Instructions

1. Solve the questions below **INDIVIDUALLY. Do not take help from Internet resources or your friends, seniors, professors, etc. You could be assigned '0' in the project if you don't abide by this rule.**

2. **This project is to be done only using SQL Statements. Points will be deducted if you don't submit all the relevant SQL statements.**

3. **If you are facing problem with the syntax or looking for some functions to solve the queries, please refer to slides and MS-Access help (F1).**

4. Please email your MS-Access database (.accdb) file with the queries to info@ivyproschool.com / connect@ivyproschool.com (cc: prateek@ivyproschool.com). This will be your project submission.

5. Make sure you name your queries starting with the question number. Eg. Q1-CREATE_Station. I'll deduct 5% marks if the queries are not named properly.

6. Project Submission **deadline** is mentioned in the covering email.

Important Concepts

1. **Composite Key Constraint** – The syntax given in the class notes to create composite key will not work on Access 2007. The correct syntax to create a composite key is using CONSTRAINT clause. For ex –
   a. CREATE TABLE Test(ID Integer, FName CHAR(20), LName CHAR(20), CONSTRAINT Cpk PRIMARY KEY(ID, LName));

2. **UPDATE Statement** - You can use UPDATE statement to update the values (or data) of a table. This is quite useful when you want to update multiple column values from many tables, based on some condition. The syntax is as follows –
   a. UPDATE TableName
      SET ColName=ColName*0.1, Col2Name=Col2Name+5
      WHERE Col4Name='January'

Questions (Total – 13)

1. **Create a table "Station" to store information about weather observation stations:**

ID	Number	Primary Key
CITY	CHAR(20),	
STATE	CHAR(2),	
LAT_N	Number	
LONG_W	Number	

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

ID	CITY	STATE	LAT_N	LONG_W
13	Phoenix	AZ	33	112
44	Denver	CO	40	105
66	Caribou	ME	47	68

3. **Execute a query to look at table STATION in undefined order.**
4. **Execute a query to select Northern stations (Northern latitude > 39.7).**

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

Column	Data type	Remark
ID	Number	must match some STATION table ID(so name & location will be known).
MONTH	Number	Range between 1 and 12
TEMP_F	Number	in Fahrenheit degrees,Range between -80 and 150,
RAIN_I	Number	in inches, Range between 0 and 100,

There will be no Duplicate ID and MONTH combination.

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

ID	MONTH	TEMP_F	RAIN_I
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

7. **Execute a query to display temperature stats (from STATS table) for each city (from Station table).**
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.**
9. **Execute a query to look at temperatures for July from table STATS, lowest temperatures first, picking up city name and latitude.**
10. **Execute a query to show MAX and MIN temperatures as well as average rainfall for each city.**
11. **Execute a query to display each city's monthly temperature in Celcius and rainfall in Centimeter.**
12. **Update all rows of table STATS to compensate for faulty rain gauges known to read 0.01 inches low.**
13. **Update Denver's July temperature reading as 74.9**