Cosc 416: MongoDB Assignment

First off, you need a database to connect to. MongoDB doesn't have a "create database" command. Instead, it is going to create one for you when you try to save something into it. Ramon has requested that we will run on an honor system, so please refrain from trolling. We have the right to revoke marks.

Connect to the mongoDB shell on **gpu1.ddl.ok.ubc.ca** by running:
`/srv/mongodb/bin/mongo`

The mongo shell is an interactive javascript shell. So you can do things like store objects in variables to save later, store results, loop etc.

To switch to your database, run
`use ########`
(Where #### is your student ID. If you run `show dbs` *you'll see that your database still doesn't exist.)*

Part 1 (20 Marks)

A. (1 mark) Create a collection called 'games'. We're going to put some games in it.

B. (2 marks) Add 5 games to the database. Give each document the following properties:
name, genre, rating (out of 100)
If you make some mistakes and want to clean it out, use `remove()` on your collection.

C. (1 mark) Write a query that returns all the games

D. (2 marks) Write a query to find one of your games by name without using limit(). Use the findOne method. Look how much nicer it's formatted!

E. (3 marks) Write a query that returns the 3 highest rated games.

F. (6 marks) Update your two favourite games to have two achievements called 'Game

Master' and 'Speed Demon', each under a single key. Show two ways to do this. Do the first using **`update()`** and do the second using **`save()`**. Hint: for save, you might want to query the object and store it in a variable first.

G. (2 marks) Write a query that returns all the games that have both the 'Game Maser' and the 'Speed Demon' achievements.

H. (3 marks) Write a query that returns only games that have achievements. Not all of your games should have achievements, obviously.

Part 2 (15 Marks)

MapReduce question:

Use mongoDB map-reduce on the example set of game data. This data is located in the **416** db in the **games** collection. (use the command **`use 416`** to switch to the database)
http://docs.mongodb.org/manual/applications/map-reduce/

Example document:

```
{
	"_id" : ObjectId("5144e16e54f9ef8613927ec4"),
	"name" : "Ape Escape",
	"publisher" : "KOEI Co., Ltd.",
	"released" : "1995-03-27",
	"rating" : 99,
	"scores" : [
		{
			"name" : "derrick",
			"score" : 705
		},
		{
			"name" : "tim",
			"score" : 379
		},
		{
			"name" : "bryan",
			"score" : 810
		}
```

```
        ]
}
```

Write a reduce that calculates the **total score** from all games for **each player**. For the output, please use your student id in your collection name so you don't overwrite other peoples'. The easiest way to create functions in the mongo shell, is to write the methods in a text editor and paste them in, then call them within the shell.
(Note: not all games have scores)

Part 3 (15 Marks)

Use the REST API to show all the game data stored in the **416** db from the **games** collection. This should be similar to the task from the Sensei lab.

Output all of the available returned data in an html table in the following format:

Game	Publisher	Release Date	Rating	Average Score (if available)

The API can be accessed from:
http://gpu1.ddl.ok.ubc.ca:50070/
This link will fail if you don't provide it a query, read the documentation.
(Hint:Use '$json = system("curl <url here>");' to store the data as a string)

The documentation is at:
https://sites.google.com/site/mongodbjavarestserver/home