Name _____

# CSCI-UA.60-1 – Database Design & Web Implementation
## Final Exam – Version A
## Spring, 2020
## Deena Engel

Name _____

NetId _____

The file(s) that you submit with your submission must include your netid and name, e.g.

**`netid_name_database_exam.pdf -- e.g. 'de123_EngelDeena_database_exam.pdf'`**

Each file should contain the following first two lines:

```
-- CSCI-UA.60-1 Final Exam
-- By [Your name goes here]
```

Your written submission(s) should be include one or several .TXT, .SQL and/or .PDF files. If you include any image files, please follow the requirements above to name the files properly.

You may **not** submit Word files, google 'doc' or other Word processing formats.

You may **not** use MySQL Workbench.

Submit the files individually; do NOT zip them together.

Be sure to include the digitally signed honor code file with your submission.

1.) **Database Design** (25 points)
Select a web page based on the first letter of your last name:

A-F Brooklyn Botanic Garden Library: https://www.bbg.org/gardening/library including related pages e.g. botanical art, historical image collections.

G-O Chelsea Piers Sports https://www.chelseapiers.com/sports/ including one or more related pages (e.g. "Triathalon").

P-S. American Museum of Natural History Education Department: https://www.amnh.org/learn-teach

T-Z Cartoon Network (opening page of the site) https://www.cartoonnetwork.com

Describe your database by writing a .SQL script containing tables you will need for the database application above. For every table, be sure to list each field, field type, and where appropriate, the field length along with primary and foreign keys. Assume that you are designing for MySQL. If you find more than 10 tables, you may stop for the purposes of this exam.

If you wish to draw an E-R diagram to supplement your answer and clarify the primary and foreign keys and demonstrate the 1:many relationships, you are welcome to do so but you may not use MySQL Workbench as not all students have access to that software. However, you may use a drawing application of your choice and submit the file and a digital image. If you use pen or pencil on paper, please be sure to write your name and netid on the paper and you may take a photograph of the drawing with your phone and upload it with the rest of your submission. If you use a drawing application or you are working on a tablet, you are welcome to take a "screen shot" and submit it if you cannot save the image. Please be sure to write your name and netid on the drawing before submission.

2.) **Joins in MySQL (15 points)**
Take as much space as you need to reasonably answer the questions below.

(a) Define a *left join,* a *right join* and an *inner join*.

(b) Write an example of a **left join** by writing the MySQL queries necessary to INSERT at least 3 records into two of the tables above that you designed in Problem #1; and then write a SELECT query which uses a **left join**.

(c) Describe your results in several sentences and explain how this example demonstrates the results of a **left** join instead of a right join or an inner join.

**3.) MongoDB** (10 points)
Given the following *.json* file called ***finalExam.json*** which is excerpted and adapted from a NYC "Open Data" source. This file contains information about schools in the New York City area. ***(****Note: "P.S." stands for "Public School")*

{ "OBJECTID": "001", "NAME": "P.S. 3", "ID": "1117", "ADDRESS": "404 W 54th Street", "CITY": "New York", "ZIP": "10019"}
{ "OBJECTID": "002", "NAME": "P.S. 41", "ID": "1120", "ADDRESS": "396 Avenue of the Americas", "CITY": "New York", "ZIP": "10011"}
{ "OBJECTID": "003", "NAME": "P.S. 76", "ID": "1131", "ADDRESS": "16 W. 8th Street", "CITY": "New York", "ZIP": "10011"}
{ "OBJECTID": "004","NAME": "The Earth School", "ID": "1139", "ADDRESS": "75 Orchard Street, ", "CITY": "Queens", "ZIP": "11371"}
{ "OBJECTID": "005", "NAME": "P.S. 97", "ID": "1141", "ADDRESS": "505 Sixth Avenue", "CITY": "Queens", "ZIP": "11371" }
{ "OBJECTID": "006","NAME": "P.S. 1", "ID": "1175", "ADDRESS": "Battery Place & 1 Place", "CITY": "New York", "ZIP": "10280" }
{ "OBJECTID": "007", "NAME": "P.S. 11", "ID": "233", "ADDRESS": "1101 E Tremont Ave", "CITY": "Bronx", "ZIP": "10460" }
{ "OBJECTID": "008","NAME": "P.S. 40", "ID": "234", "ADDRESS": "4174 White Plains Rd", "CITY": "Bronx", "ZIP": "10466" }

You may assume that the following import statement was used to successfully import these data into mongodb:

```
mongoimport --db de123 --collection exam --host class-mongodb.cims.nyu.edu --username de123 --password xyz --type json --file finalExam.json
```

Write mongodb ***queries*** (**not** the answers!) which you would run in order to answer the following questions:

(a) How many schools are in this file?

(b) How many schools listed are in the Bronx?

(c) Display the name and address of all of the schools in the 11371 zip code.

(d) Display the name for all of the schools in Brooklyn and Queens in alphabetical order.

(e) List the school name and zip code for all schools in the zip code range of 10001 – 10011 in order by zip code and then by school.

4.) Compare and contrast SQLite and working in Python for working with data. You may consider the Python modules that we covered in class e.g. *pandas, matplotlib, csv* and any others. (15 points)

| | **SQLite** | **Python** |
|---|---|---|
| Similarity #1: | | |
| Similarity #2: | | |
| Difference #1: | | |
| Difference #2: | | |
| Difference #3: | | |