COMP3311 21T1	Assignment 2 SQL, Python, SQLite  Last updated: Monday 29th Mar 05:47pm (most recent updates are in [])	Database System
Aims This assignment aims to give you practice in	Due: Friday 16th April 17:00	
<ul> <li>use of SQL in SQLite (i.e., sqlite3)</li> <li>writing scripts in Python that interact with a database</li> <li>populating a RDBMS with a larger dataset, and analysing the data;</li> <li>making test data for testing database applications;</li> </ul>		
Your task is to complete the functionality of some command-line tools via a combination of database code and Python code.  Summary		
Submission: Login to Course Web Site > Assignments > Assignment 2 > Assignment 2 Specification > Make Submission > upload required file Required Files (total 3 files): msearch, toprank, shortest  You may also submit the assignment via the give command from CSE machines: give cs3311 a2 msearch toprank shortest	files > [Submit]	
Deadline: Friday 16 April 2021 @ 17:00  Late Penalty: Late submissions will have marks deducted from the maximum achievable mark at the rate of 2% of the total mark per hour that	at they are late (i.e., 48% per day).	
This assignment contributes 20 marks toward your total mark for this course.  Downloads: a2.tgz, a2.zip  Note that a2.tgz and a2.zip contain the same material.		
Each archive contains the IMDB database dump a2.db, plus a sample Python code file called Sample.  What To Do Now:  read this specification carefully and completely login to a CSE linux machine		
<ul> <li>rogate a directory for this assignment</li> <li>unpack the supplied zip file into this directory</li> <li>get familiar with the schema and data by exploring and querying the provided database using the command: sqlite3 a2.db</li> <li>familiarise (read the code) yourself with the provided sample Python code file called sample</li> <li>try out the sample code by running: ./sample YEAR where YEAR is a number representing a year, e.g., 1989</li> <li>make sure sample is executable (by chmod u+x sample) and run it</li> <li>complete the assignment tasks using sample as a reference template</li> <li>submit all these files (the 'Required Files') via WebCMS3 or give as described above</li> </ul>		
Details of the above steps are further elaborated below. You can edit and run the Python files on any CSE machines.  Introduction		
guarantee a highly rated imdb score. This assignment is based on an IMDB dataset to build several small Python commands to show interesting	rs (such as good directors, experienced actors, etc) are important for creating good movies. Nevertheless, famous directors and actors usually bring an attractive boing results.  nousands of actors/actresses. It also contains the IMDB rating score, numbers of votes and various facebook likes. To give you a feel for the kind of data that you are	
cs3311@wagner:~/sqlite/a2\$ sqlite3 a2.db SQLite version 3.27.2 2019-02-25 16:06:06 Enter ".help" for usage hints. sqlite>		
<pre>sqlite&gt; select * from movie limit 10; 1 Avatar 2009 PG-13 178 English USA 760505847 237000000 735 2 Pirates of the Caribbean: At World's End 2007 PG-13 169 English USA 309404152 300000000 555 3 Spectre 2015 PG-13 148 English UK 200074175 245000000 1776 4 The Dark Knight Rises 2012 PG-13 164 English USA 448130642 250000000 1078</pre>		
6 John Carter 2012 PG-13 132 English USA 73058679 263700000 2217 7 Spider-Man 3 2007 PG-13 156 English USA 336530303 258000000 625 8 Tangled 2010 PG 100 English USA 200807262 260000000 922 9 Avengers: Age of Ultron 2015 PG-13 141 English USA 458991599 250000000 1410 10 Harry Potter and the Half-Blood Prince 2009 PG 153 English UK 301956980 250000000 1117		
11 Batman v Superman: Dawn of Justice 2016 PG-13 183 English USA 330249062 2500000000 2180 sqlite> sqlite> .headers on sqlite> sqlite> select * from movie limit 10;		
id title year content_rating duration lang country gross budget director_id 1 Avatar 2009 PG-13 178 English USA 760505847 237000000 735 2 Pirates of the Caribbean: At World's End 2007 PG-13 169 English USA 309404152 300000000 555 3 Spectre 2015 PG-13 148 English UK 200074175 2450000000 1776 4 The Dark Knight Rises 2012 PG-13 164 English USA 448130642 250000000 1078		
6 John Carter 2012 PG-13 132 English USA 73058679 263700000 2217 7 Spider-Man 3 2007 PG-13 156 English USA 336530303 258000000 625 8 Tangled 2010 PG 100 English USA 200807262 260000000 922 9 Avengers: Age of Ultron 2015 PG-13 141 English USA 458991599 250000000 1410 10 Harry Potter and the Half-Blood Prince 2009 PG 153 English UK 301956980 250000000 1117		
11 Batman v Superman: Dawn of Justice 2016 PG-13 183 English USA 330249062 250000000 2180 sqlite> sqlite> select * from rating limit 10; movie_id num_critic_for_reviews num_user_for_reviews num_voted_users movie_facebook_likes cast_total_f1 723 3054 886204 33000 4834 7.9	facebook_likes imdb_score	
2 302 1238 471220 0 48350 7.1 3 602 994 275868 85000 11700 6.8 4 813 2701 1144337 164000 106759 8.5 6 462 738 212204 24000 1873 6.6 7 392 1902 383056 0 46055 6.2		
8 324 387 294810 29000 2036 7.8 9 635 1117 462669 118000 92000 7.5 10 375 973 321795 10000 58753 7.5 11 673 3018 371639 197000 24450 6.9 sqlite>		
<pre>sqlite&gt; .mode column sqlite&gt; sqlite&gt; select * from director limit 10; id</pre>		
1 JK Youn 2 2 David S. W 42 3 James Fraw 21 4 Kar-Wai Wo 0 5 Brian Tren 53		
6 Perry Lang 17 7 Jeff Burr 155 8 Al Sillima 0 9 Morten Tyl 77 10 Hue Rhodes 0 sqlite>		
sqlite> select * from actor limit 10; id name facebook_likes  1 Maureen McCormick 458 2 Andrew Fiscella 137000		
3 Brittany Daniel 861 4 Michael Smiley 177 5 Najarra Townsend 538 6 Gustaf Skarsgrd 908 7 Laila Haley 1000		
8 Isaac C. Singleto 312 9 Veronica Ferres 30000 10 Chris 'Wonder' Sc 0 sqlite> sqlite> select * from acting limit 10;		
movie_id actor_id 		
2846 195 3421 738 3645 1186 2430 211 4823 1299		
1737 786 2282 866 sqlite> sqlite> select * from genre limit 10; movie_id genre		
407 Adventure 407 Comedy 407 Family 3699 Drama		
3699 Horror 3699 Mystery 3699 Sci-Fi 3699 Thriller 3016 Drama		
2846 Comedy sqlite> sqlite> select * from keyword limit 10; movie_id keyword		
407 dog 407 parole 407 parole 407 parole off 407 prison 407 puppy 3699 alien		
3699 bunker 3699 car crash 3699 kidnapping 3699 minimal ca sqlite>		
sqlite> .width 10 30 sqlite> sqlite> select * from keyword limit 10; movie_id keyword		
407 dog 407 parole 407 parole 407 parole officer 407 prison 407 puppy		
3699 alien 3699 bunker 3699 car crash 3699 kidnapping 3699 minimal cast		
sqlite> You may also explore the schema and any table information using the sqlite system catalog table called sqlite_master, or the sqlite PRAGMA fu cs3311@wagner:~/sqlite/a2\$ sqlite3 a2.db SQLite version 3.27.2 2019-02-25 16:06:06	functions (please see the sqlite documentation for details), as shown in the examples below:	
<pre>Enter ".help" for usage hints. sqlite&gt; .tables acting actor director genre keyword movie rating sqlite&gt; pragma table_info(movie); 0 id integer 1 1</pre>		
1 title character varying(256) 1  0 2 year integer 0  0 3 content_rating character varying(9) 0  0 4 duration integer 0  0 5 lang character varying(10) 0  0		
6 country character varying(20) 0  0 7 gross bigint 0  0 8 budget bigint 0  0 9 director_id integer 0  0 sqlite>		
<pre>sqlite&gt; select * from sqlite_master where name="movie"; table movie 6 CREATE TABLE movie (    id integer NOT NULL,    title character varying(256) NOT NULL,    year integer,</pre>		
<pre>content_rating character varying(9),   duration integer,   lang character varying(10),   country character varying(20),   gross bigint,</pre>		
<pre>budget bigint,   director_id integer,   PRIMARY KEY (id),   FOREIGN KEY (director_id) REFERENCES director(id) )</pre>		
sqlite> The sample Python program		
By following the steps above, you can experience the provided sample program and study its code as follows:  cs3311@wagner:~/sqlite/a2\$ ./sample Usage: ./sample YEAR cs3311@wagner:~/sqlite/a2\$ ./sample 1989		
A Nightmare on Elm Street 5: The Dream Child (1989) Back to the Future Part II (1989) Batman (1989) Bill & Ted's Excellent Adventure (1989) Black Rain (1989)		
Born on the Fourth of July (1989) Dead Poets Society (1989) Do the Right Thing (1989) Driving Miss Daisy (1989) Friday the 13th Part VIII: Jason Takes Manhattan (1989)		

Glory (1989) Halloween 5 (1989) Henry V (1989)

Major League (1989) New York Stories (1989) Pet Sematary (1989) Road House (1989) Roger & Me (1989) Sea of Love (1989)

Tango & Cash (1989) The Abyss (1989)

#!/usr/bin/python3

import sqlite3,sys

year = sys.argv[1]

cur = con.cursor()

t = cur.fetchone()
if t == None:

break

Tasks

wagner\$

wagner\$ More examples:

wagner\$

Task B: Top ranked movies (6 marks)

./toprank Genres MinRating

wagner\$ ./toprank "Action&Sci-Fi&Adventure" Usage: ./toprank Genres MinRating wagner\$ ./toprank "Action&Sci-Fi&Adventure" 8.5

UHF (1989) Warlock (1989)

Indiana Jones and the Last Crusade (1989) Licence to Kill (1989)

Sex, Lies, and Videotape (1989) Star Trek V: The Final Frontier (1989)

The Blood of Heroes (1989) The Toxic Avenger Part II (1989) Troop Beverly Hills (1989)

We're No Angels (1989) When Harry Met Sally... (1989) cs3311@wagner:~/sqlite/a2\$ ls

cs3311@wagner:~/sqlite/a2\$ cs3311@wagner:~/sqlite/a2\$ cat sample

if len(sys.argv) != 2 :
 print("Usage:", sys.argv[0], "YEAR")
 sys.exit(1)

con = sqlite3.connect('a2.db')

x,y = t print('{} ({})'.format(x,y))

We will auto-mark your submission on wagner as follows:

• place your submitted files in that directory and make them executable

manually inspect your submitted Python code (if time allows)

conjunctive selection query (i.e., via AND). For example, consider msearch SUBSTRING-1 SUBSTRING-2 SUBSTRING-3 ...

wagner\$ ./msearch "james fRAnco"

• run a series of tests using your submitted msearch, toprank and shortest scripts

Task A: List movie information by its title, actor or director substrings (6 marks)

by title (ascending). The required output format is elaborated further by the examples below:

count to measure the size of an output to determine its preliminary correctness. For example:

wagner\$ ./msearch "james franco"|wc -1 26 wagner\$ ./msearch "john smith" |wc -l

con.close()
cs3311@wagner:~/sqlite/a2\$

Submission and Testing

· create a testing subdirectory

This is a sample minimal program for COMP3311 21T1 Assignment 2 to illustrate how to write an

executable python code, take in a command line argument, and connect to a sqlite3 db. For simplicity (easier to read and follow), it does not include error checking,

cur.execute('SELECT title, year FROM movie WHERE year ={} ORDER BY title'.format(year))

• place a modified a2.db file (with the same schema but probably slightly modified data) into that directory

unless it is specified explicitly, we assume 'movies' will include any titles stored in the movie table (i.e., including movies and TV shows etc).

wagner\$./msearch "james fRAnco"

1. Sausage Party (2016, R, 7.5) [Adventure, Animation, Comedy, Fantasy]

2. The Little Prince (2015, PG, 7.8) [Adventure, Animation, Drama, Family, Fantasy]

3. The Interview (2014, R, 6.6) [Comedy, Fantasy]

4. This Is the End (2013, R, 6.7) [Action, Crime, Drama, Thriller]

6. Oz the Great and Powerful (2013, PG, 6.4) [Adventure, Family, Fantasy]

7. Palo Alto (2013, R, 6.5) [Dation, Crime, Drama, Thriller]

8. The Iceman (2012, R, 6.9) [Biography, Crime, Drama]

9. Spring Breakers (2012, R, 6.3) [Enime, Drama]

10. Rise of the Planet of the Apes (2011, PG-13, 7.6) [Action, Drama, Sci-Fi, Thriller]

11. Your Highness (2011, R, 5.6) [Adventure, Comedy, Fantasy]

12. 127 Hours (2014, R, 7.6) [Adventure, Biography, Drama, Thriller]

13. Date Night (2019, R, 7.6) [Adventure, Biography, Drama, Thriller]

14. Eat Pray Love (2019, PG-13, 5.7) [Drama, Romance, Thriller]

15. Milk (2008, R, 7.6) [Biography, Drama, History]

16. Pineapple Express (2008, R, 7.0) [Action, Comedy, Crime]

17. In the Valley of Elan (2007, R, 7.2) [Crime, Drama, Mystery, Thriller]

18. Spider-Man 3 (2007, PG-13, 6.2) [Action, Adventure, Romance]

19. The Dead Girl (2006, R, 6.7) [Crime, Drama, History, Romance, War]

21. The Great Raid (2005, R, 6.7) [Action, Drama, History, Romance, War]

22. Spider-Man 2 (2004, PG-13, 7.3) [Action, Adventure, Fantasy, Romance]

23. Spider-Man 2 (2004, PG-13, 7.3) [Action, Adventure, Fantasy, Romance]

24. City by the Sea (2002, PG-13, 7.3) [Action, Adventure, Fantasy, Romance]

25. Deuces Wild (2002, R, 6.2) [Crime, Drama, Mystery, Thriller]

25. Deuces Wild (2002, R, 6.2) [Crime, Drama, Mystery, Thriller]

26. Whatever It Takes (2009, PG-13, 5.5) [Comedy, Drama]

26. Whatever It Takes (2009, PG-13, 5.5) [Comedy, Drama, Romance]

27. Star Wars: Episode II - Attack of the Clones (2002, PG, 6.7) [Action, Adventure, Fantasy, Sci-Fi]

28. Star Wars: Episode II - Attack of the Clones (2009, PG, 6.7) [Action, Adventure, Fantasy, Sci-Fi]

39. Star Wars: Episode II - Attack of the

Wore examples:

wagners - ./msearch "happy"
1. Happy Christmas (2014, R, 5.6) [Comedy, Drama]
2. Another Happy Day (2011, R, 6.0) [Comedy, Drama]
3. Happy Feet 2 (2011, PG, 5.9) [Animation, Comedy, Family, Musical]
4. The Pursuit of Happy Day (2011, R, 6.0) [Eliography, Drama]
5. Happy Feet (2006, PG, 6.5) [Animation, Comedy, Family, Music, Romance]
6. Happy, Texas (1998, PG-13, 6.3) [Comedy, Crime, Romance]
7. Happy Glimore (1998, PG-13, 7.0) [Comedy, Sport]
8. Happy Slimore (1998, PG-13, 7.0) [Comedy, Sport]
9. The Big Short (2015, R, 7.8) [Biography, Drama, Sport]
1. The Secret Life of Pets (2016, PG, 6.8) [Animation, Comedy, Family]
2. The Big Short (2015, R, 7.8) [Biography, Drama, Sport]
4. A Most Violent Year (2014, R, 7.0) [Action, Crime, Drama, History]
3. Concussion (2015, PG-13, 7.1) [Biography, Drama, Sport]
4. A Most Violent Year (2014, R, 7.0) [Action, Crime, Drama, Thriller]
5. Baggage Claim (2013, PG-13, 7.8) [Action, Adventure, Thriller]
7. Drive (2011, R, 7.8) [Crime, Drama]
8. Albert Nobbs (2011, R, 6.7) [Drama]
9. The Book of Eli (2018, R, 6.9) [Action, Adventure, Drama, Thriller]
10. The Sompsons Movie (2007, PG-13, 7.1) [Adventure, Animation, Comedy]
11. The Simpsons Movie (2007, PG-13, 7.1) [Adventure, Animation, Comedy]
12. Once in a Lifetime: The Extraordianry Story of the New York Cosmos (2006, PG-13, 7.3) [Documentary, Sport]
14. A Good Year (2006, PG-13, 6.9) [Comedy, Drama, Romance]
16. A Very Long Engagement (2004, R, 7.7) [Drama, Mystery, Romance, Warl
17. Fat Albert (2004, PG, 4.3) [Drama, Romance]
18. Big Fish (2003, PG-13, 8.0) [Oredy, Family, Fantasy, Romance]
19. From Hell (2001, R, 6.8) [Horror, Mystery, Thriller]
20. From Hell (2001, R, 6.8) [Horror, Mystery, Thriller]
21. Frie Brockovich (2008, R, 7.3) [Erime, Drama, Romance]
22. Hencersible (2008, R, 7.3) [Crime, Drama, Romance]
23. The Tother's McWillent (1998, R, 6.6) [Comedy, Mar]
24. Breakfast of Champions (1999, R, 7.5) [Crime, Drama, Romance]
25. Menace II Society (1998, R, 7.6) [Crime, Drama, Romance]
26. Menace II

36. Tom Jones (1963, Unrated, 6.8) [Adventure, Comedy, History]
37. The Longest Day (1962, G, 7.8) [Action, Drama, History, War]
wagner\$ ./msearch "albert" "david"
1. A Most Violent Year (2014, R, 7.0) [Action, Crime, Drama, Thriller]
2. Baggage Claim (2013, PG-13, 5.0) [Comedy]
3. The Simpsons Movie (2007, PG-13, 7.4) [Adventure, Animation, Comedy]
wagner\$ ./msearch "albert" "david" "simpson"
1. The Simpsons Movie (2007, PG-13, 7.4) [Adventure, Animation, Comedy]
wagner\$ ./msearch "tom hanks" "spielberg" "Ryan"
1. Bridge of Spies (2015, PG-13, 7.6) [Drama, History, Thriller]
2. Saving Private Ryan (1998, R, 8.6) [Action, Drama, War]
wagner\$ ./msearch "star war" "Natalie PORTMAN" "Lucas"
1. Star Wars: Episode III - Revenge of the Sith (2005, PG-13, 7.6) [Action, Adventure, Fantasy, Sci-Fi]
2. Star Wars: Episode I - The Phantom Menace (1999, PG, 6.5) [Action, Adventure, Fantasy, Sci-Fi]
wagner\$ ./msearch "star war" "Natalie PORTMAN" "Lucas" "revenge"
1. Star Wars: Episode III - Revenge of the Sith (2005, PG-13, 7.6) [Action, Adventure, Fantasy, Sci-Fi]
wagner\$ ./msearch "star war" "Natalie PORTMAN" "Lucas" "revenge"
1. Star Wars: Episode III - Revenge of the Sith (2005, PG-13, 7.6) [Action, Adventure, Fantasy, Sci-Fi]
wagner\$ ./msearch "star war" "Natalie PORTMAN" "Lucas" "revenge"
1. Star Wars: Episode III - Revenge of the Sith (2005, PG-13, 7.6) [Action, Adventure, Fantasy, Sci-Fi]
wagner\$

an empty string, perform the same ranking but on movies with any genres. The required output format is elaborated further by the examples below:

18. Daredevil (TV-MA, English) [8.8, 213483] 19. It's Always Sunny in Philadelphia (TV-MA, English) [8.8, 133415] wagner\$

Task C: Six degrees of Kevin Bacon (8 marks)

The toprank script ranks movies based on their IMDB rating. It considers movies with given genres and the minimum acceptable IMDB score to be listed in the output. It takes in 2 commandline arguments:

A movie will be in its output if each of SUBSTRING-1, SUBSTRING-2, SUBSTRING-3 ... appears in its title, actor or director.

Time Requirement: in order to avoid indefinitely running scripts that block the marking process, any scripts that run for more than 2 minutes for a particular task on wagner will be terminated and considered failed for that test.

Before you submit, you should test out whether the files you submit will work by following a similar sequence of steps to those noted above on wagner (wagner.cse.unsw.edu.au).

does not pass the auto-marking tests due to extra debugging messages, etc), you will be penalised by an additional 30% administrative penalty. A testing script for formatting sanity check (based on the examples below) will be available in Week 8.

Your submitted code must be complete so that when we do the above, your Python will work just as it did in your assignment directory and with a database with the identical schema (with either the same or slightly modified dataset) to yours (a2.db). For submission simplicity, please only submit the 3 required Python files. In other words, do NOT break a Python program into multiple Python files (so each submitted file should be a self-contained Python program). If your code does not work when installed for testing on wagner, as described above (for example, it works on your home machine perfectly, but we need to edit one line in order to have it running properly on a CSE machine; OR it

For each task, you are required to implement an executable Python program command that takes in commandline arguments and displays the result in a specified format (which we will use for auto-marking). We assume the sample output format for each task below to test your programs. If any field of your output is empty, you should hide that field

The msearch (movie search) script lists the movie title, year, content rating, IMDB score and genres of those movies with the title, actor, or director matching the given substring (case insensitive), one per line. It accepts any number of given substrings (but at least one). If more than one substring are provided, they should be considered as a

The output of multiple genres of a movie should be concatenated in one line delimited by a comma and sorted alphabetically in ascending order, rows with empty year at the end), then by IMDB rating (descending, rows with empty rating at the end if any) and finally

Please note the output formatting of the last movie above (No. 7), i.e., it does not have year. For all the tasks in this assignment, as mentioned previously, your program shall output without any extra lines or messages. Therefore, we may also use the line

where Genres is a list of genres (based on case insensitive matching) separated by '&', MinRating is the minimum acceptable IMDB score and then by the number of votes (both in descending order). We interpret '&' as conjunction, i.e., the selected movies shall contain all the specified genres. When Genres is

This task is inspired by the Six degrees of Kevin Bacon. The shortest path (up to Six Degrees of Separation) between two given actors are not connected within six degrees, you can assume that they are not connected at all (i.e.

2. Brad Pitt was in Ocean's Thirteen (2007) with Matt Damon; Matt Damon was in The Legend of Bagger Vance (2000) with Will Smith
3. Brad Pitt was in True Romance (1993) with Michael Rapaport; Michael Rapaport was in Hitch (2005) with Will Smith
wagners'./shortest "chris evans" "bill clinton"

1. Chris Evans was in Captain America: Civil War (2016) with Robert Downey Jr.; Robert Downey Jr. was in Kiss Kiss Bang Bang (2005) with Larry Miller; Larry Miller was in What's the Worst That Could Happen? (2001) with Richard Schiff; Richard Schiff was in
3. Chris Evans was in Captain America: Civil War (2016) with Robert Downey Jr.; Robert Downey Jr. was in Lucky You (2007) with Robert Duvall; Robert Duvall was in The 6th Day (2000) with Michael Rapaport was in Lucky Numbers (2000) with Michael Rapaport; Michael Rapaport

Chris Evans was in The Iceman (2012) with James Franco; James Franco was in City by the Sea (2002) with Robert De Niro; Robert De Niro was in Men of Honor (2000) with Michael Rapaport; Michael Rapaport was in Lucky Numbers (2000) with Michael Moore; Mi 46. Chris Evans was in The Iceman (2012) with James Franco; James Franco was in City by the Sea (2002) with Robert De Niro; Robert De Niro was in Men of Honor (2000) with Michael Rapaport; Michael Rapaport was in Lucky Numbers (2000) with Michael Moore; Mi 47. Chris Evans was in The Iceman (2012) with James Franco; James Franco was in In the Valley of Elah (2007) with Charlize Theron was in Men of Honor (2000) with Michael Rapaport; Michael Rapaport was in Lucky Numbers (2000) with Michael Michael Moore; Michael Moore was in The Losers (2010) with Jason Patric; Jason Patric; Jason Patric was in The Lost Boys (1987) with Dianne Wiest; Dianne Wiest was in I Am Sam (2001) with Richard Schiff; Richar

output nothing). The output will be a list of actors, the movies and the years, as illustrated in the examples below. If there are more than one 'shortest paths', output all of them (all output lines are sorted ASCII-betically in ascending order). Reference (Wikipedia): Six Degrees of Kevin Ba

wagner\$ ./shortest "tom cruise" "Jeremy Renner"

1. Tom Cruise was in Mission: Impossible - Ghost Protocol (2011) with Jeremy Renner

2. Tom Cruise was in Mission: Impossible - Rogue Nation (2015) with Jeremy Renner

3. Tom Cruise was in Mission: Impossible - Rogue Nation (2015) with Jeremy Renner

4. Shortest "chris evans" "Scarlett Johansson"

5. Chris Evans was in Captain America: Civil War (2016) with Scarlett Johansson

6. Chris Evans was in Captain America: The Winter Soldier (2014) with Scarlett Johansson

7. Abortest "tom cruise" "Robert Downey Jr."

7. Tom Cruise was in Days of Thunder (1990) with Robert Duvall; Robert Duvall was in Lucky You (2007) with Robert Downey Jr.

8. Tom Cruise was in Jack Reacher (2012) with Robert Duvall; Robert Duvall was in The Judge (2014) with Robert Downey Jr.

8. Tom Cruise was in Jack Reacher (2012) with Robert Duvall; Robert Duvall was in The Judge (2014) with Robert Downey Jr.

9. Tom Cruise was in Mission: Impossible (1996) with Kristin Scott Thomas; Kristin Scott Thomas was in Richard III (1995) with Robert Downey Jr.

9. Tom Cruise was in Mission: Impossible (1996) with Eddie Marsan; Eddie Marsan was in Sherlock Holmes (2009) with Robert Downey Jr.

9. Tom Cruise was in Mission: Impossible III (2006) with Eddie Marsan; Eddie Marsan was in Sherlock Holmes: A Game of Shadows (2011) with Robert Downey Jr.

9. Tom Cruise was in The Firm (1993) with Holly Hunter; Holly Hunter was in Home for the Hollidays (1995) with Robert Downey Jr.

10. Brad Pitt was in Burn After Reading (2008) with Matt Damon; Matt Damon was in The Legend of Bagger Vance (2000) with Will Smith

2. Brad Pitt was in True Romance (1993) with Michael Rapaport; Michael Rapaport was in Hitch (2005) with Will Smith

For each task below, output nothing if there is no result returned from the database. If the task does not specify particular output order, your output, your output should follow the same column order as the sample output presented in each task below.