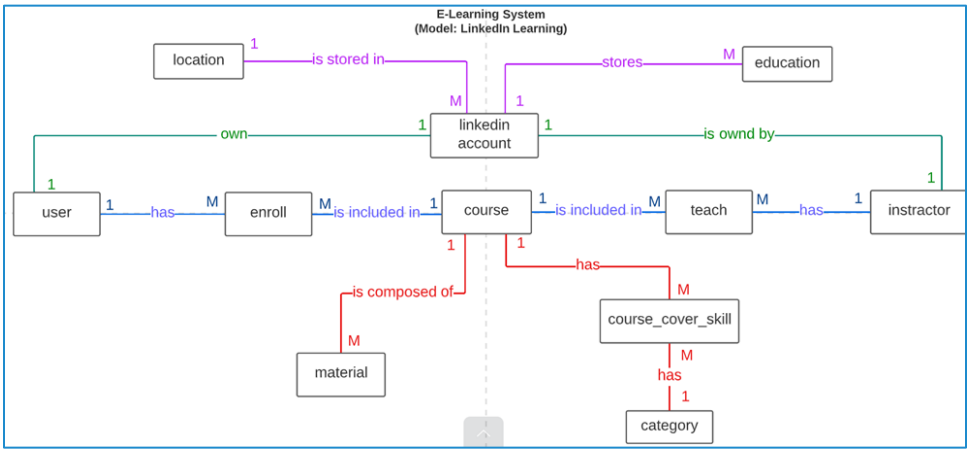
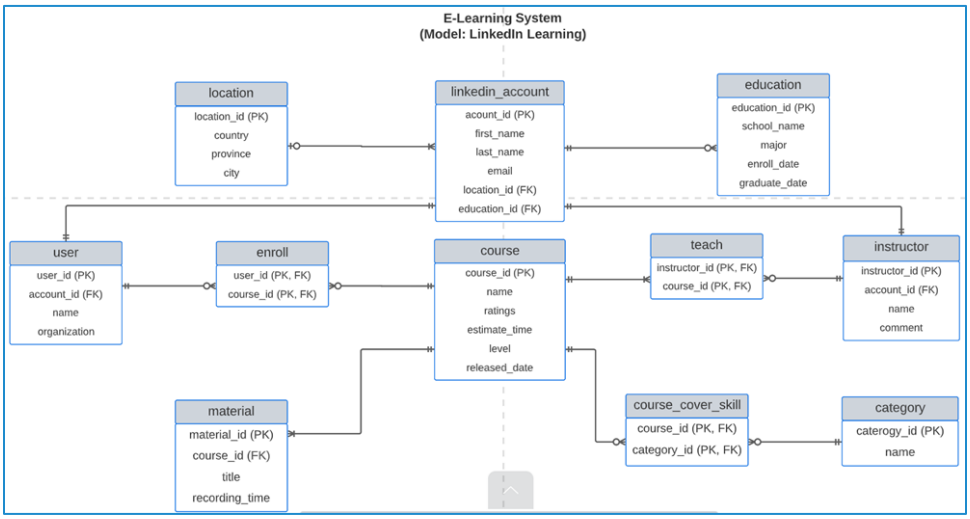
Project02

Group 01: Ran Arino - Solmaz Heidar Nassab - Anna Jazayeri

ER Model from project 01



Schema Design (MongoDB) from project 01



1. Implementation on Neo4j

```
CREATE (cou0:Course {name: 'Introduction of Data Science', ratings: 4.5,
estimate_time: 30, level: 'Beginner', released_date: '2020-10-01'}), (cou1:Course
{name: 'Structured Database Design', ratings: 4.75, estimate_time: 100, level:
'Beginner', released_date: '2018-01-19'}), (cou2:Course {name: 'Advanced Database
Design', ratings: 4.0, estimate_time: 100, level: 'Intermidiate', released_date:
'2018-02-23'}), (cou3:Course {name: 'Advanced Machine Learning', ratings: 4.8,
estimate_time: 120, level: 'Advanced', released_date: '2019-12-02'}),
(cou4:Course {name: 'Introduction of Python', ratings: 4.5, estimate_time: 90,
level: 'Beginner', released_date: '2017-05-16'}), (cou5:Course {name: 'Financial
Economics', ratings: 4.0, estimate_time: 120, level: 'Advanced', released_date:
```

'2019-08-27'}), (cou6:Course {name: 'Excel Skills for Business', ratings: 3.5, estimate_time: 45, level: 'Beginner', released_date: '2020-09-30'}), (cou7:Course {name: 'Mathematics for Data Science', ratings: 4.0, estimate_time: 90, level: 'Intermidiate', released_date: '2019-07-11'}), (cou8:Course {name: 'Python for Stock Market Analysis', ratings: 4.0, estimate_time: 90, level: 'Intermidiate', released_date: '2020-12-12'}), (cou9:Course {name: 'Advanced Deep Learning', ratings: 3.6, estimate_time: 60, level: 'Advanced', released_date: '2019-06-27'}), (mat0:Material {title: 'What is Data Science and Analysis?', recording_time: 10}), (mat1:Material {title: 'Fundamental Statistical Methods', recording_time: 10}), (mat2:Material {title: 'How to Apply Data Science Strategies to the Real World.', recording_time: 10}), (mat3:Material {title: 'What is Physical Relational Database?', recording_time: 20}), (mat4:Material {title: 'What is Logical Relational Database?', recording_time: 20}), (mat5:Material {title: 'How to Write Entity Relationship Diagram', recording_time: 20}), (mat6:Material {title: 'Function of Database Management System', recording_time: 20}), (mat7:Material {title: 'Introduction of SQL', recording_time: 20}), (mat8:Material {title: 'What is NoSQL?', recording_time: 20}), (mat9:Material {title: 'Explanation of Sharding and Replication', recording_time: 20}), (mat10:Material {title: 'What is Document Database?', recording_time: 30}), (mat11:Material {title: 'Implementation of MongoDB', recording_time: 30}), (mat12:Material {title: 'Multiple Linear Model', recording_time: 20}), (mat13:Material {title: 'Probability: Bayesian theory', recording_time: 20}), (mat14:Material {title: 'Application of Classification', recording_time: 20}), (mat15:Material {title: 'Data Mining & Dimensionality Reduction', recording_time: 20}), (mat16:Material {title: 'Sampling Methods', recording_time: 20}), (mat17:Material {title: 'Introduction of Deep Learning', recording_time: 20}), (mat18:Material {title: 'Variables and Functions', recording_time: 10}), (mat19:Material {title: 'Fundamental Math Operations', recording_time: 10}), (mat20:Material {title: 'If statement and For Loop', recording_time: 15}), (mat21:Material {title: 'Four Collection Data Types', recording_time: 20}), (mat22:Material {title: 'Numpy and Pandas', recording_time: 20}), (mat23:Material {title: 'Data Visualization: Matplotlib', recording_time: 15}), (mat24:Material {title: 'Basic Concepts of Microeconomics', recording_time: 30}), (mat25:Material {title: 'Basic Concepts of Macroeconomics', recording_time: 30}), (mat26:Material {title: 'The development of Stock Market', recording_time: 30}), (mat27:Material {title: 'Risk Management', recording_time: 30}), (mat28:Material {title: 'Excel Skills: Introduction', recording_time: 10}), (mat29:Material {title: 'Excel Skills: Intermidiate (1)', recording_time: 10}), (mat30:Material {title: 'Excel Skills: Intermidiate (2)', recording_time: 10}), (mat31:Material {title: 'Excel Skills: Advanced', recording_time: 15}), (mat32:Material {title: 'Limits, Derivatives, Intergrals', recording_time: 30}), (mat33:Material {title: 'Linear Systems', recording_time: 15}), (mat34:Material {title: 'Vector & Matrices', recording_time: 15}), (mat35:Material {title: 'Eigenvalues & Eigenvectors', recording_time: 15}), (mat36:Material {title: 'Regressions', recording_time: 15}), (mat37:Material {title: 'Visualizing Stock Data in Python', recording_time: 30}), (mat38:Material {title: 'Linear Regression Model for Stock Analysis', recording_time: 30}), (mat39:Material {title: 'Specifying Pattern of Stocks', recording_time: 30}), (mat40:Material {title:

```
'Basic Concepts of Deep Learning', recording_time: 15}), (mat41:Material {title:
'Deep Learning Methods in Python', recording_time: 15}), (mat42:Material {title:
'Implementation of Deep Learning', recording_time: 30}), (cou0)-[:Has_Material]-
>(mat0), (cou0)-[:Has_Material]->(mat1), (cou0)-[:Has_Material]->(mat2),

(cou1)-[:Has_Material]->(mat3), (cou1)-[:Has_Material]->(mat4), (cou1)-
[:Has_Material]->(mat5), (cou1)-[:Has_Material]->(mat6), (cou1)-[:Has_Material]-
>(mat7),

(cou2)-[:Has_Material]->(mat8), (cou2)-[:Has_Material]->(mat9), (cou2)-
[:Has_Material]->(mat10), (cou2)-[:Has_Material]->(mat11),

(cou3)-[:Has_Material]->(mat12), (cou3)-[:Has_Material]->(mat13), (cou3)-
[:Has_Material]->(mat14), (cou3)-[:Has_Material]->(mat15), (cou3)-
[:Has_Material]->(mat16), (cou3)-[:Has_Material]->(mat17),

(cou4)-[:Has_Material]->(mat18), (cou4)-[:Has_Material]->(mat19), (cou4)-
[:Has_Material]->(mat20), (cou4)-[:Has_Material]->(mat21), (cou4)-
[:Has_Material]->(mat22), (cou4)-[:Has_Material]->(mat23),

(cou5)-[:Has_Material]->(mat24), (cou5)-[:Has_Material]->(mat25), (cou5)-
[:Has_Material]->(mat26), (cou5)-[:Has_Material]->(mat27),

(cou6)-[:Has_Material]->(mat28), (cou6)-[:Has_Material]->(mat29), (cou6)-
[:Has_Material]->(mat30), (cou6)-[:Has_Material]->(mat31),

(cou7)-[:Has_Material]->(mat32), (cou7)-[:Has_Material]->(mat33), (cou7)-
[:Has_Material]->(mat34), (cou7)-[:Has_Material]->(mat35), (cou7)-
[:Has_Material]->(mat36),

(cou8)-[:Has_Material]->(mat37), (cou8)-[:Has_Material]->(mat38), (cou8)-
[:Has_Material]->(mat39),

(cou9)-[:Has_Material]->(mat40), (cou9)-[:Has_Material]->(mat41), (cou9)-
[:Has_Material]->(mat42),

(cat0:Category {name: 'Data Science'}),

(cat1:Category {name: 'Data Analysis'}),

(cat2:Category {name: 'Statistics'}),

(cat3:Category {name: 'Machine Learning'}),

(cat4:Category {name: 'Database'}),

(cat5:Category {name: 'SQL'}),

(cat6:Category {name: 'Database Management System'}),

(cat7:Category {name: 'NoSQL'}),

(cat8:Category {name: 'Programming'}),

(cat9:Category {name: 'Deep Learning'}),
```

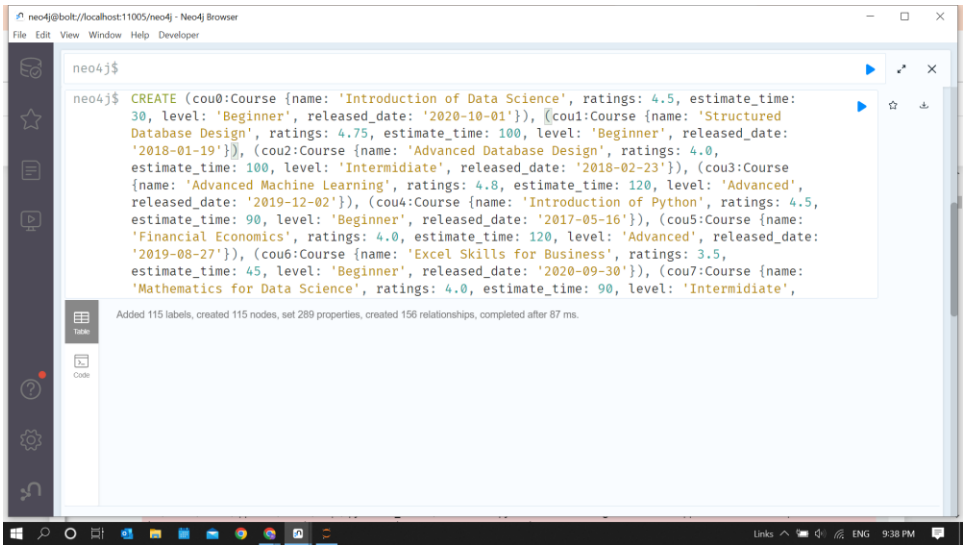
```
(cat10:Category {name: 'Computer Science'}),

(cat11:Category {name: 'Python'}),

(cat12:Category {name: 'Economics'}),

(cat13:Category {name: 'Finance'}),

(cat14:Category {name: 'Stock Market'}),

(cat15:Category {name: 'Excel'}),

(cat16:Category {name: 'Mathematics'}),

(cou0)-[:Relates_To]->(cat0), (cou0)-[:Relates_To]->(cat1), (cou0)-[:Relates_To]-
>(cat2), (cou0)-[:Relates_To]->(cat3),

(cou1)-[:Relates_To]->(cat1), (cou1)-[:Relates_To]->(cat4), (cou1)-[:Relates_To]-
>(cat5), (cou1)-[:Relates_To]->(cat6),

(cou2)-[:Relates_To]->(cat1), (cou2)-[:Relates_To]->(cat4), (cou2)-[:Relates_To]-
>(cat7), (cou2)-[:Relates_To]->(cat8),

(cou3)-[:Relates_To]->(cat1), (cou3)-[:Relates_To]->(cat0), (cou3)-[:Relates_To]-
>(cat3), (cou3)-[:Relates_To]->(cat2), (cou3)-[:Relates_To]->(cat9),

(cou4)-[:Relates_To]->(cat10), (cou4)-[:Relates_To]->(cat8), (cou4)-
[:Relates_To]->(cat11),

(cou5)-[:Relates_To]->(cat12), (cou5)-[:Relates_To]->(cat13), (cou5)-
[:Relates_To]->(cat14),

(cou6)-[:Relates_To]->(cat15), (cou6)-[:Relates_To]->(cat1),

(cou7)-[:Relates_To]->(cat16), (cou7)-[:Relates_To]->(cat0), (cou7)-
[:Relates_To]->(cat11),

(cou8)-[:Relates_To]->(cat11), (cou8)-[:Relates_To]->(cat13), (cou8)-
[:Relates_To]->(cat14),

(cou9)-[:Relates_To]->(cat11), (cou9)-[:Relates_To]->(cat9), (cou9)-
[:Relates_To]->(cat3),

(acc0:Account {first_name: 'Ran', last_name: 'Arino', email:
'rarino@myseneca.ca'}),

(acc1:Account {first_name: 'Christy', last_name: 'Collins', email:
'ccollins@myseneca.ca'}),

(acc2:Account {first_name: 'Jonathan', last_name: 'Mcdaniel', email:
'jmcdaniel@myseneca.ca'}),

(acc3:Account {first_name: 'David', last_name: 'Abbott', email:
'dabbott@myseneca.ca'}),

(acc4:Account {first_name: 'Natalie', last_name: 'Washington', email:
'nwashington@gmail.com'}),
```

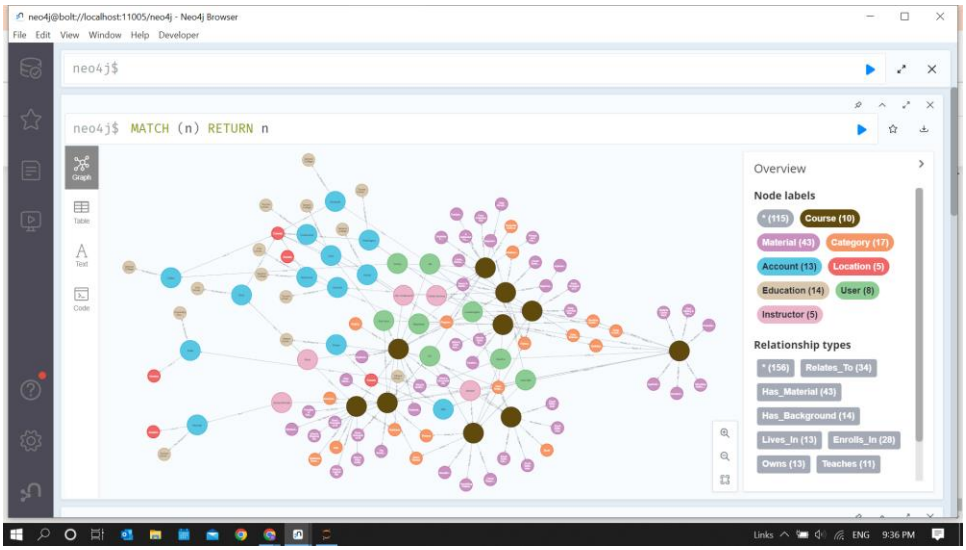
```
(acc5:Account {first_name: 'Stephanie', last_name: 'Richmond', email:
'srichmond@gmail.com'}),

(acc6:Account {first_name: 'Terri', last_name: 'Parker', email:
'tparker@gmail.com'}),

(acc7:Account {first_name: 'John', last_name: 'Ball', email: 'jball@gmail.com'}),

(acc8:Account {first_name: 'Kristie', last_name: 'Osborne', email:
'kosborne@gmail.com'}),

(acc9:Account {first_name: 'Paris', last_name: 'Byrd', email:
'pbyrd@gmail.com'}),

(acc10:Account {first_name: 'Alec', last_name: 'Underwood', email:
'aunderwood@gmail.com'}),

(acc11:Account {first_name: 'Norbert', last_name: 'Patel', email:
'npatel@gmail.com'}),

(acc12:Account {first_name: 'Sandy', last_name: 'Mcbride', email:
'smcbride@gmail.com'}),

(loc0:Location {country: 'Canada', province: 'Ontario', city: 'North York'}),
(loc1:Location {country: 'Canada', province: 'Ontario', city: 'Ottawa'}),
(loc2:Location {country: 'Canada', province: 'Ontario', city: 'Mississauga'}),
(loc3:Location {country: 'America', province: 'California', city: 'Los
Angeles'}),

(loc4:Location {country: 'America', province: 'Florida', city: ' Jacksonville'}),

(acc0)-[:Lives_In]->(loc0),

(acc1)-[:Lives_In]->(loc0),

(acc2)-[:Lives_In]->(loc0),

(acc3)-[:Lives_In]->(loc0),

(acc4)-[:Lives_In]->(loc0),

(acc5)-[:Lives_In]->(loc0),

(acc6)-[:Lives_In]->(loc1),

(acc7)-[:Lives_In]->(loc1),

(acc8)-[:Lives_In]->(loc2),

(acc9)-[:Lives_In]->(loc0),

(acc10)-[:Lives_In]->(loc0),

(acc11)-[:Lives_In]->(loc3),

(acc12)-[:Lives_In]->(loc4),
```

```
(edu0:Education {school_name: 'Seneca College', major: 'Data Science and
Analysis', enroll_date: '2022-01-13', graduate_date: ''}),

(edu1:Education {school_name: 'Seneca College', major: 'Computer Science',
enroll_date: '2021-09-06', graduate_date: ''}),

(edu2:Education {school_name: 'Seneca College', major: 'Data Science and
Analysis', enroll_date: '2021-09-06', graduate_date: ''}),

(edu3:Education {school_name: 'Seneca College', major: 'Data Science and
Analysis', enroll_date: '2018-09-03', graduate_date: '2022-04-22'}),

(edu4:Education {school_name: 'Toronto University', major: 'Computer Science',
enroll_date: '2016-09-03', graduate_date: '2020-04-22'}),

(edu5:Education {school_name: 'Seneca College', major: 'Liberal Arts',
enroll_date: '2020-09-05', graduate_date: '2021-08-15'}),

(edu6:Education {school_name: 'York University', major: 'Computer Science',
enroll_date: '2022-09-06', graduate_date: ''}),

(edu7:Education {school_name: 'Ottawa University', major: 'Finance', enroll_date:
'2020-09-03', graduate_date: ''}),

(edu8:Education {school_name: 'Ottawa University', major: 'Finance', enroll_date:
'2022-09-06', graduate_date: ''}),

(edu9:Education {school_name: 'Toronto University', major: 'Computer Science',
enroll_date: '2010-09-03', graduate_date: '2014-04-22'}),

(edu10:Education {school_name: 'York University', major: 'Computer Science',
enroll_date: '2012-09-03', graduate_date: '2016-04-22'}),

(edu11:Education {school_name: 'Toronto University', major: 'Finance',
enroll_date: '2007-09-01', graduate_date: '2011-04-24'}),

(edu12:Education {school_name: 'University of Southern California', major: 'Data
Science', enroll_date: '2008-09-01', graduate_date: '2012-04-30'}),

(edu13:Education {school_name: 'Toronto University', major: 'Finance',
enroll_date: '2000-09-03', graduate_date: '2004-04-22'}),

(acc0)-[:Has_Background]->(edu0),

(acc1)-[:Has_Background]->(edu1),

(acc2)-[:Has_Background]->(edu2),

(acc3)-[:Has_Background]->(edu3),

(acc4)-[:Has_Background]->(edu4),

(acc5)-[:Has_Background]->(edu5),

(acc5)-[:Has_Background]->(edu6),

(acc6)-[:Has_Background]->(edu7),
```

```
(acc7)-[:Has_Background]->(edu8),

(acc8)-[:Has_Background]->(edu9),

(acc9)-[:Has_Background]->(edu10),

(acc10)-[:Has_Background]->(edu11),

(acc11)-[:Has_Background]->(edu12),

(acc12)-[:Has_Background]->(edu13),

(user0:User {name: 'Ran Arino', organization: 'Seneca College'}),

(user1:User {name: 'Christy', organization: 'Seneca College'}),

(user2:User {name: 'JM', organization: 'Seneca College'}),

(user3:User {name: 'David.A', organization: 'AAA Corporation'}),

(user4:User {name: 'n-washington', organization: 'BBB Corporation'}),

(user5:User {name: 'Stephanie', organization: 'York University'}),

(user6:User {name: 'T.P.', organization: 'Ottawa University'}),

(user7:User {name: 'John Ball', organization: 'Ottawa University'}),

(user0)-[:Enrolls_In]->(cou0), (user0)-[:Enrolls_In]->(cou2), (user0)-
[:Enrolls_In]->(cou5),

(user1)-[:Enrolls_In]->(cou4), (user1)-[:Enrolls_In]->(cou9),

(user2)-[:Enrolls_In]->(cou0), (user2)-[:Enrolls_In]->(cou2), (user2)-
[:Enrolls_In]->(cou7),

(user3)-[:Enrolls_In]->(cou3), (user3)-[:Enrolls_In]->(cou5), (user3)-
[:Enrolls_In]->(cou8), (user3)-[:Enrolls_In]->(cou9),

(user4)-[:Enrolls_In]->(cou2), (user4)-[:Enrolls_In]->(cou6), (user4)-
[:Enrolls_In]->(cou7), (user4)-[:Enrolls_In]->(cou9),

(user5)-[:Enrolls_In]->(cou0), (user5)-[:Enrolls_In]->(cou1), (user5)-
[:Enrolls_In]->(cou4), (user5)-[:Enrolls_In]->(cou9),

(user6)-[:Enrolls_In]->(cou4), (user6)-[:Enrolls_In]->(cou5),

(user6)-[:Enrolls_In]->(cou6), (user6)-[:Enrolls_In]->(cou8),

(user7)-[:Enrolls_In]->(cou3), (user7)-[:Enrolls_In]->(cou5), (user7)-
[:Enrolls_In]->(cou7), (user7)-[:Enrolls_In]->(cou8),

(user0)-[:Owns]->(acc0),

(user1)-[:Owns]->(acc1),

(user2)-[:Owns]->(acc2),

(user3)-[:Owns]->(acc3),
```

```
(user4)-[:Owns]->(acc4),

(user5)-[:Owns]->(acc5),

(user6)-[:Owns]->(acc6),

(user7)-[:Owns]->(acc7),

(inst0:Instructor {name: 'Kristie Osborne', comment: 'Kristie is a professor in
the data science and analysis field.'}),

(inst1:Instructor {name: 'Paris', comment: 'Paris is a researcher in computer
science and has experienced database manager in company.'}),

(inst2:Instructor {name: 'Alec Underwood', comment: 'Alec is a professor in the
finance and the statistics field.'}),

(inst3:Instructor {name: 'Norbert', comment: 'Nobert is familiar with data
science and stock market.'}),

(inst4:Instructor {name: 'Sandy Mcbride', comment: 'Sandy is teaching finance in
Toronto University for more than 15 years.'}),

(inst0)-[:Teaches]->(cou0), (inst0)-[:Teaches]->(cou4), (inst0)-[:Teaches]-
>(cou9),

(inst1)-[:Teaches]->(cou1), (inst1)-[:Teaches]->(cou2),

(inst2)-[:Teaches]->(cou5), (inst2)-[:Teaches]->(cou7),

(inst3)-[:Teaches]->(cou3), (inst3)-[:Teaches]->(cou6), (inst3)-[:Teaches]-
>(cou8),

(inst4)-[:Teaches]->(cou5),

(inst0)-[:Owns]->(acc8),

(inst1)-[:Owns]->(acc9),

(inst2)-[:Owns]->(acc10),

(inst3)-[:Owns]->(acc11),

(inst4)-[:Owns]->(acc12)
```

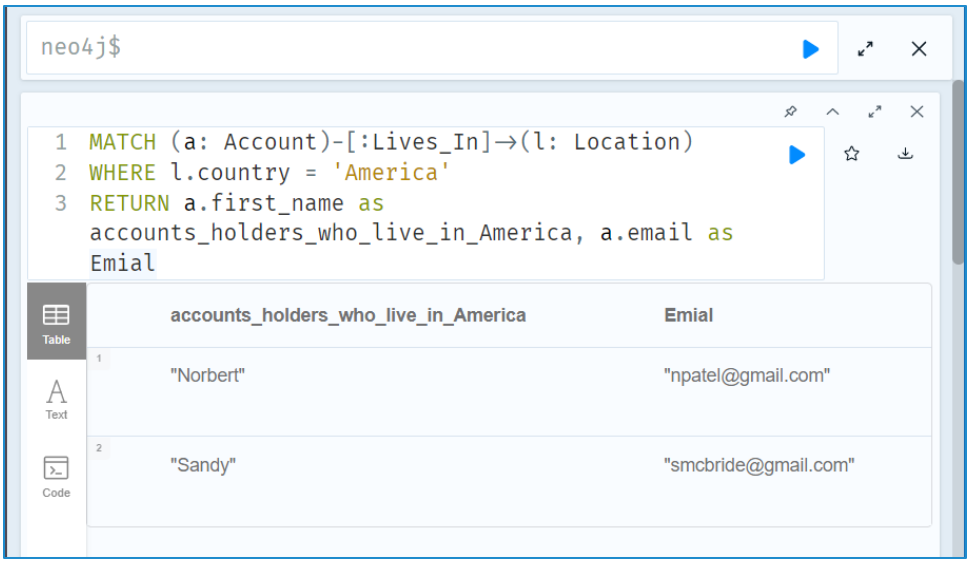


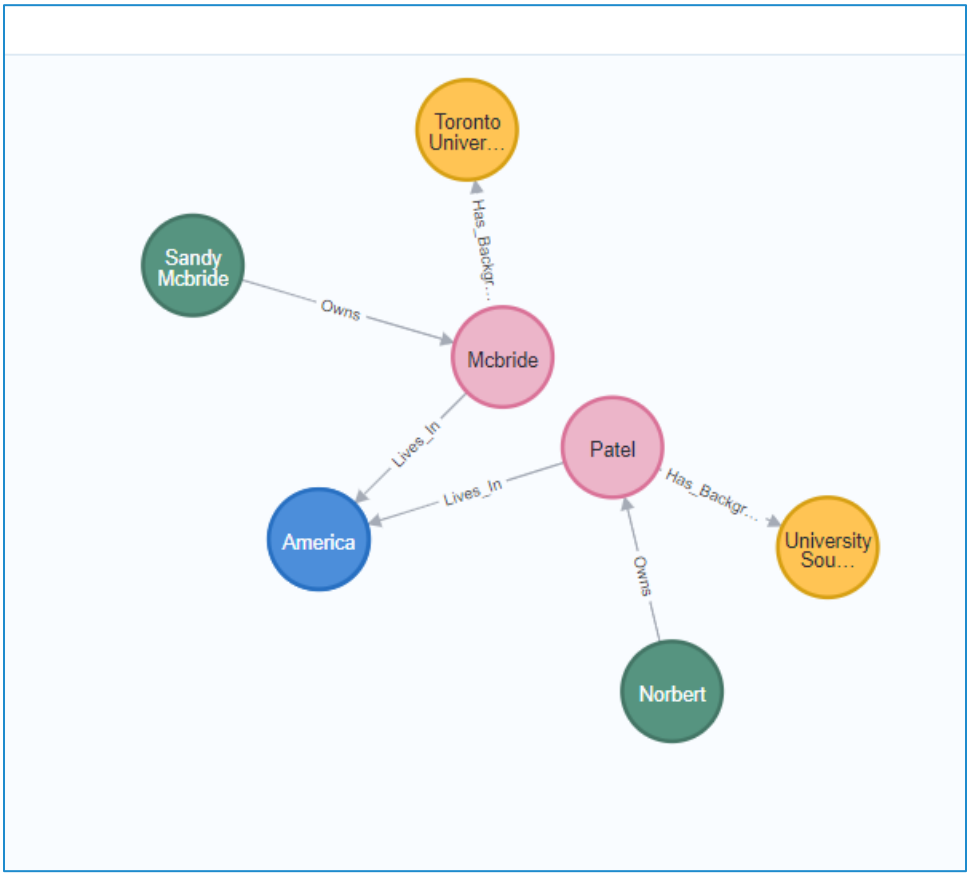


2. Simple Queries

1- Retrieve all account holders who live in America. Return their first names and their Emails.

```
MATCH (a: Account)-[:Lives_In]->(l: Location)

WHERE l.country = 'America'

RETURN a.first_name as accounts_holders_who_live_in_America, a.email as Emial
```





2- Retrieve those users who enroll in courses with ranking rate 4.5, return their name and their organization.

MATCH (n: User)

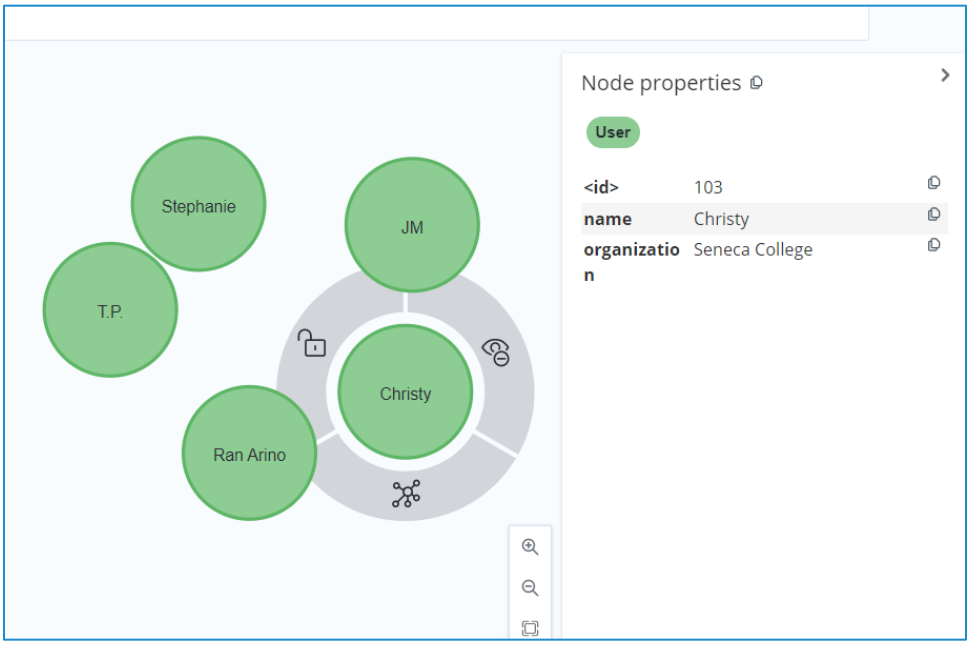
WHERE (n)-[:Enrolls_In]-({ratings: 4.5})

RETURN n.name, n. organization

```
1  MATCH (n: User)
2  WHERE (n)-[:Enrolls_In]-({ratings: 4.5})
3  RETURN n.name, n. organization
4
```

Table | Text | Code

"n.name"	"n. organization"
"Ran Arino"	"Seneca College"
"Christy"	"Seneca College"
"JM"	"Seneca College"
"Stephanie"	"York University"
"T.P."	"Ottawa University"

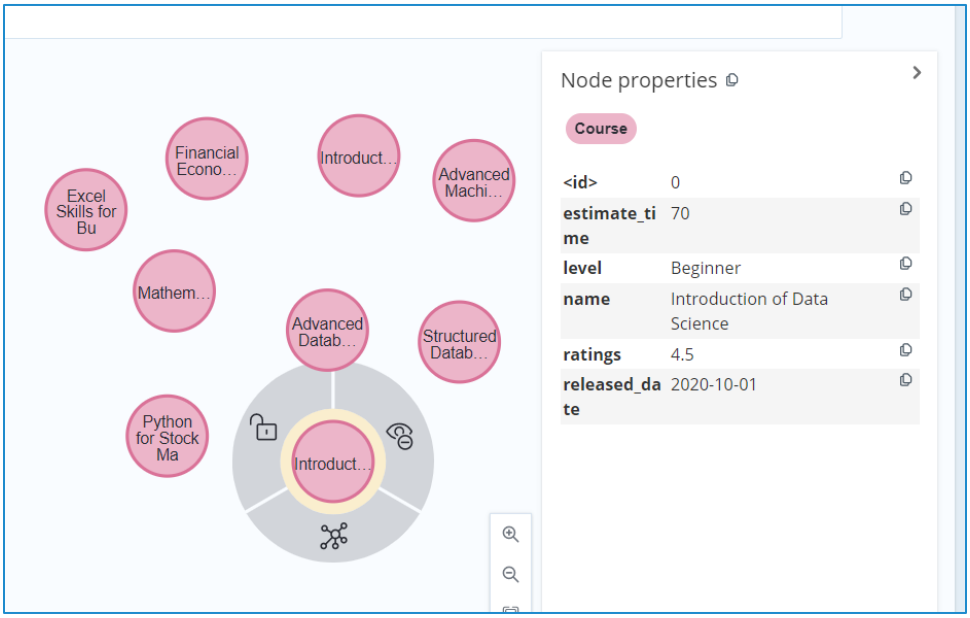


3- Retrieve courses that have estimate time of greater equal 80 or their levels are ‘Beginner’.

```
MATCH (n: Course)
WHERE n.estimate_time>=80 OR  n.level = "Beginner"
RETURN  n.estimate_time, n.name, n.level
```

```
MATCH (n: Course)

WHERE n.estimate_time >= 80 OR  n.level = "Beginner"

RETURN  n.estimate_time, n.name, n.level
```

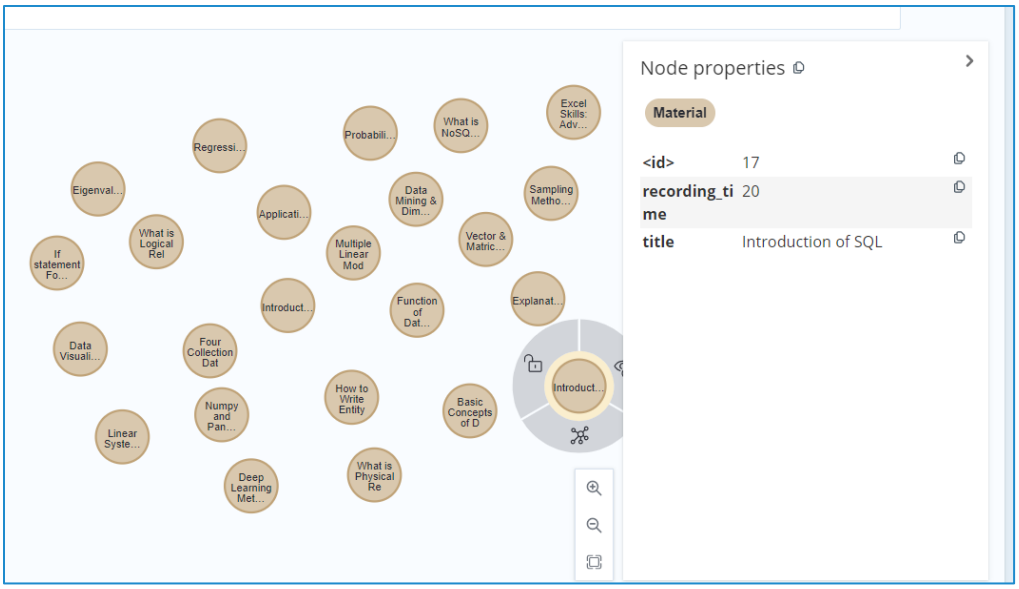
"n.estimate_time"	"n.name"	"n.level"
70	"Introduction of Data Science"	"Beginner"
100	"Structured Database Design"	"Beginner"
100	"Advanced Database Design"	"Intermidiate"
120	"Advanced Machine Learning"	"Advanced"
90	"Introduction of Python"	"Beginner"
120	"Financial Economics"	"Advanced"
45	"Excel Skills for Business"	"Beginner"
90	"Mathematics for Data Science"	"Intermidiate"
90	"Python for Stock Market Analysis"	"Intermidiate"



4- Retrieve all materials of courses which have either recording time of greater than 10 or title of 'Numpy and Pandas. Return in order of recording time

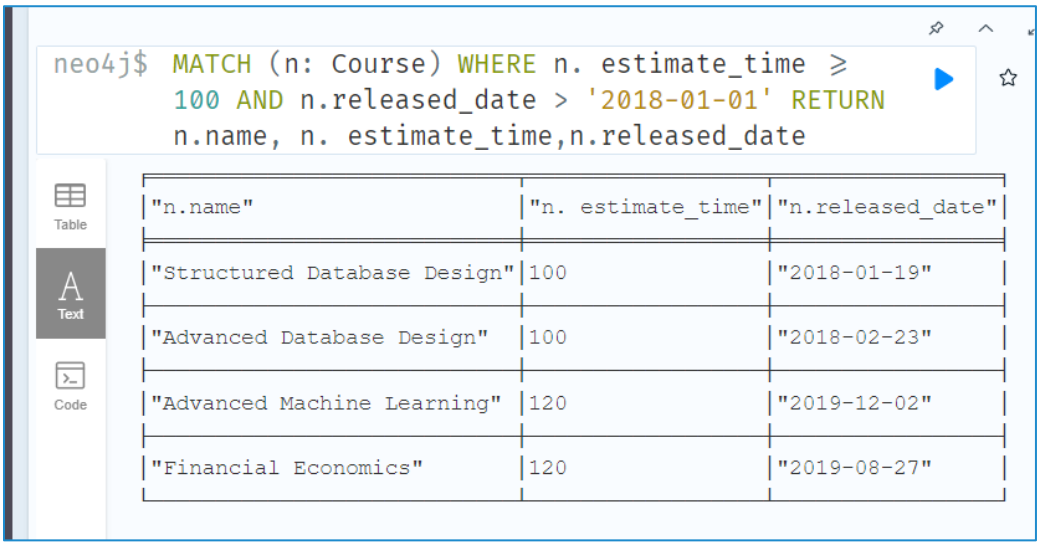
```
MATCH (n: Material)

WHERE 10 <n.recording_time <=20 OR n.title = "Numpy and Pandas"

RETURN n.title, n.recording_time

ORDER BY n.recording_time
```

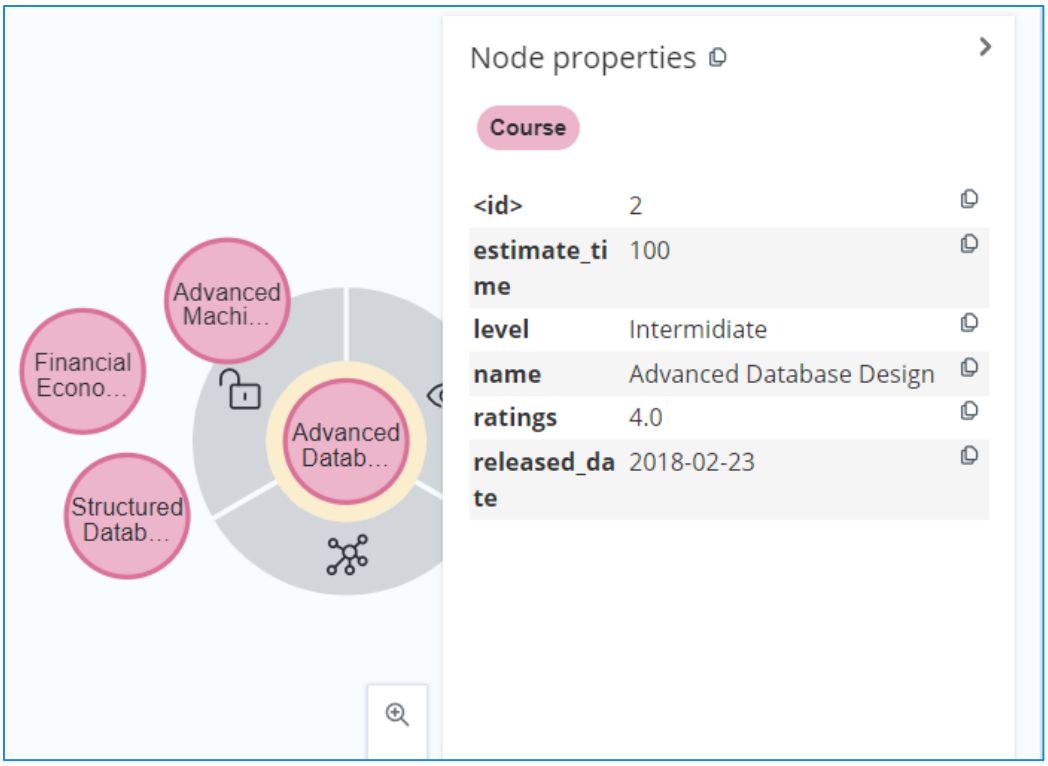
"n.title"	"n.recording_time"
"If statement and For Loop"	15
"Data Visualization: Matplotlib"	15
"Excel Skills: Advanced"	15
"Linear Systems"	15
"Vector & Matrices"	15
"Eigenvalues & Eigenvectors"	15
"Regressions"	15
"Basic Concepts of Deep Learning"	15
"Deep Learning Methods in Python"	15
"What is Physical Relational Database?"	20
"What is Logical Relational Database?"	20
"How to Write Entity Relationship Diagram"	20
"Function of Database Management System"	20
"Introduction of SQL"	20
"What is NoSQL?"	20
"Explanation of Sharding and Replication"	20
"Multiple Linear Model"	20
"Probability: Bayesian theory"	20
"Application of Classification"	20
"Data Mining & Dimensionality Reduction"	20
"Sampling Methods"	20
"Introduction of Deep Learning"	20
"Four Collection Data Types"	20
"Numpy and Pandas"	20



5- Retrieve all courses that estimate time is equal greater 100 and the released date is 2019-01-01.

```
MATCH (n: Course) WHERE n. estimate_time >= 100 AND n.released_date > '2018-01-01'
RETURN n.name, n. estimate_time,n.released_date
```





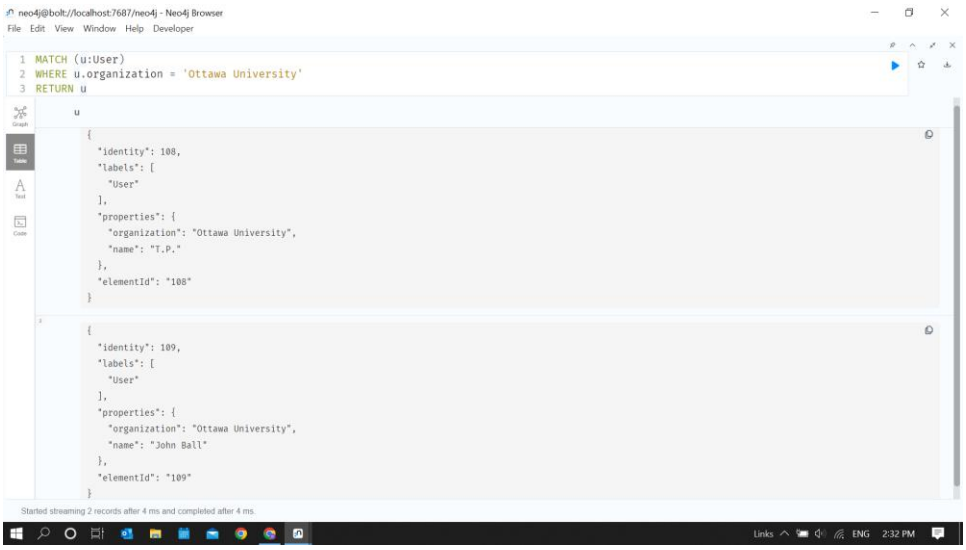
6- Retrieve average course estimate time for users who take intermediate level courses.

```
Match (u: User) – [r: Enrolls_In] -> (c: Course)
Where  c.level='Intermidiate'
Return u.name, collect(c.name), AVG(c.estimate_time)
```

"u.name"	"collect(c.name)"	"AVG(c.estimate_time)"
"Ran Arino"	["Advanced Database Design"]	100.0
"JM"	["Advanced Database Design","Mathematics for Data Science"]	95.0
"n-washington"	["Advanced Database Design","Mathematics for Data Science"]	95.0
"John Ball"	["Mathematics for Data Science","Python for Stock Market Analysis"]	90.0
"T.P."	["Python for Stock Market Analysis"]	90.0
"David.A"	["Python for Stock Market Analysis"]	90.0

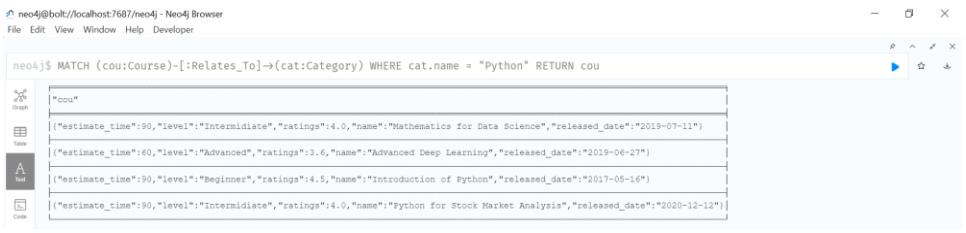
7- Who is enrolling in Ottawa University now?

```
MATCH (u:User)
WHERE u.organization = 'Ottawa University'
RETURN u
```



8- What courses are related to learning Python? (Showing course name, ratings by other learners, and estimated time to complete each course)

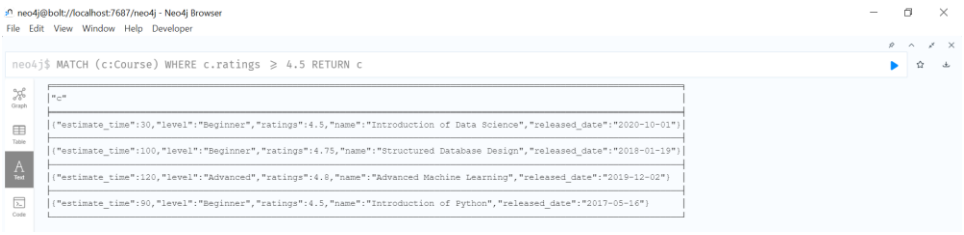
```
MATCH (cou:Course)-[:Relates_To]->(cat:Category)
WHERE cat.name = "Python"
RETURN cou
```



9- What are courses whose rating is greater than or equal to 4.5? (Showing course name, ratings by other learners, estimated time to complete, and course level)

```
MATCH (c:Course)
```

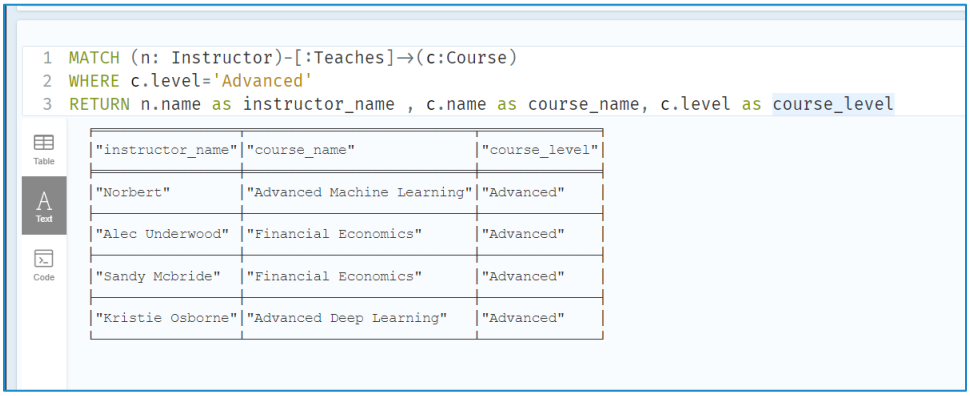
```
WHERE c.ratings >= 4.5

RETURN c
```



3. Advance Queries

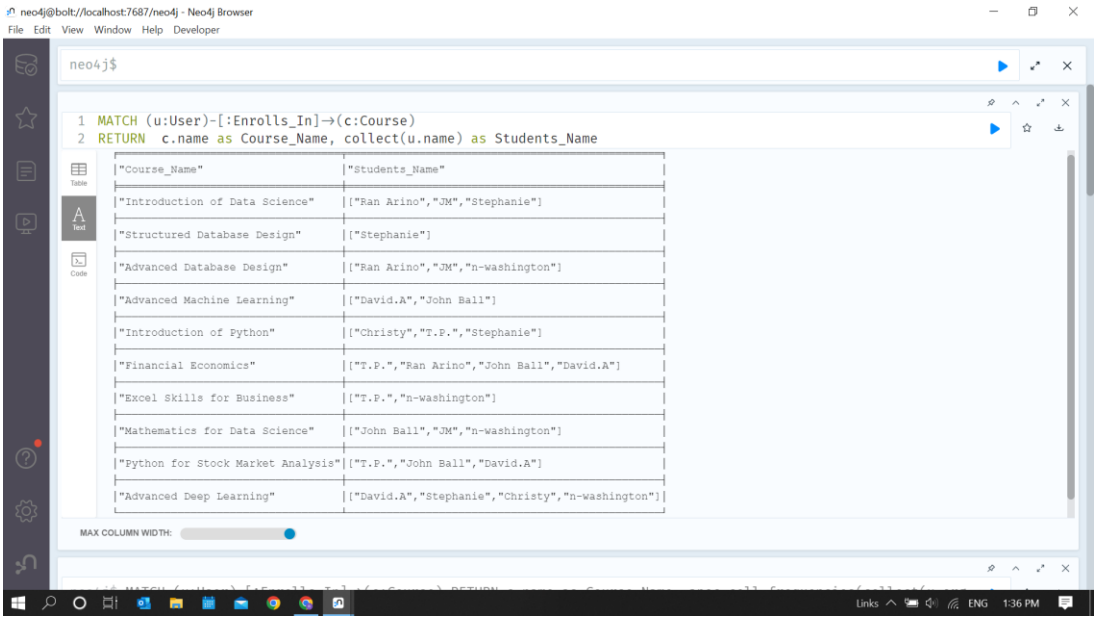
1- Retrieve all instructors who teach Advanced courses. Return their name with their courses.

```
MATCH (n: Instructor)-[:Teaches]->(c:Course)

WHERE c.level='Advanced'

RETURN n.name as instructor_name , c.name as course_name, c.level as course_level
```



2- Retrieve every course name and the name of users who are enrolling in each course.

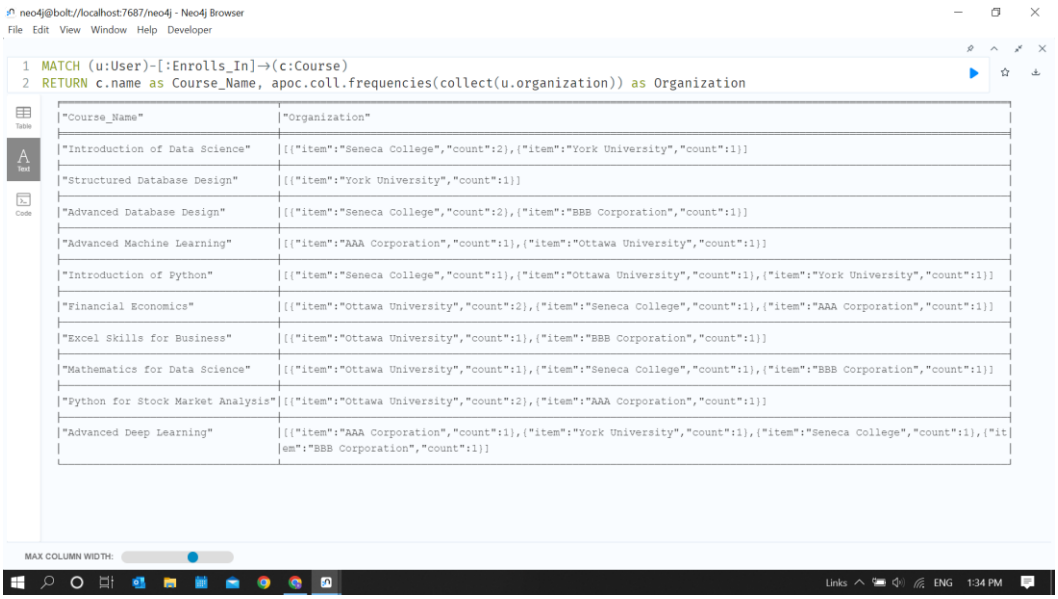
```
MATCH (u:User)-[:Enrolls_In]->(c:Course)

RETURN  c.name as Course_Name, collect(u.name) as Students_Name
```



3- Retrieve frequency of organizations of each course.

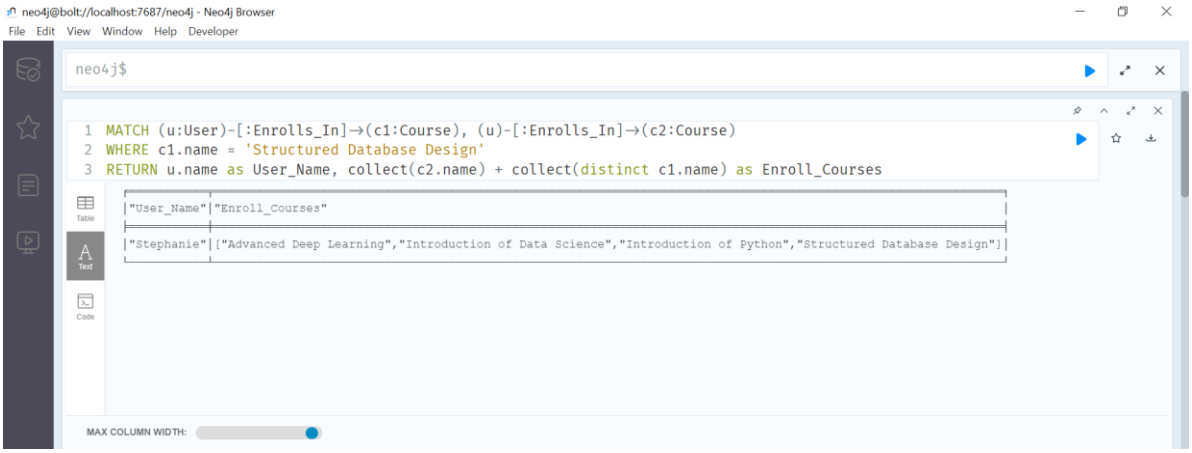
MATCH (u:User)-[:Enrolls_In]->(c:Course)

RETURN c.name as Course_Name, apoc.coll.frequencies(collect(u.organization)) as Organization



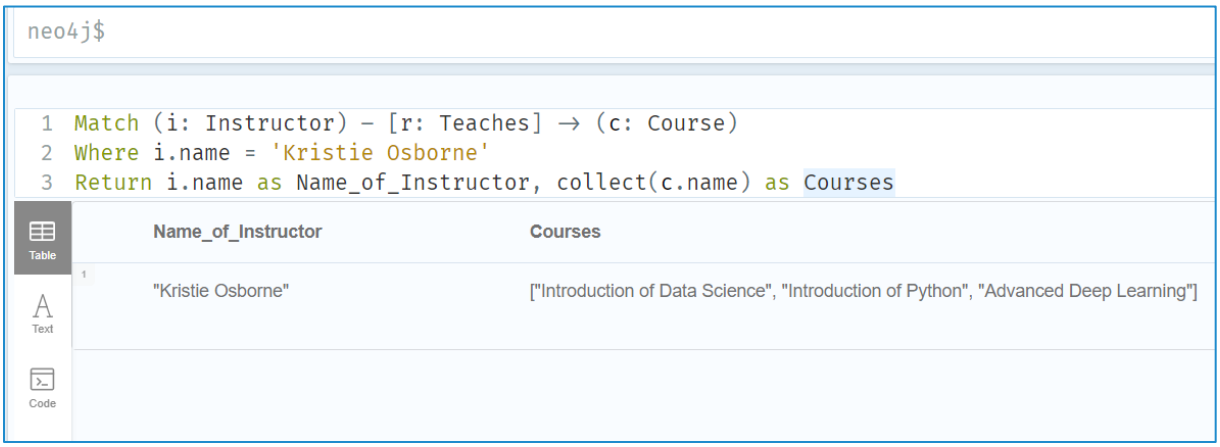
4- Retrieve all users who take ‘Structured Database Design’.

```
MATCH (u:User)-[:Enrolls_In]->(c1:Course), (u)-[:Enrolls_In]->(c2:Course)
WHERE c1.name = 'Structured Database Design'
RETURN u.name as User_Name, collect(c2.name) + collect(distinct c1.name) as Enroll_Courses
```



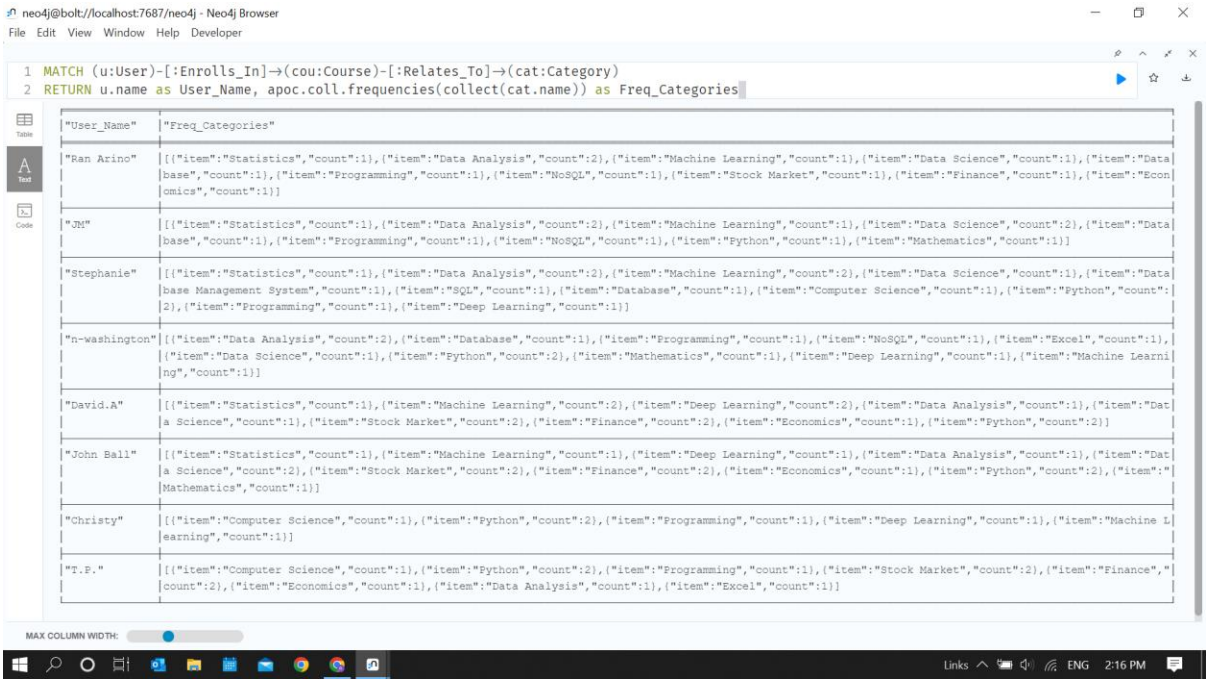
5- Retrieve all courses of instructor ‘Kristie Osborne'.

```
Match (i: Instructor) – [r: Teaches] -> (c: Course)
Where i.name = 'Kristie Osborne'
Return i.name as Name_of_Instructor, collect(c.name) as Courses
```



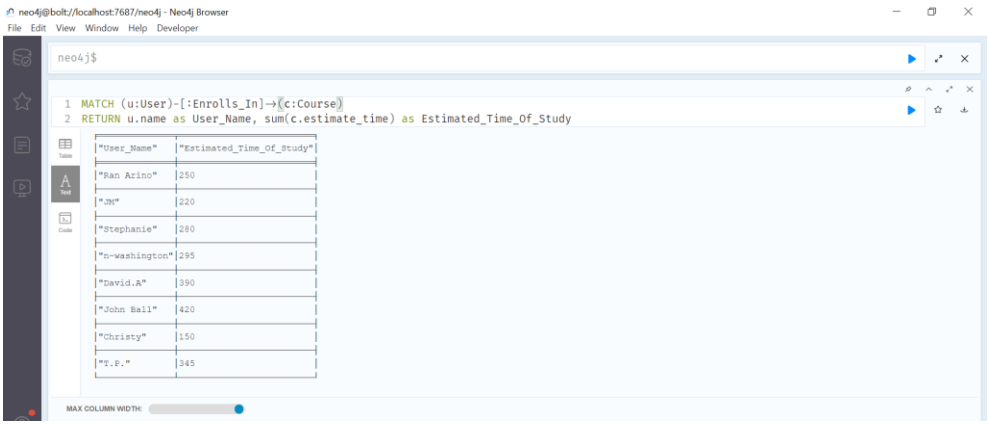
6- Showing the frequency of the categories (skills related to the course) of each user.

```
MATCH (u:User)-[:Enrolls_In]->(cou:Course)-[:Relates_To]->(cat:Category)
RETURN u.name as User_Name, apoc.coll.frequencies(collect(cat.name)) as Freq_Categories
```



7- How many minutes will each user need to study to complete all enrolling courses?

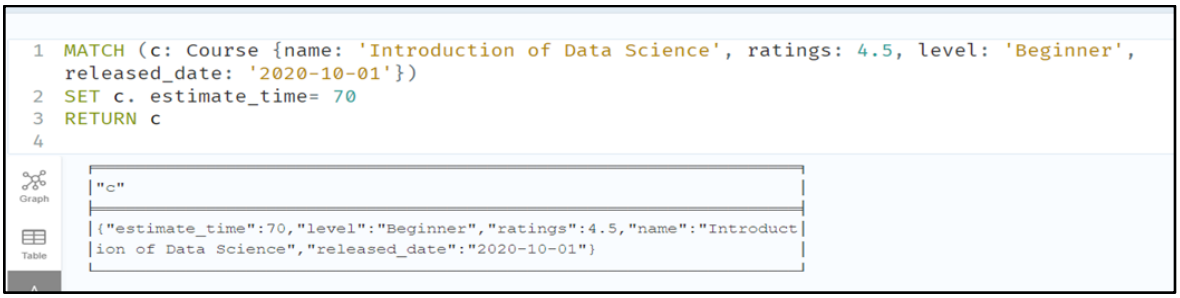
```
MATCH (u:User)-[:Enrolls_In]->(c:Course)
RETURN u.name as User_Name, sum(c.estimate_time) as Estimated_Time_Of_Study
```

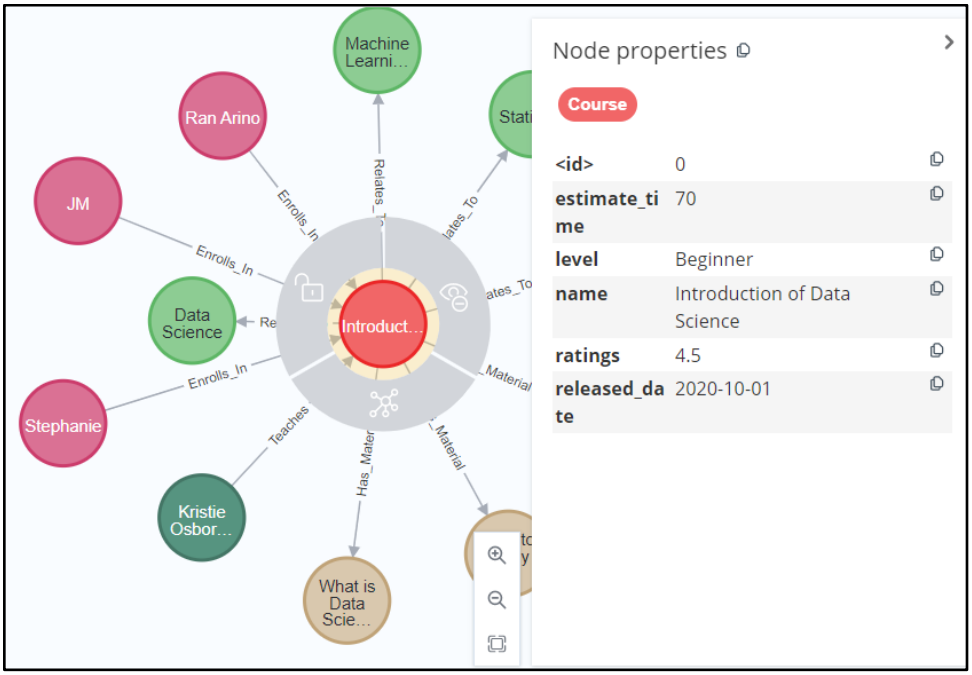


4. **Updating** and **deleting** database content:

1- Update: updating estimate_time to 70 for course 'Introduction of Data Science', ratings: 4.5, level: 'Beginner', released_date: '2020-10-01'.

```
MATCH (c: Course {name: 'Introduction of Data Science', ratings: 4.5, level: 'Beginner', released_date: '2020-10-01'})

SET c. estimate_time= 70

RETURN c
```

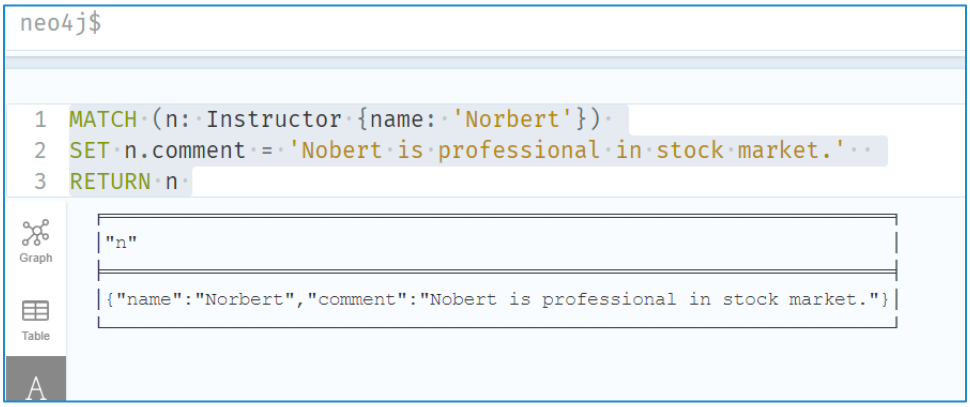


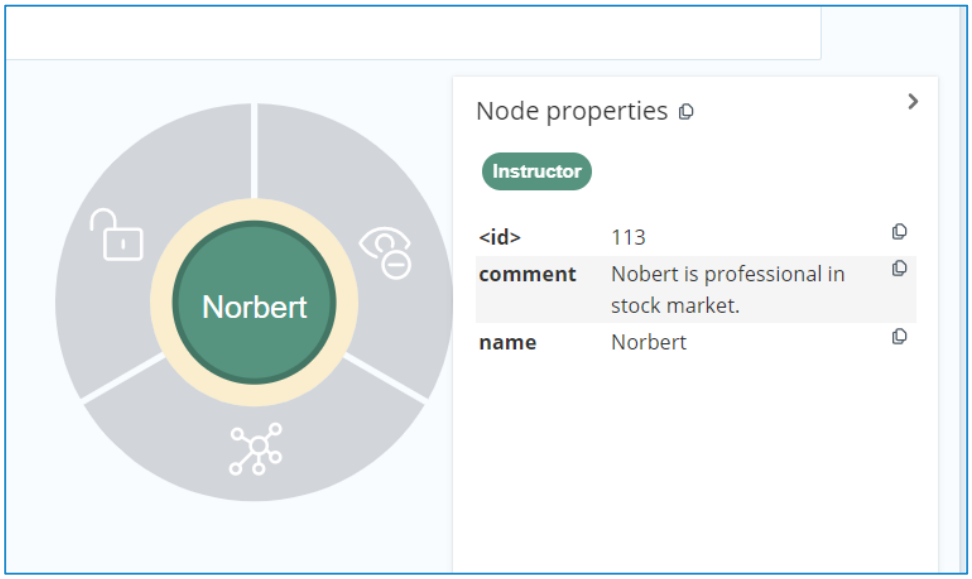


2. updating comment for instructor ‘Norbert’.

```
MATCH (n: Instructor {name: 'Norbert'})

SET n.comment = 'Nobert is professional in stock market.'
```

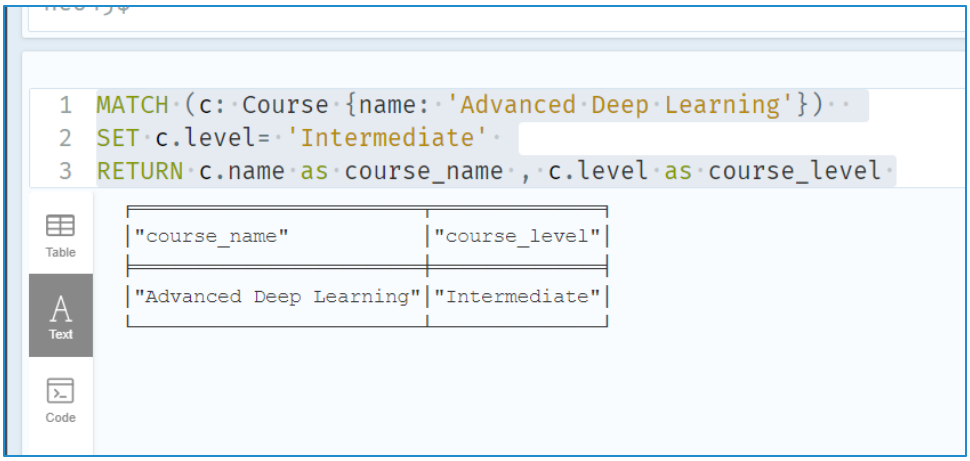
```
RETURN n
```

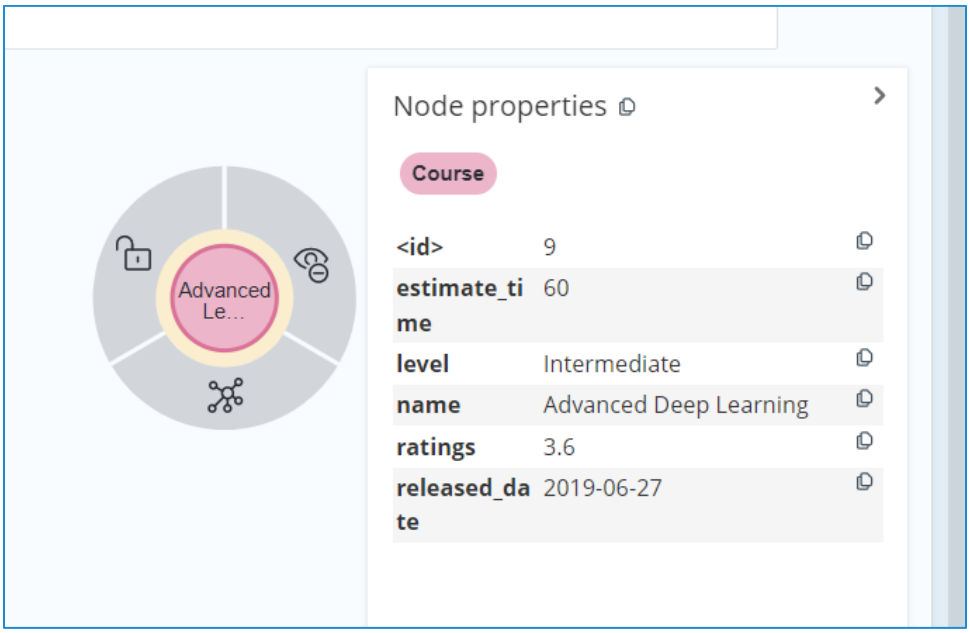




3- Updating level of course ‘Advanced Deep Learning’ to ‘Intermediate’.

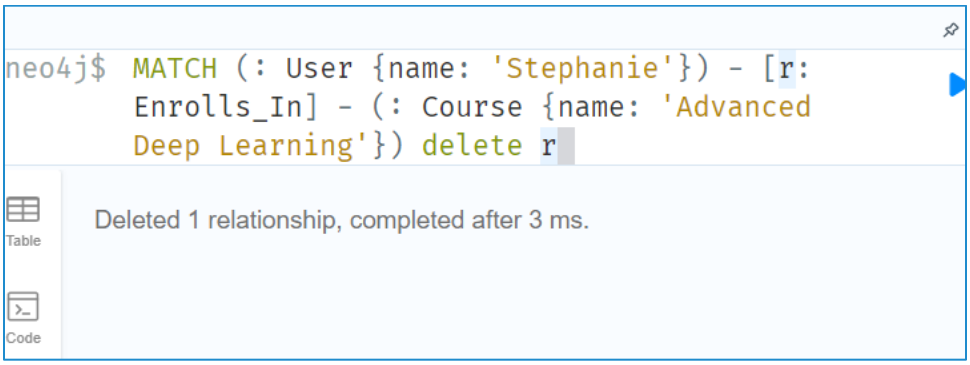
```
MATCH (c: Course {name: 'Advanced Deep Learning'})
SET c.level= 'Intermediate'
RETURN c.name as course_name , c.level as course_level
```





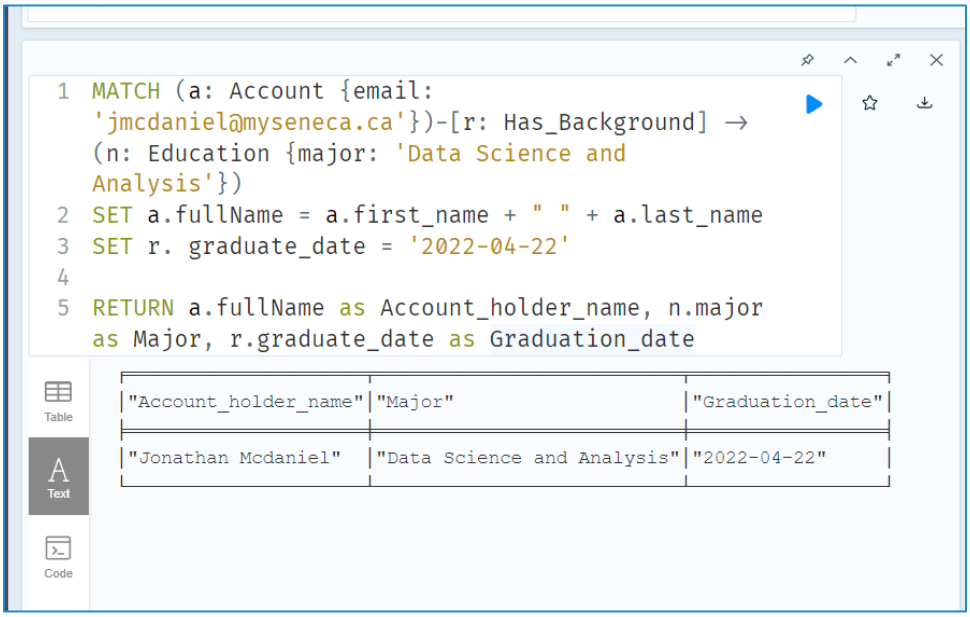
4. Remove user ‘Stephanie’ from course ‘Advanced Deep Learning’.

MATCH (: User {name: 'Stephanie'}) - [r: Enrolls_In] - (: Course {name: 'Advanced Deep Learning'}) Delete r



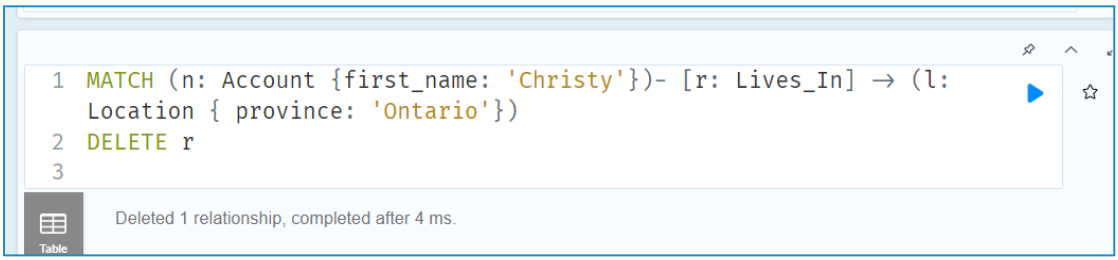
5- Updating graduation data for email ‘jmcdaniel@myseneca.ca’ to ‘2022-04-22'.

```
MATCH (a: Account {email: 'jmcdaniel@myseneca.ca'})-[r: Has_Background] -> (n: Education {major: 'Data Science and Analysis'})

SET a.fullName = a.first_name + " " + a.last_name

SET r. graduate_date = '2022-04-22'

RETURN a.fullName as Account_holder_name, n.major as Major, r.graduate_date as Graduation_date
```



6- Deleting a relationship. Removing relationship of "Lives_in" from 'Christy' account name.

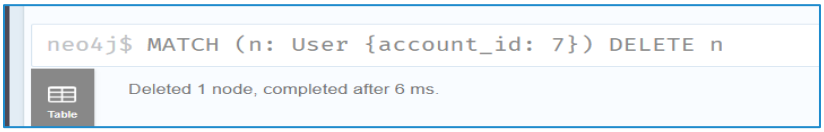
```
MATCH (n: Account {first_name: 'Christy'})- [r: Lives_In] -> (l: Location {province: 'Ontario'})

DELETE r
```



7. Delete a Node

```
MATCH (n: User {account_id: 7})
```

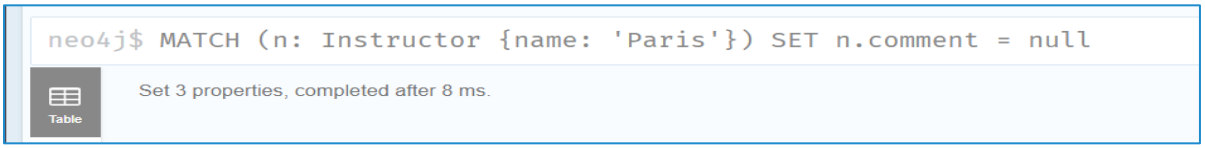
DELETE n



8. Delete property with SET to null value.

MATCH (n: Instructor {name: 'Paris'})

SET n.comment = null



9. Delete a Node and Relationship

MATCH (n: Category {name: 'Computer Science'})

DETACH DELETE n