Assignment-4

Name :- Bibek Chand Sah

Roll No. :- 22054029

Section :- CSE-05

1. WAQ to display the current date.

➔ SELECT SYSDATE AS current_date FROM dual;

```
SQL> SELECT SYSDATE AS current_date FROM dual;

CURRENT_D
---------
05-FEB-24
```

2. Assume that employees serve a six month provisional period starting from their DOJ. After 6 months they get permanent status. Display all the employees' names, doj and the date on which they received their permanent status as permanent_date.

➔ SELECT f_name, l_name, doj, doj + INTERVAL '6' MONTH AS permanent_date FROM employee;

3. WAQ to display the last date of this current month.

➔ SELECT LAST_DAY(SYSDATE) AS last_date_of_month FROM dual;

```
SQL> SELECT LAST_DAY(SYSDATE) AS last_date_of_month FROM dual;

LAST_DATE
---------
29-FEB-24
```

4. For all the employees display their emp_id, f_name and their total experience in months.

➔ SELECT employee_id, first_name, last_name, MONTHS_BETWEEN(SYSDATE, doj) AS total_experience_months FROM employee;

```
SQL> SELECT employee_id, first_name, last_name, MONTHS_BETWEEN(SYSDATE, doj) AS total_experience_month
s FROM employee;

EMPLOYEE_ID FIRST_NAME LAST_NAME  TOTAL_EXPERIENCE_MONTHS
----------- ---------- ---------- -----------------------
          1 arun       khan                   313.058495
          2 barun      kumar                  311.897205
          3 chitra     kapoor                 312.929463
          4 dheeraj    mishra                  265.31656
          5 emma       dutt                   262.542366
          6 floki      dutt                   282.671398
          7 dheeraj    kumar                  91.1552692
          8 saul       good                   112.993979
         10 sunny      deol                   274.187527
         11 bobby      deol                   75.6391402
         13 anand      patil                  84.1875273

EMPLOYEE_ID FIRST_NAME LAST_NAME  TOTAL_EXPERIENCE_MONTHS
----------- ---------- ---------- -----------------------
         14 anandi     patel                  82.1552692

12 rows selected.
```

5. WAQ to display the date of next TUESDAY.

➔ SELECT NEXT_DAY(SYSDATE, 'TUESDAY') AS next_tuesday_date FROM dual;

```
SQL> SELECT NEXT_DAY(SYSDATE, 'TUESDAY') AS next_tuesday_date FROM dual;

NEXT_TUES
---------
06-FEB-24
```

6. WAQ to extract the current month.

➔ SELECT EXTRACT(MONTH FROM SYSDATE) AS current_month FROM dual;

```
SQL> SELECT EXTRACT(MONTH FROM SYSDATE) AS current_month FROM dual;

CURRENT_MONTH
-------------
            2
```

7. WAQ to extract the current year.

➔ SELECT EXTRACT(YEAR FROM SYSDATE) AS current_year FROM dual;

```
SQL> SELECT EXTRACT(YEAR FROM SYSDATE) AS current_year FROM dual;

CURRENT_YEAR
------------
        2024
```

8. WAQ to display the absolute value of -505.

➔ SELECT ABS(-505) AS absolute_value FROM dual;

```
SQL> SELECT ABS(-505) AS absolute_value FROM dual;

ABSOLUTE_VALUE
--------------
           505
```

9. WAQ to display the ceiling of 10.44 and 10.50 and 10.65.

➔ SELECT CEIL(10.44) AS ceil_1, CEIL(10.50) AS ceil_2, CEIL(10.65) AS ceil_3 FROM dual;

```
SQL> SELECT CEIL(10.44) AS ceil_1, CEIL(10.50) AS ceil_2, CEIL(10.65) AS ceil_3 FROM dual;

    CEIL_1     CEIL_2     CEIL_3
---------- ---------- ----------
        11         11         11
```

10. WAQ to display the floor value of 10.44 and 10.50 and 10.65.

➔ SELECT FLOOR(10.44) AS floor_1, FLOOR(10.50) AS floor_2, FLOOR(10.65) AS floor_3 FROM dual;

```
SQL> SELECT FLOOR(10.44) AS floor_1, FLOOR(10.50) AS floor_2, FLOOR(10.65) AS floor_3 FROM dual;

   FLOOR_1    FLOOR_2    FLOOR_3
---------- ---------- ----------
        10         10         10
```

11. Find the logarithmic value of 10 base 2.

➔ SELECT LOG(10, 2) AS logarithmic_value FROM dual;

```
SQL> SELECT LOG(10, 2) AS logarithmic_value FROM dual;

LOGARITHMIC_VALUE
-----------------
        .301029996
```

12. Display the remainder in 594/7.

➔ SELECT MOD(594, 7) AS remainder FROM dual;

```
SQL> SELECT MOD(594, 7) AS remainder FROM dual;

 REMAINDER
----------
         6
```

13. WAQ to display the value of 8 to the power 3.

➔ SELECT POWER(8, 3) AS power_value FROM dual;

```
SQL> SELECT POWER(8, 3) AS power_value FROM dual;

POWER_VALUE
-----------
        512
```

14. WAQ to display the square root of 3481.

➔ SELECT SQRT(3481) AS square_root FROM dual;

```
SQL> SELECT SQRT(3481) AS square_root FROM dual;

SQUARE_ROOT
-----------
         59
```

15. Display the following rounding operations- round(45.923,2), round(45.923,0), round(45.923,-1), round(45.923,2), round(45.923,0), round(45.923,-2).

➔ SELECT ROUND(45.923, 2) AS round_1, ROUND(45.923, 0) AS round_2, ROUND(45.923, -1) AS round_3, ROUND(45.923, -2) AS round_4 FROM dual;

```
SQL> SELECT ROUND(45.923, 2) AS round_1, ROUND(45.923, 0) AS round_2, ROUND(45.923, -1) AS round_3, RO
UND(45.923, -2) AS round_4 FROM dual;

   ROUND_1     ROUND_2     ROUND_3     ROUND_4
---------- ----------- ----------- -----------
     45.92          46          50           0
```

16. Display the following rounding operations- trunc(45.888,2), trunc(56.758,0), trunc(49.245,-2), trunc(45.888,2), round(45.888,2).

➔ SELECT TRUNC(45.888, 2) AS trunc_1, TRUNC(56.758, 0) AS trunc_2, TRUNC(49.245, -2) AS trunc_3 FROM dual;

```
SQL> SELECT TRUNC(45.888, 2) AS trunc_1, TRUNC(56.758, 0) AS trunc_2, TRUNC(49.245, -2) AS trunc_3 FRO
M dual;

   TRUNC_1     TRUNC_2     TRUNC_3
---------- ----------- -----------
     45.88          56           0
```

17. WAQ to return the sign of 20 and -67.60 and 0.

➔ SELECT SIGN(20) AS sign_1, SIGN(-67.60) AS sign_2, SIGN(0) AS sign_3 FROM dual;

```
SQL> SELECT SIGN(20) AS sign_1, SIGN(-67.60) AS sign_2, SIGN(0) AS sign_3 FROM dual;

    SIGN_1     SIGN_2     SIGN_3
---------- ---------- ----------
         1         -1          0
```

18. Display the value of cos(45), sin(45), and tan(45).

➔ SELECT COS(45) AS cosine, SIN(45) AS sine, TAN(45) AS tangent FROM dual;

```
SQL> SELECT COS(45) AS cosine, SIN(45) AS sine, TAN(45) AS tangent FROM dual;

    COSINE       SINE    TANGENT
---------- ---------- ----------
.525321989 .850903525 1.61977519
```

19. Display the ASCII character corresponding to the integer 79.

➔ SELECT CHR(79) AS ascii_character FROM dual;

```
SQL> SELECT CHR(79) AS ascii_character FROM dual;

A
-
O
```

20.Display the f_name and l_name together using the concat() function.

➔ SELECT CONCAT ('Bibek', 'Sah') FROM DUAL;

```
SQL> SELECT CONCAT ('Bibek', 'Sah') FROM DUAL;

CONCAT('
--------
BibekSah
```

21. Display all the f_names in capital letters.

➔ SELECT UPPER(first_name) AS capital_f_name FROM employee;

```
SQL> SELECT UPPER(first_name) AS capital_f_name FROM employee;

CAPITAL_F_
----------
ARUN
BARUN
CHITRA
DHEERAJ
EMMA
FLOKI
DHEERAJ
SAUL
SUNNY
BOBBY
ANAND

CAPITAL_F_
----------
ANANDI

12 rows selected.
```

22. Find the length of the first name and last name of all the employees who work in the sales department.

➔ SELECT LENGTH(first_name) AS first_name_length, LENGTH(last_name) AS last_name_length FROM employee WHERE dept = 'sales';

```
SQL> SELECT LENGTH(first_name) AS first_name_length, LENGTH(last_name) AS last_name_length FROM employ
ee WHERE dept = 'sales';

FIRST_NAME_LENGTH LAST_NAME_LENGTH
----------------- ----------------
                7                6
```

23. Determine the tax-rate for each employee based on their monthly salary. The tax-rates are

as per the following table. If salary is < 2000 then tax rate is 0 %, 20000<=salary<40000 9%, 40000<=salary<60000 20%, 60000<=salary<80000 30%, salary>=80000 45%

Monthly Salary Range	Tax Rate (%)
salary<20000	0
20000<=salary<40000	9
40000<=salary<60000	20
60000<=salary<80000	30
salary>=80000	45

➔ SELECT employee_id, first_name, last_name, salary, CASE WHEN salary < 2000 THEN 0 WHEN salary >= 20000 AND salary < 40000 THEN 9 WHEN salary >= 40000 AND salary < 60000 THEN 20

WHEN salary >= 60000 AND salary < 80000 THEN 30 WHEN salary >= 80000 THEN 45 END AS tax_rate FROM employee;

```
SQL> SELECT employee_id, first_name, last_name, salary, CASE WHEN salary < 2000 THEN 0 WHEN salary >=
20000 AND salary < 40000 THEN 9 WHEN salary >= 40000 AND salary < 60000 THEN 20 WHEN salary >= 60000 A
ND salary < 80000 THEN 30 WHEN salary >= 80000 THEN 45 END AS tax_rate FROM employee;

EMPLOYEE_ID FIRST_NAME LAST_NAME      SALARY    TAX_RATE
----------- ---------- ---------- ---------- ----------
          1 arun       khan            90000          45
          2 barun      kumar           80000          45
          3 chitra     kapoor          60000          30
          4 dheeraj    mishra          75000          30
          5 emma       dutt            55000          20
          6 floki      dutt            70000          30
          7 dheeraj    kumar           40000          20
          8 saul       good            60000          30
         10 sunny      deol            20000           9
         11 bobby      deol            35000           9
         13 anand      patil           28000           9

EMPLOYEE_ID FIRST_NAME LAST_NAME      SALARY    TAX_RATE
----------- ---------- ---------- ---------- ----------
         14 anandi     patel           12000

12 rows selected.
```

24. Find the average salary, maximum salary, minimum salary and the sum of salaries from the employee table.

➔ SELECT AVG(salary) AS average_salary, MAX(salary) AS maximum_salary, MIN(salary) AS minimum_salary, SUM(salary) AS total_salary FROM employee;

```
SQL> SELECT AVG(salary) AS average_salary, MAX(salary) AS maximum_salary, MIN(salary) AS minimum_salar
y, SUM(salary) AS total_salary FROM employee;

AVERAGE_SALARY MAXIMUM_SALARY MINIMUM_SALARY TOTAL_SALARY
-------------- -------------- -------------- ------------
    52083.3333          90000          12000       625000
```

25. Find the average salary, maximum salary, minimum salary and the sum of salaries of the

employees who work for the sales department.

➔ SELECT AVG(salary) AS average_salary, MAX(salary) AS maximum_salary, MIN(salary) AS minimum_salary, SUM(salary) AS total_salary FROM employee WHERE dept = 'sales';

```
SQL> SELECT AVG(salary) AS average_salary, MAX(salary) AS maximum_salary, MIN(salary) AS minimum_salar
y, SUM(salary) AS total_salary FROM employee WHERE dept = 'sales';

AVERAGE_SALARY MAXIMUM_SALARY MINIMUM_SALARY TOTAL_SALARY
-------------- -------------- -------------- ------------
         75000          75000          75000        75000
```

26. Find the newest and oldest employee.

➔ SELECT MAX(doj) AS newest_employee, MIN(doj) AS oldest_employee FROM employee;

```
SQL> SELECT MAX(doj) AS newest_employee, MIN(doj) AS oldest_employee FROM employee;

NEWEST_EM OLDEST_EM
--------- ---------
17-OCT-17 04-JAN-98
```

27. Find those two employees whose l_name comes first and last in alphabetical order.

➔ SELECT MIN(last_name) AS first_employee, MAX(last_name) AS last_employee FROM employee;

```
SQL> SELECT MIN(last_name) AS first_employee, MAX(last_name) AS last_employee FROM employee;

FIRST_EMPL LAST_EMPLO
---------- ----------
deol       patil
```

28. Find the number of engineers.

➔ SELECT COUNT(*) AS num_engineers FROM employee WHERE job_types = 'engineer';

```
SQL> SELECT COUNT(*) AS num_engineers FROM employee WHERE job_types = 'engineer';

NUM_ENGINEERS
-------------
            5
```

29. Find the number of departments from the employee table.

➔ SELECT COUNT(DISTINCT dept) AS num_departments FROM employee;

```
SQL> SELECT COUNT(DISTINCT dept) AS num_departments FROM employee;

NUM_DEPARTMENTS
---------------
              7
```

30. Find the average commission from the employee table.

➔ SELECT AVG(commission) AS average_commission FROM employee;

```
SQL> SELECT AVG(commission) AS average_commission FROM employee;

AVERAGE_COMMISSION
------------------
        4166.66667
```