Section 2

1. Aggregation & Join

Schema:

CREATE TABLE Population (  
    rank INTEGER,  
    country VARCHAR(30) PRIMARY KEY,  
    population DOUBLE,  
    percentage FLOAT  
);

CREATE TABLE GDP (  
    rank INTEGER,  
    country VARCHAR(30) PRIMARY KEY,  
    gdp DOUBLE  
);

CREATE TABLE Airport (  
    code VARCHAR(30) PRIMARY KEY,  
    name VARCHAR(30),  
    country VARCHAR(30)  
);
**Problems:**

-- What is the total population of earth?

Total_Population

6778067375

-- What is the percentage of the population from the top 10 populated countries?

Top_Sum

58.9241749607129

-- How many countries do have less than 1,000,000 population?

Small_Countries

68

-- How many countries have airports?

Airport_Count

247

-- Top 10 countries with most airports, in descending order

<table>
<thead>
<tr>
<th>country</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>2238</td>
</tr>
<tr>
<td>Australia</td>
<td>617</td>
</tr>
</tbody>
</table>
Canada 533
Papua New Gui 380
Brazil 288
Indonesia 205
China 187
Colombia 167
United Kingdo 151
France 144

-- Order the top 10 countries by total GDP per capita (gdp / population)

<table>
<thead>
<tr>
<th>country</th>
<th>GDP_per_capita</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seychelles</td>
<td>0.282666666666667</td>
</tr>
<tr>
<td>&quot;Saint Kitts and Nevis&quot;</td>
<td>0.256076923076923</td>
</tr>
<tr>
<td>&quot;Antigua and Barbuda&quot;</td>
<td>0.196681818181818</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>0.158883485309017</td>
</tr>
<tr>
<td>Dominica</td>
<td>0.152507462686567</td>
</tr>
<tr>
<td>Brunei</td>
<td>0.119825</td>
</tr>
<tr>
<td>Iceland</td>
<td>0.118570005575638</td>
</tr>
<tr>
<td>Grenada</td>
<td>0.102855769230769</td>
</tr>
<tr>
<td>&quot;Saint Vincent and the Grenadines&quot;</td>
<td>0.0929082568807339</td>
</tr>
<tr>
<td>Barbados</td>
<td>0.087</td>
</tr>
</tbody>
</table>
2. Join & Aggregation

**Schema:**

CREATE TABLE Class (  
department VARCHAR(6),  
number INTEGER,  
title VARCHAR(75),  
PRIMARY KEY (department, number)
);

CREATE TABLE Instructor (  
username VARCHAR(8),  
fname VARCHAR(50),  
lname VARCHAR(50),  
started_on CHAR(10),  
PRIMARY KEY (username)
);

CREATE TABLE Teaches (  
username VARCHAR(8),  
department VARCHAR(6),  
number INTEGER,  
PRIMARY KEY (username, department, number),  
FOREIGN KEY (username) REFERENCES Instructor(username),  
FOREIGN KEY (department, number) REFERENCES Class(department, number)
);
Problems:

/* Review of joins */

-- Who teaches CSE 451?

fname     lname
---------- ----------
Tom        Anderson
John       Zahorjan
Hank       Levy

-- What courses does Zahorjan teach?

department  number
---------- ----------
CSE         378
CSE         451
CSE         461

-- Which courses do both Levy and Zahorjan teach?

department  number  title
---------- ---------- ---------------------------------
CSE         451      Introduction to Operating Systems
/* Queries using aggregation functions */

-- How many classes are there in the course catalog?
COUNT(*)

---------
3

-- What are the highest and lowest class numbers?
MIN(number) MAX(number)

---------  ---------
378        461

/* Queries with both grouping and aggregation */

-- How many instructors teach each class?
department number teacher_count

---------  ---------  -----------
CSE        378       1
CSE        451       3
CSE        461       3
- Order the instructors by who teaches in the most departments

<table>
<thead>
<tr>
<th>username</th>
<th>Department_Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>djw</td>
<td>1</td>
</tr>
<tr>
<td>levy</td>
<td>1</td>
</tr>
<tr>
<td>tom</td>
<td>1</td>
</tr>
<tr>
<td>zahorjan</td>
<td>1</td>
</tr>
</tbody>
</table>