

DATA ANALYST INTERNSHIP

Task 3: SQL for Data Analysis:

- Objective: Use SQL queries to extract and analyze data from a database.
- Tools: MySQL or PostgreSQL or SQLite
- Deliverables: SQL queries in a SQL file + screenshots of output
- Hints/Mini Guide:
 - a. Use SELECT, WHERE, ORDER BY, GROUP BY
 - b. Use JOINS (INNER, LEFT, RIGHT)
 - c. Write subqueries
 - d. Use aggregate functions (SUM, AVG)
 - e. Create views for analysis
 - f. Optimize queries with indexes
- Dataset: Ecommerce_SQL_Database(or any data set of your choice)
- Outcome: Learn to manipulate and query structured data using SQL.

Tool Used : SQLPlus

Schema Used : SCOTT

```
SQL> SELECT *
2 FROM EMP;
```

| EMPNO | ENAME | JOB | MGR | HIREDATE | SAL | COMM | DEPTNO |
|-------|--------|-----------|------|-----------|------|------|--------|
| 7369 | SMITH | CLERK | 7902 | 17-DEC-80 | 800 | | 20 |
| 7499 | ALLEN | SALESMAN | 7698 | 20-FEB-81 | 1600 | 300 | 30 |
| 7521 | WARD | SALESMAN | 7698 | 22-FEB-81 | 1250 | 500 | 30 |
| 7566 | JONES | MANAGER | 7839 | 02-APR-81 | 2975 | | 20 |
| 7654 | MARTIN | SALESMAN | 7698 | 28-SEP-81 | 1250 | 1400 | 30 |
| 7698 | BLAKE | MANAGER | 7839 | 01-MAY-81 | 2850 | | 30 |
| 7782 | CLARK | MANAGER | 7839 | 09-JUN-81 | 2450 | | 10 |
| 7788 | SCOTT | ANALYST | 7566 | 19-APR-87 | 3000 | | 20 |
| 7839 | KING | PRESIDENT | | 17-NOV-81 | 5000 | | 10 |
| 7844 | TURNER | SALESMAN | 7698 | 08-SEP-81 | 1500 | 0 | 30 |
| 7876 | ADAMS | CLERK | 7788 | 23-MAY-87 | 1100 | | 20 |
| 7900 | JAMES | CLERK | 7698 | 03-DEC-81 | 950 | | 30 |
| 7902 | FORD | ANALYST | 7566 | 03-DEC-81 | 3000 | | 20 |
| 7934 | MILLER | CLERK | 7782 | 23-JAN-82 | 1300 | | 10 |

14 rows selected.

```
SQL> SELECT *
2 FROM DEPT;
```

| DEPTNO | DNAME | LOC |
|--------|------------|----------|
| 10 | ACCOUNTING | NEW YORK |
| 20 | RESEARCH | DALLAS |
| 30 | SALES | CHICAGO |
| 40 | OPERATIONS | BOSTON |

1. SELECT with WHERE and ORDER BY

```
SQL> SELECT * FROM EMP
2 WHERE SAL > 2000
3 ORDER BY HIREDATE DESC;
```

| EMPNO | ENAME | JOB | MGR | HIREDATE | SAL | COMM | DEPTNO |
|-------|-------|-----------|------|-----------|------|------|--------|
| 7788 | SCOTT | ANALYST | 7566 | 19-APR-87 | 3000 | | 20 |
| 7902 | FORD | ANALYST | 7566 | 03-DEC-81 | 3000 | | 20 |
| 7839 | KING | PRESIDENT | | 17-NOV-81 | 5000 | | 10 |
| 7782 | CLARK | MANAGER | 7839 | 09-JUN-81 | 2450 | | 10 |
| 7698 | BLAKE | MANAGER | 7839 | 01-MAY-81 | 2850 | | 30 |
| 7566 | JONES | MANAGER | 7839 | 02-APR-81 | 2975 | | 20 |

6 rows selected.

```
SQL> -- 2. GROUP BY with aggregate
SQL> SELECT DEPTNO, AVG(SAL) AS AVG_SALARY
2 FROM EMP
3 GROUP BY DEPTNO;
```

| DEPTNO | AVG_SALARY |
|--------|------------|
| 30 | 1566.66667 |
| 20 | 2175 |
| 10 | 2916.66667 |

2. b. JOINS (INNER, LEFT, RIGHT)

```
SQL> -- 3. INNER JOIN between EMP and DEPT
SQL> SELECT E.ENAME, E.SAL, D.DNAME, D.LOC
2 FROM EMP E
3 INNER JOIN DEPT D ON E.DEPTNO = D.DEPTNO;
```

| ENAME | SAL | DNAME | LOC |
|--------|------|------------|----------|
| CLARK | 2450 | ACCOUNTING | NEW YORK |
| KING | 5000 | ACCOUNTING | NEW YORK |
| MILLER | 1300 | ACCOUNTING | NEW YORK |
| JONES | 2975 | RESEARCH | DALLAS |
| FORD | 3000 | RESEARCH | DALLAS |
| ADAMS | 1100 | RESEARCH | DALLAS |
| SMITH | 800 | RESEARCH | DALLAS |
| SCOTT | 3000 | RESEARCH | DALLAS |
| WARD | 1250 | SALES | CHICAGO |
| TURNER | 1500 | SALES | CHICAGO |
| ALLEN | 1600 | SALES | CHICAGO |
| JAMES | 950 | SALES | CHICAGO |
| BLAKE | 2850 | SALES | CHICAGO |
| MARTIN | 1250 | SALES | CHICAGO |

14 rows selected.

```
SQL> -- 4. LEFT JOIN between EMP and DEPT
SQL> SELECT E.ENAME, D.DNAME
  2   FROM EMP E
  3  LEFT OUTER JOIN DEPT D ON E.DEPTNO = D.DEPTNO;
```

| ENAME | DNAME |
|--------|------------|
| MILLER | ACCOUNTING |
| KING | ACCOUNTING |
| CLARK | ACCOUNTING |
| FORD | RESEARCH |
| ADAMS | RESEARCH |
| SCOTT | RESEARCH |
| JONES | RESEARCH |
| SMITH | RESEARCH |
| JAMES | SALES |
| TURNER | SALES |
| BLAKE | SALES |
| MARTIN | SALES |
| WARD | SALES |
| ALLEN | SALES |

14 rows selected.

```
SQL> -- 5. RIGHT JOIN between DEPT and EMP
SQL> SELECT D.DNAME, E.ENAME
  2   FROM DEPT D
  3  RIGHT OUTER JOIN EMP E ON D.DEPTNO = E.DEPTNO;
```

| DNAME | ENAME |
|------------|--------|
| ACCOUNTING | MILLER |
| ACCOUNTING | KING |
| ACCOUNTING | CLARK |
| RESEARCH | FORD |
| RESEARCH | ADAMS |
| RESEARCH | SCOTT |
| RESEARCH | JONES |
| RESEARCH | SMITH |
| SALES | JAMES |
| SALES | TURNER |
| SALES | BLAKE |
| SALES | MARTIN |
| SALES | WARD |
| SALES | ALLEN |

14 rows selected.

3. c. Subqueries

```
SQL> -- 6. Subquery: Get employee names who work in 'DALLAS'
SQL> SELECT ENAME
2  FROM EMP
3  WHERE DEPTNO IN (
4      SELECT DEPTNO FROM DEPT
5      WHERE LOC = 'DALLAS'
6  );
```

ENAME

JONES
FORD
ADAMS
SMITH
SCOTT

4. d. Aggregate functions (SUM, AVG)

```
SQL> -- 7. Total and Average salary per department
SQL> SELECT DEPTNO, SUM(SAL) AS TOTAL_SAL, AVG(SAL) AS AVG_SAL
2  FROM EMP
3  GROUP BY DEPTNO;
```

| DEPTNO | TOTAL_SAL | AVG_SAL |
|--------|-----------|------------|
| ----- | ----- | ----- |
| 30 | 9400 | 1566.66667 |
| 20 | 10875 | 2175 |
| 10 | 8750 | 2916.66667 |

5. e. Create Views for Analysis

```
SQL> -- 8. Create a view for employees with high salaries
SQL> CREATE OR REPLACE VIEW high_salary_emp AS
2  SELECT ENAME, SAL, DEPTNO
3  FROM EMP
4  WHERE SAL > 3000;
```

View created.

```
SQL> -- 9. Query the view
SQL> SELECT * FROM high_salary_emp
2  ORDER BY SAL DESC;
```

| ENAME | SAL | DEPTNO |
|-------|-------|--------|
| ----- | ----- | ----- |
| KING | 5000 | 10 |

6. f. Optimize Queries with Indexes

```
SQL> CREATE INDEX idx_emp_dept_sal ON EMP(DEPTNO, SAL);  
Index created.  
  
SQL> SELECT ENAME FROM EMP WHERE DEPTNO = 10 AND SAL > 3000;  
  
ENAME  
-----  
KING  
  
SQL> SPOOL OFF  
SQL> SPOOL OFF;  
not spooling currently  
SQL> |
```