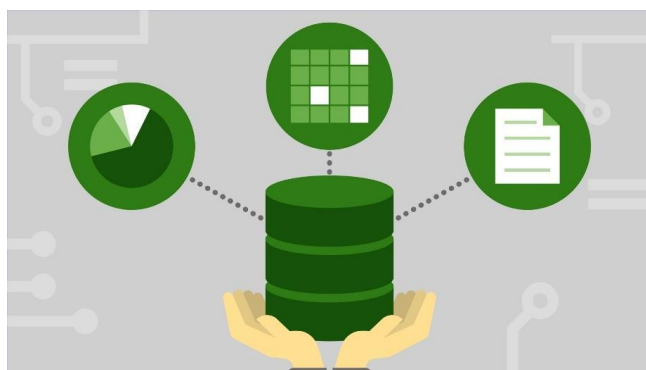


به نام خدا



دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده برق و کامپیوتر



آزمایشگاه پایگاه داده

دستور کار شماره چهارم (آشنایی با دیتابیس های گراف محور)

امیرعلی رایگان

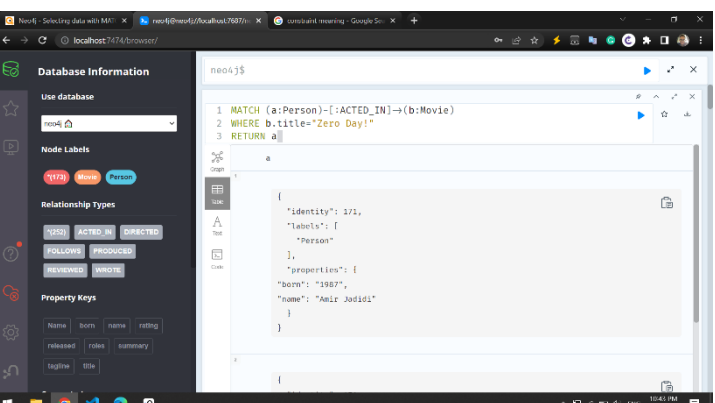
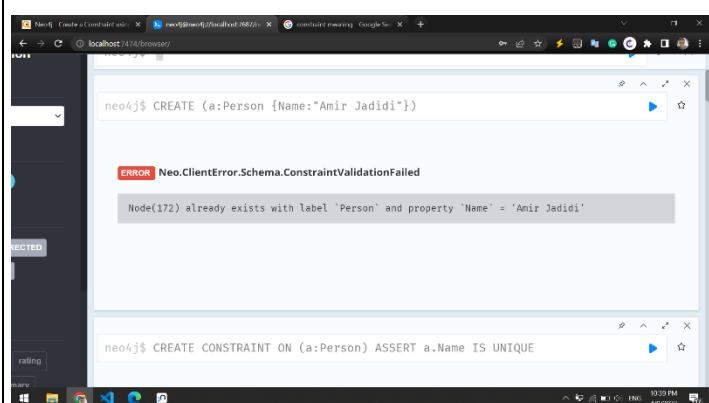
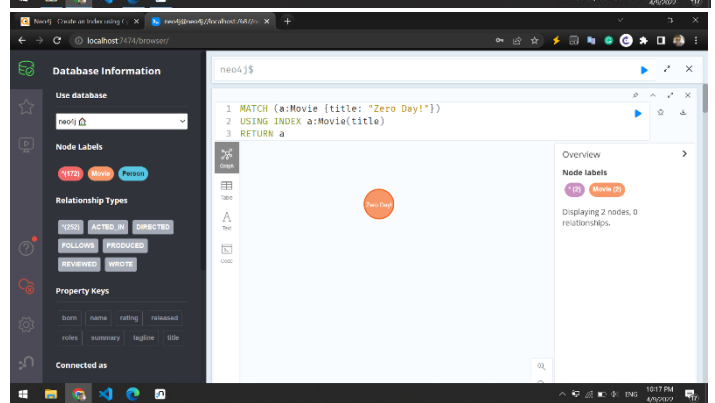
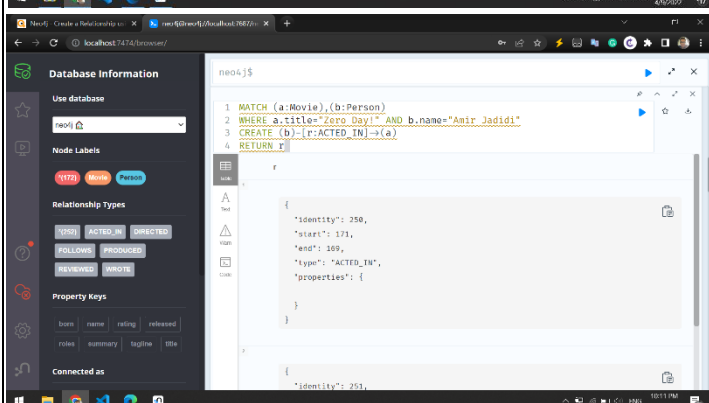
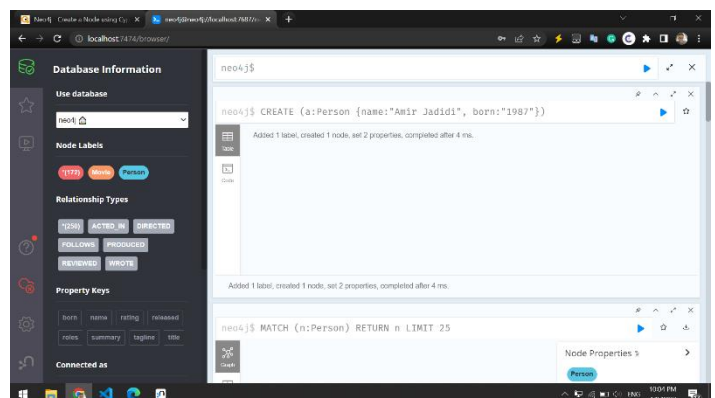
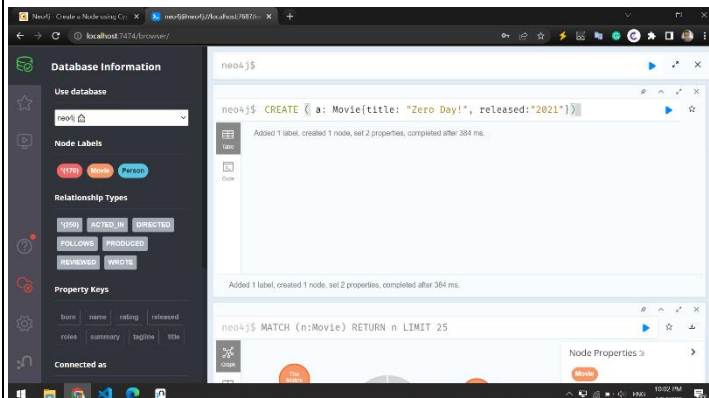
۸۱۰۱۹۷۶۲۳

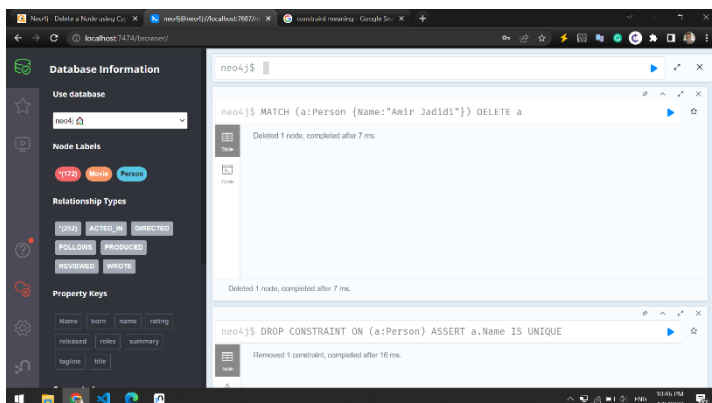
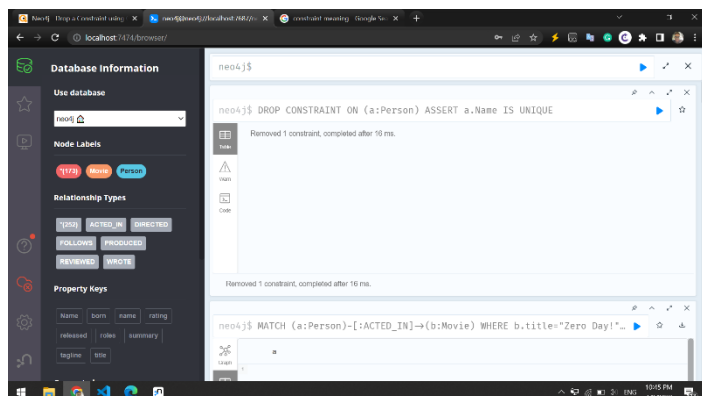
بهار ۱۴۰۱

گزارش دستورکار انجام شده

گام اول. آشنایی با Neo4j و زبان Cypher

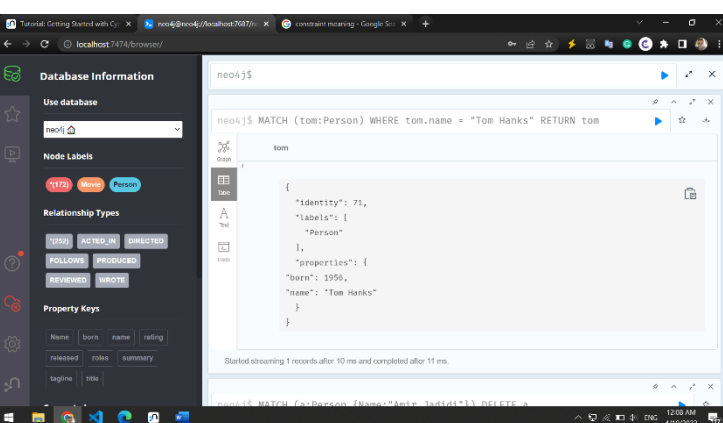
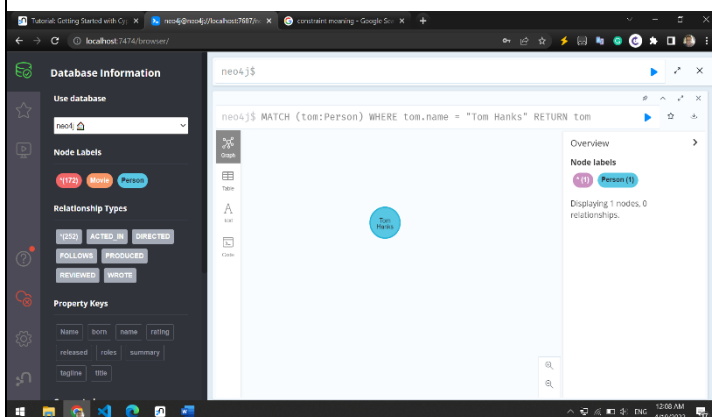
در ابتدا برای آشنایی با یک پایگاه داده گراف محور از طریق آموزش های ابتدایی سایت quackit.com با زبان cypher که زبان دستور های دیتابیس Neo4j می باشد آشنا شدیم. عکس های زیر مراحل این آشنایی را نشان می دهد.



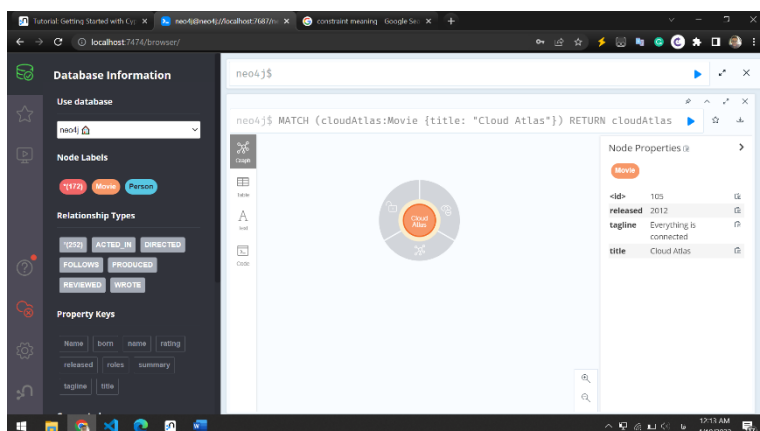


گام دوم. آموزش زبان Cypher از سایت neo4j.com

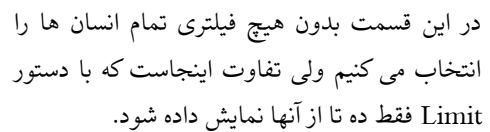
در هر قسمت از این آموزش عکس هایی به همراه توضیحات دستورات قرار داده شده است.



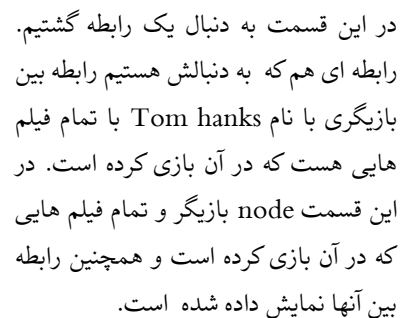
در شکل های بالا قابل مشاهده است که دستوری که وارد کردیم تمام Node هایی از جنس بازیگر که نامشان Tom Hanks می باشد را با نام Tom انتخاب کرده و نمایش داده است. در تصویر سمت چپ اطلاعات انتخاب شده به شکل گراف و در شکل سمت راست به صورت جدولی قابل مشاهده می باشد.



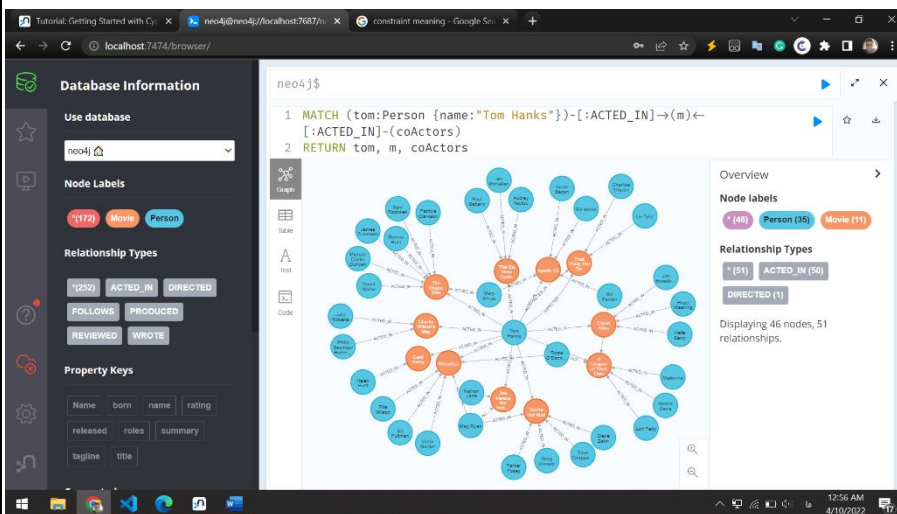
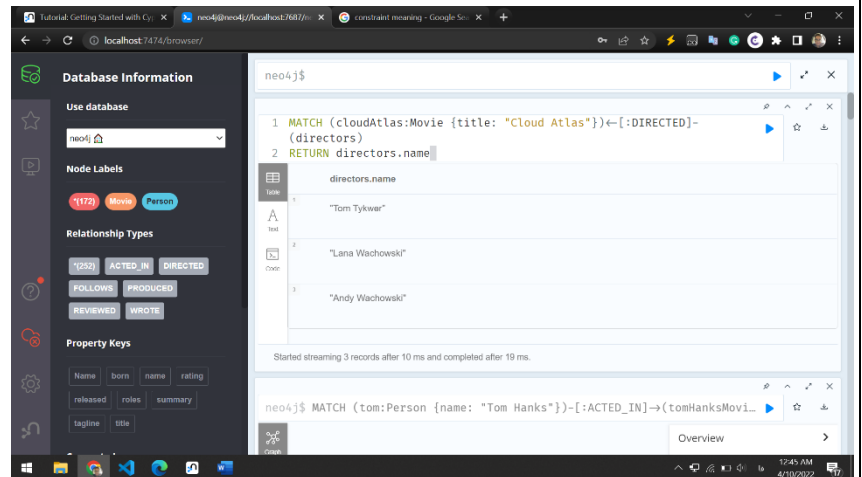
در شکل سمت چپ هم مثل مرحله قبل یک انتخاب صورت گرفته. از بین تمامی فیلم ها فیلمی که نامش مشخص شده جست و جو داده شده و نمایش داده شده است.



The screenshot shows a web browser with two tabs. The active tab is titled 'Tutorial: Getting Started With Cy...' and shows the Neo4j 'Database Information' page. The page has a dark sidebar with navigation icons and a main content area. The 'Use database' section shows 'neo4j' selected. The 'Node Labels' section displays 'Movie' and 'Person' as red and blue circles respectively. The 'Relationship Types' section shows 'DIRECTED', 'ACTED_IN', 'PRODUCED', 'REVIEWED', and 'WROTE' as buttons. The 'Property Keys' section shows buttons for 'Name', 'born', 'name', 'rating', 'released', 'roles', 'summary', 'tagline', and 'title'. The second tab is titled 'localhost:7474/browser/'. Below the browser, a terminal window is open, showing a Cypher query: 'neo4j\$ MATCH (nineties:Movie) WHERE nineties.released > 1990 AND nineti...'. The terminal output shows a table of results with 6 rows, each containing a movie title. The titles are: 'The Matrix', 'The Devil's Advocate', 'A Few Good Men', 'As Good as It Gets', 'What Dreams May Come', and 'Snow Falling on Cedars'. At the bottom of the terminal, it says 'Started streaming 19 records after 24 ms and completed after 26 ms.'

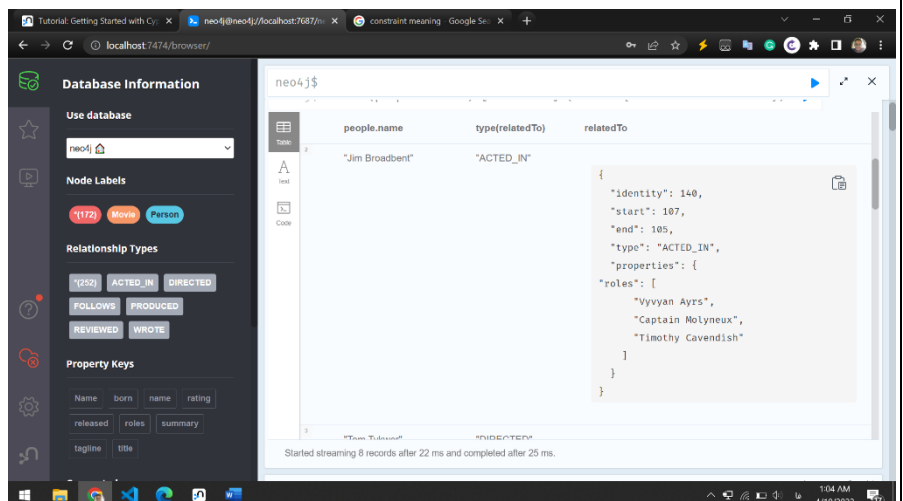


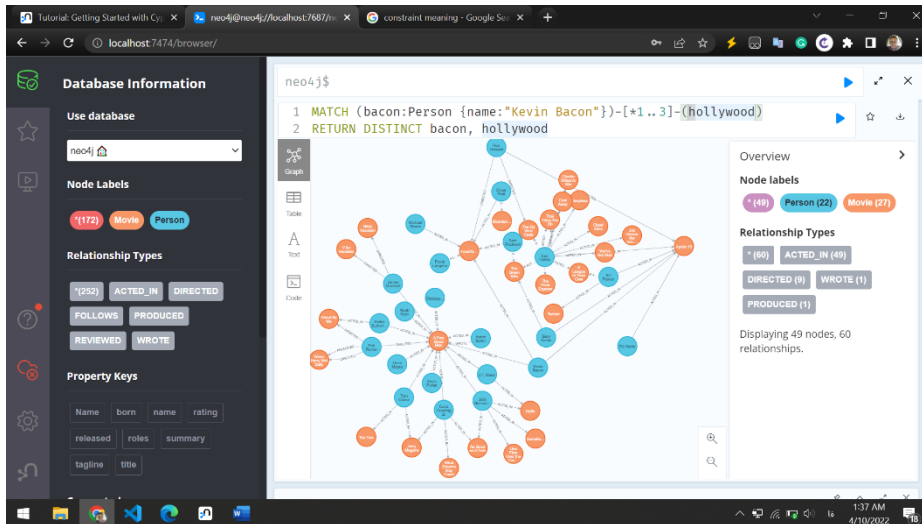
در این قسمت هم شبیه قسمت قبل یک رابطه انتخاب کردیم. ابتدا تمام کارگردان هایی که فیلمی با نام cloud atlas را کارگردانی کرده اند را پیدا کردیم و در نهایت نام آن کارگردان ها را نمایش داده ایم.



در این قسمت یک رابطه ی سه تایی را انتخاب کرده ایم. به این صورت که تمامی فیلم هایی که tom hanks در آن بازی کرده است را مشخص می کنیم. سپس تمام بازیگران مکمل آن فیلم ها را پیدا می کنیم. در نهایت خود tom hanks و فیلم هایی که بازی کرده و همچنین تمام بازیگران مکمل در شکل نشان داده شده است.

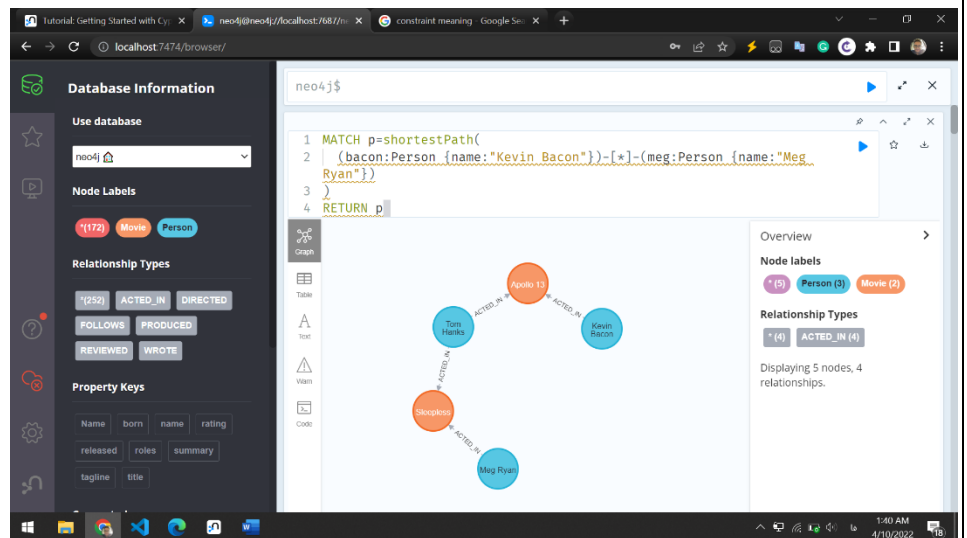
در این جا به دنبال رابطه ی پیچیده تری گشتیم. آن هم اینکه تمام کسانی که به نوعی با فیلم cloud atlas در رابطه بودند را به همراه نوع رابطه شان پیدا کردیم و در انتها چاپ کردیم. به نوعی تمامی دست اندرکاران این فیلم با نقششان مشخص شده اند.





در این بخش به دنبال رابطه ی پیچیده تری هستیم که شاید نتوان خارج از دیتابیس آن را توضیح داد. رابطه ای که به دنبال تمام افراد با فیلم هایی است که کمتر از ۴ فاصله به شخصی با نام kevin bacon مرتبط می شوند. از این جست و جو بیشتر در مدل دیتابیس های گراف محور می توان استفاده کرد.

در این قسمت می خواهیم پل ارتباطی بین دو بازیگر با نام های kevin bacon و meg ryan را پیدا کنیم. به این معنا که به فیلم یا بازیگر مشترکی برسیم که جفت این بازیگر ها به نوعی در ارتباط بوده اند. در شکل کوتاه ترین مسیر ارتباطی بین این دو قابل مشاهده است.



گام سوم. کوئری های جدید روی دیتابیس Movie

در این قسمت با توجه به دانسته های قبلی چند کوئری جدید به همراه پاسخ های آنها در گزارش آورده می شود.

- می خواهیم تمام همکاری بین نویسندگان و کارگردان ها را پیدا کنیم. به این منظور کوئری خاصی می زنیم تا پیدا کنیم هر نویسنده با هر کارگردان در چه فیلمی همکاری کرده اند.

The screenshot shows the Neo4j Browser interface. On the left, there are filters for Node Labels (Movie, Person) and Relationship Types (ACTED_IN, DIRECTED, etc.). The main area displays a Cypher query:

```
1 MATCH (a:Person),(b:Person),(c:Movie)
2 WHERE (a)-[:WROTE]→(c) AND (b)-[:DIRECTED]→(c)
3 RETURN (a.name), (b.name), (c.title)
```

The results are shown in a table with 7 rows:

	(a.name)	(b.name)	(c.title)
2	"Andy Wachowski"	"Lana Wachowski"	"Speed Racer"
3	"Andy Wachowski"	"James Marshall"	"V for Vendetta"
4	"Lana Wachowski"	"Andy Wachowski"	"Speed Racer"
5	"Lana Wachowski"	"Lana Wachowski"	"Speed Racer"
6	"Lana Wachowski"	"James Marshall"	"V for Vendetta"
7	"Aaron Sorkin"	"Rob Reiner"	"A Few Good Men"

- می خواهیم تمام افرادی که تجربه حضور در فیلمی با نمره ی بالای ۹ دارند را به همراه نقش آنها در آن فیلم پیدا کنیم. (امتیاز فیلم ها در دیتابیس وارد نشده و متأسفانه نتیجه ای نمیبینیم 😞)

The screenshot shows the Neo4j Browser interface. The Cypher query is:

```
1 MATCH (a:Person)-[role]-(b:Movie)
2 WHERE (b.rating > 9)
3 RETURN (a.name), (type(role)), (b.title)
```

The results show "(no changes, no records)". Below the query editor, there is a preview of the next query:

```
neo4j$ MATCH (a:Person),(b:Person),(c:Movie) WHERE (a)-[:WROTE]→(c) AND...
```

The preview table has columns (a.name), (b.name), and (c.title).

- می خواهیم تمام دست اندرکاران فیلم هایی که یک review با نمره بالا تر از ۹۰ را به خود اختصاص داده اند را به همراه نقششان ببینیم.

The screenshot shows the Neo4j Browser interface. On the left, there are panels for Node Labels (Movie, Person), Relationship Types (ACTED_IN, DIRECTED, etc.), Property Keys, and Connected as (neo4j). The main area displays a Cypher query and its results in a table.

```

1 MATCH (a:Person)-[jori:REVIEWED]-(b:Movie)-[relate]-(c:Person)
2 WHERE (jori.rating > 90)
3 RETURN (c.name), (type(relate)), (b.title), (jori.rating)

```

	(c.name)	(type(relate))	(b.title)	(jori.rating)
1	"Tom Hanks"	"ACTED_IN"	"Cloud Atlas"	95
2	"Jim Broadbent"	"ACTED_IN"	"Cloud Atlas"	95
3	"Tom Tykwer"	"DIRECTED"	"Cloud Atlas"	95
4	"Lana Wachowski"	"DIRECTED"	"Cloud Atlas"	95
5	"Halle Berry"	"ACTED_IN"	"Cloud Atlas"	95
6	"Hugo Weaving"	"ACTED_IN"	"Cloud Atlas"	95

- می خواهیم ۵ تا از همبازی های tom hanks بین سال های ۱۹۹۰ تا ۲۰۰۰ را پیدا کنیم و نام آنها، فیلمی که بازی کردند و سال انتشار آن فیلم را ببینیم.

The screenshot shows the Neo4j Browser interface. On the left, there are panels for Node Labels (Movie, Person), Relationship Types (ACTED_IN, DIRECTED, etc.), Property Keys, and Connected as (neo4j). The main area displays a Cypher query and its results in a table.

```

1 MATCH (a:Person {name:"Tom Hanks"})-[:ACTED_IN]-(b:Movie)
2 WHERE (1990 < b.released < 2000)
3 RETURN (c.name), (b.title), (b.released)
4 LIMIT 5

```

	(c.name)	(b.title)	(b.released)
1	"Parker Posey"	"You've Got Mail"	1998
2	"Greg Kinnear"	"You've Got Mail"	1998
3	"Meg Ryan"	"You've Got Mail"	1998
4	"Steve Zahn"	"You've Got Mail"	1998
5	"Dave Chappelle"	"You've Got Mail"	1998

- می خواهیم نام فیلم، نمره فیلم از نگاه کارشناس و خلاصه فیلم هایی که قبل از سال ۲۰۰۰ منتشر شده را مشاهده کنیم.

The screenshot shows the Neo4j web interface in a browser. The left sidebar contains navigation options like Node Labels, Relationship Types, and Property Keys. The main area displays a Cypher query and its results.

```

neo4j$
1 MATCH (a:Person)-[jori:REVIEWED]-(m:Movie)
2 WHERE (m.released<2000)
3 RETURN (m.title), (jori.rating), (jori.summary)
  
```

	(m.title)	(jori.rating)	(jori.summary)
1	"Unforgiven"	85	"Dark, but compelling"
2	"The Birdcage"	45	"Slapstick redeemed only by the Robin Williams and Gene Hackman's stellar performances"

Started streaming 2 records in less than 1 ms and completed in less than 1 ms.

```

neo4j$ MATCH (n) WHERE EXISTS(n.summary) RETURN DISTINCT "node" as entit...
  
```

	entity	summary
--	--------	---------

مشکلات و توضیحات تکمیلی

مشکلی به وجود نیامد.

آنچه آموختم / پیشنهادات

آشنایی نسبتاً کاملی با دیتابیس های گراف محور پیدا کردیم و به صورت عملی با آنها کار کردیم. چون آشنایی قبلی ای با این گونه دیتابیس ها نداشتم کاملاً برایم جدید و بسیار جالب بود. به نظر آزمایش خیلی مفید و پر بازدهی بود.