

CONDUIRA DATA ANALYST-PYTHON

Final Assessment

SQL:

Please use MSSQL/MYSQL/Oracle for the following, First question(Q1) is mandatory:

1. Write a SQL query to create these tables in your database and insert the data into these tables with the following characteristics:

a. Add the primary key “Emp_ID” to the Employees Table. Also, mention what are the constraints used in SQL.

b. Add foreign key “EMP_REF_ID” in Variables Details and Designation Table that references “Emp_ID” in Employees Table

SQL Queries:

```
CREATE DATABASE company;
```

```
USE company;
```

```
CREATE TABLE employees(  
    EMP_ID INT(10) NOT NULL,  
    FIRST_NAME VARCHAR(25) NOT NULL,  
    LAST_NAME VARCHAR(25) NOT NULL,  
    SALARY INT(10),  
    JOINING_DATE DATE,  
    DEPARTMENT VARCHAR(25),  
    primary key(EMP_ID)  
)
```

```
INSERT INTO employees  
(EMP_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT)  
VALUES(001,'Manish','Agarwal',700000,'2019-04-20 09:00:00','HR');
```

```
INSERT INTO employees  
(EMP_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT)  
VALUES(002,'Niranjan','Bose',20000,'2019-02-11 09:00:00','DA');
```

```
INSERT INTO employees
(EMP_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT)
VALUES(003,'Vivek','Singh',100000,'2019-01-20 09:00:00','DA');
```

```
INSERT INTO employees
(EMP_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT)
VALUES(004,'Asutosh','Kapoor',700000,'2019-03-20 09:00:00','HR');
```

```
INSERT INTO employees
(EMP_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT)
VALUES(005,'Vihaan','Benerjee',300000,'2019-06-11 09:00:00','DA');
```

```
INSERT INTO employees
(EMP_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT)
VALUES(006,'Atul','Diwedi',400000,'2019-05-11 09:00:00','Account');
```

```
INSERT INTO employees
(EMP_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT)
VALUES(007,'Satyendra','Tripathi',95000,'2019-03-20 09:00:00','Account');
```

```
INSERT INTO employees
(EMP_ID,FIRST_NAME,LAST_NAME,SALARY,JOINING_DATE,DEPARTMENT)
VALUES(008,'Pritika','Bhatt',80000,'2019-02-11 09:00:00','DA');
```

```
SELECT * FROM employees;
```

```
commit;
```

b.)

```
use company;
```

```
create table variablesdetails(
```

```
    EMP_REF_ID INT(10) NOT NULL,
```

```
    VARIABLES_DATE DATETIME,
```

```
    VARIABLES_AMOUNT INT(10)
```

```
    foreign key(EMP_REF_ID)
```

```
);
```

```
INSERT INTO variablesdetails(EMP_REF_ID,VARIABLES_DATE,VARIABLES_AMOUNT) VALUES
(1,'2019-02-20 00:00:00',15000);
```

```
INSERT INTO variablesdetails(EMP_REF_ID,VARIABLES_DATE,VARIABLES_AMOUNT) VALUES
(2,'2019-06-11 00:00:00',30000);
```

```
INSERT INTO variablesdetails(EMP_REF_ID,VARIABLES_DATE,VARIABLES_AMOUNT) VALUES  
(3,'2019-02-20 00:00:00',42000);
```

```
INSERT INTO variablesdetails(EMP_REF_ID,VARIABLES_DATE,VARIABLES_AMOUNT) VALUES  
(4,'2019-02-20 00:00:00',14500);
```

```
INSERT INTO variablesdetails(EMP_REF_ID,VARIABLES_DATE,VARIABLES_AMOUNT) VALUES  
(5,'2019-06-11 00:00:00',23500);
```

```
commit;
```

```
use company;
```

```
CREATE TABLE designation(  
    EMP_REF_ID INT(100),  
    EMP_TITLE VARCHAR(25),  
    AFFECTEDFROM DATETIME,  
    foreign key(EMP_REF_ID)  
);
```

```
INSERT INTO designation(EMP_REF_ID,EMP_TITLE,AFFECTEDFROM) VALUES (1,'Asst.  
Manager','2019-02-20 00:00:00');
```

```
INSERT INTO designation(EMP_REF_ID,EMP_TITLE,AFFECTEDFROM) VALUES (2,'Senior  
Analyst','2019-01-11 00:00:00');
```

```
INSERT INTO designation(EMP_REF_ID,EMP_TITLE,AFFECTEDFROM) VALUES (8,'Senior  
Analyst','2019-04-06 00:00:00');
```

```
INSERT INTO designation(EMP_REF_ID,EMP_TITLE,AFFECTEDFROM) VALUES  
(5,'Manager','2019-10-06 00:00:00');
```

```
INSERT INTO designation(EMP_REF_ID,EMP_TITLE,AFFECTEDFROM) VALUES (4,'Asst.  
Manager','2019-12-06 00:00:00');
```

```
INSERT INTO designation(EMP_REF_ID,EMP_TITLE,AFFECTEDFROM) VALUES (7,'Team  
Lead','2019-06-06 00:00:00');
```

```
INSERT INTO designation(EMP_REF_ID,EMP_TITLE,AFFECTEDFROM) VALUES (6,'Team  
Lead','2019-09-06 00:00:00');
```

```
INSERT INTO designation(EMP_REF_ID,EMP_TITLE,AFFECTEDFROM) VALUES (3,'Senior  
Analyst','2019-08-06 00:00:00');
```

```
commit;
```

2. Name the four different types of joins? Give examples of each by performing all the joins on the Employees table and Designation Table.

a. Write a query to get the employee details(columns - full name and department) of those who received the highest and the least variables

b. Write a query to get the designation which has got the highest and second lowest amount (salary + variables) for the whole year of 2019. Get the corresponding amount values.

c. What is cross join? Write a query to give an example of the same by performing it on the Employees table and Designation table.

d. What are the clauses used with Select statements and what are the preference orders of it?

SQL JOINS are used to retrieve data from multiple tables. A SQL JOIN is performed whenever two or more tables are listed in a SQL statement.

There are 4 different types of SQL joins:

- SQL INNER JOIN
- SQL LEFT OUTER JOIN
- SQL RIGHT OUTER JOIN
- SQL FULL OUTER JOIN

Queries:

a. SELECT (FIRST_NAME + LAST_NAME) as FULL_NAME, department FROM employees

left join variablesdetails on employees.EMP_ID = variablesdetails.EMP_REF_ID

WHERE VARIABLES_AMOUNT = ALL(SELECT max(VARIABLE_AMOUNT) AND

VARIABLE_AMOUNT ALL(SELECT min(VARIABLE_AMOUNT)

FROM variablesdetails GROUP BY department_id);

b. SELECT EMP_TITLE FROM designation

WHERE (SELECT T1.salary as SALARY, T2.VARIABLES_AMOUNT as VARIABLE, T1.salary +
T2.VARIABLES_AMOUNT AS AMOUNT

FROM employees as T1 JOIN variablesdetails as T2 on T1.EMP_ID = T2.variablesdetails)

ORDER BY AMOUNT DESC WHERE rownum = 2 AND rownum = -1

JOIN employees ON employees.EMP_ID = designation.EMP_REF_ID

JOIN variablesdetails ON designation.EMP_REF_ID = variablesdetails.EMP_REF_ID;

- c. In SQL, the **Cross Join** is used to combine each row of the first table with each row of the second table. It is also known as the Cartesian join since it returns the Cartesian product of the sets of rows from the joined tables.

SELECT * FROM employees CROSS JOIN designation;

- d. The SELECT statement has the following clauses:

- SELECT
- FROM
- WHERE
- GROUP BY
- HAVING
- ORDER BY
- OFFSET
- FETCH FIRST
- UNION
- INTERSECT
- EXCEPT
- WITH

The preference order of the SELECT clause follows the above order.

3. What is the stored procedure?

- a. Write a query to get the employee details who got their designations updated in the second half of the year 2019(July to December), sorted by the "variables_amount" (highest to lowest)
- b. Write a stored procedure to call the query that you have written for Q2.

Stored Procedure:

A stored procedure is a prepared SQL code that you can save, so the code can be reused repeatedly. So, if you have an SQL query that you write repeatedly, save it as a stored procedure, and then just call it to execute it. You can also pass parameters to a stored procedure, so that the stored procedure can act based on the parameter value(s) that is passed.

a.

- b. CREATE PROCEDURE employee_details

AS

SELECT (FIRST_NAME + LAST_NAME) as FULL_NAME, department FROM employees

left join variablesdetails on employees.EMP_ID = variablesdetails.EMP_REF_ID

WHERE VARIABLES_AMOUNT = ALL(SELECT max(VARIABLE_AMOUNT) AND

VARIABLE_AMOUNT ALL(SELECT min(VARIABLE_AMOUNT)

FROM variablesdetails GROUP BY department_id)

Go;

