'AI', 203, 10000, 10, 0)
INTO EMPLOYEES VALUES(105, 'DARSHAN', 'SDE', 201, 5000, 40, 0)
INTO EMPLOYEES VALUES(106, 'MEET', 'WEB_DEV', 202, 7000, 30, 0)
SELECT * FROM DUAL;
```

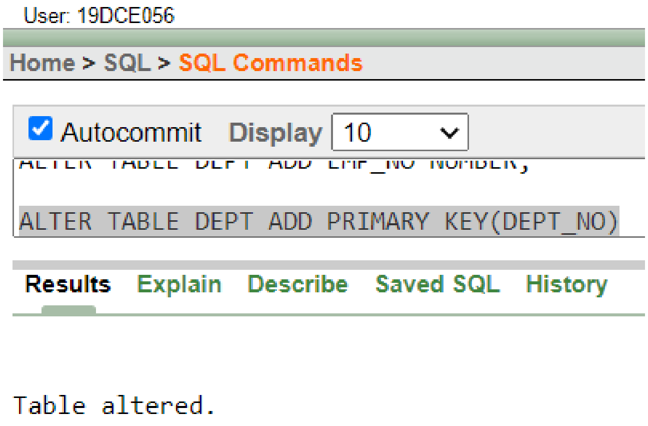


```
INSERT ALL
INTO EMPLOYEES VALUES(102, 'YASH', 'SDE', 201, 5000, 20, 0)
INTO EMPLOYEES VALUES(103, 'YUG', 'WEB_DEV', 202, 7000, 50, 0)
INTO EMPLOYEES VALUES(104, 'TANMAY', 'AI', 203, 10000, 10, 0)
INTO EMPLOYEES VALUES(105, 'DARSHAN', 'SDE', 201, 5000, 40, 0)
INTO EMPLOYEES VALUES(106, 'MEET', 'WEB_DEV', 202, 7000, 30, 0)
SELECT * FROM DUAL;
```

```
5 row(s) inserted.
```

ALTER TABLE DEPT ADD PRIMARY KEY(DEPT_NO)

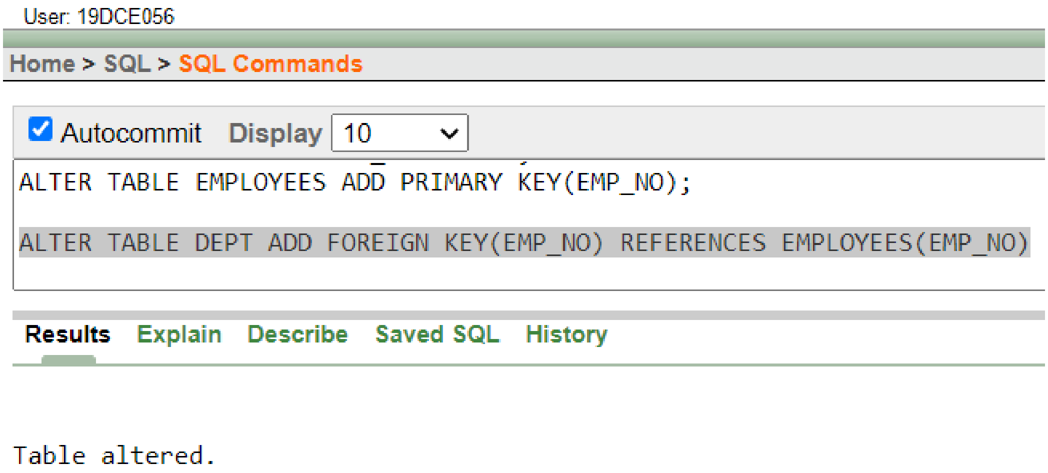
ALTER TABLE EMPLOYEES ADD PRIMARY KEY(EMP_NO);


User: 19DCE056
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History

```
ALTER TABLE DEPT ADD EMP_NO NUMBER;

ALTER TABLE DEPT ADD PRIMARY KEY(DEPT_NO)
```

```
Table altered.
```

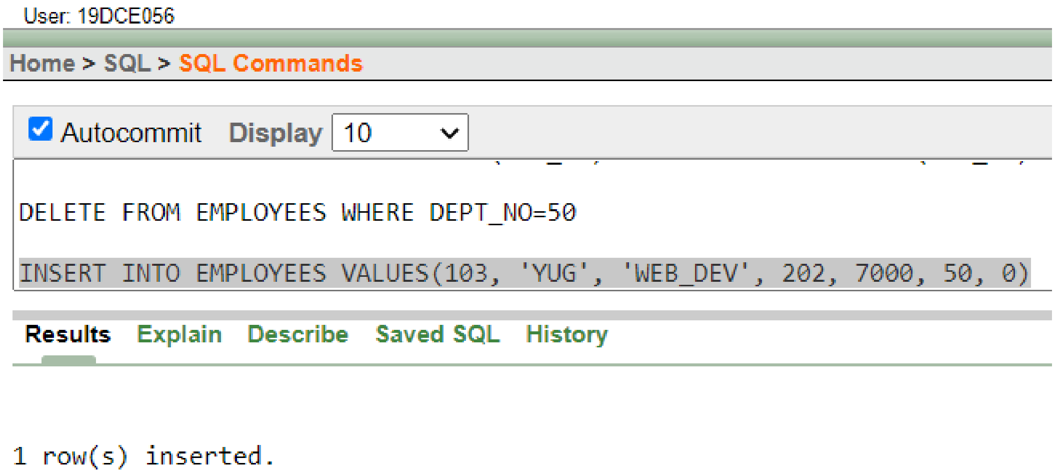
ALTER TABLE DEPT ADD FOREIGN KEY(EMP_NO) REFERENCES EMPLOYEES(EMP_NO)


User: 19DCE056
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History

```
ALTER TABLE EMPLOYEES ADD PRIMARY KEY(EMP_NO);

ALTER TABLE DEPT ADD FOREIGN KEY(EMP_NO) REFERENCES EMPLOYEES(EMP_NO)
```

```
Table altered.
```

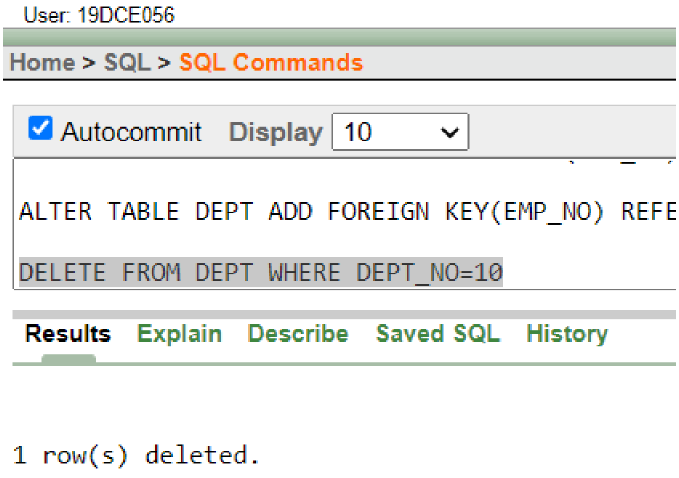
INSERT INTO EMPLOYEES VALUES(103, 'YUG', 'WEB_DEV', 202, 7000, 50, 0)


User: 19DCE056
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History

```
DELETE FROM EMPLOYEES WHERE DEPT_NO=50

INSERT INTO EMPLOYEES VALUES(103, 'YUG', 'WEB_DEV', 202, 7000, 50, 0)
```

```
1 row(s) inserted.
```

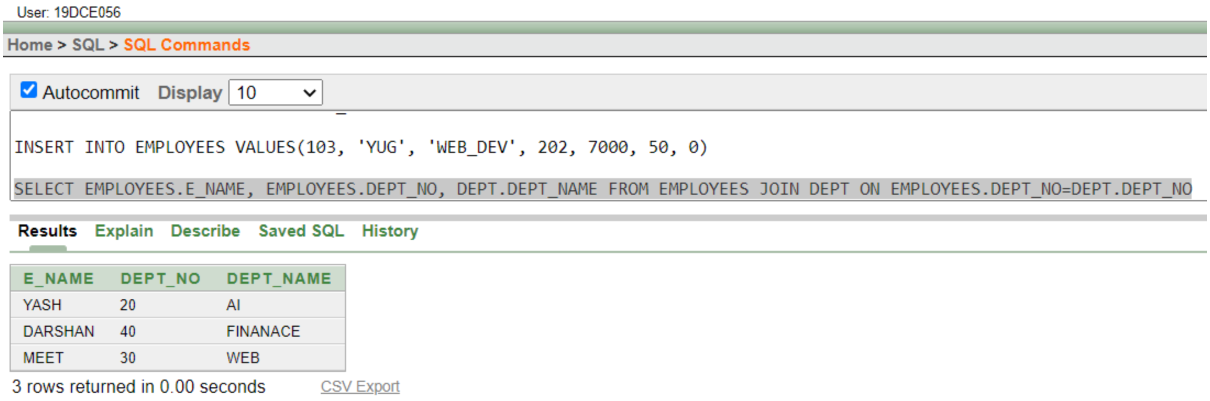
DELETE FROM DEPT WHERE DEPT_NO=10


User: 19DCE056
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History

```
ALTER TABLE DEPT ADD FOREIGN KEY(EMP_NO) REFE

DELETE FROM DEPT WHERE DEPT_NO=10
```

```
1 row(s) deleted.
```

SELECT EMPLOYEES.E_NAME, EMPLOYEES.DEPT_NO, DEPT.DEPT_NAME FROM EMPLOYEES JOIN DEPT ON EMPLOYEES.DEPT_NO=DEPT.DEPT_NO


User: 19DCE056
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History
CSV Export

```
INSERT INTO EMPLOYEES VALUES(103, 'YUG', 'WEB_DEV', 202, 7000, 50, 0)

SELECT EMPLOYEES.E_NAME, EMPLOYEES.DEPT_NO, DEPT.DEPT_NAME FROM EMPLOYEES JOIN DEPT ON EMPLOYEES.DEPT_NO=DEPT.DEPT_NO
```

E_NAME	DEPT_NO	DEPT_NAME
YASH	20	AI
DARSHAN	40	FINANACE
MEET	30	WEB

3 rows returned in 0.00 seconds

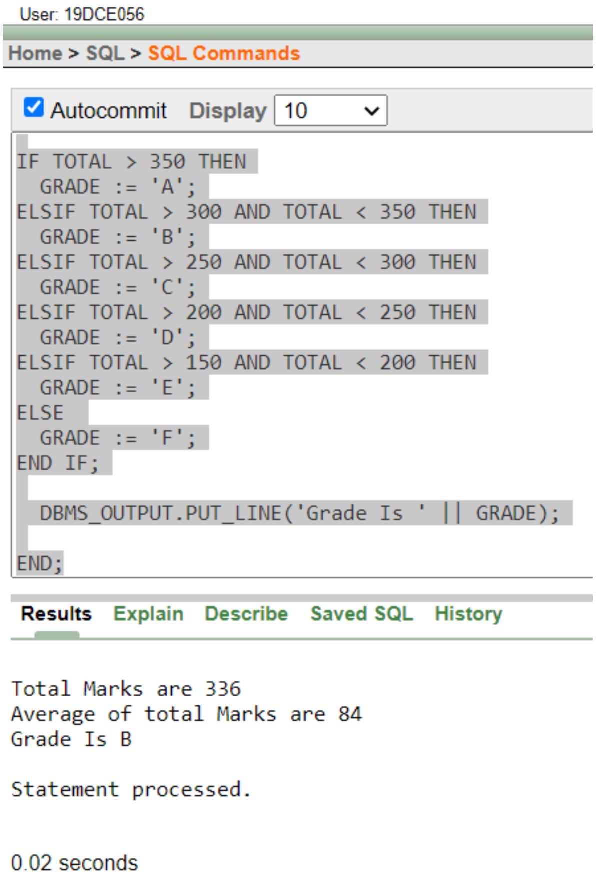
PRACTICAL – 2

AIM: Write a PL/SQL program to find sum and average of 4 subjects and display grade.

CODE:

```
DECLARE

SUB_1 NUMBER;
SUB_2 NUMBER;
SUB_3 NUMBER;
SUB_4 NUMBER;
TOTAL NUMBER;
AVERAGE NUMBER;
GRADE VARCHAR2(1);

BEGIN

-- USER SHOULD ENTER MARKS OUT OF 100
SUB_1 := :ENTER_SUBJECT_1_MARKS;
SUB_2 := :ENTER_SUBJECT_2_MARKS;
SUB_3 := :ENTER_SUBJECT_3_MARKS;
SUB_4 := :ENTER_SUBJECT_4_MARKS;

TOTAL := SUB_1 + SUB_2 + SUB_3 + SUB_4;
AVERAGE := TOTAL/4;
DBMS_OUTPUT.PUT_LINE('Total Marks are ' || TOTAL);
DBMS_OUTPUT.PUT_LINE('Average of total Marks are ' || AVERAGE);

IF TOTAL > 350 THEN
  GRADE := 'A';
ELSIF TOTAL > 300 AND TOTAL < 350 THEN
  GRADE := 'B';
ELSIF TOTAL > 250 AND TOTAL < 300 THEN
  GRADE := 'C';
ELSIF TOTAL > 200 AND TOTAL < 250 THEN
  GRADE := 'D';
ELSIF TOTAL > 150 AND TOTAL < 200 THEN
  GRADE := 'E';
ELSE
  GRADE := 'F';
END IF;

  DBMS_OUTPUT.PUT_LINE('Grade Is ' || GRADE);

END;
```

OUTPUT:


User: 19DCE056
Home > SQL > SQL Commands
☑ Autocommit Display 10
Results Explain Describe Saved SQL History

```
IF TOTAL > 350 THEN
  GRADE := 'A';
ELSIF TOTAL > 300 AND TOTAL < 350 THEN
  GRADE := 'B';
ELSIF TOTAL > 250 AND TOTAL < 300 THEN
  GRADE := 'C';
ELSIF TOTAL > 200 AND TOTAL < 250 THEN
  GRADE := 'D';
ELSIF TOTAL > 150 AND TOTAL < 200 THEN
  GRADE := 'E';
ELSE
  GRADE := 'F';
END IF;

  DBMS_OUTPUT.PUT_LINE('Grade Is ' || GRADE);

END;
```

```
Total Marks are 336
Average of total Marks are 84
Grade Is B

Statement processed.


0.02 seconds
```