

AIM: WORKING WITH Neo4j DATABASE**➤ What is Neo4j?**

Neo4j is a graph database management system that stores and manages data as nodes, edges, and properties.

➤ Uses

- Neo4j is used in a wide range of applications, such as social networking, recommendation engines, fraud detection, and supply chain management.
- It is particularly useful for applications that involve complex relationships between data points.

➤ Pros:

- Neo4j's strength is its ability to query complex relationships and quickly traverse large datasets.
- It is a highly scalable database system and can handle large amounts of data with ease.
- Its query language, Cypher, is easy to learn and use.

➤ Cons:

- Neo4j can be more complex to set up and manage compared to traditional relational databases.
- It may not be suitable for applications that require heavy write operations.

➤ Installation:

- set NEO4J_HOME=C:\neo4j-community-3.5.15
- set PATH=C:\neo4j-community-3.5.15\bin;%PATH%
- set NEO4J_HOME= C:\Program Files\Java\jdk-11.0.10
- set PATH= C:\Program Files\Java\jdk-11.0.10\bin;%PATH%
- neo4j.bat install-service
- neo4j.bat start

```
C:\Windows\system32\cmd.exe
C:\>set PATH=C:\neo4j-community-4.2.3\bin;%PATH%
'et' is not recognized as an internal or external command,
operable program or batch file.

C:\>set PATH=C:\neo4j-community-4.2.3\bin;%PATH%

C:\>set java_HOME=C:\Program Files\Java\jdk-11.0.10
Environment variable java_HOME-C:\Program Files\Java\jdk-11.0.10 not defined

C:\>set JAVA_HOME=C:\Program Files\Java\jdk-11.0.10
Environment variable JAVA_HOME-C:\Program Files\Java\jdk-11.0.10 not defined

C:\>set JAVA_HOME=C:\Program Files\Java\jdk-11.0.10
Environment variable JAVA_HOME-C:\Program Files\Java\jdk-11.0.10 not defined

C:\>set JAVA_HOME=C:\Program Files\Java\jdk-11.0.10

C:\>set PATH=C:\Program Files\Java\jdk-11.0.10\bin;%PATH%

C:\>neo4j.bat
Usage: neo4j { console | start | stop | restart | status | install-service | uninstall-service | update-service } < -Verbose >

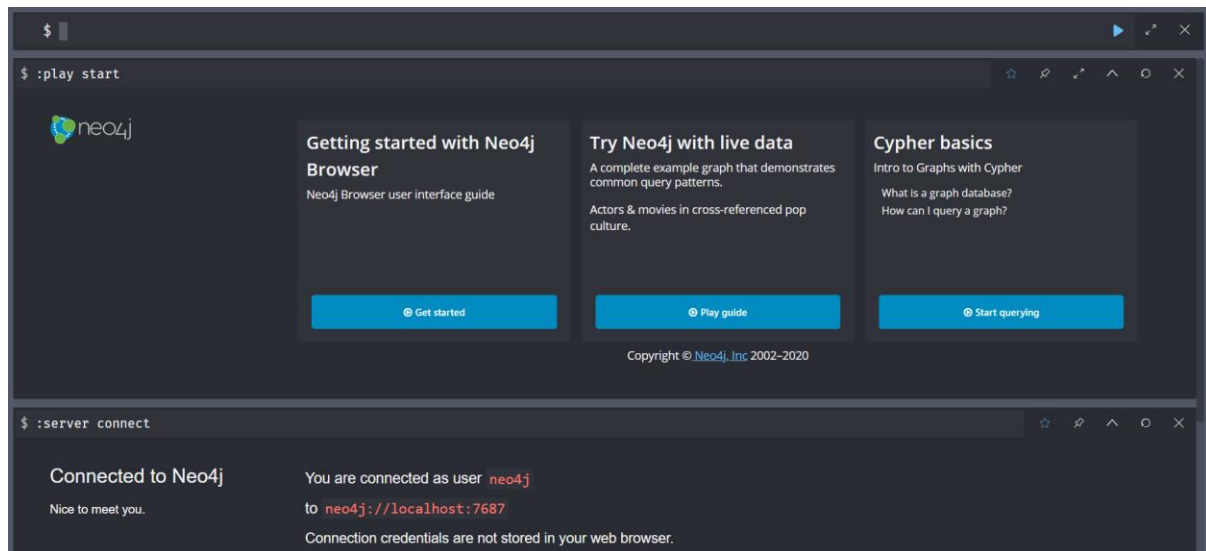
C:\>neo4j.bat install-service
Neo4j service installed

C:\>neo4j.bat start
Neo4j service started

C:\>
```

Type in browser:

<http://localhost:7474>



1. Create 7 nodes in people label having properties name, department, address, age.

```
CREATE (:people {name: "Guru", department: "IT", address: "1234 Main Street, Mumbai, India", age: 22}),
```

```
(:people {name: "Ansh", department: "CE", address: "5678 Second Street, New Delhi, India", age: 20}),
```

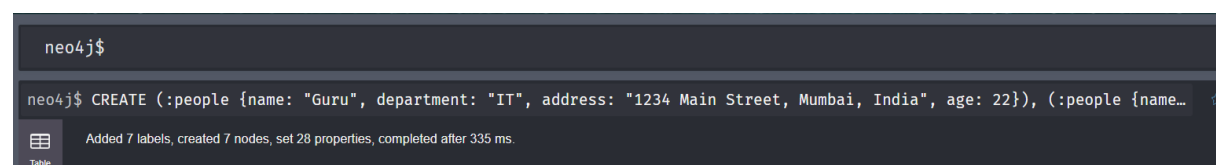
```
(:people {name: "Aman", department: "IT", address: "910 Third Street, Bangalore, India", age: 21}),
```

```
(:people {name: "Dhruv", department: "CE", address: "1212 Fourth Street, Kolkata, India", age: 19}),
```

```
(:people {name: "Aryan", department: "IT", address: "3434 Fifth Street, Pune, India", age: 20}),
```

```
(:people {name: "Avi", department: "CE", address: "5656 Sixth Street, Chennai, India", age: 20}),
```

```
(:people {name: "Het", department: "IT", address: "7878 Seventh Street, Hyderabad, India", age: 21})
```



2. create 7 nodes in film label having properties title, releasedyear,directorname.

```
CREATE (:film {title: "3 Idiots", releasedyear: 2009, directorname: "Rajkumar Hirani"}),
```

```
(:film {title: "Lagaan", releasedyear: 2001, directorname: "Ashutosh
Gowariker"}),
(:film {title: "Taare Zameen Par", releasedyear: 2007, directorname: "Aamir
Khan"}),
(:film {title: "Dilwale Dulhania Le Jayenge", releasedyear: 1995, directorname:
"Aditya Chopra"}),
(:film {title: "Kabhi Khushi Kabhie Gham", releasedyear: 2001, directorname:
"Karan Johar"}),
(:film {title: "PK", releasedyear: 2014, directorname: "Rajkumar Hirani"}),
(:film {title: "Bahubali: The Beginning", releasedyear: 2015, directorname: "S. S.
Rajamouli"})
```

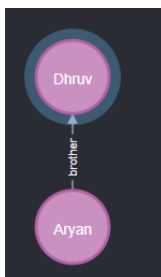
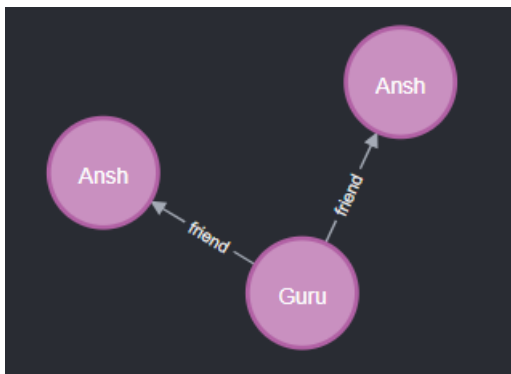
```
CREATE (:film {title: "3 Idiots", releasedyear: 2009, directorname: "Rajkumar Hirani"}),
Added 7 labels, created 7 nodes, set 21 properties, completed after 10 ms.
```

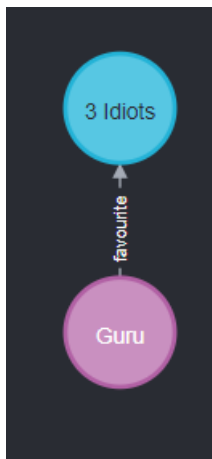
3. Create relationships called friend, brother, and favourite nodes.

```
match (a088:people {name:"Guru"}),(b088:people {name:"Ansh"}) merge (a088)-
[r:friend]->(b088) return a088,b088,r
```

```
MATCH (a088:people {name: "Aryan"}), (b088:people {name: "Dhruv"})MERGE
(a088)-[r:brother]->(b088)RETURN a088, b088, r
```

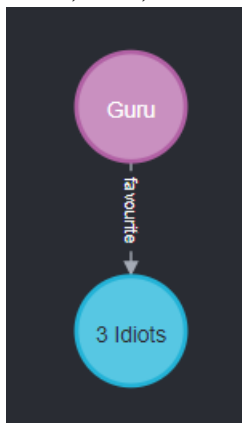
```
MATCH (a088:people {name: "Guru"}), (b088:film {title: "3 Idiots"})MERGE
(a088)-[r:favourite]->(b088)RETURN a088, b088, r
```





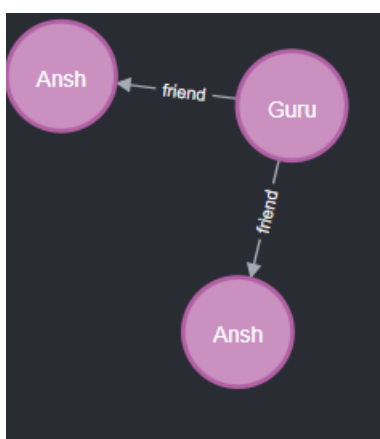
4. What are the favourite movies of Aman

```
MATCH (a088:people {name: "Guru"})-[r:favourite]->(b088:film)RETURN  
b088,a088,r
```



5. Who are friends of Aman

```
MATCH (a088:people {name: "Guru"})-[r:friend]-(b088:people)RETURN  
b088,a088,r
```



6. Add a property called mobile_no for person Aman and Ansh

```
MATCH (a088:people {name: "Guru"})SET a088.mobile_no =
"93156325646"RETURN a088.mobile_no
```

```
MATCH (a088:people {name: "Ansh"})SET a088.mobile_no =
"1233157890"RETURN a088.mobile_no
```

```
j$ MATCH (a088:people {name: "Guru"})
```

	a088.mobile_no
1	"93156325646"

```
neo4j$ MATCH (a088:people {name: "Ansh"})SET a088.mobile_no = "1233157890"RETURN a088.mobile_no
```

	a088.mobile_no
1	"1233157890"

7. Remove mobile _ no property for person Aman

```
MATCH (a088:people {name: "Guru"})REMOVE a088.mobile_no RETURN a088
```



8. Delete a relationship called brother

```
MATCH (a088:people {name: "Aryan"})-[r:brother]-(b088:people {name:
"Dhruv"})DELETE r
```

```
MATCH (a088:people {name: "Aryan"})-[r:brother]-(b088:people {name: "Dhruv"}) DELETE r
```

Deleted 1 relationship, completed after 3 ms.

9. Delete a node having age 23

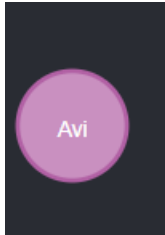
```
MATCH (a088:people {age: 23})DELETE a088
```

```
neo4j$ MATCH (a088:people {age: 23})DELETE a088
```

	(no changes, no records)
--	--------------------------

10. Display details of node whose name is Avi and age is 23

```
MATCH (a088:people {name: "Avi", age: 20})RETURN a088
```

**11. display details of node having name Avi or Aman**

match (a088:people) where a088.name="Avi" or a088.name="Guru" return a088

