#1: SQL ASSESSMENT

1. Create a SQL statement to list all managers and their titles.

```sql
SELECT CONCAT(e.first_name, ' ', e.last_name) AS department_manager,
title
FROM employees e
INNER JOIN dept_manager dm ON dm.emp_no = e.emp_no
LEFT JOIN titles t ON t.emp_no = dm.emp_no;
```

- Used an INNER JOIN to pull matching records from both employees and dept_manager tables
- Used a LEFT JOIN to pull all records from the dept_manager including records that matched with the titles table

2. Create a SQL statement to show the salary of all employees and their department name.

```sql
SELECT CONCAT(first_name, ' ',last_name) AS full_name, s.salary,
d.dept_name
FROM employees e
LEFT JOIN salaries s ON e.emp_no = s.emp_no
LEFT JOIN dept_emp de ON e.emp_no = de.emp_no
LEFT JOIN departments d ON de.dept_no = d.dept_no;

```

- SELECTED full name, salary, and dept_name of all employees
- Used LEFT JOIN to pull all/matching records from employees, salaries, dept_emp, and departments table

3. Create a SQL statement to show the hire date and birth date of those who belong to the HR department

```sql
SELECT dept_name, hire_date, birth_date
FROM employees e
INNER JOIN dept_manager dm ON e.emp_no = dm.emp_no
INNER JOIN departments d ON dm.dept_no = d.dept_no
WHERE dept_name = "Human Resources"

UNION

SELECT dept_name, hire_date, birth_date
FROM employees e
INNER JOIN dept_emp de ON e.emp_no = de.emp_no
INNER JOIN departments d ON de.dept_no = d.dept_no
WHERE dept_name = "Human Resources";
```

- Used INNER JOIN to select all matching records from employee, dept_manager, and departments tables
- Used WHERE clause to display managers in the HR department

UNION - selects distinct values from both statements and joins them together.

- Used INNER JOIN to select all matching records from employees, dept_emp, and departments tables
- Used WHERE clause to display employees in the HR department

4. Create a SQL statement to show all departments and their department's managers.

```sql
SELECT d.dept_name, CONCAT(e.first_name," ",e.last_name)
AS department_manager
FROM departments d
LEFT JOIN dept_manager dm ON dm.dept_no = d.dept_no
LEFT JOIN employees e ON e.emp_no = dm.emp_no;
```

- Used LEFT JOIN to select all records from departments and dept_manager tables including records that match with the employees table.

5. Create a SQL statement to show a list of HR employees who were hired after 1986

```sql
SELECT CONCAT(first_name, ' ', last_name) AS full_name, dept_name,
hire_date
FROM employees e
INNER JOIN dept_emp de ON de.emp_no= e.emp_no
```

- Used an INNER JOIN to select all matching records from employees, dept_emp, and departments tables

```
INNER JOIN departments d ON d.dept_no= de.dept_no
WHERE dept_name = 'Human Resources'
AND YEAR(e.hire_date)> '1986';
```

- Used WHERE to filter out employees in the HR department only
- ‘=>’ greater than operator to filter hire_dates after 1986

6. Create a SQL statement to increase any employee’s salary up to 2%. Assume the the employee has just phoned in with his/her last name.

```
DELIMITER $$
CREATE PROCEDURE salary_increase(last_name VARCHAR(30))
BEGIN
SELECT e.last_name, e.emp_no, s.salary AS current_salary,
s.salary*1.02 AS new_salary
FROM employees e
LEFT JOIN salaries s on e.emp_no = s.emp_no
WHERE e.last_name = last_name;
END $$
DELIMITER;

CALL salary_increase('last_name');
```

- Created a stored procedure to increase an employee's salary by 2% with just their last name. - can be used in the future for other employees using just their last name
- LEFT JOIN to pull all records from the employees' table and include matching records from the salaries table
- Used WHERE clause to filter records to display salaries according to only their last_name

7. Create a SQL statement to delete an employee’s record who belongs to the marketing department and name start with A.

```
CREATE TEMPORARY TABLE marketing_na mn (
SELECT d.*, e.*
FROM employees e
LEFT JOIN dept_emp de ON de.emp_no = e.emp_no
LEFT JOIN dept_manager dm ON dm.emp_no = e.emp_no
LEFT JOIN departments d ON d.dept_no = de.dept_no
OR d.dept_no = dm.dept_no
);

DELETE FROM marketing_na mn
WHERE mn.dept_name = 'Marketing' AND mn.first_name LIKE 'A%';
```

- Created a temporary table so no records are altered in the original database.
- SELECTED all data from departments and employees table, as they need to be deleted.
- Used LEFT JOIN to link records from dept_emp, dept_manager, and departments table to list all employees/managers alongside their departments.
- Used WHERE clause to filter out employees in Marketing and used the “A%” wildcard to find names beginning with 'A'
- Used DELETE to remove all records
- OUTCOME: empty set

8. Create a database view to list the full names of all departments’ managers and their salaries.

```
CREATE VIEW manager_sal AS
SELECT CONCAT(e.first_name, ' ', e.last_name) AS full_name,
d.dept_name, s.salary
FROM dept_manager dm
INNER JOIN employees e ON dm.emp_no = e.emp_no
INNER JOIN departments d ON dm.dept_no = d.dept_no
INNER JOIN salaries s ON dm.emp_no = s.emp_no;

SELECT * FROM manager_sal;
```

- VIEW - to create a view
- Used INNER JOIN to select all matching records from dept_manager, employees, departments, and salaries tables
- OUTCOME:

9. Create a database view to list all departments and their department managers, who were hired between 1980 and 1990.

```
CREATE VIEW manager_depts AS

SELECT d.dept_name, CONCAT(e.first_name, ' ', e.last_name) AS
department_manager, YEAR(e.hire_date) AS year_hired
FROM dept_manager dm
INNER JOIN employees e ON dm.emp_no = e.emp_no
INNER JOIN departments d ON dm.dept_no = d.dept_no
WHERE YEAR(e.hire_date) BETWEEN '1980' AND '1990';

SELECT * FROM manager_depts;
```

- VIEW - to create a view
- INNER JOIN - select matching records from dept_manager, employees, and departments tables
- WHERE - filter hire_date using YEAR, which only displays the year.
- BETWEEN - used to select data from a range, between the year 1980 and 1990.

10. Create a SQL statement to increase salaries of all department managers up to 10% who are working since 1990.

```
CREATE TEMPORARY TABLE managers_inc(

SELECT CONCAT(e.first_name, ' ', e.last_name) AS name,
d.dept_name, e.hire_date as hired,
s.salary * 1.1 AS new_salary
FROM employees e
LEFT JOIN dept_manager dm ON dm.emp_no = e.emp_no
LEFT JOIN departments d ON d.dept_no = dm.dept_no
LEFT JOIN salaries s ON s.emp_no = e.emp_no
WHERE YEAR(hire_date)>=1990);

SELECT * FROM managers_inc;
```

- Created a TEMPORARY TABLE so information in the database is not altered.
- *1.1 (100 +10) - to increase salary by 10%
- Used LEFT JOIN to pull all records from employees, dept_manager, and departments tables including matching records
- Used WHERE clause to filter hire-date, equals to or less than 1990