



## | Business Analyst Career Program (ChatGPT Included) |

|| Gourav Kinker || [gouravkinker111@gmail.com](mailto:gouravkinker111@gmail.com) ||

### SQL [Major]

**Q1) Create a table "STATION" to store information about weather observation stations.**

**Solution ->**

create table STATION(

ID int primary key,

CITY char(20),

STATE char(2),

LAT\_N int,

LONG\_W int );

Input -

```
3  -- Q1) Create a table "STATION" to store information about weather observation stations.
4
5  • create table STATION(
6    ID int primary key,
7    CITY char(20),
8    STATE char(2),
9    LAT_N int,
10   LONG_W int );
```

Output -

Result Grid					
Filter Rows: <input type="text"/>					
Edit:					
	ID	CITY	STATE	LAT_N	LONG_W
*	NULL	NULL	NULL	NULL	NULL

station 3 x

## Q2) Insert the following records into the table

Solution ->

insert into STATION (ID ,CITY ,STATE ,LAT\_N , LONG\_W)

Values (13 , 'PHOENIX' , 'AZ' , 33 , 112),

(44 , 'DENVER' , 'CO' ,40 ,105 ),

(66 , 'CARIBOU' , 'ME' , 47 , 68 );

Input -

```

12  -- Q2) Insert the following records into the table
13
14  •  insert into STATION (ID ,CITY ,STATE ,LAT_N , LONG_W)
15  Values (13 , 'PHOENIX' , 'AZ' , 33 , 112),
16  (44 , 'DENVER' , 'CO' ,40 ,105 ),
17  (66 , 'CARIBOU' , 'ME' , 47 , 68 );
18

```

Output -

```

12  -- Q2) Insert the following records into the table
13
14  •  insert into STATION (ID ,CITY ,STATE ,LAT_N , LONG_W)
15  Values (13 , 'PHOENIX' , 'AZ' , 33 , 112),
16  (44 , 'DENVER' , 'CO' ,40 ,105 ),
17  (66 , 'CARIBOU' , 'ME' , 47 , 68 );
18

```

Result Grid					
		Filter Rows:		Edit:	
	ID	CITY	STATE	LAT_N	LONG_W
▶	13	PHOENIX	AZ	33	112
	44	DENVER	CO	40	105
	66	CARIBOU	ME	47	68
•	NULL	NULL	NULL	NULL	NULL

**Q3) Execute a query to look at table STATION in undefined order.**

Solution ->

Select \* from station;

Input –



```
19      -- Q3) Execute a query to look at table STATION in undefined order.  
20  
21 •    Select * from station;  
22
```

Output-




```
21 • Select * from station;
22
```

---

Result Grid

Filter Rows:

Edit:   

Export/Imp

	ID	CITY	STATE	LAT_N	LONG_W
▶	13	PHOENIX	AZ	33	112
	44	DENVER	CO	40	105
	66	CARIBOU	ME	47	68
•	NULL	NULL	NULL	NULL	NULL

**Q4) Execute a query to select Northern stations (Northern latitude>39.7)**

Solution ->



Select \* from STATION

Where LAT\_N > 39.7;

Input -

```
23      -- Q4) Execute a query to select Northern stations (Northernlatitude>39.7)
24
25 •    Select * from STATION
26      Where LAT_N > 39.7;
27
```

Output-

Result Grid					
Filter Rows: <input type="text"/>					
Edit:  					
	ID	CITY	STATE	LAT_N	LONG_W
▶	44	DENVER	CO	40	105
	66	CARIBOU	ME	47	68
*	NULL	NULL	NULL	NULL	NULL



## Q6) Populate the table STATS with some statistics for January and July

Solution ->

```
INSERT INTO STATS (ID, MONTH, TEMP_F, RAIN_IN)
```

```
VALUES
```

```
(13, 1, 57.4, .31),
```

```
(13, 7, 91.7, 5.15),
```

```
(44, 1, 27.3, .18),
```

```
(44, 7, 74.8, 2.11),
```

```
(66, 1, 6.7, 2.1),
```

```
(66, 7, 65.8, 4.52);
```

Input –

```
39 • INSERT INTO STATS (ID, MONTH, TEMP_F, RAIN_IN)
40 VALUES
41 (13, 1, 57.4, .31),
42 (13, 7, 91.7, 5.15),
43 (44, 1, 27.3, .18),
44 (44, 7, 74.8, 2.11),
45 (66, 1, 6.7, 2.1),
46 (66, 7, 65.8, 4.52);
47
```

Output –

Result Grid				
Filter Rows:				
	ID	MONTH	TEMP_F	RAIN_IN
▶	13	1	57.4	0.31
	13	7	91.7	5.15
	44	1	27.3	0.18
	44	7	74.8	2.11
	66	1	6.7	2.1
	66	7	65.8	4.52

STATS 2 ×

**Q7) Execute a query to display temperature stats (from the STATS table) for each city (from the STATION table).**

**Solution ->**

```
select st.city , avg(temp_f) As avg_Temp ,
max(temp_f) As Max_Temp,
min(temp_f) As Min_Temp from STATS As s
Join STATION As st
on s.id = st.id
Group By st.city;
```

**Input-**

```
50  -- Q7) Execute a query to display temperature stats (from the STATS table) for each city (from the STATION table).
51
52 • select st.city , avg(temp_f) As avg_Temp , max(temp_f) As Max_Temp, min(temp_f) As Min_Temp from STATS As s
53   Join STATION As st
54   on s.id = st.id
55   Group By st.city;
56
```

**Output-**

Result Grid					Filter Rows:		Export:	Wrap Cell Content:
	city	avg_Temp	Max_Temp	Min_Temp				
▶	PHOENIX	74.54999923706055	91.7	57.4				
	DENVER	51.05000114440918	74.8	27.3				
	CARIBOU	36.250001430511475	65.8	6.7				

**Q8) Execute a query to look at the table STATS, ordered by month and greatest rainfall, with columns rearranged. It should also show the corresponding cities.**



```
select st.city , s.month , s.rain_in ,s.temp_f from stats as s
join station as st
on st.id = s.id
order by s.month , s.rain_in desc ;
```


input-

```
/*Q8) Execute a query to look at the table STATS, ordered by month and greatest rainfall,
with columns rearranged. It should also show the corresponding cities.*/
```

```
select st.city , s.month , s.rain_in ,s.temp_f from stats as s
join station as st
on st.id = s.id
order by s.month , s.rain_in desc ;
```

Output-

Result Grid   Filter Rows: <input type="text"/>				
	city	month	rain_in	temp_f
▶	CARIBOU	1	2.1	6.7
	PHOENIX	1	0.31	57.4
	DENVER	1	0.18	27.3
	PHOENIX	7	5.15	91.7
	CARIBOU	7	4.52	65.8
	DENVER	7	2.11	74.8

Result 7 



**Q9) Execute a query to look at temperatures for July from table STATS, lowest temperatures first, picking up city name and latitude.**

**Solution->**

```
select st.city , s.month , st.lat_n , s.temp_f from stats as s
join station as st
on st.id = s.id
where s.month = 7
order by s.temp_f asc;
```

input-

```
65  /*Q9) Execute a query to look at temperatures for July from table STATS,
66  lowest temperatures first, picking up city name and latitude.*/
67
68  •  select st.city , s.month , st.lat_n , s.temp_f  from stats as s
69      join station as st
70      on st.id = s.id
71      where s.month = 7
72      order by s.temp_f asc;
```

Output-

Result Grid					Filter Rows:	Ex
	city	month	lat_n	temp_f		
▶	CARIBOU	7	47	65.8		
	DENVER	7	40	74.8		
	PHOENIX	7	33	91.7		

**Q10) Execute a query to show MAX and MIN temperatures as well as average rainfall for each city.**

**Solution->**

```
select st.city , max(s.temp_f) as Max_Temp, min(s.temp_f) as Min_Temp, Avg(s.rain_in)
from stats As s
join station as st
on st.id =s.id
group by st.city;
```

input-

```
74  -- Q10) Execute a query to show MAX and MIN temperatures as well as average rainfall for each city.
75
76  • select st.city , max(s.temp_f) as Max_Temp, min(s.temp_f) as Min_Temp, Avg(s.rain_in) from stats As s
77    join station as st
78    on st.id =s.id
79    group by st.city;
80
```

output-

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	city	Max_Temp	Min_Temp	Avg(s.rain_in)
▶	PHOENIX	91.7	57.4	2.7300000488758087
	DENVER	74.8	27.3	1.1449999511241913
	CARIBOU	65.8	6.7	3.309999942779541

Result 11

### Q11) Execute a query to display each city's monthly temperature in Celsius and rainfall in Centimeter.



**Solution->**


```
select st.city , s.month ,round((s.temp_f -32 ) * 5/9 ,2) as Temp_celsi ,
round(s.rain_in * 2.54 , 2) as rain_cent from stats as s
join station as st
on st.id =s.id;
```

**Input\***

```
81  -- Q11) Execute a query to display each city's monthly temperature in Celsius and rainfall in Centimeter.
82
83
84  • select st.city , s.month ,round((s.temp_f -32 ) * 0.5555 ,2) as Temp_celsi ,
85    round(s.rain_in * 2.54 , 2) as rain_cent from stats as s
86    join station as st
87    on st.id =s.id;
88
```

**Output-**

Result Grid   Filter Rows: <input type="text"/> Export:				
	city	month	Temp_celsi	rain_cent
▶	PHOENIX	1	14.11	0.79
	PHOENIX	7	33.16	13.08
	DENVER	1	-2.61	0.46
	DENVER	7	23.78	5.36
	CARIBOU	1	-14.05	5.33
	CARIBOU	7	18.78	11.48

Result 12 

**Q12) Update all rows of table STATS to compensate for faulty rain gauges known to read 0.01 inches slow.**

**Solution-<**

update stats rain\_in

set rain\_in = rain\_in + 0.01

input-

```

89  -- Q12) Update all rows of table STATS to compensate for faulty rain gauges known to read 0.01 inches slow.
90
91  • update stats rain_in
92    set rain_in = rain_in + 0.01
93

```

Output-

**Before Add rain\_in +0.001**

Result Grid				
Filter Rows:				
	ID	MONTH	TEMP_F	RAIN_IN
▶	13	1	57.4	0.31
	13	7	91.7	5.15
	44	1	27.3	0.18
	44	7	74.8	2.11
	66	1	6.7	2.1
	66	7	65.8	4.52

STATS 13 ×

**After Add rain\_in +0.001**

Result Grid				
Filter Rows:				
	ID	MONTH	TEMP_F	RAIN_IN
▶	13	1	57.4	0.32
	13	7	91.7	5.16
	44	1	27.3	0.19
	44	7	74.8	2.12
	66	1	6.7	2.11
	66	7	65.8	4.53

STATS 18 ×

### Q13) Update Denver's July temperature reading as 74.9.

Solution->

update STATS set TEMP\_F = 74.9

where id = 44 and month = 7;

input-



```

96  -- Q13) Update Denver's July temperature reading as 74.9.
97
98  •  update STATS set TEMP_F = 74.9
99  where id = 44 and month = 7;

```



Output-

**Before Update Denver's July TEMP\_F = 74.8**

Result Grid   Filter Rows: <input type="text"/>				
	ID	MONTH	TEMP_F	RAIN_IN
▶	13	1	57.4	0.32
	13	7	91.7	5.16
	44	1	27.3	0.19
	44	7	74.8	2.12
	66	1	6.7	2.11
	66	7	65.8	4.53

STATS 16 ×

**After Update Denver's July TEMP\_F = 74.9**

Result Grid   Filter Rows: <input type="text"/>				
	ID	MONTH	TEMP_F	RAIN_IN
▶	13	1	57.4	0.32
	13	7	91.7	5.16
	44	1	27.3	0.19
	44	7	74.9	2.12
	66	1	6.7	2.11
	66	7	65.8	4.53

STATS 15 ×