

REPORT

I Have completed my SQL lab with help of notes provided during the regular online classes as well as reference book shared in class that is “book **BeginningSQLqueries** by Clare Churcher”.

I got required the output (correct format) after rectifying silly error in almost each queries. I Refer website such as <https://stackoverflow.com/>, <https://www.geeksforgeeks.org/> <https://www.w3schools.com/>, to get know different ways of writing queies.

Department of Data Science, Bishop Heber College Tiruchirappalli
NoSQL Database Management Lab

Lab3. India Weather Analytics using Historical Data Part-II

Objectives:

In this lab, you are going to explore further India Weather Dataset with additional query operators such as GROUPBY, HAVING and ORDERBY

Tasks To Be Completed:

Question1: Write 3 Queries using WHERE and GROUPBY clause

Question2: Write 3 Queries using WHERE, GROUPBY and HAVING clause

Question3: Write 3 Queries using WHERE, GROUPBY, HAVING and ORDERBY clause

Sample Queries

1. What is the average weather in May of all cities?. Display city and average temperature in descending order.

SQL> Select city, avg(temperature) from weather-india where month=5 group by city order by city desc;

2. Show the average historic temperature (from year 1995 to Feb 2020, entire table) in each city in ascending order of city name:

SQL> Select city, avg(temperature) from weather-india where year between 1995 and 2020 and month=2 group by city order by city;

3. Show lowest, highest and average temperature in Kolkata during 2010 to 2020.

SQL> select min(temperature), max(temperature), avg(temperature)
from weather-india where year between 2010 and 2020
2 and city = 'Kolkata';

4. Find cities and average temperature which recorded atleast 40 degree Celsius in April 2019.

SQL> select city, avg(temperature) from weather-india where
temperature <= 40 and year = 2019 and month = 4 group
by city;

5. Show monthwise average temperature in Chennai in 2019. Print month name and average
temperature values

SQL> select mon, month, avg(temperature) from india-weather
data where year = 2019 and city = 'chennai' group by
mon, month order by month asc;

6. Show year wise average temperature of Mumbai. Print year and corresponding average temperature values, in descending order

SQL> Select year, avg(temperature) from weather-india where city = 'mumbai' group by year order by year desc;

7. Show city wise yearly average temperature values for the years 2017, 2018 and 2019. City names as rows and years as columns. Each cell will denote its average temperature value.

SQL> select city, year, avg(temperature) from weather-india where year between 2017 and 2019 group by city, year order by city, year asc;

LAB:3 INDIA WEATHER ANALYTICS USING HISTORICAL DATA PART-||

Question1: Write 3 queries using WHERE and GROUPBY clause

(1) select avg(temperature), city from weather_india where day=9 group by city;

AVG(TEMPERATURE) CITY

82.8665563	chennai
81.2976821	mumbai
75.7834437	delhi
79.1301325	kolkata

(2) select city, count(city) from weather_india where temperature='90.0' group by city;

CITY COUNT(CITY)

chennai	38
mumbai	5
delhi	34
kolkata	18

(3) select city, max(temperature) from weather_india where year between 2010 and 2015 group by city;

CITY MAX(TEMPERATURE)

chennai	96
mumbai	92.6
delhi	102.5
kolkata	95.2

Question2: Write 3 Queries using WHERE, GROUPBY and HAVING clause

- (1) select sum(temperature), month from weather_india where year between 2000 and 2010 group by month having month in (10,11,12,1,2,3);

SUM(TEMPERATURE) MONTH

SUM(TEMPERATURE)	MONTH
93502.3	1
100515.6	11
91268.2	2
107512.6	3
110221.6	10
95674.2	12

- (2) select count(city) as "no. of days", city from weather_india where year between 1999 and 2017 and temperature < 100 group by city having city not in 'chennai';

no. of days CITY

6941 mumbai
6898 delhi
6941 kolkata

- (3) select avg(temperature), year from weather_india where city not in 'chennai' group by year having year >= 2000;

AVG(TEMPERATURE) YEAR

AVG(TEMPERATURE)	YEAR
78.9787215	2005
79.7507763	2009
79.7721461	2010
79.1148858	2013
78.4954338	2018
78.5444749	2001
79.6859361	2006
79.349589	2014
80.3903196	2017
76.5238356	2019

77.5646575 2007

AVG(TEMPERATURE) YEAR

78.7945355 2000
78.5621005 2011
79.0432877 2003
79.4233151 2004
79.5721311 2015
79.9728597 2016
67.6222222 2020
77.5352511 2002
78.3162113 2008
79.5935337 2012

Question3: Write 3 Queries using WHERE, GROUPBY, HAVING and ORDERBY clause

- (1) select min(temperature),year from weather_india where temperature > -99 group by year having year <= 2000 order by year;

MIN(TEMPERATURE) YEAR

50.7 1995
50.3 1996
45.4 1997
49.1 1998
48.9 1999
47.7 2000

- (2) select city,avg(temperature) from weather_india where year between 1995 and 2005 group by city having avg(temperature) >50 order by city;

CITY AVG(TEMPERATURE)

chennai 82.4170817
delhi 75.6950722
kolkata 78.5704331
mumbai 81.0530114

(3) select month,sum(temperature) from weather_india where year between 2000 and 2015 group by month having sum(temperature) >50000 order by month;

MONTH SUM(TEMPERATURE)

1	135863.9
2	131723
3	157025.5
4	162124
5	174671.3
6	164098.9
7	169385.1
8	166761.4
9	160322.3
10	161255
11	146546.3
12	138721.3

INDIAN WEATHER ANALYTICS USING HISTORICAL DATA PART-||

SAMPLE QUERIES:

Question1: SQL> select city,avg(temperature)from weather_india where month=5 group by city order by city desc;

CITY	AVG(TEMPERATURE)

mumbai	85.2651613
kolkata	85.888
delhi	89.6534194
chennai	88.636

Question2: SQL> select city,avg(temperature) from weather_india where year between 1995 and 2020 and month=2 2 group by city order by city;

CITY	AVG(TEMPERATURE)

chennai	79.140192
delhi	62.8658436
kolkata	72.6458162
mumbai	78.2491084

Question3: SQL> select min(temperature),max(temperature),avg(temperature) from weather_india where year between 2010 and 2020 2 and city='kolkata';

MIN(TEMPERATURE)	MAX(TEMPERATURE)	AVG(TEMPERATURE)
-99	96.3	79.0960345

Question4: SQL> select city,avg(temperature) from weather_india where temperature<=40 and year=2019 and month=4 group by city;

CITY	AVG(TEMPERATURE)
------	------------------

chennai	-99
mumbai	-99
delhi	-99
kolkata	-99

Question5: SQL> select mon,month,avg(temperature) from indian_weatherdata where year=2019 and city='chennai' group by mon,month order by month asc;

MON	MONTH	AVG(TEMPERATURE)
-----	-------	------------------

January	1	77.3451613
February	2	82.2678571
March	3	85.9064516
April	4	82.9
May	5	73.6064516
June	6	92.5466667
July	7	88.8967742
August	8	88.3709677
September	9	85.2066667
October	10	83.2806452
November	11	82.52

MON	MONTH	AVG(TEMPERATURE)
-----	-------	------------------

December	12	79.6225806
----------	----	------------

12 rows selected.

Question6: SQL> select year,avg(temperature) from weather_india where city='mumbai' group by year order by year desc;

YEAR AVG(TEMPERATURE)

YEAR	AVG(TEMPERATURE)
2020	78.962963
2019	79.8649315
2018	82.2526027
2017	83.4043836
2016	81.8393443
2015	82.6166667
2014	82.2515068
2013	81.7391781
2012	81.6964481
2011	82.2846575
2010	82.6871233

YEAR AVG(TEMPERATURE)

YEAR	AVG(TEMPERATURE)
2009	82.5021918
2008	80.492623
2007	81.4682192
2006	81.3005479
2005	81.2624658
2004	80.6027322
2003	81.4369863
2002	80.1052055
2001	81.0630137
2000	81.7103825
1999	81.2789041

YEAR AVG(TEMPERATURE)

```
-----  
1998    80.0279452  
1997    81.7857534  
1996    81.745082  
1995    80.5621918
```

26 rows selected.

Question7: SQL> select city,year,avg(temperature) from weather_india where year between 2017 and 2019 group by city,year order by city,year asc;

CITY YEAR AVG(TEMPERATURE)

```
-----  
chennai               2017   84.7586301  
chennai               2018   83.8887671  
chennai               2019   83.5249315  
delhi                 2017   77.9082192  
delhi                 2018   75.099726  
delhi                 2019   73.4953425  
kolkata               2017   79.8583562  
kolkata               2018   78.1339726  
kolkata               2019   76.2112329  
mumbai                2017   83.4043836  
mumbai                2018   82.2526027
```

CITY YEAR AVG(TEMPERATURE)

```
-----  
mumbai               2019   79.8649315
```

12 rows selected.

REPORT

I Have completed my SQL lab with help of notes provided during the regular online classes as well as reference book shared in class that is “book **BeginningSQLqueries** by Clare Churcher”.

I got required the output (correct format) after rectifying silly error in almost each queries. I Refer website such as <https://stackoverflow.com/>, <https://www.geeksforgeeks.org/> <https://www.w3schools.com/>, to get know different ways of writing queies.

Department of Data Science, Bishop Heber College Tiruchirappalli
NoSQL Database Management Lab

Lab4. Retail Sales Analytics Part-I

Objectives

In this lab, you will be performing analytics on retail sales of a particular enterprise. The sales of several of their stores are represented using three Excel files.

The Problem

One challenge of modelling retail data is the need to make decisions based on limited history. Holidays and select major events come once a year, and so does the chance to see how strategic decisions impacted the bottom line. In addition, markdowns are known to affect sales – the challenge is to predict which departments will be affected and to what extent.

Data Description

You are provided with historical sales data for 45 stores located in different regions - each store contains a number of departments. The company also runs several promotional markdown events throughout the year. These markdowns precede prominent holidays, the four largest of which are the Super Bowl, Labor Day, Thanksgiving, and Christmas. The weeks including these holidays are weighted five times higher in the evaluation than non-holiday weeks.

Attributes of Stores, Features and Sales Tables

1. Stores

Anonymized information about the 45 stores, indicating the type and size of store

2. Features

Contains additional data related to the store, department, and regional activity for the given dates.

- Store - the store number
- Date - the week
- Temperature - average temperature in the region
- Fuel_Price - cost of fuel in the region
- MarkDown1-5 - anonymized data related to promotional markdowns. MarkDown data is only available after Nov 2011, and is not available for all stores all the time. Any missing value is marked with an NA
- CPI - the consumer price index
- Unemployment - the unemployment rate
- IsHoliday - whether the week is a special holiday week

3. Sales

Historical sales data, which covers from 2010-02-05 to 2012-11-01. Within this tab you will find the following fields:

- Store - the store number
- Dept - the department number
- Date - the week
- Weekly_Sales - sales for the given department in the given store
- IsHoliday - whether the week is a special holiday week

Some Sample Tasks

1. Predict the department-wide sales for each store for the following year
2. Model the effects of markdowns on holiday weeks
3. Provide recommended actions based on the insights drawn, with prioritization placed on largest business impact

Tasks To be Completed:

Question1: Describe the data, making note of the interesting attributes and relationships in the data. Include 3-5 rows of sample data from each file. Very important: explain what insights you hope to gain from analyzing the data.

Question2: Create tables from Excel files. Relate the tables via primary key and foreign key wherever required.

Question3: Write atleast 2 queries to explore each table by using WHERE and ORDERBY

Question:2

SQL> alter table sales add primary key (Sales_id);
Table altered.

SQL> alter table store add foreign key (Store_id) references
sales (Sales_id);
Table altered.

Question 3:

SQL> Select store-size from store where type = 'A' order by
store-size asc;

SQL> Select store-id from store where type = 'B' order by
store-size;

SQL> Select dept, ^{Sales}date from sales where dept = 21 and
isholiday = 'TRUE' order by weekly-Sales;

SQL> Select storeid,dept, weekly-Sales from sales where
sales-id between 20 & 30 order by weekly-Sales desc;

SQL> Select store, unemployment, temperature from feature
where store = 21 and unemployment < 7 order by
temperature;

LAB4. RETAIL SALES ANALYTICS PART-1

IMPORTING THE STORE, SALES AND FEATURE TABLE

STEP 1: Choose the feature excel sheet file from the local disk and click next.

Data Import Wizard - Step 1 of 5

Data Preview

Source: Local File

File: D:\NOSQL\Documents\Features data set.csv

File Format

Header: After Skip Skip Rows: 0

Format: csv Preview Row Limit: 100

Encoding: Cp1252

Delimiter: , Line Terminator: standard: CR LF, CR or LF

Left Enclosure: " Right Enclosure: "

File Contents

Store	Date	Temperature	Fuel_Price	MarkDown1	MarkDown2	MarkDown3	MarkDown4	MarkDown5	CPI	Unemployment
1	5/2/2010	42.31	2.572						211.0963582	8.106
1	12/2/2010	38.51	2.548						211.2421698	8.106
1	19-02-2010	39.93	2.514						211.2891429	8.106
1	26-02-2010	46.63	2.561						211.3196429	8.106
1	5/3/2010	46.5	2.625						211.3501429	8.106
1	12/3/2010	57.79	2.667						211.3806429	8.106
1	19-03-2010	54.58	2.72						211.215635	8.106
1	26-03-2010	51.45	2.732						211.0180424	8.106
1	2/4/2010	62.27	2.719						210.8204499	7.808
1	9/4/2010	65.86	2.77						210.6228574	7.808
1	16-04-2010	66.32	2.808						210.4391228	7.808
1	23-04-2010	64.84	2.795						210.3895456	7.808
1	30-04-2010	67.41	2.78						210.3399684	7.808
1	7/5/2010	72.55	2.835						210.3374261	7.808
1	14-05-2010	74.78	2.854						210.6170934	7.808
1	21-05-2010	76.44	2.826						210.8967606	7.808
1	28-05-2010	80.44	2.759							

STEP 2: Give the name of the table and click next.

Data Import Wizard - Step 2 of 4

Import Method

Specify the method for importing data. For External Table method, an external table will be created to read the data in the file. For Staging External Table method, an external table will be created as a staging table for importing the target table. For other methods, a new table is created and the data is imported.

Import Method: Insert

Send Create Script to SQL Worksheet:

Table Name: feature

Import Row Limit: 100

File Contents

Store	Date	Temperature	Fuel_Price	MarkDown1	MarkDown2	MarkDown3	MarkDown4	MarkDown5	CPI	Unemployment
1	5/2/2010	42.31	2.572						211.0963582	8.106
1	12/2/2010	38.51	2.548						211.2421698	8.106
1	19-02-2010	39.93	2.514						211.2891429	8.106
1	26-02-2010	46.63	2.561						211.3196429	8.106
1	5/3/2010	46.5	2.625						211.3501429	8.106
1	12/3/2010	57.79	2.667						211.3806429	8.106
1	19-03-2010	54.58	2.72						211.215635	8.106
1	26-03-2010	51.45	2.732						211.0180424	8.106
1	2/4/2010	62.27	2.719						210.8204499	7.808
1	9/4/2010	65.86	2.77						210.6228574	7.808
1	16-04-2010	66.32	2.808						210.4391228	7.808
1	23-04-2010	64.84	2.795						210.3895456	7.808
1	30-04-2010	67.41	2.78						210.3399684	7.808
1	7/5/2010	72.55	2.835						210.3374261	7.808
1	14-05-2010	74.78	2.854						210.6170934	7.808
1	21-05-2010	76.44	2.826						210.8967606	7.808
1	28-05-2010	80.44	2.759							

STEP 3: Select the column to import the data set and arrange them in the order we want.

The screenshot shows the 'Choose Columns' step of the Data Import Wizard. On the left, a navigation pane lists steps: Data Preview, Import Method, Choose Columns (which is selected), Column Definition, and Finish. The main area has two columns: 'Available Columns' (left) and 'Selected Columns' (right). The 'Selected Columns' list contains: Store, Date, Temperature, Fuel_Price, Markdown1, Markdown2, Markdown3, Markdown4, Markdown5, CPI, Unemployment, and IsHoliday. Below these lists is a 'File Contents' preview table:

Store	Date	Temperature	Fuel_Price	Markdown1	Markdown2	Markdown3	Markdown4	Markdown5	CPI	Unemployment
1	24-09-2010	80.94	2.624						211.5972246	7.787
1	1/10/2010	71.89	2.603						211.6719895	7.838
1	8/10/2010	63.93	2.633						211.7467544	7.838

Buttons at the bottom include Help, Back, Next >, Finish, and Cancel.

STEP 4: For each column on left, define the column details of the database table that will be created to import the data into.

The screenshot shows the 'Column Definition' step of the Data Import Wizard. The left sidebar includes steps: Data Preview, Import Method, Choose Columns, Column Definition (selected), and Finish. The main area shows 'Source Data Columns' on the left and 'Target Table Columns' on the right. The 'Source Data Columns' list includes: Store, Date, Temperature, Fuel_Price, Markdown1, Markdown2, Markdown3, Markdown4, Markdown5, CPI, Unemployment, and IsHoliday. The 'Target Table Columns' section for 'IsHoliday' is detailed below:

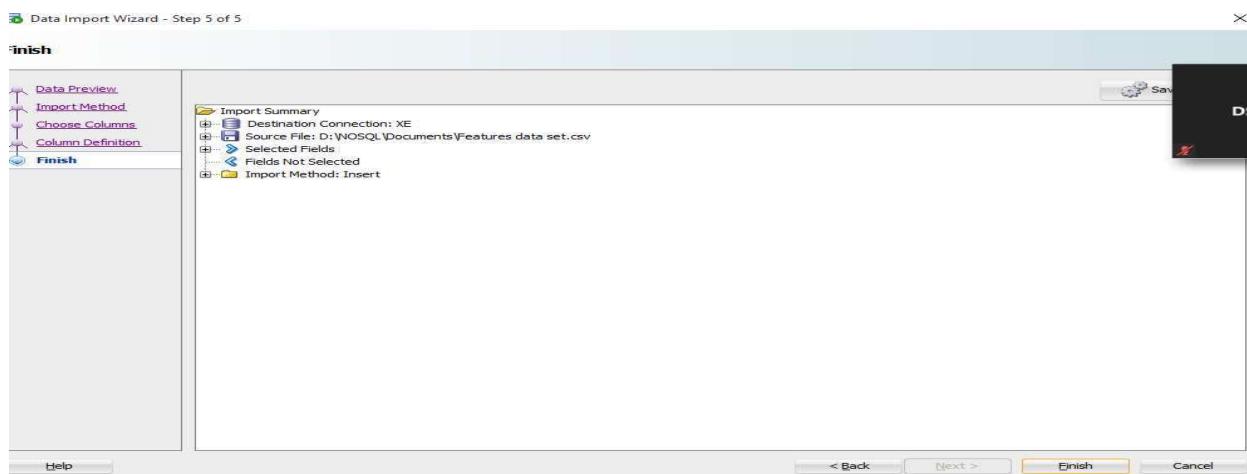
Name	IsHoliday
Data Type	VARCHAR2
Size/Precision	26
Nullable?	<input checked="" type="checkbox"/> Default
Comment	(empty)

Below this is a 'Data' preview table:

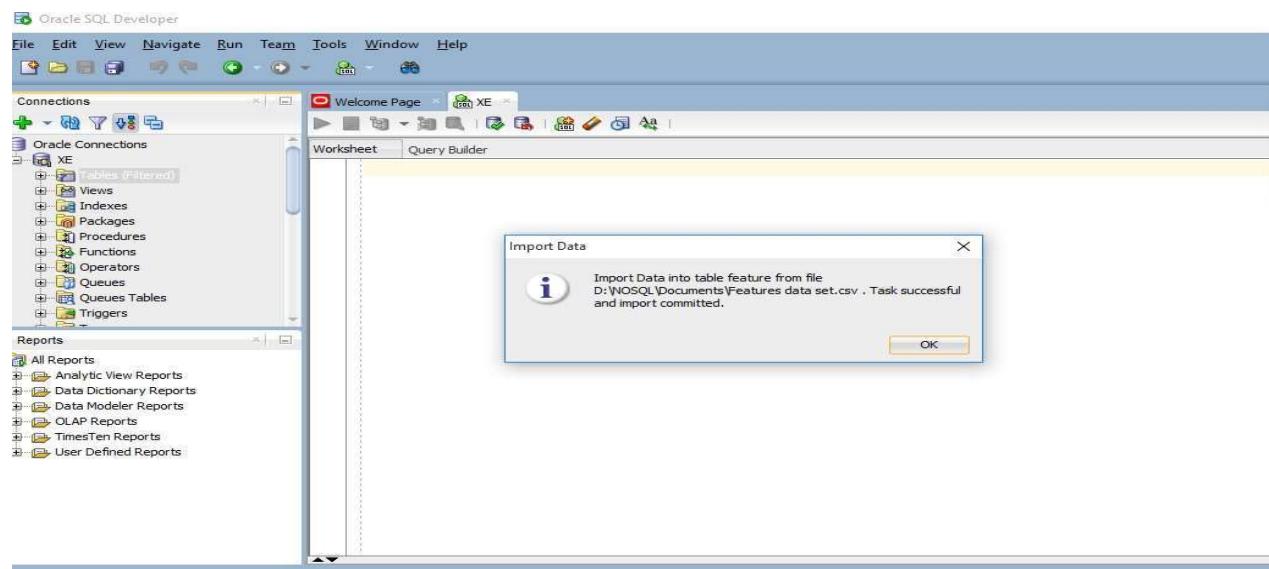
FALSE
TRUE
FALSE
FAI SF

Buttons at the bottom include Help, Back, Next >, Finish, and Cancel.

STEP 5: It will show the connection ,source file and click finish.



STEP 6: Once the data is imported it will be show the notification like task successful and imported committed.



Similarly we can import **sales and store file** in a sql developer.

Question2: Create tables from Excel files. Relate the tables via primary key and foreign key wherever required.

SQL> alter table sales add primary

key(sales_id); Table altered.

SQL>desc sales;

Name	Null?	Type
SALES_ID	NOT NULL	NUMBER(38)
STORE		NUMBER(38)
DEPT		NUMBER(38)
F_DATE		VARCHAR2(26)
WEEKLY_SALES		NUMBER(38,2)
ISHOLIDAY		VARCHAR2(26)

SQL> alter table store add foreign key(store_id) references

sales(sales_id); Table altered.

SQL>desc store;

Name	Null?	Type
STORE_ID		NUMBER(38)
TYPE		VARCHAR2(26)
STORE_SIZE		NUMBER(38)

Question3: Write atleast 2 queries to explore each table by using WHERE and ORDERBY

SQL> select store_size from store where type='A' order by store_sizeasc;

STORE_SIZE

39690

39910

151315

152513

155078

155083

158114

184109

196321

200898

202307

STORE_SIZE

202505

203007

203742

203750

203819

205863

206302

22 rows selected.

SQL> select store_id from store where type='B' order by store_size;

STORE_ID

5
3
16
7
17
29
35
12
23
45
22

STORE_ID

18
15
10
25
21

17 rows selected.

```
SQL> select dept, sales_date from sales where dept=21 and isholiday='TRUE'  
order by weekly_sales;
```

DEPT SALES_DAT

```
-----  
21 11-DEC-30  
21 12-SEP-07  
21 12-FEB-10  
21 12-FEB-10  
21 10-DEC-31  
21 11-DEC-30  
21 11-FEB-11  
21 12-SEP-07  
21 11-NOV-25  
21 11-FEB-11  
21 10-FEB-12
```

DEPT SALES_DAT

```
-----  
21 11-SEP-09  
21 11-SEP-09  
21 10-SEP-10  
21 10-DEC-31  
21 10-NOV-26  
21 10-FEB-12  
21 10-NOV-26  
21 10-SEP-10  
21 11-NOV-25
```

21 10-DEC-31

21 11-FEB-11

DEPT SALES_DAT

21 11-DEC-30

21 12-FEB-10

21 10-FEB-12

21 10-SEP-10

21 10-DEC-31

21 11-DEC-30

21 11-SEP-09

21 10-DEC-31

21 11-FEB-11

21 11-DEC-30

21 11-NOV-25

DEPT SALES_DAT

21 12-SEP-07

21 10-NOV-26

21 11-SEP-09

21 11-NOV-25

21 10-NOV-26

21 12-SEP-07

21 12-FEB-10

21 10-SEP-10

21 12-SEP-07

21 11-SEP-09

21 10-FEB-12

DEPT SALES_DAT

21 11-DEC-30

21 12-SEP-07

21 11-DEC-30

21 10-DEC-31

21 10-SEP-10

21 12-FEB-10

21 10-DEC-31

21 11-SEP-09

21 11-SEP-09

21 12-FEB-10

21 11-DEC-30

DEPT SALES_DAT

21 11-FEB-11

21 10-DEC-31

21 12-SEP-07

21 12-SEP-07

21 10-FEB-12

21 12-FEB-10

21 12-FEB-10

21 10-SEP-10

21 10-NOV-26

21 11-FEB-11

21 11-FEB-11

DEPT SALES_DAT

21 10-SEP-10

21 11-FEB-11

21 12-FEB-10

21 12-FEB-10

21 11-DEC-30

21 10-NOV-26

21 12-SEP-07

21 11-SEP-09

21 10-DEC-31

21 11-FEB-11

21 11-NOV-25

DEPT SALES_DAT

21 11-NOV-25

21 11-DEC-30

21 10-FEB-12

21 10-SEP-10

21 11-FEB-11

21 11-SEP-09

21 12-FEB-10

21 10-DEC-31

21 11-SEP-09

21 10-FEB-12

21 11-FEB-11

DEPT SALES_DAT

21 12-SEP-07

21 10-SEP-10

21 10-FEB-12

21 10-SEP-10

21 11-NOV-25

21 10-DEC-31

21 10-NOV-26

21 10-NOV-26

21 10-SEP-10

21 11-NOV-25

21 11-NOV-25

DEPT SALES_DAT

21 11-NOV-25

21 12-SEP-07

21 10-DEC-31

21 10-NOV-26

21 10-FEB-12

21 10-DEC-31

21 10-NOV-26

21 11-SEP-09

21 10-DEC-31

21 10-FEB-12

21 10-DEC-31

DEPT SALES_DAT

21 11-NOV-25

21 10-NOV-26

21 10-FEB-12

21 12-FEB-10

21 12-SEP-07

21 11-FEB-11

21 12-FEB-10

21 10-SEP-10

21 10-DEC-31

21 11-DEC-30

21 12-SEP-07

DEPT SALES_DAT

21 11-DEC-30

21 12-FEB-10

21 10-DEC-31

21 10-SEP-10

21 10-DEC-31

21 11-FEB-11

21 12-SEP-07

21 10-DEC-31

21 12-SEP-07

21 10-DEC-31

21 12-FEB-10

DEPT SALES_DAT

21 10-NOV-26

21 10-DEC-31

21 11-SEP-09

21 11-SEP-09

21 11-DEC-30

21 11-DEC-30

21 11-FEB-11

21 11-DEC-30

21 11-FEB-11

21 10-SEP-10

21 10-FEB-12

DEPT SALES_DAT

21 11-SEP-09

21 10-FEB-12

21 11-FEB-11

21 11-DEC-30

21 11-SEP-09

21 10-FEB-12

21 12-SEP-07

21 10-DEC-31

21 10-DEC-31

21 11-DEC-30

21 11-DEC-30

DEPT SALES_DAT

21 11-SEP-09

21 10-DEC-31

21 11-FEB-11

21 12-FEB-10

21 12-FEB-10

21 11-SEP-09

21 10-DEC-31

21 11-SEP-09

21 10-SEP-10

21 11-SEP-09

21 11-DEC-30

DEPT SALES_DAT

21 10-NOV-26

21 12-SEP-07

21 10-NOV-26

21 12-FEB-10

21 10-NOV-26

21 12-SEP-07

21 10-DEC-31

21 11-DEC-30

21 11-FEB-11

21 10-SEP-10

21 10-DEC-31

DEPT SALES_DAT

21 12-SEP-07

21 10-FEB-12

21 12-SEP-07

21 10-DEC-31

21 11-FEB-11

21 12-SEP-07

21 11-SEP-09

21 11-NOV-25

21 12-SEP-07

21 10-DEC-31

21 11-SEP-09

DEPT SALES_DAT

21 10-FEB-12

21 10-FEB-12

21 12-FEB-10

21 11-SEP-09

21 11-FEB-11

21 11-DEC-30

21 12-FEB-10

21 11-SEP-09

21 12-SEP-07

21 11-DEC-30

21 10-NOV-26

DEPT SALES_DAT

21 10-SEP-10

21 12-SEP-07

21 10-SEP-10

21 11-NOV-25

21 11-NOV-25

21 12-FEB-10

21 12-FEB-10

21 12-FEB-10

21 10-NOV-26

21 12-FEB-10

21 11-NOV-25

DEPT SALES_DAT

21 10-SEP-10

21 12-FEB-10

21 11-FEB-11

21 10-NOV-26

21 10-NOV-26

21 10-FEB-12

21 11-FEB-11

21 12-SEP-07

21 11-SEP-09

21 12-SEP-07

21 10-SEP-10

DEPT SALES_DAT

21 12-SEP-07

21 11-DEC-30

21 10-FEB-12

21 11-SEP-09

21 11-NOV-25

21 12-FEB-10

21 10-SEP-10

21 10-SEP-10

21 11-FEB-11

21 12-SEP-07

21 12-FEB-10

DEPT SALES_DAT

21 11-DEC-30

21 10-SEP-10

21 11-DEC-30

21 11-SEP-09

21 11-DEC-30

21 10-FEB-12

21 10-SEP-10

21 11-FEB-11

21 11-NOV-25

21 10-DEC-31

21 11-DEC-30

DEPT SALES_DAT

21 10-DEC-31

21 10-NOV-26

21 11-SEP-09

21 11-NOV-25

21 11-DEC-30

21 12-FEB-10

21 10-FEB-12

21 10-DEC-31

21 11-DEC-30

21 11-SEP-09

21 10-NOV-26

DEPT SALES_DAT

21 11-FEB-11

21 10-SEP-10

21 10-FEB-12

21 10-FEB-12

21 10-SEP-10

21 12-FEB-10

21 11-SEP-09

21 11-FEB-11

21 10-SEP-10

21 10-NOV-26

21 11-FEB-11

DEPT SALES_DAT

21 11-NOV-25

21 11-NOV-25

21 10-DEC-31

21 11-FEB-11

21 12-FEB-10

21 12-SEP-07

21 10-SEP-10

21 10-DEC-31

21 11-FEB-11

21 10-SEP-10

21 11-FEB-11

DEPT SALES_DAT

21 10-FEB-12

21 11-DEC-30

21 10-DEC-31

21 11-FEB-11

21 11-NOV-25

21 11-SEP-09

21 12-FEB-10

21 10-NOV-26

21 10-FEB-12

21 11-FEB-11

21 10-DEC-31

DEPT SALES_DAT

21 12-FEB-10

21 11-NOV-25

21 11-NOV-25

21 11-NOV-25

21 10-SEP-10

21 11-SEP-09

21 10-NOV-26

21 11-DEC-30

21 11-FEB-11

21 12-FEB-10

21 10-DEC-31

DEPT SALES_DAT

21 10-FEB-12

21 11-FEB-11

21 11-FEB-11

21 11-SEP-09

21 10-DEC-31

21 10-NOV-26

21 10-FEB-12

21 11-NOV-25

21 10-NOV-26

21 10-NOV-26

21 11-SEP-09

DEPT SALES_DAT

21 10-FEB-12

21 12-FEB-10

21 10-NOV-26

21 10-FEB-12

21 11-NOV-25

21 10-SEP-10

21 11-DEC-30

21 11-SEP-09

21 10-NOV-26

21 12-SEP-07

21 12-SEP-07

DEPT SALES_DAT

21 11-NOV-25

21 11-SEP-09

21 11-SEP-09

21 12-SEP-07

21 10-SEP-10

21 12-SEP-07

21 10-DEC-31

21 10-SEP-10

21 11-DEC-30

21 12-FEB-10

21 11-DEC-30

DEPT SALES_DAT

21 10-FEB-12

21 10-FEB-12

21 11-NOV-25

21 11-DEC-30

21 10-FEB-12

21 10-DEC-31

21 11-NOV-25

21 10-FEB-12

21 10-SEP-10

21 11-SEP-09

21 12-SEP-07

DEPT SALES_DAT

21 11-DEC-30

21 10-FEB-12

21 12-SEP-07

21 12-SEP-07

21 10-FEB-12

21 12-SEP-07

21 11-FEB-11

21 11-NOV-25

21 12-FEB-10

21 11-NOV-25

21 12-FEB-10

DEPT SALES_DAT

21 11-DEC-30

21 12-FEB-10

21 11-DEC-30

21 10-DEC-31

21 10-NOV-26

21 10-DEC-31

21 11-FEB-11

21 10-SEP-10

21 11-SEP-09

21 11-FEB-11

21 11-DEC-30

DEPT SALES_DAT

21 10-FEB-12

21 11-DEC-30

21 11-DEC-30

21 10-SEP-10

21 11-SEP-09

21 10-NOV-26

21 11-FEB-11

21 10-SEP-10

21 11-DEC-30

21 10-SEP-10

21 10-NOV-26

DEPT SALES_DAT

21 12-FEB-10

21 10-NOV-26

21 12-SEP-07

21 11-SEP-09

21 10-NOV-26

21 11-FEB-11

21 11-SEP-09

21 12-FEB-10

21 11-NOV-25

21 12-SEP-07

21 12-FEB-10

DEPT SALES_DAT

21 11-NOV-25

21 12-SEP-07

21 11-SEP-09

21 12-SEP-07

21 10-NOV-26

21 10-FEB-12

21 10-SEP-10

21 11-NOV-25

21 11-FEB-11

21 10-DEC-31

21 10-SEP-10

DEPT SALES_DAT

21 11-FEB-11

21 10-SEP-10

21 12-SEP-07

21 10-DEC-31

21 10-FEB-12

21 10-NOV-26

21 10-SEP-10

21 10-FEB-12

21 11-DEC-30

21 12-SEP-07

21 11-FEB-11

DEPT SALES_DAT

21 12-FEB-10

21 12-SEP-07

21 12-FEB-10

21 11-SEP-09

21 11-SEP-09

21 10-FEB-12

21 10-SEP-10

21 12-FEB-10

21 11-DEC-30

21 11-NOV-25

21 11-NOV-25

DEPT SALES_DAT

21 11-NOV-25

21 10-SEP-10

21 10-NOV-26

21 11-FEB-11

21 10-DEC-31

21 11-NOV-25

21 10-FEB-12

21 10-FEB-12

21 10-NOV-26

21 10-NOV-26

21 11-NOV-25

DEPT SALES_DAT

21 10-NOV-26

21 10-FEB-12

21 10-NOV-26

21 10-FEB-12

21 11-NOV-25

21 11-NOV-25

21 12-SEP-07

21 11-NOV-25

21 11-DEC-30

21 10-NOV-26

21 11-FEB-11

DEPT SALES_DAT

21 11-SEP-09

21 10-NOV-26

21 10-SEP-10

21 11-NOV-25

21 12-FEB-10

21 11-NOV-25

21 10-NOV-26

21 10-FEB-12

21 10-NOV-26

21 11-NOV-25

450 rows selected.

SQL> select store_id, dept, weekly_sales from sales where sales_id between 20 and 30 order by weekly_sales desc;

STORE_ID DEPT WEEKLY_SALES

1 1 17688.76

1 1 17508.41

1 1 17150.84

1 1 16333.14

1 1 16328.72

1 1 16216.27

```
1      1  15793.87  
1      1  15740.13  
1      1  15536.4  
1      1  15381.82  
1      1  15360.45
```

11 rows selected.

SQL> select store_id, unemployment, temperature from feature_data where store_id=2 and unemployment<7 order by temperature;

STORE_ID UNEMPLOYMENT TEMPERATURE

```
-----  
2    6.237    39.12  
2    6.237    40.98  
2    6.17     41.56  
2    6.237    46.08  
2    6.237    47.42  
2    6.17     47.69  
2    6.237    48.16  
2    6.237    48.92  
2    6.237    50.54  
2    6.237    51.12  
2    6.237    51.33
```

STORE_ID UNEMPLOYMENT TEMPERATURE

```
-----  
2    6.17     52.64  
2    6.17     52.72
```

2	6.237	54.75
2	6.17	55.14
2	6.237	55.14
2	6.237	56.08
2	6.17	56.4
2	6.17	58.06
2	6.112	58.13
2	6.112	58.3
2	6.237	59.97

STORE_ID UNEMPLOYMENT TEMPERATURE

2	6.17	60.97
2	6.112	61.23
2	6.17	61.9
2	6.17	64.19
2	6.891	65.69
2	6.112	67.05
2	6.891	67.2
2	6.17	68.08
2	6.891	68.08
2	6.891	68.43
2	6.17	69.79

STORE_ID UNEMPLOYMENT TEMPERATURE

2	6.17	70.27
2	6.565	71.09

2	6.891	71.27
2	6.891	73.87
2	6.565	75.88
2	6.891	76.73
2	6.565	76.91
2	6.891	78.19
2	6.891	78.38
2	6.891	78.69
2	6.565	79.45

STORE_ID UNEMPLOYMENT TEMPERATURE

2	6.565	80.17
2	6.891	80.56
2	6.891	81.04
2	6.565	82.64
2	6.565	83.23
2	6.565	84.2
2	6.565	84.79
2	6.891	86.32
2	6.565	86.37
2	6.565	87.65
2	6.565	88.55

STORE_ID UNEMPLOYMENT TEMPERATURE

2	6.565	90.22
---	-------	-------

56 rows selected.