1. Write a Oracle SQL query to display the designation (job id) and total number of employees working in each designation.

**Select job\_id, count(job\_id) from employees group by job\_id;**

2. Write a query to list the name of all the employees who are not working in department number 20.

**SELECT empname, department\_id FROM employees**

**WHERE department\_id =20;**

3. Write a query to list the name of all the employees who are working as account manager and drawing a salary more than $5000.

**SELECT empname, salary FROM employees**

**WHERE job\_id='AC\_MGR' and salary>5000;**

4. Write a query to list the names (first and last), salary of those employees whose salary is four digit number ending with Zero.

**SELECT empname, salary FROM employees**

**WHERE length (salary) = 4 and salary like ’%0’;**

5. Write a Oracle SQL query to get the department number with more than 10 employees in each department.

**SELECT department\_id, count(department\_id) as "Number of Employees" from employees group by department\_id having count(\*) > 10;**

6. Write a Oracle SQL query to get the designations (jobs) along with the total salary for those designations where total salary is more than 50000.

**SELECT job\_id, sum(salary) as "Total Salary of Designation" from employees group by job\_id having sum(salary) > 50000;**

7. Write a query that displays the difference between the highest and lowest salaries of employees. Label the column DIFFERENCE.

**SELECT MAX(salary) - MIN(salary) DIFFERENCE FROM employees;**

8. Create a query to display the employee id and names of all employees who earn more than the average salary. Sort the results in descending order of salary.(Subquery)

**SELECT employee\_id, empname FROM employees**

**WHERE salary > (SELECT AVG(salary) FROM employees)**

**ORDER BY salary DESC;**

9. Display the last name, department number, and job ID of all employees whose department location ID is 1700. (Subquery)

**select empname, department\_id, job\_id from employees**

**where department\_id in (select department\_id from departments where location\_id =1700);**

10. Add a table-level PRIMARY KEY constraint to the EMP table on the ID column. The constraint should be named at creation. Name the constraint my\_emp\_id\_pk

**ALTER TABLE emp**

**ADD CONSTRAINT my\_emp\_id\_pk PRIMARY KEY (id);**

11. Add a column DEPT\_ID to the EMP table. Add a foreign key reference on the EMP table that ensures that the employee is not assigned to a nonexistent department. Name the constraint my\_emp\_dept\_id\_fk.

**ALTER TABLE emp ADD (dept\_id NUMBER(7));**

**ALTER TABLE emp ADD CONSTRAINT my\_emp\_dept\_id\_fk**

**FOREIGN KEY (dept\_id) REFERENCES dept(id);**

12. Modify the EMP table. Add a COMMISSION column of NUMBER data type, precision 2, scale 2.Add a constraint to the commission column that ensures that a commission value is greater than zero.

**ALTER TABLE EMP ADD commission NUMBER(2,2)**

**CONSTRAINT my\_emp\_comm\_ck CHECK (commission >= 0;**

13. Create a view called EMPLOYEES\_VU based on the employee numbers, employee names, and department numbers from the EMPLOYEES table. Change the heading for the employee name to EMPLOYEE.

**CREATE OR REPLACE VIEW employees\_vu AS**

**SELECT employee\_id, last\_name employee, department\_id**

**FROM employees;**

**Display the contents of the EMPLOYEES\_VU view.**

**SELECT \* FROM employees\_vu;**

14. Write a PL/SQL program to check whether a number is even or odd.

15. Write a PL/SQL program to find the average of first 10 numbers

16. Write a PL/SQL program to display the grade(Nested if)

17. Write a program in PL/SQL to display a cursor based detail information of employees from employees table.