**CSC 341**

**Project**

**DUE by Friday May 6th at 11:59 PM**

**You will be submitting a project report (Word document or PDF) in addition to files (Python code, data files, etc.)**

**Database**

* The database for your project should have at least four entities.
* Each table should include a few properties (columns).
* Each table should have a few records (rows).
* Primary keys should be indicated.
* It should have at least one one-to-many relationship.
* It should have at least one many-to-many relationship.
* Your project report must include a statement about each relationship in two ways, such as, "A Congress person may be on many committees" and "A committee consists of many congress persons."
* Your project report must include an ER diagram for your database (hand-drawn and scanned in is okay).

**Python, SQLAlchemy & Flask**

* Write a Python/SQLAlchemy file that defines the database and will be imported into subsequent files as needed.
* Write a Python file or files that inserts some data. At least some of the data should come from
  + A simple delimited text file such as a .csv file
  + A JSON file
* Make some pages/routes for project. You should include a list of your routes in your project report. I am not going to specify the number of pages, but the following features and functionality should be present
  + **INSERT**
    - A (WTF) form that inserts data into a database table that has no foreign keys
    - A (WTF) form that inserts data into a database table that has a foreign key provided by a drop-down list populated by the appropriate tables
    - Between these insert forms there should be at least two validated fields (input required or length or email – some sort of validation)
  + **SELECT**
    - Display the result of a query from a single table (e.g. Painters)
    - Display the result of a query from two related tables (e.g. Painters and their paintings)
    - Display the result of a query in an HTML table (or grid)
    - Display the result of a query in a Datatable (jQuery or otherwise) that is searchable and sortable