

# SQL

**HARI PRASATH S**

**225229110**

## **LAB 4 Retail sales analytics part 1**

**1).**

SQL> desc store\_data;

Name	Null?	Type
-----		
STORE_ID	NOT NULL	NUMBER(38)
TYPE		VARCHAR2(26)
SIZE36		NUMBER(38)

SQL> desc sales\_data;

Name	Null?	Type
-----		
SALES_ID	NOT NULL	NUMBER(38)
STORE_ID		NUMBER(38)
DEPT		NUMBER(38)
DATE36		DATE
WEEKLY_SALES		NUMBER(38,2)
ISHOLIDAY		VARCHAR2(26)

SQL> desc features\_data;

Name	Null?	Type
-----		
STORE_ID		NUMBER(38)
DATE36		DATE

TEMPERATURE	NUMBER(38,2)
FUEL_PRICE	NUMBER(38,3) MARKDOWN1
MARKDOWN2	NUMBER(38,2)
MARKDOWN3	NUMBER(38,2)
MARKDOWN4	NUMBER(38,2)
MARKDOWN5	NUMBER(38,2)
CPI	NUMBER(38,7)
UNEMPLOYMENT	NUMBER(38,3)
ISHOLIDAY	VARCHAR2(26)

## 2).

```
SQL> alter table store_data add primary key(store_id);
```

Table altered.

```
SQL> alter table features_data add foreign key(store_id) references store_data(store_id);
```

Table altered.

```
SQL> alter table sales_data add foreign key(store_id) references store_data(store_id);
```

Table altered.

## 3).

```
SQL> select sales_id,dept from sales_data where sales_id<=30 order by sales_id;
```

SALES_ID	DEPT
1	1
2	1

3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1

SALES_ID	DEPT
12	1
13	1
14	1
15	1
16	1
17	1
18	1
19	1
20	1
21	1
22	1

SALES_ID	DEPT
23	1
24	1
25	1
26	1
27	1

28	1
29	1
30	1

30 rows selected.

```
SQL> select sales_id,dept,weekly_sales from sales_data where weekly_sales<10 group by
sales_id,dept,weekly_sales order by weekly_sales;
```

SALES_ID	DEPT	WEEKLY_SALES
267731	6	-4988.94
336496	47	-3924
417802	47	-1750
153917	47	-1699
271301	32	-1321.48
16292	47	-1098
16296	47	-1098
16298	47	-1098
24831	32	-1008.96
35562	47	-898
6049	47	-863

SALES_ID	DEPT	WEEKLY_SALES
93574	47	-798
93582	47	-798
93592	47	-798
193326	47	-798
193321	47	-778.5
332895	16	-771.9
273031	47	-705

6050	47	-698
54836	47	-698

6727 rows selected.

## NOSQL Database Management Lab

### Lab5. Retail Sales Analytics Part-II

**Question1:** Write 3 queries with at least 1 join per query.

```
SQL> select stores.store_id,stores.type,sales.dept from stores left join sales on  
stores.store_id=sales_id where stores.store_id<=2;
```

STORE_ID	TYPE	DEPT
1	A	1
2	A	1

```
SQL> select stores.store_id,stores.type,sales.dept from stores inner join sales on  
stores.store_id=sales_id where stores.store_id<=2 and sales_id<=2;
```

STORE_ID	TYPE	DEPT
1	A	1
2	A	1

```
SQL> select stores.store_id,stores.type,sales.dept from stores inner join sales on  
stores.store_id=sales_id where stores.type='B';
```

STORE_ID	TYPE	DEPT
3	B	1
5	B	1
7	B	1
9	B	1
10	B	1

12	B	1
15	B	1
16	B	1
17	B	1
18	B	1
21	B	1
22	B	1
23	B	1
25	B	1
29	B	1
35	B	1
45	B	1

**Question2:** Write at least 3 queries that must use outer joins.

SQL> select sales.sales\_id,weekly\_sales from sales full outer join stores on stores.store\_id=sales\_id where sales\_id<=2 and stores.store\_id<=2;

SALES_ID	WEEKLY_SALES
1	24924.5
2	46039.49

SQL> select sales.sales\_id,weekly\_sales from sales full outer join stores on dept=stores.store\_id where sales\_id<=2 and stores.type='A';

SALES_ID	WEEKLY_SALES
1	24924.5
2	46039.49

SQL> select sales.sales\_id,weekly\_sales from sales full outer join stores on sales\_id=stores.store\_id where isholiday='TRUE' and sales\_id<=10;

SALES_ID	WEEKLY_SALES
2	46039.49

## Lab 6: NoSQL Database Management Lab

**Question:** Develop aggregate queries in SQL on Retail dataset as follows

- Write 6 SQL queries with aggregation.
- At least 1 aggregate function per query.

- At least 2/5 aggregate functions among the 6 queries.
- At least 4 GROUP BY clauses among the 6 queries.
- At least 3 HAVING clauses among the 6 queries.

SQL> select count(store\_id),count(sales\_id),count(dept) from sales;

COUNT(STORE_ID)	COUNT(SALES_ID)	COUNT(DEPT)
421570	421570	421570

SQL> select store\_id,max(temperature),min(temperature) from features where store\_id=5 group by store\_id;

STORE_ID	MAX(TEMPERATURE)	MIN(TEMPERATURE)
5	91.07	37.74

SQL> select avg(weekly\_sales),count(store\_id) from sales group by store\_id having store\_id=5;

AVG(WEEKLY_SALES)	COUNT(STORE_ID)
5053.41581	8999

SQL> select avg(temperature),avg(fuel\_price) from features group by store\_id having store\_id=5;

AVG(TEMPERATURE)	AVG(FUEL_PRICE)
68.2245055	3.25924176

SQL> select max(fuel\_price),min(fuel\_price) from features group by store\_id having store\_id=5;

MAX(FUEL_PRICE)	MIN(FUEL_PRICE)
3.907	2.514

SQL> select avg(size\_),max(size\_),min(size\_) from stores;

AVG(SIZE_)	MAX(SIZE_)	MIN(SIZE_)
130287.6	219622	34875