

# SQL Major Assignment

**Answers**

# 1. Create a table “Station” to store information about weather observation stations

SQL Worksheet

 Clear

 Find

Actions ▼

 Save

Run 

```
1 CREATE TABLE STATION
2 (ID INTEGER PRIMARY KEY,
3  CITY CHAR(20),
4  STATE CHAR(2),
5  LAT_N INT,
6  LONG_W INT);
```

Table created.

## 2. Insert the following records into the table:

SQL Worksheet

 Clear

 Find

Actions ▼

```
1 v INSERT INTO STATION (ID, CITY, STATE, LAT_N, LONG_W)
2   VALUES(13, 'PHOENIX', 'AZ', 33, 112);
3 v INSERT INTO STATION (ID, CITY, STATE, LAT_N, LONG_W)
4   VALUES(44, 'DENVER', 'CO', 40, 105);
5 v INSERT INTO STATION (ID, CITY, STATE, LAT_N, LONG_W)
6   VALUES(66, 'CARIBOU', 'ME', 47, 68);
```


1 row(s) inserted.


1 row(s) inserted.

1 row(s) inserted.


### 3. Execute a query to look at table STATION in undefined order


**SQL Worksheet**

 Clear

 Find

Actions ▾

 Save

Run 

1 `SELECT * FROM STATION`

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

Download CSV

3 rows selected.

## 4. Execute a query to select Northern stations (Northern latitude > 39.7)

### SQL Worksheet

```
1 SELECT * FROM STATION
2 WHERE LAT_N > 39.7;
```

ID	CITY	STATE	LAT_N	LONG_W
44	Denver	CO	40	105
66	Caribou	ME	47	68

[Download CSV](#)

2 rows selected.

5. Create another table, 'STATS', to store normalized temperature and precipitation data:

SQL Worksheet


```
1 CREATE TABLE STATS
2 (ID INTEGER REFERENCES STATION(ID),
3  MONTH INTEGER CHECK (MONTH BETWEEN 1 AND 12),
4  TEMP_F REAL CHECK (TEMP_F BETWEEN -80 AND 150),
5  RAIN_I REAL CHECK (RAIN_I BETWEEN 0 AND 100),
6  PRIMARY KEY (ID, MONTH));
```

Table created.

## 6. Populate the table STATS with some statistics for January and July

SQL Worksheet

 Clear

 Find

Actions ▼

 Save

```
1 v INSERT INTO STATS(ID,"MONTH",TEMP_F,RAIN_I)
2   VALUES(13,1,57.4,31);
3 v INSERT INTO STATS(ID,"MONTH",TEMP_F,RAIN_I)
4   VALUES(13,7,91.7,5.15);
5 v INSERT INTO STATS(ID,"MONTH",TEMP_F,RAIN_I)
6   VALUES(44,1,27.3,.18);
7 v INSERT INTO STATS(ID,"MONTH",TEMP_F,RAIN_I)
8   VALUES(44,7,74.8,2.11);
9 v INSERT INTO STATS(ID,"MONTH",TEMP_F,RAIN_I)
10  VALUES(66,1,6.7,2.1);
11 v INSERT INTO STATS(ID,"MONTH",TEMP_F,RAIN_I)
12  VALUES(66,7,65.8,4.52);
```

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

# 7. Execute a query to display temperature stats (from STATS table) for each city (from Station table).

**SQL Worksheet**

```
1 SELECT CITY, TEMP_F FROM STATION A INNER JOIN STATS B
2 ON A.ID = B.ID
```

Phoenix	91.7
Denver	27.3
Denver	74.8
Caribou	6.7
Caribou	65.8

[Download CSV](#)

6 rows selected.



8. 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.

#### SQL Worksheet

```
1 SELECT MONTH, RAIN_I, CITY
2 FROM STATS A JOIN STATION B ON A.ID = B.ID
3 ORDER BY MONTH, RAIN_I DESC;
```

MONTH	RAIN_I	CITY
1	2.1	Caribou
1	.31	Phoenix
1	.18	Denver
7	5.15	Phoenix
7	4.52	Caribou
7	2.11	Denver

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

#### SQL Worksheet

```
1 SELECT MONTH, CITY, TEMP_F, LAT_N
2 FROM STATS, STATION
3 WHERE MONTH = 7
4 AND STATS.ID = STATION.ID
5 ORDER BY TEMP_F;
```

MONTH	CITY	TEMP_F	LAT_N
7	Caribou	65.8	47
7	Denver	74.8	40
7	Phoenix	91.7	33

Download CSV

# 10. Execute a query to show MAX and MIN temperatures as well as average rainfall for each city.

## SQL Worksheet

```
1 SELECT CITY, MAX(TEMP_F), MIN(TEMP_F), AVG(RAIN_I)
2 FROM STATS A JOIN STATION B
3 ON A.ID = B.ID
4 GROUP BY CITY;
```

CITY	MAX(TEMP_F)	MIN(TEMP_F)	AVG(RAIN_I)
Caribou	65.8	6.7	3.31
Denver	74.8	27.3	1.145
Phoenix	91.7	57.4	2.73

Download CSV

3 rows selected.

11. Execute a query to display each city's monthly temperature in Celcius and rainfall in Centimeter.

## SQL Worksheet

```
1 | SELECT CITY,  
2 |      ((TEMP_F - 32) * (5/9)) AS TEM_C,  
3 |      (RAIN_I * 2.54) AS RAIN_CM  
4 | FROM STATS A JOIN STATION B  
5 | ON A.ID=B.ID;
```

[illegible]

## 12. Update all rows of table STATS to compensate for faulty rain gauges known to read 0.01 inches low.

### SQL Worksheet

```
1 UPDATE STATS SET RAIN_I = RAIN_I + 0.01;
```

6 row(s) updated.

### SQL Worksheet

```
1 SELECT * FROM STATS;
```

ID	MONTH	TEMP_F	RAIN_I
13	1	57.4	.32
13	7	91.7	5.16
44	1	27.3	.19
44	7	74.8	2.12
66	1	6.7	2.11
66	7	65.8	4.53

# 13. Update Denver's July temperature reading as 74.9

## SQL Worksheet

```
1 ✓ UPDATE STATS SET TEMP_F = 74.9
2 WHERE ID = 44
3 AND MONTH = 7;
```

1 row(s) updated.

## SQL Worksheet

```
1 SELECT * FROM STATS
```

ID	MONTH	TEMP_F	RAIN_I
13	1	57.4	.32
13	7	91.7	5.16
44	1	27.3	.19
44	7	74.9	2.12
66	1	6.7	2.11
66	7	65.8	4.53