Assignment 2 Data Cleaning and Integration

Brief

- **Due date:** 11:59PM 10/06/2016
- **Data:** class.txt, superbowl.html
- **Handin:** follow the handin instruction
- **Required files:** p1/README, p1/clean.py, p1/cleaned.txt, p1/query.py, p2/transform.py, p2/result.csv, p2/README

Entity Resolution: UIC Courses

In an unfortunate series of events, the UIC registrar's course catalog has been corrupted, and must be rebuilt. You have been selected to help with rebuilding it, using various data sources that were acquired from various websites. Your job is to develop (and implement) a series of transformation rules that can be used to recreate the registrar's catalog.

In this assignment, you are given a file `class.txt` that shows the courses taught by UIC professors. However, since data is not clean, there might be cases that multiple versions of professor/course names actually refer to the same professor/course. Your task is to clean the dataset using proper transformation rules, and analyze the cleaned dataset by answering some queries below.

First of all, we suggest that you take a look at the dataset and get a sense of why this dataset is dirty. Then, try to come up with some transformation rules that you would like to use when you clean the dataset. The format of the input data is:

```
professor_name - course_1|course_2|...course_n
```

After you have determined the appropriate transformation rules, write a Python script, `clean.py`, to read the dataset, apply transformation rules, and output a cleaned dataset `cleaned.txt`. Note that the cleaned dataset should have the same format as the dirty dataset, and professors should be listed in alphabetical order based on their last name (**do not include their first name in the cleaned dataset**). For each professor, the courses that he/she teaches should also be listed in alphabetical order.

Finally, using the cleaned dataset, write a python script `query.py` with 3 functions (one for of the following queries, named q1,q2, and q3) to answer the following questions. Each of the

functions should take only the cleaned, raw `cleaned.txt` file as input, and print the solution to the console.

```
1: How many distinct courses does this dataset contain?

2: List all the courses (in alphabetical order) taught by Professor Mitchell Theys
in comma-separated form.

3: For professors who have taught at least 5 courses, using Jaccard distance to
determine which two professors have the most aligned teaching interests based on
course titles. Note that you should implement the function to calculate Jaccard
distance instead of using an existing package.
```

Note that `class.txt` is only a subset of the full dataset, and we will run your `clean.py` on our full dataset to evaluate its quality. i.e., it has to be run with the input: `python3 clean.py [any_file].txt`
, as well as your `query.py` script.

Hint

- You can assume that the professor's last name is a unique identifier for the name of the professor.
- Normally, the professor's last name comes after their first name. But when the name is in comma separated form, the first name comes after the last name.
- Note that since the dataset is made up, it does not have to represent the real situation in our department.
- Pay attention to abbreviations!

Note:
You are not supposed to use any external libraries. In case you are stuck, you can confirm your approach/ logic with TA. Points will be given based on the approach you choose to solve the problems. Extra points for solutions that are as generic as possible!!

What to turn in

1. clean.py - Python script that applies your transformation rules.
2. cleaned.txt - Output from `clean.py`.
3. query.py - Python script with 3 functions that answer the above queries.
4. README - Description of transformation rules, any other relevant information.

Reformatting Data: Super Bowl Champions

The Super Bowl is an annual American football game that determines the **champion of the National Football League (NFL)**. In this assignment, you are given the source file `superbowl.html`

which is the HTML source code of the whole Wikipedia webpage. This data is in HTML format, and we would like to transform it into a more usable format. There are few tables in the mentioned web page; however, the objective is to use python and the BeautifulSoup library to scrape the data from the second table and save it into a CSV file `result.csv`. Each line in `result.csv` should contain 6 fields: Game number, year, winning team, score, losing team, and venue. Note that the number after every team (and venue) indicates the number of times that the team (or venue) has been in the Super Bowl.

First, extract relevant portion of `superbowl.html`. Note that an HTML table is divided into rows with the <tr> tag, and each row is divided into data cells with the <td> tag. Using the converted HTML file, write a python script `transform.py` to pull data out of the relevant HTML portion (second table) and save it into a CSV file, named `result.csv` which should match the output shown below. As shown below, the CSV file headings should match those of the table's and the rest rows in the file would be the first 50 rows of the mentioned table on Wikipedia.

```
Game number,year,winning team,score,losing team,venue
I,1967,Green Bay Packers 01,35-10,Kansas City Chiefs 01,Los Angeles Memorial Coliseum
01
II,1968,Green Bay Packers 02,33-14,Oakland Raiders 01,Orange Bowl 01
III,1969,New York Jets 01,16-7,Indianapolis Colts 01,Orange Bowl 02
IV,1970,Kansas City Chiefs 02,23-7,Minnesota Vikings 01,Tulane Stadium 01
V,1971,Indianapolis Colts 02,16-13,Dallas Cowboys 01,Orange Bowl 03
VI,1972,Dallas Cowboys 02,24-3,Miami Dolphins 01,Tulane Stadium 02
VII,1973,Miami Dolphins 02,14-7,Washington Redskins 01,Los Angeles Memorial Coliseum
02
VIII,1974,Miami Dolphins 03,24-7,Minnesota Vikings 02,Rice Stadium 01
IX,1975,Pittsburgh Steelers 01,16-6,Minnesota Vikings 03,Tulane Stadium 03
X,1976,Pittsburgh Steelers 02,21-17,Dallas Cowboys 03,Orange Bowl 04
XI,1977,Oakland Raiders 02,32-14,Minnesota Vikings 04,Rose Bowl 01
XII,1978,Dallas Cowboys 04,27-10,Denver Broncos 01,Louisiana Superdome 01
XIII,1979,Pittsburgh Steelers 03,35-31,Dallas Cowboys 05,Orange Bowl 05
XIV,1980,Pittsburgh Steelers 04,31-19,St. Louis Rams 01,Rose Bowl 02
XV,1981,Oakland Raiders 03,27-10,Philadelphia Eagles 01,Louisiana Superdome 02
XVI,1982,San Francisco 49ers 01,26-21,Cincinnati Bengals 01,Pontiac Silverdome 01
XVII,1983,Washington Redskins 02,27-17,Miami Dolphins 04,Rose Bowl 03
XVIII,1984,Oakland Raiders 04,38-9,Washington Redskins 03,Tampa Stadium 01
XIX,1985,San Francisco 49ers 02,38-16,Miami Dolphins 05,Stanford Stadium 01
XX,1986,Chicago Bears 01,46-10,New England Patriots 01,Louisiana Superdome 03
XXI,1987,New York Giants 01,39-20,Denver Broncos 02,Rose Bowl 04
XXII,1988,Washington Redskins 04,42-10,Denver Broncos 03,Jack Murphy Stadium 01
XXIII,1989,San Francisco 49ers 03,20-16,Cincinnati Bengals 02,Joe Robbie Stadium 01
XXIV,1990,San Francisco 49ers 04,55-10,Denver Broncos 04,Louisiana Superdome 04
XXV,1991,New York Giants 02,20-19,Buffalo Bills 01,Tampa Stadium 02
XXVI,1992,Washington Redskins 05,37-24,Buffalo Bills 02,Metrodome 01
XXVII,1993,Dallas Cowboys 06,52-17,Buffalo Bills 03,Rose Bowl 05
XXVIII,1994,Dallas Cowboys 07,30-13,Buffalo Bills 04,Georgia Dome 01
XXIX,1995,San Francisco 49ers 05,49-26,San Diego Chargers 01,Joe Robbie Stadium 02
XXX,1996,Dallas Cowboys 08,27-17,Pittsburgh Steelers 05,Sun Devil Stadium 01
XXXI,1997,Green Bay Packers 03,35-21,New England Patriots 02,Louisiana Superdome 05
XXXII,1998,Denver Broncos 05,31-24,Green Bay Packers 04,Jack Murphy Stadium 02
XXXIII,1999,Denver Broncos 06,34-19,Atlanta Falcons 01,Joe Robbie Stadium 03
```

```
XXXIV,2000,St. Louis Rams 02,23-16,Tennessee Titans 01,Georgia Dome 02
XXXV,2001,Baltimore Ravens 01,34-7,New York Giants 03,Raymond James Stadium 01
XXXVI,2002,New England Patriots 03,20-17,St. Louis Rams 03,Louisiana Superdome 06
XXXVII,2003,Tampa Bay Buccaneers 01,48-21,Oakland Raiders 05,Jack Murphy Stadium 03
XXXVIII,2004,New England Patriots 04,32-29,Carolina Panthers 01,NRG Stadium 01
XXXIX,2005,New England Patriots 05,24-21,Philadelphia Eagles 02,ALLTEL Stadium 01
XL,2006,Pittsburgh Steelers 06,21-10,Seattle Seahawks 01,Ford Field 01
XLI,2007,Indianapolis Colts 03,29-17,Chicago Bears 02,Joe Robbie Stadium 04
XLII,2008,New York Giants 04,17-14,New England Patriots 06,University of Phoenix
Stadium 01
XLIII,2009,Pittsburgh Steelers 07,27-23,Arizona Cardinals 01,Raymond James Stadium 02
XLIV,2010,New Orleans Saints 01,31-17,Indianapolis Colts 04,Joe Robbie Stadium 05
XLV,2011,Green Bay Packers 05,31-25,Pittsburgh Steelers 08,Cowboys Stadium 01
XLVI,2012,New York Giants 05,21-17,New England Patriots 07,Lucas Oil Stadium 01
XLVII,2013,Baltimore Ravens 02,34-31,San Francisco 49ers 06,Louisiana Superdome 07
XLVIII,2014,Seattle Seahawks 02,43-8,Denver Broncos 07,MetLife Stadium 01
XLIX,2015,New England Patriots 08,28-24,Seattle Seahawks 03,University of Phoenix
Stadium 02
50,2016,Denver Broncos 08,24–10,Carolina Panthers 02,Levi's Stadium 01
```

What to turn in

1. `transform.py`
2. `result.csv`
3. README (please explain your solution and note any issues/bugs with your solution)

Handing in

Please put the required files for the first part (Entity Resolution) in a folder called p1, and the required files for the second part (Reformatting Data) in a folder called p2. For submitting to blackboard, create a folder with your NetID, put p1 and p2 in the folder and zip it into **[your NetID].zip**, not other file extension.

Late submission is not acceptable. The submission link in Blackboard will disappear immediately at the deadline. You could submit multiple times, only the latest version will be graded.

Cheating/Plagiarism Policy:

Any cheating may result in a **fail grade** in the course **and/or be reported to the university higher authorities for appropriate action**.