Introduction to Data Science
Homework 3

50 points

1. Find two tables on the Web and write a SQL query performing Data Integration over the tables that you have found, so that this query yields a better result due to *larger amount of information available from multiple data sources (2 in this case) rather than from a single one.*

For example, consider a person looking for a job in either *New York* or *Boston,* who is interested only in the positions that offer annual compensation higher than $*120'000*.

T1:

City	Salary(thousands U.S. dollars)
'New York City, NY'	120
'Pittsburgh, PA'	125
'Boston, MA'	200
'Austin, TX'	110
'Boulder, CO'	140

T2:

City	Salary(thousands U.S. dollars)
'New York City, NY'	180
'San Francisco, CA'	400
'Seattle, WA'	100
'Charlotte, NC'	90
'Boston, MA'	150

SQL query:

```
SELECT city, salary FROM T1
WHERE (city LIKE "%NEW YORK%" OR city LIKE "%BOSTON%") AND
salary > 120
UNION ALL
SELECT city, salary FROM T2
WHERE (city LIKE "%NEW YORK%" OR city LIKE "%BOSTON%") AND
salary > 120
ORDER BY salary DESCENDING;
```

Result:

City	Salary
'Boston, MA'	200
‘New York City, NY’	180
'Boston, MA'	150

Note that using multiple data sources *increases the number of records* in the output, hence provides a better result set, compared to what could be retrieved from just one source.

Find 5 such pairs of tables and turn in each pair of tables that you have found (the attribute names and 5 sample rows from each table), the SQL query that used for these two tables, the result of running the query on your tables (no more than 5 first rows from the query result), and your explanation of why using two tables yields in an enriched query result.

5 points for each pair = 25 points.

2. Find 2 tables on the Web and write a SQL query performing Data Integration over the tables that you have found, so that this query yields a better result due to *filtering on attributes that are present in only one table and absent in the other one.*

For example, consider a person looking for songs about *valentines*, preferring *blues* music more than other genres and two Web sites, one having a table including the song *lyrics* and another one having a table with the song *genre*. The person joins two tables to be able to filter out songs by *both lyrics* and *genre.*

T1:

Author	Title	Lyrics
'Eminem'	'Lose Yourself'	'His palms are sweaty, knees weak, arms are heavy...'
'Bullet For My Valentine'	'Your Betrayal'	'You were told to run away\nSoak the place, and light the flame\n...'
'Tom Waits'	'Blue Valentines'	'She sends me blue valentines\nAll the way from Philadelphia\n...'
'Bob Dylan'	'Hurricane'	'Pistol shots ring out in the barroom night\nEnter Patty Valentine from the upper hall\n...'
'Katy Perry'	'Teenage Dream'	'Now every February, you'll be my Valentine, Valentine\n...'

T2:

Author	Title	Genre	Length
'Pink Floyd'	'Another Brick in the Wall, Pt.2'	'progressive rock'	'5:58'
'Tom Waits'	'Blue Valentines'	'blues'	'5:50'
'ABBA'	'Mamma Mia'	'pop'	'3:35'
'My Chemical Romance'	'Welcome to the Black Parade'	'punk rock'	'5:11'
'Dropkick Murphys'	'State of Massachusetts'	'celtic folk'	'3:52'

SQL query:

```
SELECT author, title, genre, lyrics
FROM T1, T2
WHERE T1.author = T2.author
AND T1.title = T2.title
AND T1.lyrics LIKE "%VALENTINE%"
AND T2.genre LIKE "%BLUE%";
```

Result:

Author	Title	Genre	Lyrics
'Tom Waits'	'Blue Valentines'	'blues'	'She sends me blue valentines\nAll the way from Philadelphia\n...'

Note that using multiple attributes in a filter condition *increases the precision* of the result.

Find 5 such pairs of tables and turn in each pair that you have found (the attribute names and 5 sample rows from each table), the SQL query that uses these two tables, the result of running the query on your tables (no more than 5 first rows from the query result), and your explanation of why using two tables yields in a better query result.

5 points for each pair = 25 points