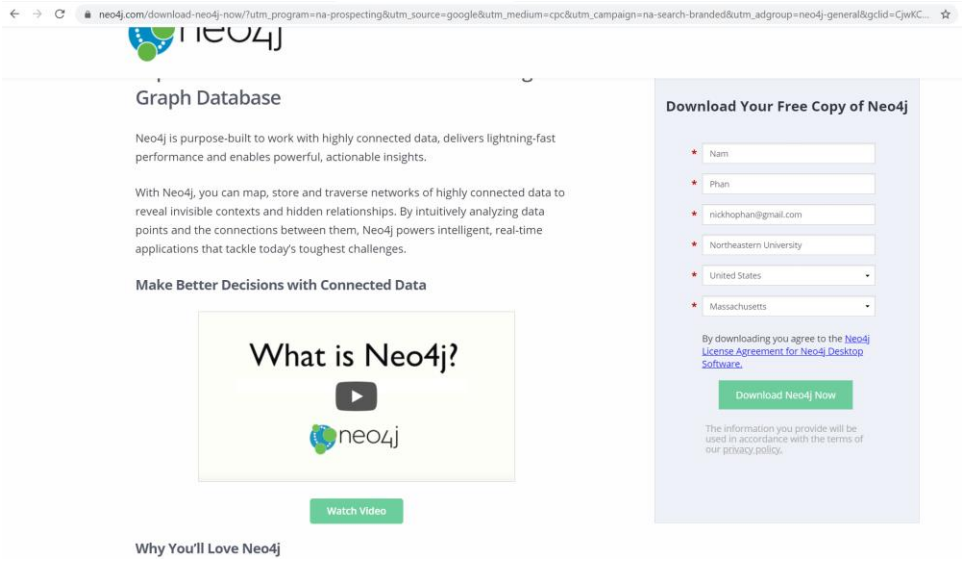
Nam Ho Phan

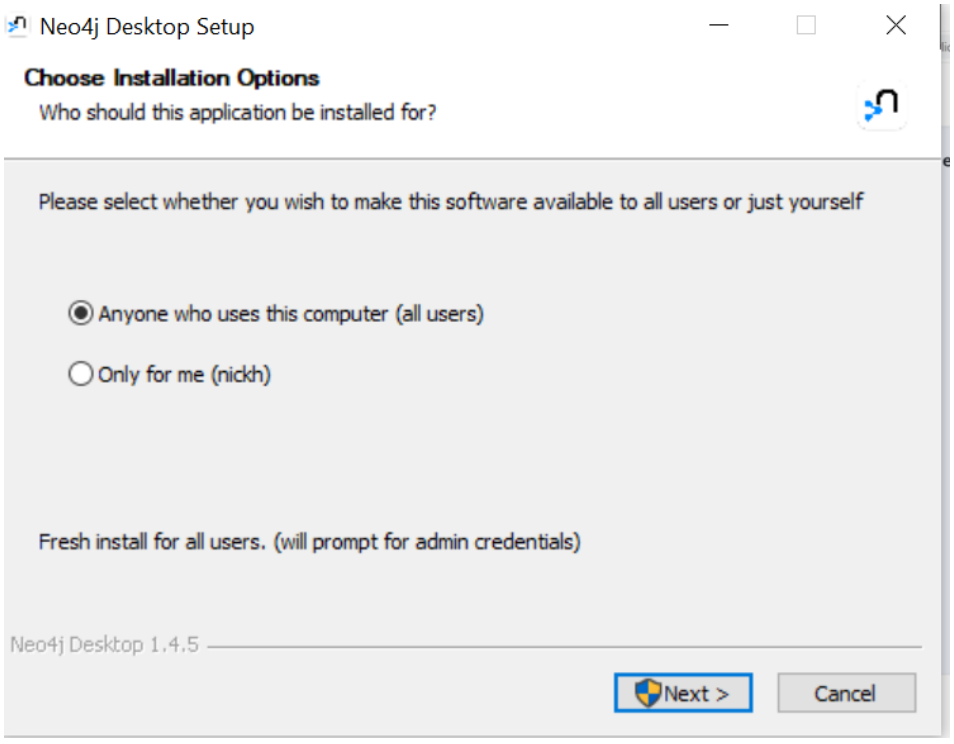
Northeastern University

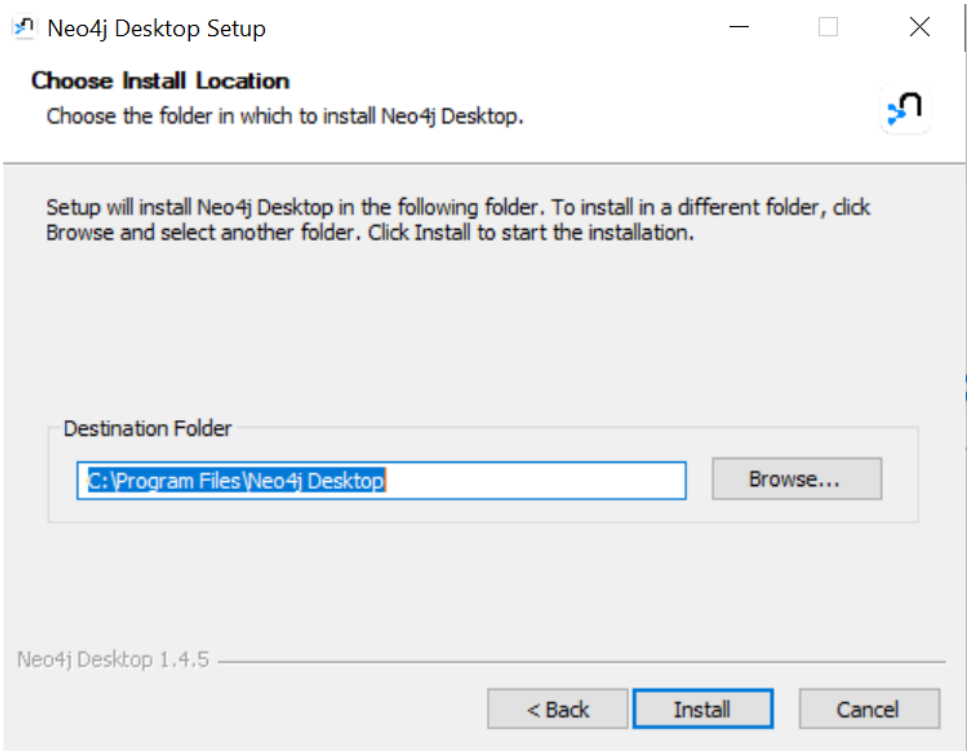
How to use neo4j to visualize real-time data

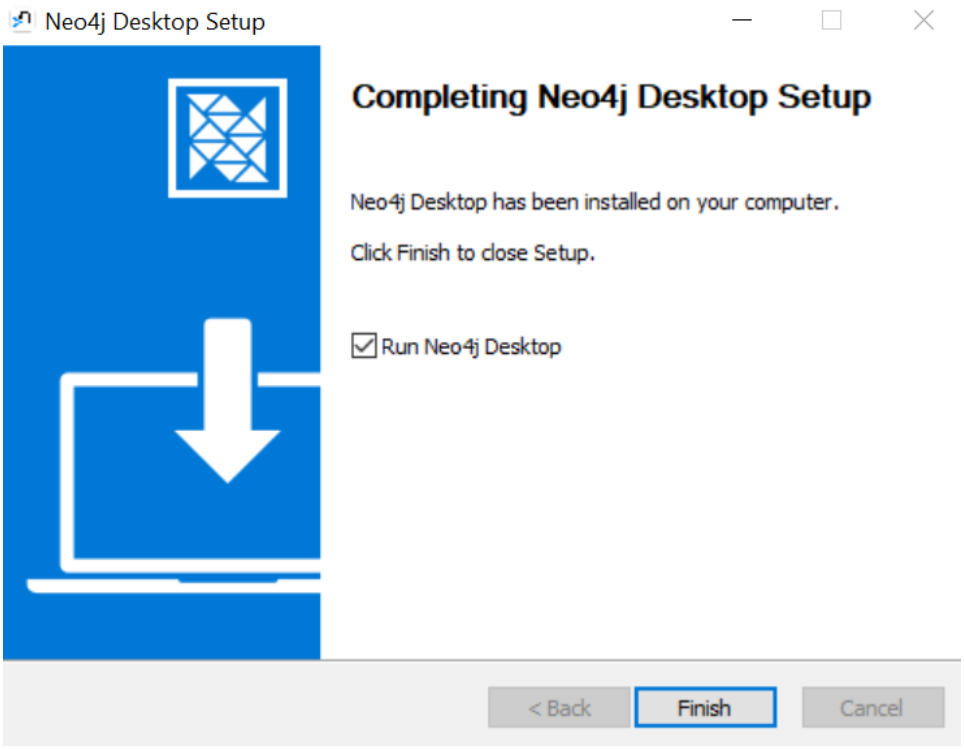
Step 1

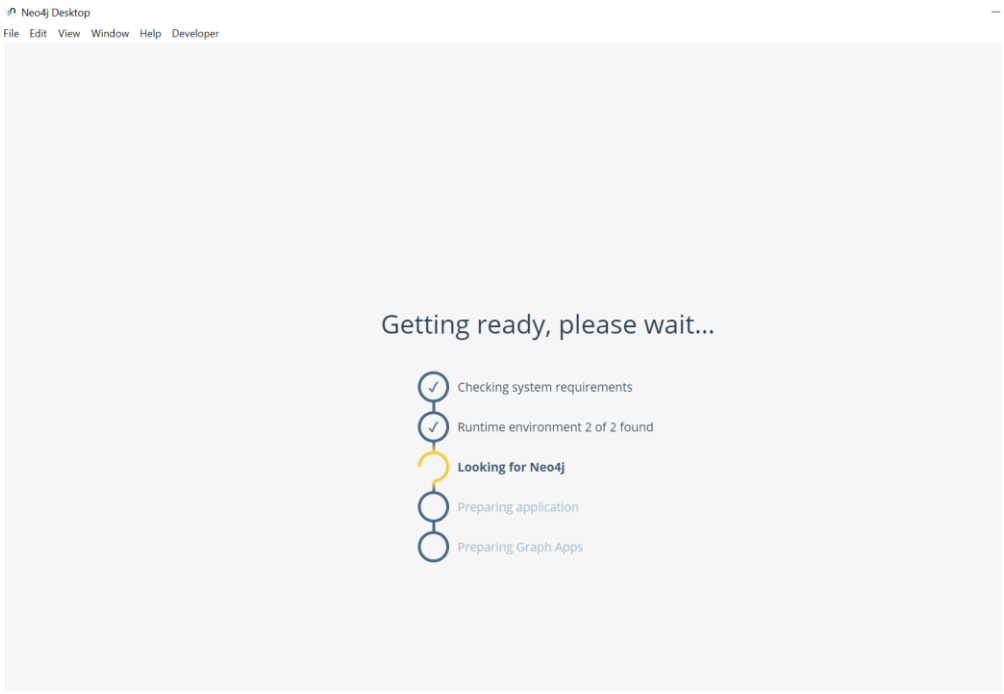
To install neo4j, I must go to its website and register by my email. After downloading the install file into computer successfully, I run that file to set up it in my desktop.






Neo4j Desktop Setup
Choose Install Location
Choose the folder in which to install Neo4j Desktop.
Setup will install Neo4j Desktop in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.
Destination Folder
C:\Program Files\Neo4j Desktop
Browse...
Neo4j Desktop 1.4.5
< Back
Install
Cancel


Neo4j Desktop Setup
Completing Neo4j Desktop Setup
Neo4j Desktop has been installed on your computer.
Click Finish to close Setup.
Run Neo4j Desktop
< Back
Finish
Cancel



Step 2:

I use CREATE function to load information into database

```
11 CREATE (keanu)-[:ACTS_IN { role : 'Neo' }]→(matrix3)
12 CREATE (laurence)-[:ACTS_IN { role : 'Morpheus' }]→
   (matrix1)
13 CREATE (laurence)-[:ACTS_IN { role : 'Morpheus' }]→
   (matrix2)
14 CREATE (laurence)-[:ACTS_IN { role : 'Morpheus' }]→
   (matrix3)
15 CREATE (carrieanne)-[:ACTS_IN { role : 'Trinity' }]→
   (matrix1)
16 CREATE (carrieanne)-[:ACTS_IN { role : 'Trinity' }]→
   (matrix2)
17 CREATE (carrieanne)-[:ACTS_IN { role : 'Trinity' }]→
```

Table
Code

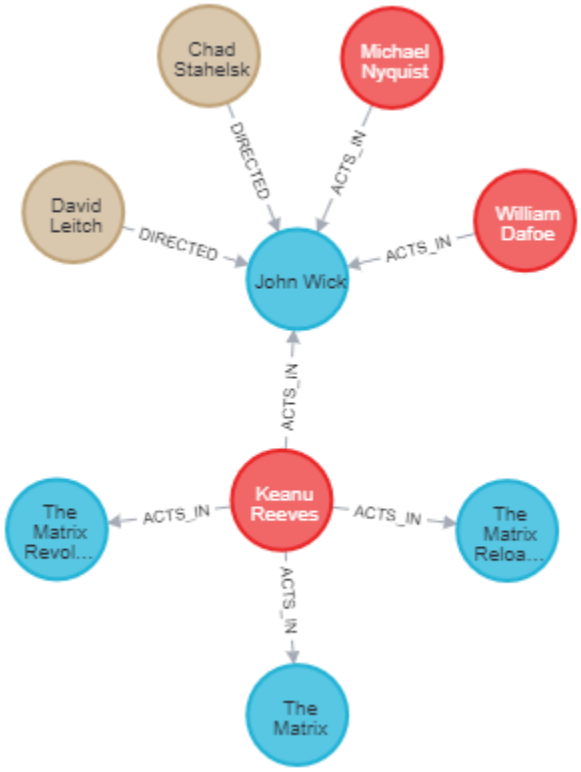
Server version	Neo4j/4.2.1
Server address	localhost:7687
Query	CREATE (matrix1:Movie { title : 'The Matrix', year : '1999-03-31' }) CREATE (matrix2:Movie { title : 'The Matrix Reloaded', year : '2003-05-07' }) CREATE (matrix3:Movie { title : 'The Matrix Revolutions', year : '2003-10-27' }) CREATE (keanu:Actor { name:'Keanu Reeves' }) CREATE (laurence:Actor { name:'Laurence Fishburne' }) CREATE (carrieanne:Actor { name:'Carrie-Anne Moss' }) CREATE (keanu)-[:ACTS_IN { role : 'Neo' }]->(matrix1) CREATE (keanu)-[:ACTS_IN { role : 'Neo' }]->(matrix2) CREATE (keanu)-[:ACTS_IN { role : 'Neo' }]->(matrix3) CREATE (laurence)-[:ACTS_IN { role : 'Morpheus' }]->(matrix1) CREATE (laurence)-[:ACTS_IN { role : 'Morpheus' }]->(matrix2) CREATE (laurence)-[:ACTS_IN { role : 'Morpheus' }]->(matrix3) CREATE (carrieanne)-[:ACTS_IN { role : 'Trinity' }]->(matrix1) CREATE (carrieanne)-[:ACTS_IN { role : 'Trinity' }]->(matrix2) CREATE (carrieanne)-

Step 3:

I also use CREATE to set up information with the title of movie, relevant actors, and directors.

```
[:ACTS_IN { role : 'Trinity' }]->(matrix3) CREATE (JohnWick:Movie { title : 'John Wick'}) CREATE
(William:Actor { name:'William Dafoe' }) CREATE (Micheal:Actor { name:'Michael Nyquist' }) CREATE
(Chad:Director { name:'Chad Stahelsk' }) CREATE (David:Director { name:'David Leitch' }) CREATE
(keanu)-[:ACTS_IN] -> (JohnWick) CREATE (William)-[:ACTS_IN] -> (JohnWick) CREATE (Micheal)-
[:ACTS_IN] -> (JohnWick) CREATE (Chad) - [:DIRECTED] -> (JohnWick) CREATE (David) -
[:DIRECTED] -> (JohnWick)
```

To visualize the node John Wick, I use MATCH function and it will give back to me the nodes and relationship related to John Wick.



Step 4:

To find co actors, I use MATCH function with name as ‘Keanu Reeves’, and use [:ACTS_IN] variable to set the relation.

```
neo4j$ MATCH (keanu:Actor {name: 'Keanu Reeves'}) -
       [:ACTS_IN]→(m)←[:ACTS_IN]-(coActors) RETURN
       coActors.name
```

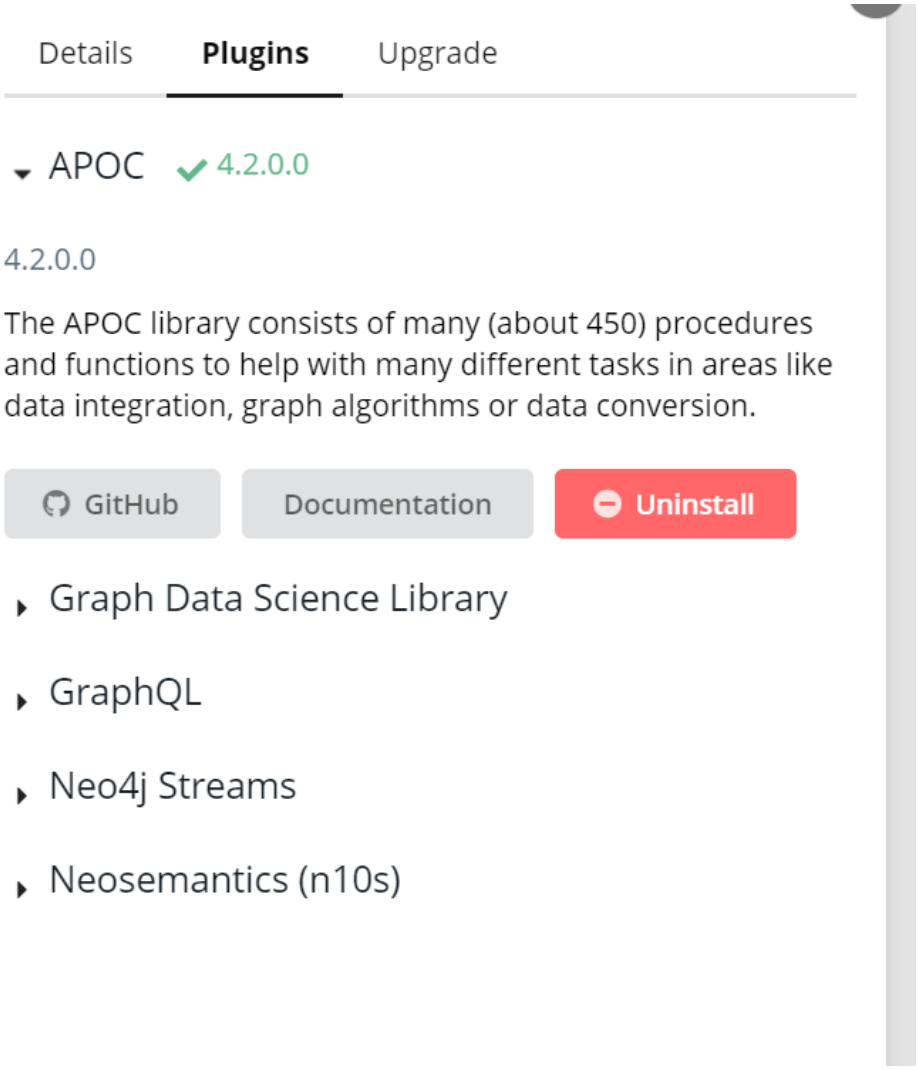
	coActors.name
1	"Laurence Fishburne"
2	"Carrie-Anne Moss"
3	"Laurence Fishburne"
4	"Carrie-Anne Moss"
5	"Carrie-Anne Moss"
6	"Laurence Fishburne"

```
neo4j$ MATCH (keanu:Actor {name: 'Keanu Reeves'}) -
       [:ACTS_IN]→(m)← [:DIRECTED]- (directors) RETURN
       directors.name
```

	directors.name
1	"Chad Stahelsk"
2	"David Leitch"

Problem 5:

To be able to export and import the csv file, I must install APOC in plugins.

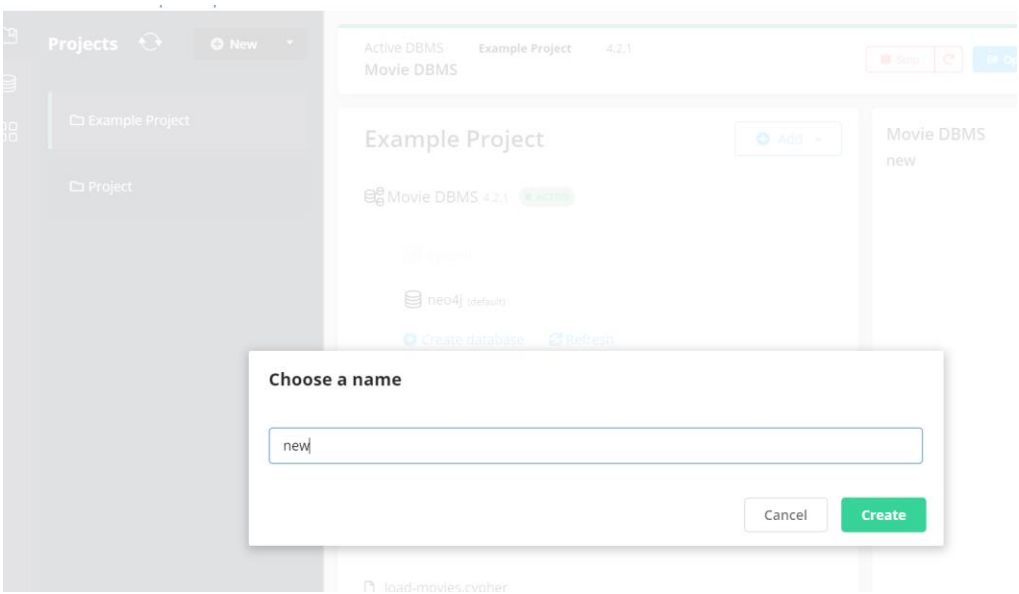


To export the database, I will use CALL, YIELD, RETURN to save file into folder.

```
neo4j$ CALL apoc.export.csv.all("movies.csv", {})
```

```
1 MATCH (keanu:Person {name:"Keenu Reeves"})-
  [:ACTED_IN]→(m)←[:ACTED_IN]-(coActors)
2 WITH collect(coActors) AS coActors
3 CALL apoc.export.csv.data(coActors, [],
  "coActors.csv", {})
4 YIELD file, source, format, nodes, relationships,
  properties, time, rows, batchSize, batches, done,
  data
5 RETURN file, source, format, nodes, relationships,
  properties, time, rows, batchSize, batches, done,
  data
```

After that, I create new project as ‘new’ to open the database, then I use LOAD CSV from ‘file:///file.csv’ AS row to extract file from folder.




```
LOAD CSV FROM 'file:///directors.csv' AS row
WITH toInteger(row[0]) AS ID, row[1] AS label,
row[2] AS name
RETURN ID, label, name
```

"ID"	"label"	"name"
null	"_labels"	"name"
65	":Director"	"David Leitch"
64	":Director"	"Chad Stahelsk"

```
LOAD CSV FROM 'file:///coActors.csv' AS row
WITH toInteger(row[0]) AS ID, row[1] AS label,
row[2] AS name
RETURN ID, label, name
```

"ID"	"label"	"name"
null	"_labels"	"name"
59	":Actor"	"Laurence Fishburne"
60	":Actor"	"Carrie-Anne Moss"
59	":Actor"	"Laurence Fishburne"
60	":Actor"	"Carrie-Anne Moss"
59	":Actor"	"Laurence Fishburne"
60	":Actor"	"Carrie-Anne Moss"
63	":Actor"	"Michael Nyquist"
62	":Actor"	"William Dafoe"

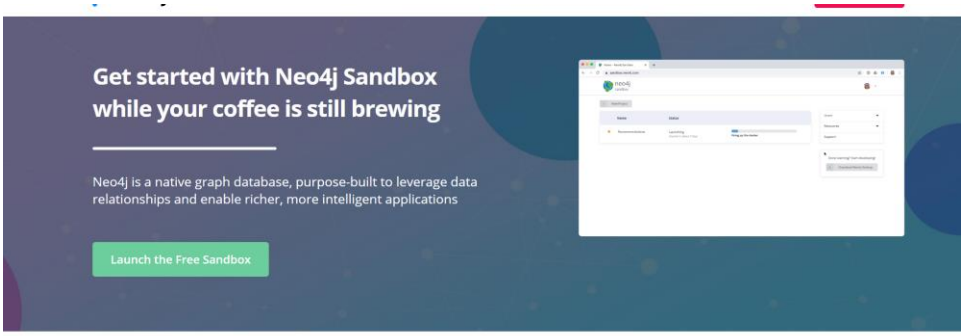
```
1 LOAD CSV FROM 'file:///movies.csv' AS row
2 WITH toInteger(row[0]) AS ID, row[1] AS label,
  row[2] AS name,row[3] AS title,row[4] AS year,
  row[7] AS type, row[8] AS role
3 RETURN ID, label, name,title,year,type,role
```

Table | Text | Code

58	":Actor"	"Keanu Reeves"	""	""	null	null
59	":Actor"	"Laurence Fishburne"	""	""	null	null
60	":Actor"	"Carrie-Anne Moss"	""	""	null	null
61	":Movie"	""	"John Wick"	""	null	null
62	":Actor"	"William Dafoe"	""	""	null	null
63	":Actor"	"Michael Nyquist"	""	""	null	null
64	":Director"	"Chad Stahelsk"	""	""	null	null
65	":Director"	"David Leitch"	""	""	null	null
null	null	null	null	null	"ACTS_IN"	"Neo"
null	null	null	null	null	"ACTS_IN"	"Neo"
null	null	null	null	null	"ACTS_IN"	"Neo"
null	null	null	null	null	"ACTS_IN"	"Morpheu

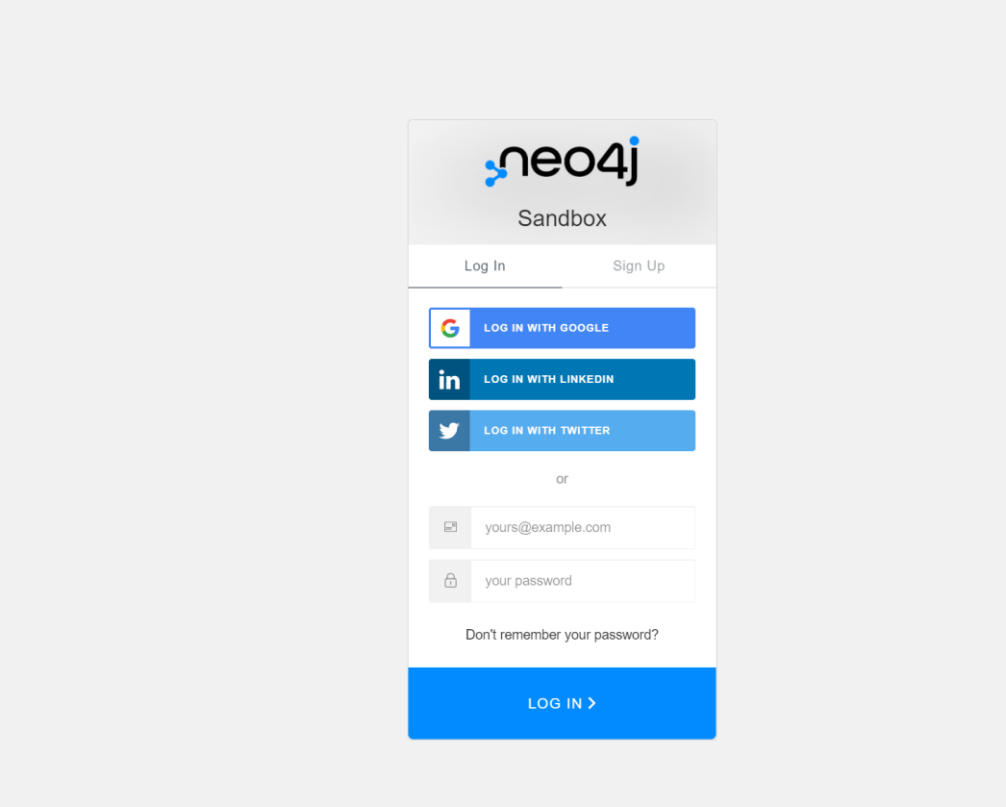
Step 6:

I click on the link of Sandbox and click on the Launch icon below. After that, I use my email to register to use the application.

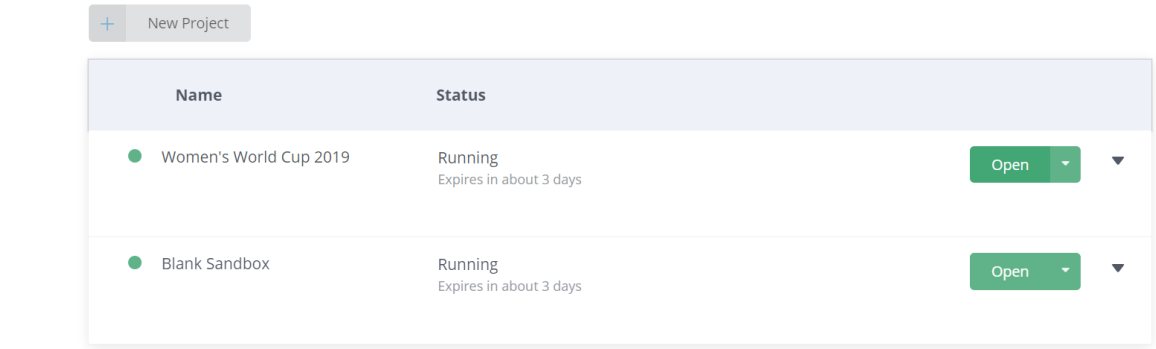


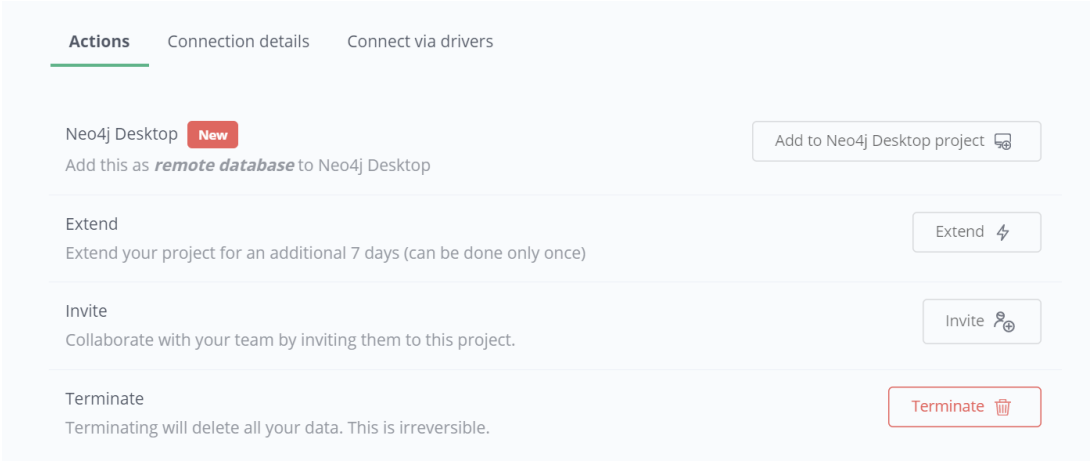
Experience Neo4j in a click with the Sandbox

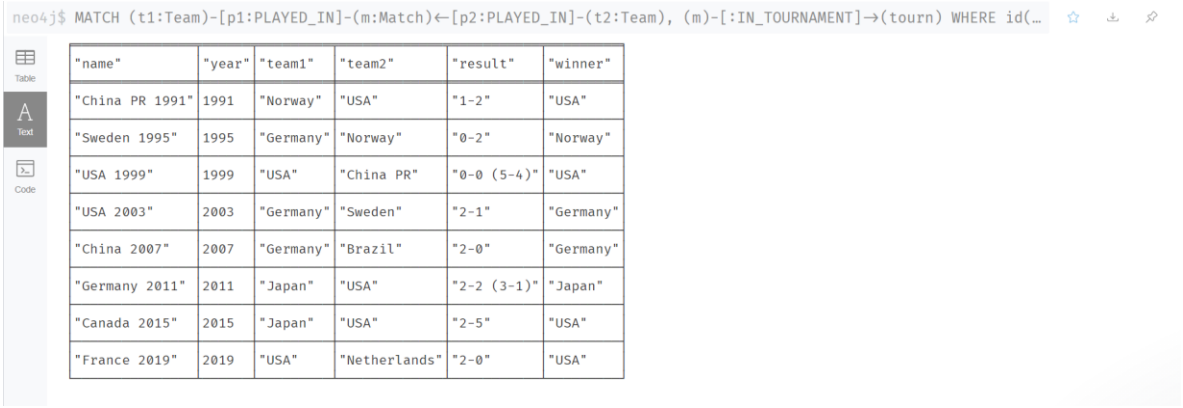
Pick a project and get started in less than 60 seconds. ***No download required.***



Below is the projects I try creating and run some codes to see the examples.







"name"	"year"	"team1"	"team2"	"result"	"winner"
"China PR 1991"	1991	"Norway"	"USA"	"1-2"	"USA"
"Sweden 1995"	1995	"Germany"	"Norway"	"0-2"	"Norway"
"USA 1999"	1999	"USA"	"China PR"	"0-0 (5-4)"	"USA"
"USA 2003"	2003	"Germany"	"Sweden"	"2-1"	"Germany"
"China 2007"	2007	"Germany"	"Brazil"	"2-0"	"Germany"
"Germany 2011"	2011	"Japan"	"USA"	"2-2 (3-1)"	"Japan"
"Canada 2015"	2015	"Japan"	"USA"	"2-5"	"USA"
"France 2019"	2019	"USA"	"Netherlands"	"2-0"	"USA"

```
neo4j$ MATCH (t:Tournament)←[:PARTICIPATED_IN]-(team) RETURN t.name, t.year, count(*) ORDER BY t.year
```

Table | Text | Code

"t.name"	"t.year"	"count(*)"
"China PR 1991"	1991	12
"Sweden 1995"	1995	12
"USA 1999"	1999	16
"USA 2003"	2003	16
"China 2007"	2007	16
"Germany 2011"	2011	16
"Canada 2015"	2015	24

```
neo4j$ MATCH path = (t:Tournament {year: 2019})←[:PARTICIPATED_IN]-(team) RETURN path
```

Graph | Table | Text | Code

*(25) Tournament(1) Team(24)

*(24) PARTICIPATED_IN(24)