

NoSQL Database Management Lab

Lab Manual with Student Lab Record

Developed By

Dr. K. Rajkumar

Dr. M. Lovelin Ponn Feliciah



Roll No	205229119
Name	Maheşvaran. S
Class	I MSc Data Science

Department of Data Science
Bishop Heber College (Autonomous)
Tiruchirappalli 620017, INDIA

June 2020

Copyright © Bishop Heber College



Department of Data Science
Bishop Heber College (Autonomous)
Tiruchirappalli 620017

BONAFIDE CERTIFICATE

Name: Maheshvaran. S.

Reg. No: 205229119 Class: I - M.Sc Datasience

Course Title NoSQL Database Management Lab

Certified that this is the bonafide record of work done by me during **Odd / Even**
Semester of **2020 – 2021** and submitted for the Practical Examinations on

Staff In-Charge

Head of the Department

Examiners

1. _____

2. _____

Grade Sheet

Roll No	205229119	Name	Maheshvaran .S
Year	1	Semester	1
Instructor Name			

Lab	Activity	Grade	Sign/Date
1	Designing and Querying MyRestaurant Database		
2	India Weather Analytics Using Historical Data Part-I		
3	India Weather Analytics Using Historical Data Part-II		
4	Retail Sales Analytics Part-I		
5	Retail Sales Analytics Part-II		
6	Retail Sales Analytics Part-III		
7	University Course Enrollment Data Analytics		
8	Retail Sales Analytics Part-IV		
9	Student Information System Design using MongoDB Part-I		
10	Student Information System Design using MongoDB Part-II		
11	Ecommerce Product Catalog Design Using MongoDB		
12	Neo4J Play Ground Exercise		
13	Designing Movie Graph Database using Neo4J		
14	Designing Flight Graph Database Using Neo4J		

Question: 1

SQL> Create table my-restaurants (rname varchar2(15), foodtype varchar2(10), distance number (2), lastvisit varchar2(15), ilike varchar2(5));

Table created.

Question: 2

SQL> insert into my-restaurants values ('&rname', '& foodtype', '& distance', '& lastvisit', '& ilike');

Enter value for rname: apple-leaf

Enter value for foodtype: nonveg

Enter value for distance: 15

Enter value for lastvisit: 01-Jan-2020

Enter value for ilike: 1

old 1 : insert into my-restaurants values ('&rname', '& foodtype', '& distance', '& lastvisit', '& ilike')

new 1 : insert into my-restaurants values ('apple-leaf', 'nonveg', '15', '01-Jan-2020', '1')

1 row created.

SQL> /

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

Lab1. Developing and Querying *MyRestaurants* Database

Objectives

In this lab, you will create a new table, insert tuples satisfying the constraints and perform query processing.

Tasks To Be Completed

Question1. Create a table called *MyRestaurants* with the following attributes:

- Name of the restaurant: a varchar field
- Type of food they make: a varchar field (food type: veg, nonveg, chinese)
- Distance (in minutes) from your house: an integer
- Date of your last visit: a varchar field, interpreted as date
- Whether you like it or not: an integer, interpreted as a Boolean

Question2. Insert at least five tuples using the SQL INSERT command seven (or more) times. You should insert at least one restaurant you liked, at least one restaurant you did not like, and at least one restaurant where you leave the iLike field NULL.

Sample Dataset:

name	foodtype	distance	lastvisit	ilike
apple_leaf	nonveg	15	01-Jan-2020	1
sowmyas	veg	18	20-Mar-2020	1
thinnappa	nonveg	25	20-Nov-2019	0
sribhavan	veg	18	20-Dec-2019	0
chinaworld	chinese	14	05-Mar-2020	1
littlechina	chinese	30	10-Mar-2020	0
munivilas	nonveg	20	05-Dec-2019	null
dosacorner	nonveg	10	05-Feb-2020	1

Question3. Write a SQL query that returns all restaurants in your table.

Select * from my_restaurants;

Question4. Now, experiment with a few output formats, using the SQL query you wrote for Question3.

- print the results in comma-separated form
- print the results in list form, delimited by " | "
- print the results in column form, and make each column have width 15

Question: 4

* SQL> Select listagg(rname,',') within group (order by rname)
Restaurant-name from my-restaurants;

RESTAURANT-NAME

apple-leaf, chinaworld, dosacorner, Littlechina, munivillas,
sowmyas, Sribhavan, thinnappa.

* SQL> Select listagg(rname,'1') within group (order by
rname) Restaurant-name from my-restaurants;

* SQL> set numwidth 15;

SQL> Select * from my-restaurants;

RNAME	FOODTYPE	DISTANCE	LAST VISIT	ILIKE
apple-leaf	nonveg	15	01-Jan-2020	1
sowmyas	veg	18	20-Mar-2020	1
thinnappa	nonveg	25	20-NOV-2019	0

- for each of the formats above, try printing/not printing the column headers with the results

Question5. Modify your SQL query such that it prints "I liked it" or "I hated it" for each restaurant you liked or not. Note that you are not allowed to modify the table on disk. You should be able to answer this question using only a SELECT statement. A solution that creates and uses an extra table, however, will be accepted.

Question6. Write a SQL query that returns only the name and distance of all restaurants within and including 20 minutes of your house. The query should list the restaurants in alphabetical order of names.

```
SQL> select rname, distance from my-restaurants  
      where distance <=20 order by rname asc;
```

Question7. Write a SQL query that returns the names of restaurants in descending order that makes Chinese foods.

```
SQL> select rname from my-restaurants where  
      foodtype ='chinese' order by rname desc;
```

Question8. Write a SQL query that returns the names of restaurants in ascending order which you have visited in the past 2 months

Select * from my_restaurants where ilike = '1' and last visit < sysdate order by name asc;

Question9. Write a SQL query that returns all restaurants that you like, but have not visited since more than 3 months ago.

SQL> Select rname from my_restaurants where ilike = '1' and lastvisit < sysdate;

Question10. Write a SQL query that returns all restaurants names that are within and including 15 mins from your house where you like Chinese foods.

SQL> Select rname from my_restaurants where distance <= 15 and foodtype = 'chinese';

RNAME

chinaworld

LAB1. DEVELOPING AND QUERYING MY RESTAURANTS DATABASE

Question1. Create a table called MyRestaurants with the following attributes;

- Name of the restaurant: a varchar field
- Type of food they make: a varchar field (food type: veg, nonveg, Chinese)
- Distance (in minutes) from your house: an integer
- Date of your last visit: a varchar field, interpreted as date
- Whether you like it or not: an integer, interpreted as a Boolean

```
SQL> create table my_restaurants (rname varchar2(15), foodtype varchar2(10), distance  
number (2), lastvisit varchar2(15), ilike varchar2(5));
```

Table created.

Name	Null?	Type
RNAME		VARCHAR2(30)
FOODTYPE		VARCHAR2(20)
DISTANCE		NUMBER (10)
LASTVISIT		VARCHAR2(15)
ILIKE		VARCHAR2(2)

Question2. Insert at least five tuples using the SQL insert command seven (or more) times. you should insert at least one restaurant you liked, at least one restaurant you did not like, and at least one restaurant where you leave the ilike field null;

```
SQL> insert into my_restaurants values ('& rname', '& foodtype', '& distance', '& lastvisit', '&  
ilike');
```

Enter value for rname: apple_leaf

Enter value for foodtype: nonveg

Enter value for distance: 15

Enter value for lastvisit: 01-Jan-2020

Enter value for ilike: 1

old 1: insert into my_restaurants values ('& rname', '& foodtype', '& distance', '& lastvisit', '& ilike')

new 1: insert into my_restaurants values ('apple_leaf', 'nonveg', '15', '01-Jan-2020', '1')

1 row created.

SQL> /

Enter value for rname: sowmyas

Enter value for foodtype: veg

Enter value for distance: 18

Enter value for lastvisit: 20-Mar-2020

Enter value for ilike: 1

old 1: insert into my_restaurants values ('& rname', '& foodtype', '& distance', '& lastvisit', '& ilike')

new 1: insert into my_restaurants values ('sowmyas', 'veg', '18', '20-Mar-2020', '1')

1 row created.

SQL> /

Enter value for rname: thinnappa

Enter value for foodtype: nonveg

Enter value for distance: 25

Enter value for lastvisit: 20-Nov-2019

Enter value for ilike: 0

old 1: insert into my_restaurants values ('& rname', '& foodtype', '& distance', '& lastvisit', '& ilike')

new 1: insert into my_restaurants values ('thinnappa', 'nonveg', '25', '20-Nov-2019', '0')

1 row created.

```
SQL> /  
Enter value for rname: sribhavan  
Enter value for foodtype: veg  
Enter value for distance: 18  
Enter value for lastvisit: 20-Dec-2019  
Enter value for ilike: 0  
old  1: insert into my_restaurants values ('& rname', '& foodtype', '& distance', '& lastvisit', '& ilike')  
new  1: insert into my_restaurants values ('sribhavan', 'veg', '18', '20-Dec-2019', '0')
```

1 row created.

```
SQL> /  
Enter value for rname: chinaworld  
Enter value for foodtype: chinese  
Enter value for distance: 14  
Enter value for lastvisit: 05-Mar-2020  
Enter value for ilike: 1  
old  1: insert into my_restaurants values ('& rname', '& foodtype', '& distance', '& lastvisit', '& ilike')  
new  1: insert into my_restaurants values ('chinaworld', 'chinese', '14', '05-Mar-2020', '1')
```

1 row created.

```
SQL> /  
Enter value for rname: littlechina  
Enter value for foodtype: chinese
```

Enter value for distance: 30

Enter value for lastvisit: 10-Mar-2020

Enter value for ilike: 0

old 1: insert into my_restaurants values ('& rname', '& foodtype', '& distance', '& lastvisit', '& ilike')

new 1: insert into my_restaurants values ('littlechina', 'chinese', '30', '10-Mar-2020', '0')

1 row created.

SQL> /

Enter value for rname: munivilas

Enter value for foodtype: nonveg

Enter value for distance: 20

Enter value for lastvisit: 05-Dec-2019

Enter value for ilike: null

old 1: insert into my_restaurants values ('& rname', '& foodtype', '& distance', '& lastvisit', '& ilike')

new 1: insert into my_restaurants values ('munivilas', 'nonveg', '20', '05-Dec-2019', 'null')

1 row created.

SQL> /

Enter value for rname: dosacorner

Enter value for foodtype: nonveg

Enter value for distance: 10

Enter value for lastvisit: 05-Feb-2020

Enter value for ilike: 1

old 1: insert into my_restaurants values('& rname', '& foodtype', '& distance', '& lastvisit', '& ilike')

new 1: insert into my_restaurants values('dosacorner', 'nonveg', '10', '05-Feb-2020', '1')

1 row created.

Question3. Write a SQL query that returns all restaurants in your table;

SQL> select * from my_restaurants;

RNAME	FOODTYPE	DISTANCE	LASTVISIT	ILIKE
apple_leaf	nonveg	15	01-Jan-2020	1
sowmyas	veg	18	20-Mar-2020	1
thinnappa	nonveg	25	20-Nov-2019	0
sribhavan	veg	18	20-Dec-2019	0
chinaworld	chinese	14	05-Mar-2020	1
littlechina	chinese	30	10-Mar-2020	0
munivilas	nonveg	20	05-Dec-2019	null
dosacorner	nonveg	10	05-Feb-2020	1

Question4. Now, experiment with a few output formats, using the SQL query you wrote for question3.

PRINT THE RESULTS IN COMMA-SEPARATED FORM

SQL> select listagg(rname,',') within group(order by rname) Restaurant_name from my_restaurants;

RESTAURANT_NAME

apple_leaf,chinaworld,dosacorner,littlechina,munivilas,sowmyas,sribhavan,thinnappa

PRINT THE RESULTS IN LIST FORM, DELIMITED BY “|”

```
SQL> select listagg(rname,'|')within group(order by rname) Restaurant_name from my_restaurants;
```

RESTAURANT_NAME

```
-----  
apple_leaf|chinaworld|dosacorner|littlechina|munivilas|sowmyas|sribhavan|thinnappa
```

PRINT THE RESULTS IN COLUMN FORM, AND MAKE EACH COLUMN HAVE WIDTH 15;

```
SQL> set numwidth 15;
```

```
SQL> select * from my_restaurants;
```

RNAME	FOODTYPE	DISTANCE	LASTVISIT	ILIKE
-----	-----	-----	-----	---
apple_leaf	nonveg	15	01-Jan-2020	1
sowmyas	veg	18	20-Mar-2020	1
thinnappa	nonveg	25	20-Nov-2019	0
sribhavan	veg	18	20-Dec-2019	0
chinaworld	chinese	14	05-Mar-2020	1
littlechina	chinese	30	10-Mar-2020	0
munivilas	nonveg	20	05-Dec-2019	null
dosacorner	nonveg	10	05-Feb-2020	1

Question6. Write a SQL query that returns only the name and distance of all restaurants within and including 20 minutes of you house. the query should list the restaurants in alphabetical order of names;

```
SQL> select rname,distance from my_restaurants where distance<=20 order by rname asc;
```

RNAME	DISTANCE
-------	----------

apple_leaf	15
chinaworld	14
dosacorner	10
munivilas	20
sowmyas	18
sribhavan	18

Question7. Write a SQL query that returns the names of restaurants in descending order that makes Chinese foods.

```
SQL> select rname from my_restaurants where foodtype='chinese' order by rname desc;
```

RNAME

littlechina
chinaworld

Question9. Write a SQL query that returns all restaurants that you like, but have not visited since more than 3 months ago.

```
SQL> select rname from my_restaurants where ilike='1' and lastvisit<sysdate;
```

RNAME

apple_leaf

sowmyas

chinaworld

dosacorner

Question10. Write a SQL query that returns all restaurants names that are within and including 15 mins from your house where you like chines foods.

SQL> select rname from my_restaurants where distance<=15 and foodtype='chinese';

RNAME

Chinaworld

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

Lab2. India Weather Analytics using Historical Data Part-I

Objectives

In this lab, you will explore the selection, filtering and aggregate functions to analyse the historical data of India Weather Information from 1995 to February 2020

Dataset: weather_india_1995_2020.xlsx

Data format: month, day, year, temperature, city

Tasks To Be Completed

PART-1

1. Explain the attributes of the dataset, what are they?
2. Check whether the dataset has primary key?. If not, you do need primary key?.
3. Explain what insights you hope to gain

PART-2

Question1: Import Excel data into your table (You can use any tools – Oracle, SQLite, etc)

Question2: Write 5 Queries using SELECT and WHERE clause

Question3: Write 5 queries using Aggregate functions (min, max, avg and count)

Sample Queries (Write SQL Queries to answer the following)

1. What is the lowest, highest and average temperature of your dataset (from year 1995 to February 2020)?

SQL> Select min(temperature), max(temperature), avg(temperature)
from weather_india where year > 1994 and year < 2021 and
month = 2;

2. What is the average temperature in May 2019 in Chennai?

SQL> Select avg(temperature) from weather_india where
month = 5 and year = 2019 and city = 'Chennai';

3. Which is the hottest day in 2019 in Delhi?

SQL> select max(temperature) from weather-india where year=2019 and city='delhi';

4. Which is the coldest day in 2018 in Chennai?

SQL> select min(temperature) from weather-india where year=2018 and city='chennai';

5. Which is the coldest year in December?. Print city and temperature.

SQL> select city, temperature, year from weather-india where month=12 and temperature<30;

6. Which is the hottest city in India in 2017?

SQL> Select city, temperature, year from weather_india
where year = 2017 and temperature > 100;

7. Is winter in Delhi in January 2017 colder than Mumbai?. Print average temperature of Delhi and Mumbai too.

SQL> Select avg(temperature) from weather_india where city
between 'delhi' and 'mumbai' and month = 1 and year = 2017;

8. Display the day, month, year and city of the coldest day.

SQL> Select day, month, year, city from weather_india
where temperature < 20;

LAB2. INDIAN WEATHER ANALYTICS USING HISTOCIAL DATA (PART 1)

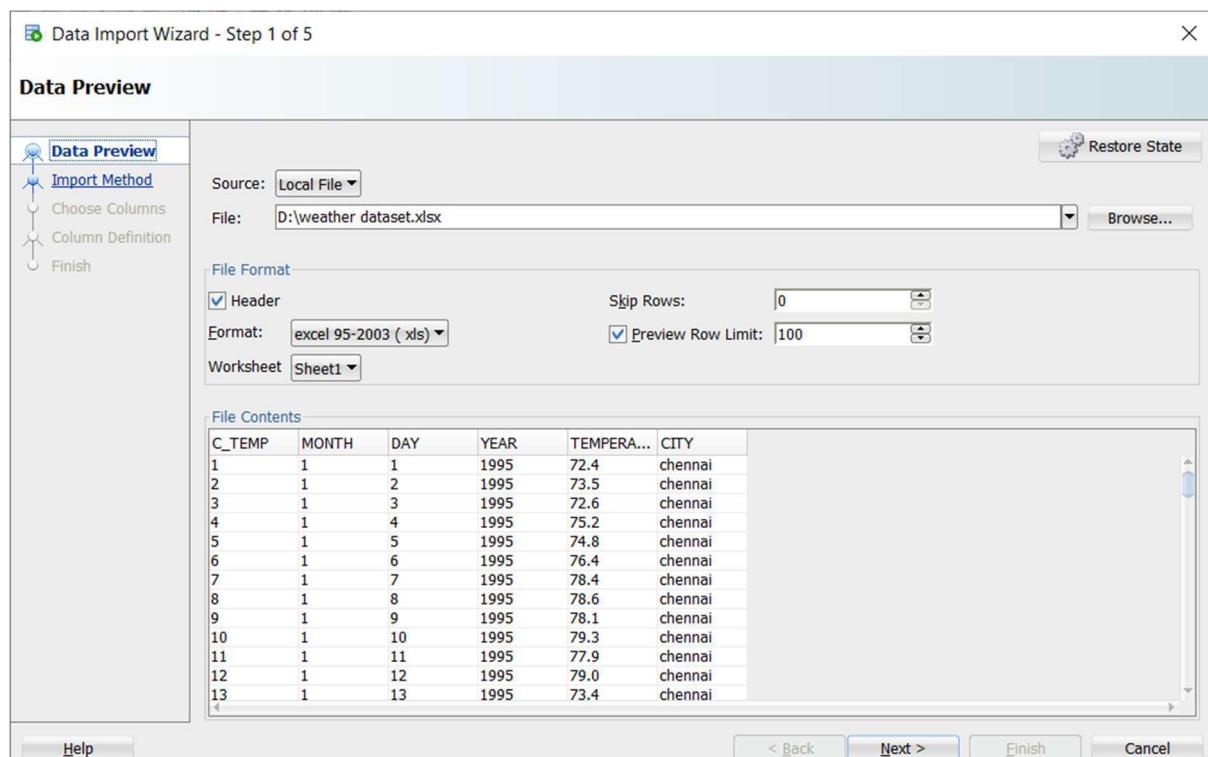
Import excel data into your table(You can use any tools – Oracle, SQLite, etc)

Step 1:

We need to install sql developer for importing data to our database schema. Unlike installing Oracle database this sql developer is pretty easy to install and also easy to work with it.

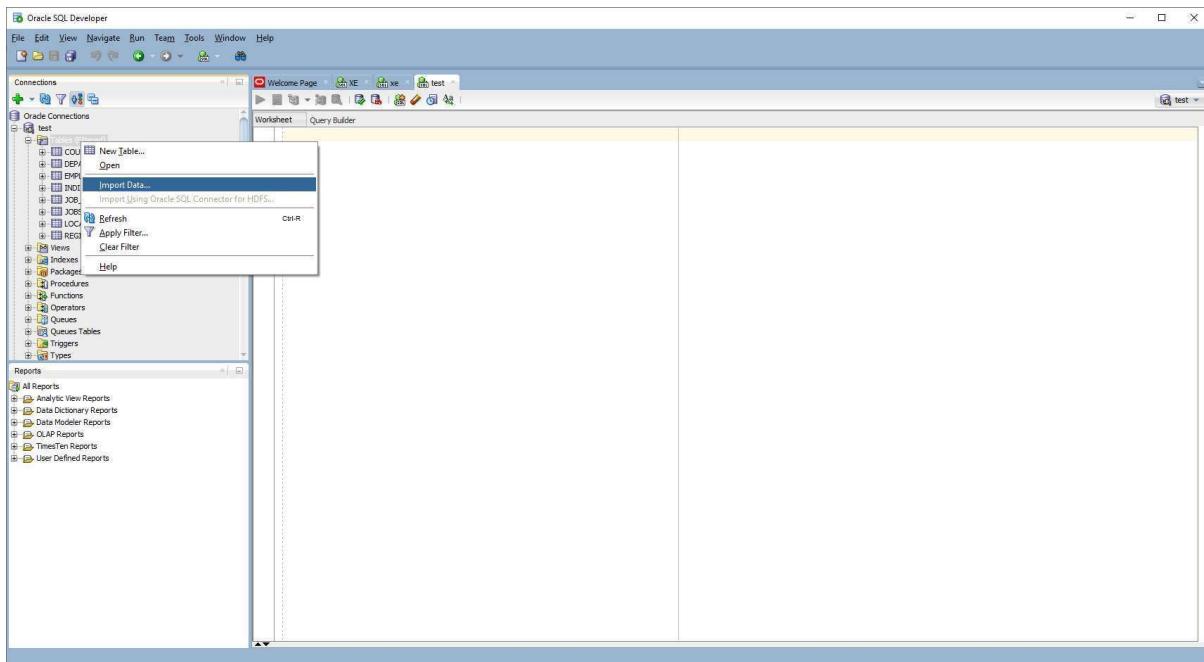
Step 2:

First we need to create a new connect using the “+” icon on the left side. Then should fill the details where displayed below: Name of the connection which can be anything for eg: your name then give the schema that you want to access that may be HR or SCOTT as a username then give the password. Finally you need to change the SID by default it will be xe you need to change that to ORCL (case sensitive). Then test the connection when the status shows success then give connect.



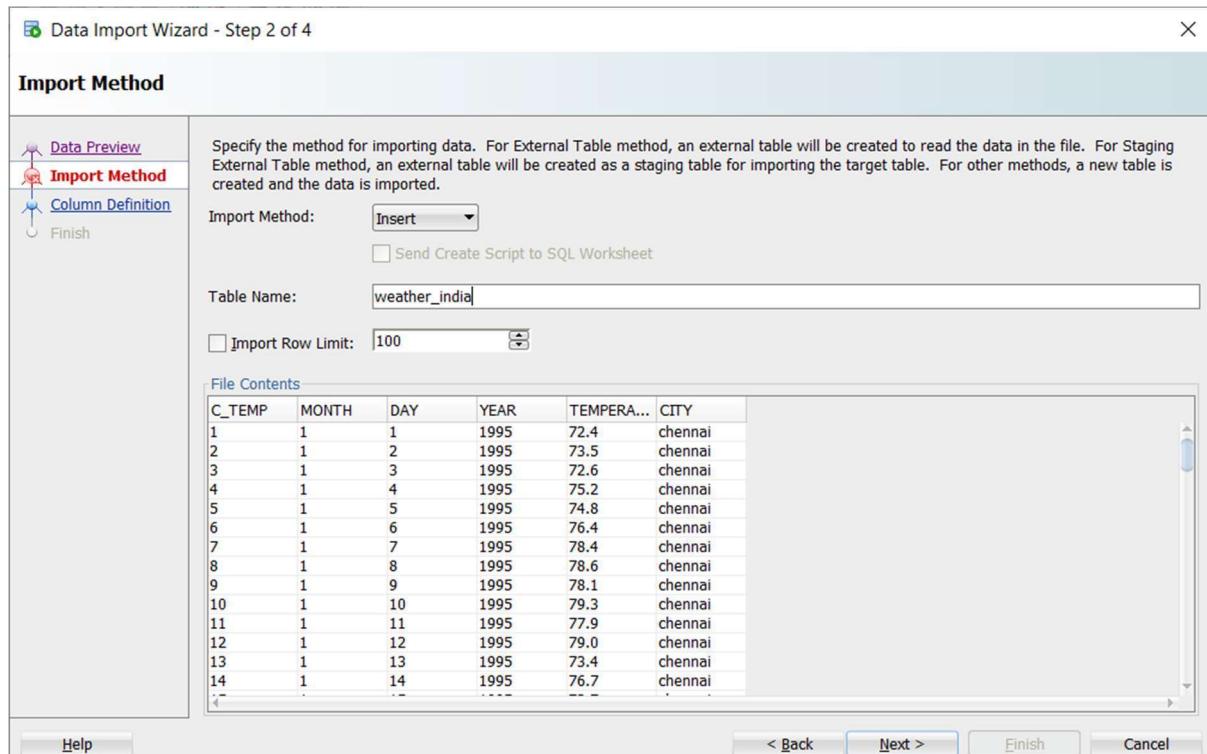
Step 3:

Click on the connection name which you created under the sub category you can see the table names where already in the Schema. Then right click on the connection name then it shows option for import data just click on that.



Step 4:

After clicking import data you can see the below screenshot. Then click for browse to select the file which you want to import. Then you can preview the data as in file contents then give.



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:

Table Name:

Import Row Limit:

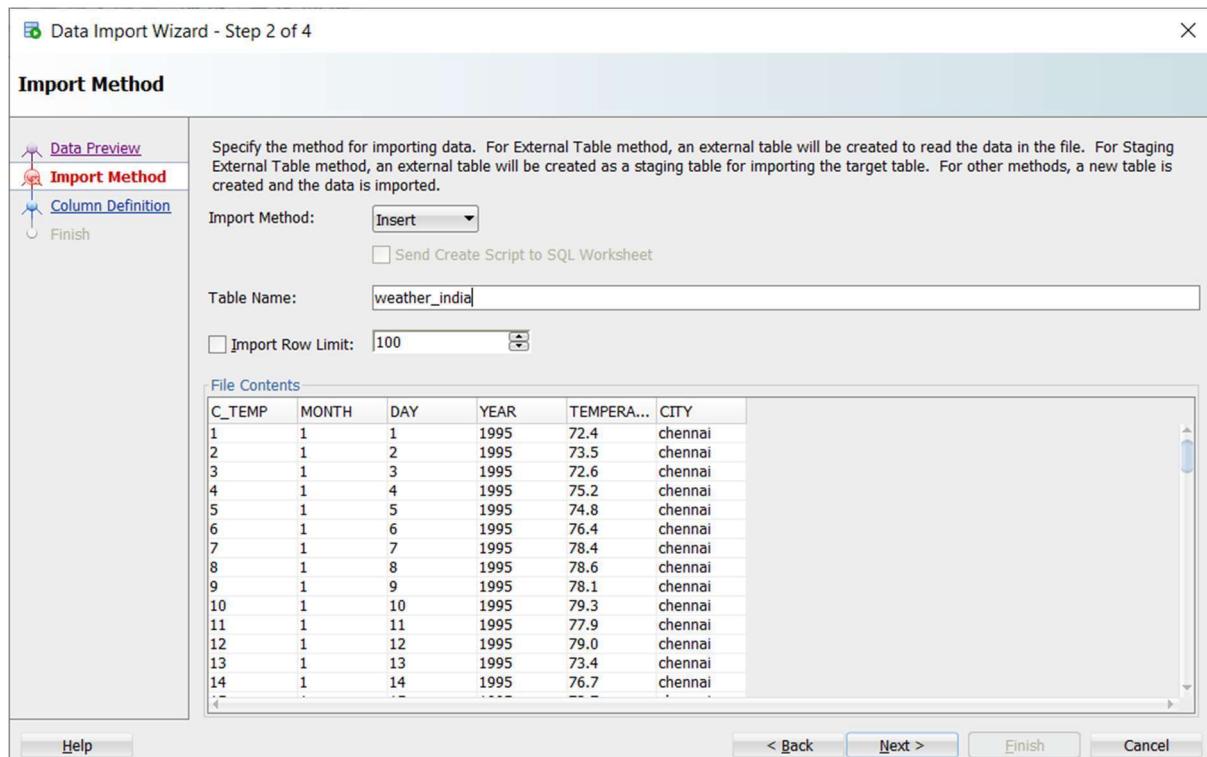
File Contents

C_TEMP	MONTH	DAY	YEAR	TEMPERA...	CITY
1	1	1	1995	72.4	chennai
2	1	2	1995	73.5	chennai
3	1	3	1995	72.6	chennai
4	1	4	1995	75.2	chennai
5	1	5	1995	74.8	chennai
6	1	6	1995	76.4	chennai
7	1	7	1995	78.4	chennai
8	1	8	1995	78.6	chennai
9	1	9	1995	78.1	chennai
10	1	10	1995	79.3	chennai
11	1	11	1995	77.9	chennai
12	1	12	1995	79.0	chennai
13	1	13	1995	73.4	chennai
14	1	14	1995	76.7	chennai

Help < Back Next > Finish Cancel

Step 5:

Here you need to create a table name and you can set the row limit to import data as like in below screenshot.

 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:

Send Create Script to SQL Worksheet

Table Name:

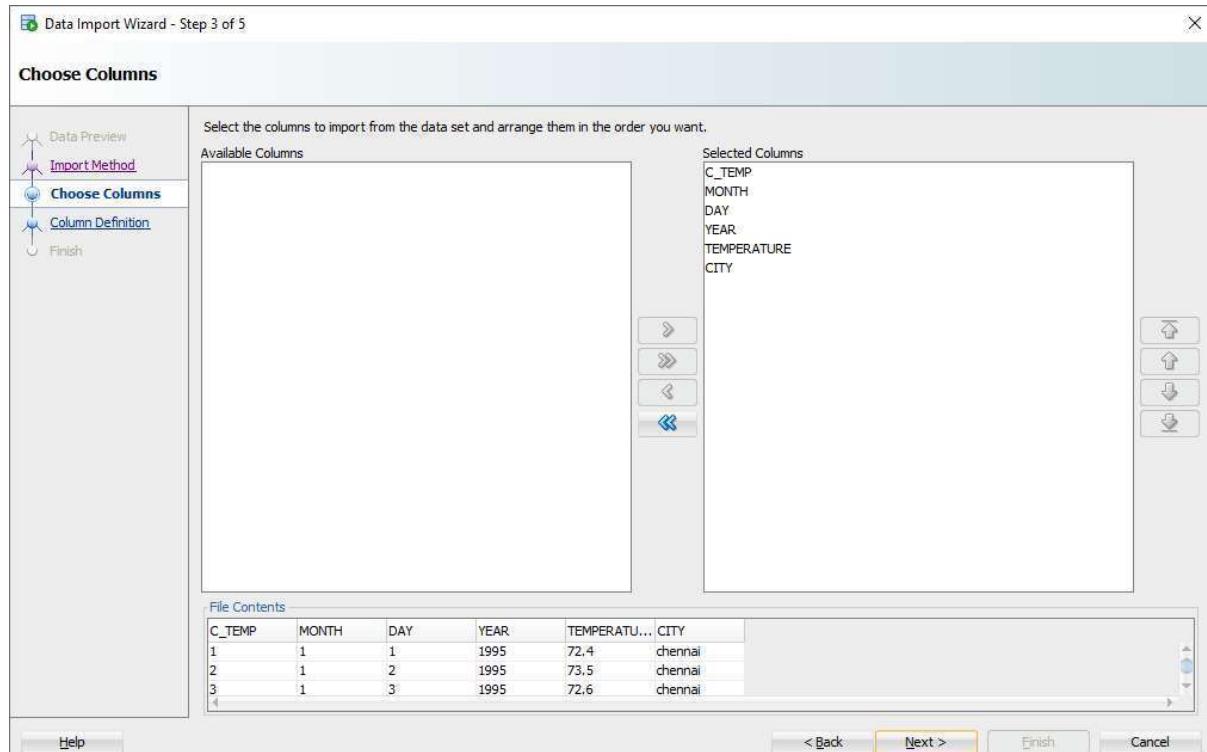
Import Row Limit:

File Contents

C_TEMP	MONTH	DAY	YEAR	TEMPERA...	CITY
1	1	1	1995	72.4	chennai
2	1	2	1995	73.5	chennai
3	1	3	1995	72.6	chennai
4	1	4	1995	75.2	chennai
5	1	5	1995	74.8	chennai
6	1	6	1995	76.4	chennai
7	1	7	1995	78.4	chennai
8	1	8	1995	78.6	chennai
9	1	9	1995	78.1	chennai
10	1	10	1995	79.3	chennai
11	1	11	1995	77.9	chennai
12	1	12	1995	79.0	chennai
13	1	13	1995	73.4	chennai
14	1	14	1995	76.7	chennai

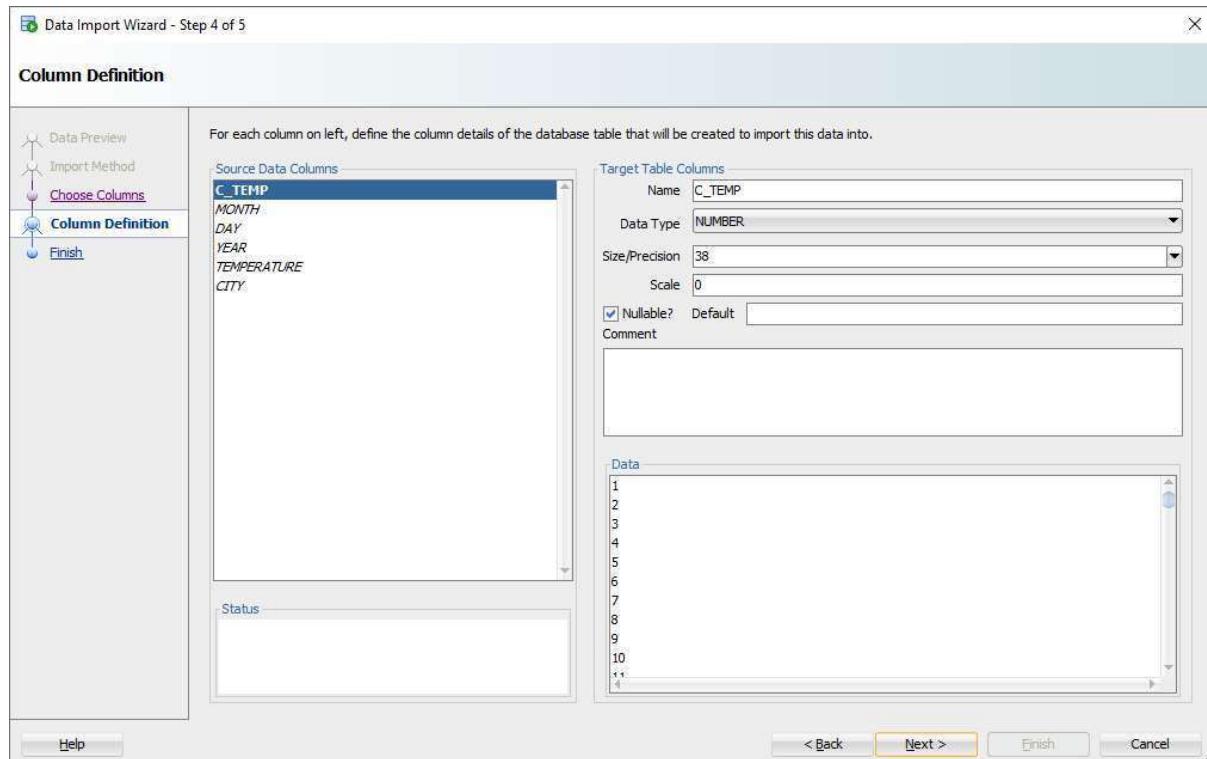
Step 6:

Here you can choose the columns that you want to add it to the table or you can order the column name as your wish. If you want to remove a particular column you can specify the column name by selecting that and Click the "<" to remove single column.



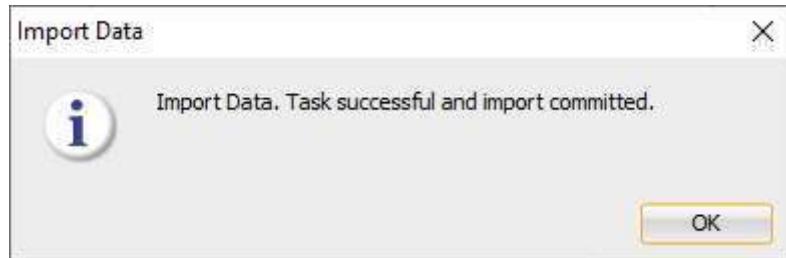
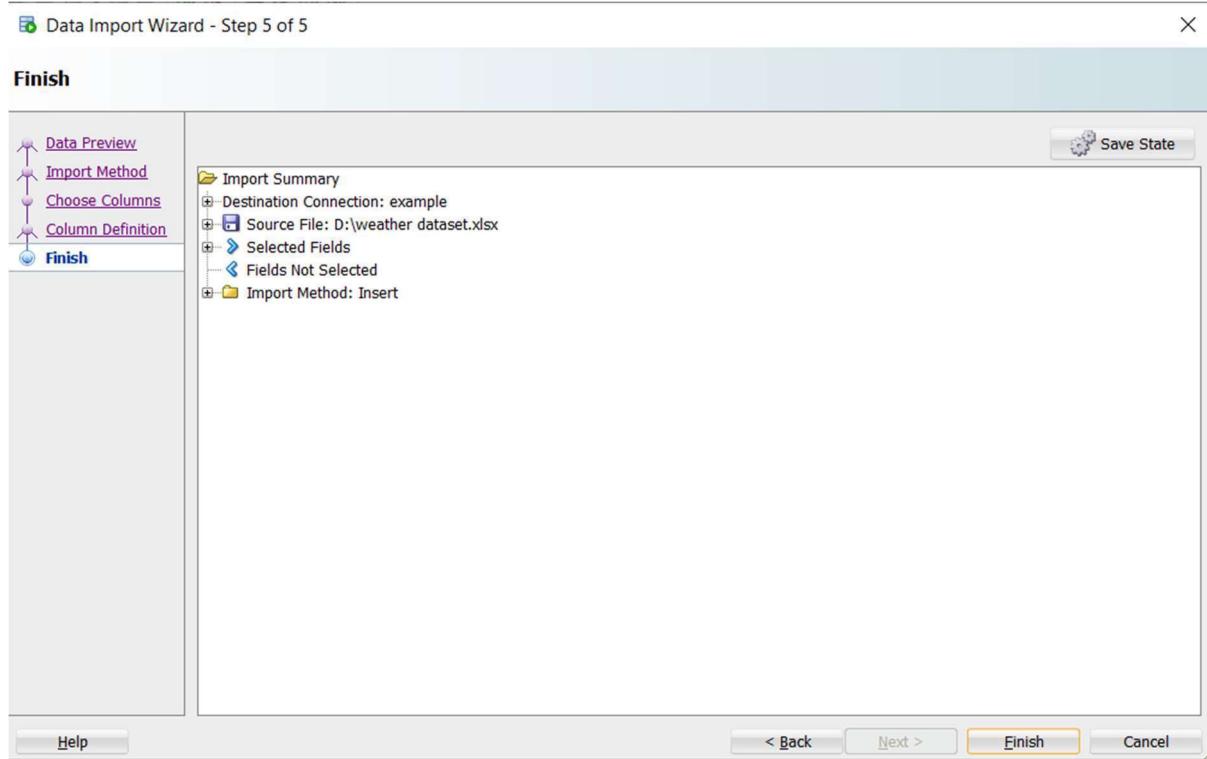
Step 7:

Here you can look up the column details for each column for eg: Name of the Column, Data type, Nullable? Size etc. check the column name one by one then click next.



Step 8:

Then finally click finish. You will get a prompt like “Import Data. Task successful and import committed.” Then click ok.



CHECK WHETHER THE DATASET HAS PRIMARY KEY?

```
SQL> desc weather_india;
```

Name	Null?	Type
C_TEMP		NUMBER(38)
MONTH		NUMBER(38)
DAY		NUMBER(38)
YEAR		NUMBER(38)
TEMPERATURE		NUMBER(38,1)
CITY		VARCHAR2(26)

QUERY TO ADD CONSTRAINT PRIMARY KEY:

```
SQL> alter table weather_india add constraint temp_pk primary key(c_temp);
```

Table altered.

```
SQL> desc weather_india;
```

Name	Null?	Type
C_TEMP	NOT NULL	NUMBER(38)
MONTH		NUMBER(38)
DAY		NUMBER(38)
YEAR		NUMBER(38)
TEMPERATURE		NUMBER(38,1)
CITY		VARCHAR2(26)

After the primary key is added to check we use describe followed by table name: describe weather_india; It shows the column that you gave as primary key as NOT NULL and these primary keys are unique, they don't have any repeated values.

Write any 5 queries using SELECT and WHERE clause:

SQL> select * from weather_india where temp_>100;

C_TEMP	MONTH	DAY	YEAR	TEMP_CITY
18523	6	2	1995	101.7 delhi
18525	6	4	1995	101.1 delhi
18528	6	7	1995	100.1 delhi
18530	6	9	1995	101.9 delhi
18531	6	10	1995	100.7 delhi
18532	6	11	1995	100.3 delhi
18533	6	12	1995	100.9 delhi
18534	6	13	1995	100.7 delhi
18535	6	14	1995	102.8 delhi
18536	6	15	1995	102.9 delhi
18537	6	16	1995	102.3 delhi

C_TEMP	MONTH	DAY	YEAR	TEMP_CITY
18538	6	17	1995	101.9 delhi
19610	5	24	1998	100.5 delhi
19612	5	26	1998	101.9 delhi
19613	5	27	1998	100.4 delhi
19614	5	28	1998	103.7 delhi
19615	5	29	1998	102.9 delhi
20345	5	28	2000	100.3 delhi
20692	5	10	2001	100.6 delhi
21066	5	19	2002	100.4 delhi

21446 6 3 2003 100.6 delhi

21447 6 4 2003 101.4 delhi

C_TEMP MONTH DAY YEAR TEMP_CITY

21798 5 20 2004 101.3 delhi

22194 6 20 2005 100.2 delhi

23658 6 23 2009 100.5 delhi

23661 6 26 2009 100.8 delhi

23662 6 27 2009 101.4 delhi

23986 5 17 2010 101.2 delhi

23987 5 18 2010 100.9 delhi

24002 6 2 2010 101.5 delhi

24020 6 20 2010 100.9 delhi

24022 6 22 2010 101.6 delhi

24023 6 23 2010 101.8 delhi

C_TEMP MONTH DAY YEAR TEMP_CITY

22907 6 3 2007 100.7 delhi

22911 6 7 2007 100.2 delhi

22912 6 8 2007 101 delhi

22913 6 9 2007 100.9 delhi

22914 6 10 2007 102.1 delhi

24731 5 31 2012 102.5 delhi

24732 6 1 2012 101.1 delhi

24746 6 15 2012 100.8 delhi

24747 6 16 2012 100.3 delhi

24748 6 17 2012 101.1 delhi

24752	6	21	2012	101.3	delhi
-------	---	----	------	-------	-------

C_TEMP	MONTH	DAY	YEAR	TEMP_CITY
--------	-------	-----	------	-----------

24762	7	1	2012	100.1	delhi
-------	---	---	------	-------	-------

25088	5	23	2013	100.3	delhi
-------	---	----	------	-------	-------

25468	6	7	2014	101.3	delhi
-------	---	---	------	-------	-------

25470	6	9	2014	100.2	delhi
-------	---	---	------	-------	-------

25481	6	20	2014	100.2	delhi
-------	---	----	------	-------	-------

25820	5	25	2015	100.1	delhi
-------	---	----	------	-------	-------

27287	5	30	2019	100.8	delhi
-------	---	----	------	-------	-------

27288	5	31	2019	101.7	delhi
-------	---	----	------	-------	-------

27296	6	8	2019	100.3	delhi
-------	---	---	------	-------	-------

27298	6	10	2019	102.6	delhi
-------	---	----	------	-------	-------

27299	6	11	2019	101.5	delhi
-------	---	----	------	-------	-------

C_TEMP	MONTH	DAY	YEAR	TEMP_CITY
--------	-------	-----	------	-----------

27303	6	15	2019	100.3	delhi
-------	---	----	------	-------	-------

27317	6	29	2019	100.6	delhi
-------	---	----	------	-------	-------

27318	6	30	2019	100.3	delhi
-------	---	----	------	-------	-------

26164	5	2	2016	100.4	delhi
-------	---	---	------	-------	-------

26197	6	4	2016	100.8	delhi
-------	---	---	------	-------	-------

26542	5	15	2017	100.3	delhi
-------	---	----	------	-------	-------

26562	6	4	2017	103.6	delhi
-------	---	---	------	-------	-------

26563	6	5	2017	103.3	delhi
-------	---	---	------	-------	-------

26564	6	6	2017	100.3	delhi
-------	---	---	------	-------	-------

64 rows selected.

```
SQL> select temp_,city from weather_india where month=9 and year=2013 and city='kolkata';
```

TEMP_CITY

87.3 kolkata

80.5 kolkata

84.1 kolkata

87.8 kolkata

88.5 kolkata

83.5 kolkata

84.9 kolkata

81.3 kolkata

81.5 kolkata

82.1 kolkata

84.2 kolkata

TEMP_CITY

86.5 kolkata

88.7 kolkata

82.1 kolkata

85.9 kolkata

86 kolkata

82.7 kolkata

83 kolkata

82.4 kolkata

82.7 kolkata

84.4 kolkata

86.1 kolkata

TEMP_CITY

87.6 kolkata

88.5 kolkata

86.8 kolkata

84.8 kolkata

83.6 kolkata

80.5 kolkata

80.4 kolkata

78.8 kolkata

30 rows selected.

SQL> select city,month from weather_india where temp_<65 and year<1996;

CITY	MONTH
------	-------

chennai	3
---------	---

chennai	6
---------	---

mumbai	3
--------	---

delhi	11
-------	----

delhi	11
-------	----

delhi	11
-------	----

delhi	11
-------	----

delhi	11
-------	----

delhi	11
-------	----

delhi	11
-------	----

delhi	11
-------	----

delhi 12

delhi 12

delhi 12

delhi 12

delhi 12

delhi 12

dolbi 12

12

13

12

delhi 12

delhi 12

dolbi 12

dolbi 12

12

13

卷之二

denli 12

12

delhi 12

delhi 12

delhi 12

delhi 12

delhi 12

delhi 12

dolbi 12

dolbi 12

12

dolbi 1

- 1 -

[View Details](#)

dem

1

1

delhi

delhi

delhi 1

CITY	MONTH
-----	-----

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

CITY	MONTH
-----	-----

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	1
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

CITY	MONTH
-----	-----

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

CITY	MONTH
-----	-----

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	2
-------	---

delhi	3
-------	---

delhi	3
-------	---

delhi	3
-------	---

delhi	3
-------	---

delhi	3
-------	---

delhi	3
-------	---

delhi	3
-------	---

CITY	MONTH
------	-------

delhi	3
-------	---

delhi	3
-------	---

delhi	3
-------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

CITY	MONTH
------	-------

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	1
---------	---

kolkata	2
---------	---

kolkata	3
---------	---

CITY	MONTH
------	-------

kolkata	6
kolkata	6
kolkata	12
kolkata	12

125 rows selected.

SQL> select temp_,city from weather_india where month=5 and day=15 and year=2000;

TEMP_CITY

93.7 chennai

86.6 mumbai

95.5 delhi

89.2 kolkata

5 rows selected.

SQL> select year , city from weather_india where temp_>103;

YEAR CITY

1998 delhi

2017 delhi

2017 delhi

3 rows selected.

Write 5 queries using Aggregate function (min, max , avg and count)

SQL> select count(day) from weather_india where temp_<60 and year=2005;

COUNT(DAY)

70

SQL> select max(temp_) from weather_india where year between 1995 and 2000;

MAX(TEMP_)

103.7

SQL> select count(distinct year) from weather_india where temp_>99;

COUNT(DISTINCTYEAR)

20

SQL> select min(temp_) from weather_india;

MIN(TEMP_)

-99

SQL> select avg(temp_) from weather_india where year=2020;

AVG(TEMP_)

70.7324074

Sample queries for India weather analytics using historical data part-1.

1.What is the lowest, highest and average temperature of your dataset (from year 1995 to February 2020)?

```
SQL> select min(temperature), max(temperature), avg(temperature) from weather_india  
where year>1994 and year<2021 and month=2;
```

MIN(TEMPERATURE) MAX(TEMPERATURE) AVG(TEMPERATURE)

```
----- ----- -----  
-99 88.9 73.2252401
```

2.What is the average temperature in May 2019 in Chennai?

```
SQL> select avg(temperature) from weather_india where month=5 and year=2019 and  
city='chennai';
```

AVG(TEMPERATURE)

```
-----  
73.6064516
```

3.Which is the hottest day in 2019 in Delhi?

```
SQL> select max(temperature) from weather_india where year=2019 and city='delhi';
```

MAX(TEMPERATURE)

```
-----  
102.6
```

4.Which is the coldest day in 2018 in Chennai?

```
SQL> select min(temperature) from weather_india where year=2018 and city='chennai';
```

```
MIN(TEMPERATURE)
```

```
-----  
-99
```

5.Which is the coldest year in December? Print city and temperature.

```
SQL> select city, temperature, year from weather_india where month=12 and  
temperature<30;
```

CITY	TEMPERATURE	YEAR
chennai	-99	1998
chennai	-99	2015
chennai	-99	2015
mumbai	-99	1998
mumbai	-99	2007

CITY	TEMPERATURE	YEAR
mumbai	-99	2015
mumbai	-99	2015
delhi	-99	1998
delhi	-99	2007
delhi	-99	2015
delhi	-99	2015
kolkata	-99	2015
kolkata	-99	2015

CITY	TEMPERATURE	YEAR
kolkata	-99	1998

26 rows selected.

6.Which is the hottest city in India in 2017?

SQL> select city, temperature, year from weather_india where year=2017 and temperature>100;

CITY	TEMPERATURE	YEAR
delhi	100.3	2017
delhi	103.6	2017
delhi	103.3	2017
delhi	100.3	2017

7.Is winter in Delhi in January 2017 colder than Mumbai? Print average temperature of Delhi and Mumbai too.

SQL> select avg(temperature) from weather_india where city between 'delhi' and 'mumbai' and month=1 and year=2017;

AVG(TEMPERATURE)

68.2494624

8.Display the day, month, year and city of the coldest day.

SQL> select day, month, year, city from weather_india where temperature<20;

DAY	MONTH	YEAR CITY
29	3	1995 chennai
22	6	1995 chennai
26	10	1996 chennai
18	6	2002 chennai
19	6	2002 chennai
20	6	2002 chennai
21	6	2002 chennai
26	10	2002 chennai
13	1	2003 chennai
16	8	1998 chennai
24	12	1998 chennai

DAY	MONTH	YEAR CITY
25	12	1998 chennai
30	12	1998 chennai
31	12	1998 chennai
10	1	1999 chennai
28	8	2007 chennai
9	4	2009 chennai
24	9	2008 chennai
6	2	2014 chennai
30	12	2015 chennai
31	12	2015 chennai

10 3 2016 chennai

DAY MONTH YEAR CITY

11 11 2016 chennai

14 11 2018 chennai

17 11 2018 chennai

24 4 2019 chennai

16 5 2019 chennai

17 5 2019 chennai

18 5 2019 chennai

29 3 1995 mumbai

24 12 1998 mumbai

25 12 1998 mumbai

30 12 1998 mumbai

DAY MONTH YEAR CITY

31 12 1998 mumbai

10 1 1999 mumbai

18 6 2002 mumbai

19 6 2002 mumbai

20 6 2002 mumbai

21 6 2002 mumbai

28 8 2007 mumbai

7 12 2007 mumbai

22 1 2008 mumbai

18 8 2008 mumbai

24 9 2008 mumbai

DAY	MONTH	YEAR CITY
-----	-------	-----------

9	4	2009 mumbai
6	2	2014 mumbai
30	12	2015 mumbai
31	12	2015 mumbai
10	3	2016 mumbai
11	11	2016 mumbai
14	11	2018 mumbai
15	11	2018 mumbai
17	11	2018 mumbai
23	1	2019 mumbai
24	4	2019 mumbai

DAY	MONTH	YEAR CITY
-----	-------	-----------

16	5	2019 mumbai
17	5	2019 mumbai
18	5	2019 mumbai
19	5	2019 mumbai
29	3	1995 delhi
16	8	1998 delhi
1	10	1998 delhi
24	12	1998 delhi
25	12	1998 delhi
30	12	1998 delhi
31	12	1998 delhi

DAY MONTH YEAR CITY

10	1	1999	delhi
29	4	1999	delhi
30	4	1999	delhi
28	4	2001	delhi
18	6	2002	delhi
19	6	2002	delhi
20	6	2002	delhi
21	6	2002	delhi
26	10	2002	delhi
28	8	2007	delhi
22	10	2007	delhi

DAY MONTH YEAR CITY

7	12	2007	delhi
24	9	2008	delhi
2	3	2009	delhi
9	4	2009	delhi
17	4	2010	delhi
18	4	2010	delhi
19	4	2010	delhi
21	4	2010	delhi
9	6	2011	delhi
10	6	2011	delhi
11	6	2011	delhi

DAY MONTH YEAR CITY

12	6	2011 delhi
13	6	2011 delhi
6	3	2007 delhi
7	3	2007 delhi
20	5	2007 delhi
30	12	2015 delhi
31	12	2015 delhi
10	3	2016 delhi
15	5	2013 delhi
6	2	2014 delhi
4	1	2018 delhi

DAY MONTH YEAR CITY

17	9	2018 delhi
24	9	2018 delhi
14	11	2018 delhi
15	11	2018 delhi
17	11	2018 delhi
23	1	2019 delhi
24	4	2019 delhi
25	4	2019 delhi
27	4	2019 delhi
30	4	2019 delhi
16	5	2019 delhi

DAY	MONTH	YEAR	CITY
17	5	2019	delhi
18	5	2019	delhi
19	5	2019	delhi
29	3	1995	kolkata
22	6	1995	kolkata
23	6	1995	kolkata
8	3	1996	kolkata
14	3	1996	kolkata
15	3	1996	kolkata
20	3	1996	kolkata
22	3	1996	kolkata

DAY	MONTH	YEAR	CITY
16	4	2016	delhi
5	5	2016	delhi
21	9	2016	delhi
11	11	2016	delhi
19	4	2017	delhi
20	4	2017	delhi
6	2	2014	kolkata
30	12	2015	kolkata
31	12	2015	kolkata
10	3	2016	kolkata
11	11	2016	kolkata

DAY	MONTH	YEAR	CITY
-----	-------	------	------

6	3	2007	kolkata
7	3	2007	kolkata
20	5	2007	kolkata
28	8	2007	kolkata
24	12	1998	kolkata
25	12	1998	kolkata
30	12	1998	kolkata
31	12	1998	kolkata
10	1	1999	kolkata
24	9	2008	kolkata
9	4	2009	kolkata

DAY	MONTH	YEAR	CITY
-----	-------	------	------

14	11	2018	kolkata
15	11	2018	kolkata
17	11	2018	kolkata
23	1	2019	kolkata
24	4	2019	kolkata
22	3	2000	kolkata
23	3	2000	kolkata
24	3	2000	kolkata
16	5	2019	kolkata
17	5	2019	kolkata
18	5	2019	kolkata

DAY	MONTH	YEAR	CITY
19	5	2019	kolkata
18	6	2002	kolkata
19	6	2002	kolkata
20	6	2002	kolkata
21	6	2002	kolkata
26	10	2002	kolkata

160 rows selected.