1. Display all employees with their commission value. Display 0 commission for employees who do not get any commission.

```
SELECT EmployeeId, Lname, Fname, NVL(Commission, 0) FROM employee;
```

```
EMPLOYEEID LNAME                FNAME                NVL(COMMISSION,0)
---------- -------------------- -------------------- -----------------
       111 Brown                Chris                                0
       222 Green                Alex                               500
       333 Taylor               Jordan                               0
       444 Martinez             Olivia                            1000
       555 Lopez                Sophia                               0
```

2. Count the total number of rooms in LOCATION.

```
SELECT COUNT(*) AS TotalRooms FROM location;
```

```
TOTALROOMS
----------
         5
```

3. Count the distinct building names in LOCATION.

```
SELECT COUNT(DISTINCT Building) AS DistinctBuildings FROM location;
```

```
DISTINCTBUILDINGS
-----------------
                5
```

4. Display all student names and birth dates. Display birth dates with the format '20 OCTOBER, 1970'.

```
SELECT First, Last, TO_CHAR(BirthDate, 'DD MONTH, YYYY') AS DOB From student;
```

```
FIRST                LAST                 DOB
-------------------- -------------------- ----------------------------------------------
Jane                 Doe                  15 JUNE     , 1998
Michael              Smith                25 AUGUST   , 1999
Sophia               Williams             05 MAY      , 2000
Emma                 Johnson              10 NOVEMBER , 1997
Isabella             Brown                12 MARCH    , 1998
```

5. Find the average, highest, and lowest age for students.

```
SELECT
  ROUND(AVG(MONTHS_BETWEEN(SYSDATE, BirthDate)) / 12) AS AverageAge,
  ROUND(MONTHS_BETWEEN(SYSDATE, MIN(BirthDate)) / 12) AS YoungestAge,
  ROUND(MONTHS_BETWEEN(SYSDATE, MAX(BirthDate)) / 12) AS OldestAge
FROM student;
```

```
AVERAGEAGE YOUNGESTAGE  OLDESTAGE
---------- ----------- ----------
        26          27         25
```

6. Display only the year value from each employee's hire date.

```
SELECT EmployeeId, EXTRACT(YEAR FROM HireDate) AS HireYear FROM employee;
```

```
EMPLOYEEID   HIREYEAR
---------- ----------
       111       2018
       222       2020
       333       2021
       444       2022
       555       2023
```

7. Find average employee commission.

i. Ignore NULLs

```
SELECT AVG(Commission) AS AvgCommission FROM employee
WHERE Commission IS NOT NULL;

AVGCOMMISSION
-------------
          750
```

ii. Do not ignore NULLs

```
SELECT AVG(NVL(Commission, 0)) AS AvgCommission FROM employee;


AVGCOMMISSION
-------------
          300
```

8. Find 2 to the power 10.

```
SELECT POWER(2, 10) AS Result FROM dual;

    RESULT
----------
      1024
```

9. Display courses and prerequisites. If there is no prerequisite, display 'none', else display 'one'.

```
SELECT Title AS Course, NVL2(PreReq, 'one', 'none') AS Prerequisite FROM course;

COURSE               PRER
-------------------- ----
Database Systems     none
Thermodynamics       one
Circuit Analysis     one
Organic Chemistry    one
Data Structures      one
```

10. Find the number of years employees have been working for. Display integer part of value only.

```
SELECT EmployeeId, FLOOR(MONTHS_BETWEEN(SYSDATE, HireDate) / 12) AS YearsWorked FROM employee;

EMPLOYEEID YEARSWORKED
---------- -----------
       111           7
       222           4
       333           3
       444           2
       555           1
```

11. Find students who are born in the month of May.

```
SELECT StudentId, First, Last, BirthDate FROM student WHERE EXTRACT(MONTH FROM BirthDate) = 5;

EMPLOYEEID YEARSWORKED
---------- -----------
       111           7
       222           4
       333           3
       444           2
       555           1
```

12. Display employee's last name and first name, followed by salary+commission if commission is not null, else display salary only.

```
SELECT Lname, Fname, Salary + NVL(Commission, 0) AS TotalEarnings FROM employee;
```

```
LNAME                FNAME                TOTALEARNINGS
-------------------- -------------------- -------------
Brown                Chris                        60000
Green                Alex                         40500
Taylor               Jordan                       45000
Martinez             Olivia                       51000
Lopez                Sophia                       75000
```

13. Display employee's full name followed by a message based on salary.

```
SELECT Fname, Lname,
       CASE
           WHEN Salary > 100000 THEN 'HIGH'
           WHEN Salary BETWEEN 50000 AND 100000 THEN 'MEDIUM'
           ELSE 'LOW'
       END AS SalaryCategory
FROM employee;
```

```
FNAME                LNAME                SALARY
-------------------- -------------------- ------
Chris                Brown                MEDIUM
Alex                 Green                LOW
Jordan               Taylor               LOW
Olivia               Martinez             MEDIUM
Sophia               Lopez                MEDIUM
```