

TY B.Tech. (CSE) – II [2022-23]

5CS372: Advanced Database System Lab.

Assignment No. 12

Name : Tanaya Mukund Bhide

PRN : 2020BTECS00011

Batch : T5

Branch: T.Y CSE

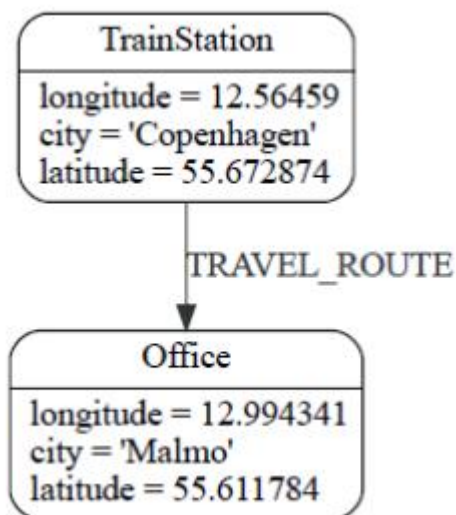
Spatial and Geographic Data

Geospatial is the natural domain for Graph Database Use Neo4j and Neo4j Spatial

Problem Statement : Finding Things Close to Other Things.

Task :

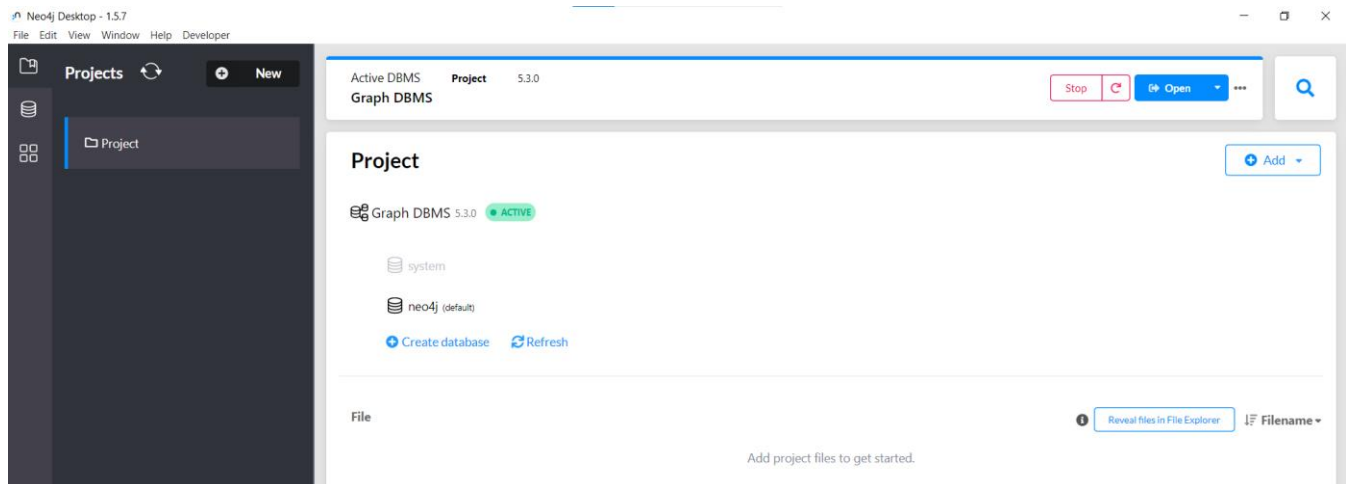
1. Use Neo4j graph database installed in previous assignments.
2. Install/configure Neo4jSpatial (<https://github.com/neo4j-contrib/spatial>) from GitHub. It is the Neo4j plug-in that facilitates geospatial operations on data stored in Neo4j.
3. Write CQL (Cypher Query Language) script to add randomly 10,000 location points as follows. Assume any data.



4. Use the point() , distance() function of Neo4j to answer the queries “which things close/nearest to which other things”.
5. Demonstrate the result by firing different cypher queries (write CQL statement Application in : location-based services on the web

Task

1:

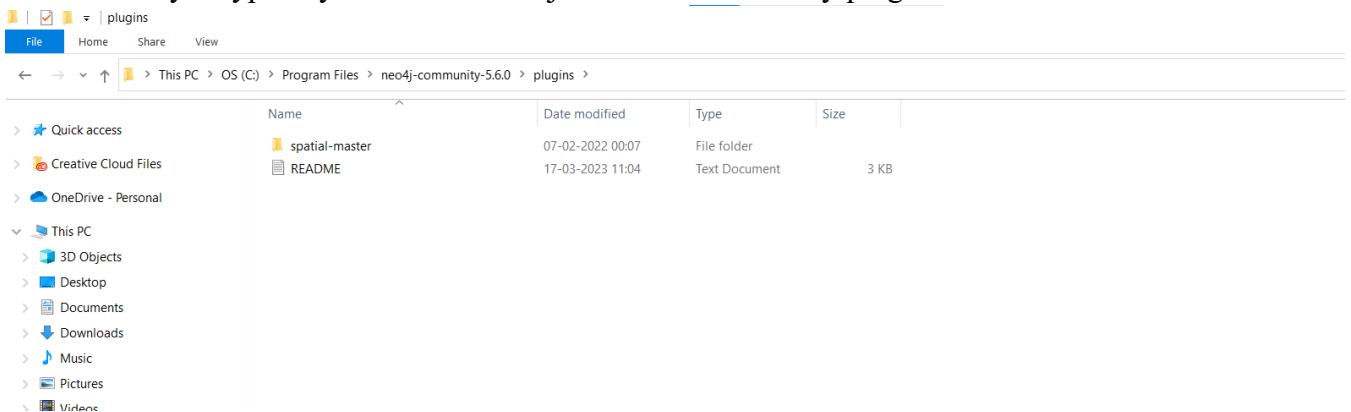


2. Configuration of the Neospatial master

- Download Neo4jSpatial from GitHub at <https://github.com/neo4j-contrib/spatial/releases>.
- You can download the latest release or a specific version that you want to use.
- Extract the downloaded file to a directory of your choice.

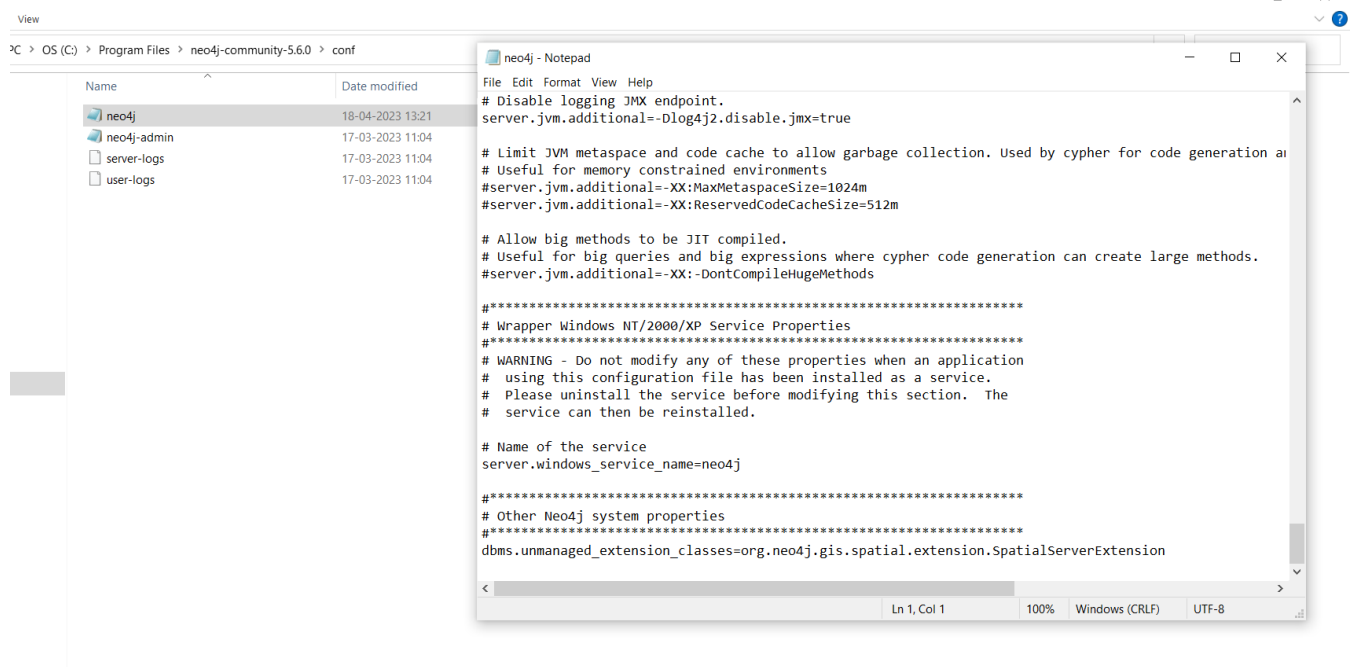


- Copy the extracted "neo4j-spatial-" directory to the "plugins" directory of your Neo4j installation.
- This directory is typically located at "neo4j-installation-directory/plugins/".



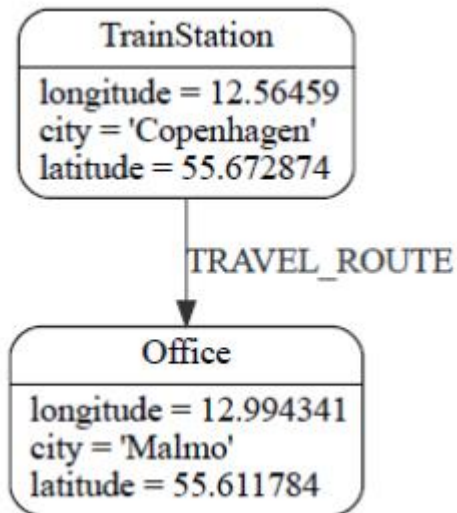
- Open the "neo4j.conf" file located in the "conf" directory of your Neo4j installation.

- Add the following line to the end of the file:
- Copy code
- `dbms.unmanaged_extension_classes=org.neo4j.gis.spatial.extension.SpatialServerExtension`
- This line enables the `SpatialServerExtension`, which provides the REST API for Neo4jSpatial.
- Save and close the "neo4j.conf" file.



- Restart Neo4j to apply the changes.

3. Write CQL (Cypher Query Language) script to add randomly 10,000 location points as follows. Assume any data.



Cypher script :

```
// Create 10,000 Train Station nodes
```

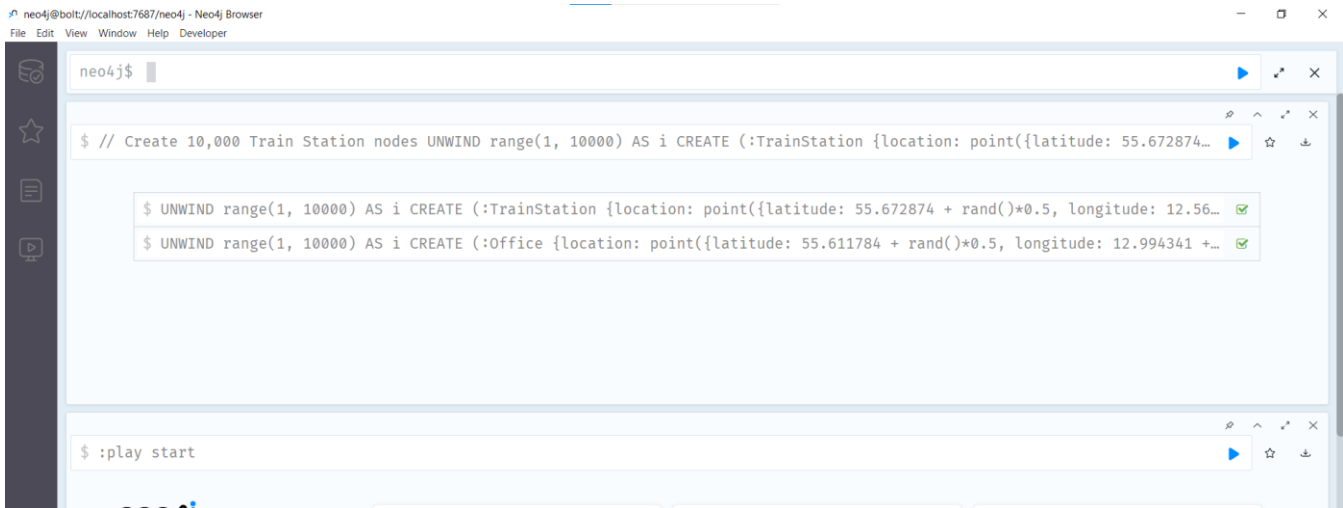
```
UNWIND range(1, 10000) AS i
```

```
CREATE (:TrainStation {location: point({latitude: 55.672874 + rand()*0.5, longitude: 12.56459 + rand()*0.5}), city: 'Copenhagen'});
```

```
// Create 10,000 Office nodes
```

```
UNWIND range(1, 10000) AS i
```

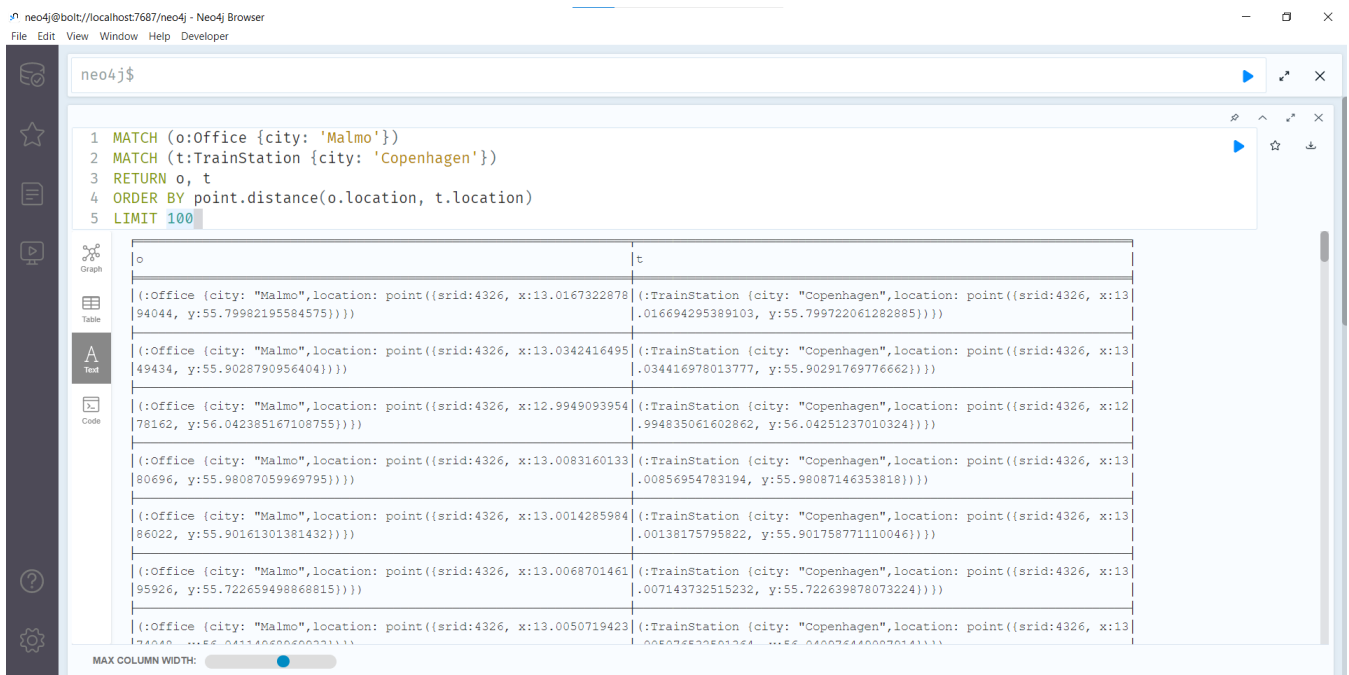
```
CREATE (:Office {location: point({latitude: 55.611784 + rand()*0.5, longitude: 12.994341 + rand()*0.5}), city: 'Malmo'});
```



4. Use the point() , distance() function of Neo4j to answer the queries “which things close/nearest to which other things”.

5. Demonstrate the result by firing different cypher queries (write CQL statement)

1. Find the nearest train station to each office in Malmo:



2. Find the closest Office to each Train Station in Copenhagen:

neo4j@bolt://localhost:7687/neo4j - Neo4j Browser

File Edit View Window Help Developer

neo4j\$

```
1 MATCH (t:TrainStation {city: 'Copenhagen'})
2 MATCH (o:Office {city: 'Malmo'})
3 RETURN t, o
4 ORDER BY point.distance(t.location, o.location)
5 LIMIT 50
6
```

t	o
(:TrainStation {city: "Copenhagen",location: point((srid:4326, x:13.016694295389103, y:55.799722061282885))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.016732287894044, y:55.79982195584575))})
(:TrainStation {city: "Copenhagen",location: point((srid:4326, x:13.034416978013777, y:55.90291769776662))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.034241649549434, y:55.9028790956404))})
(:TrainStation {city: "Copenhagen",location: point((srid:4326, x:12.994835061602862, y:56.04251237010324))})	(:Office {city: "Malmo",location: point((srid:4326, x:12.994909395478162, y:56.042385167108755))})
(:TrainStation {city: "Copenhagen",location: point((srid:4326, x:13.00856954783194, y:55.98087146353818))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.008316013380696, y:55.98087059969795))})
(:TrainStation {city: "Copenhagen",location: point((srid:4326, x:13.00138175795822, y:55.901758771110046))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.001428598486022, y:55.90161301381432))})
(:TrainStation {city: "Copenhagen",location: point((srid:4326, x:13.007143732515232, y:55.722639878073224))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.006870146195926, y:55.722659498868815))})
(:TrainStation {city: "Copenhagen",location: point((srid:4326, x:13.0015076593201324, y:55.040036488087034333))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.005071942374018, y:55.0411465688087333))})

MAX COLUMN WIDTH:

3. Find the closest Office to each other Office in Malmo.

neo4j@bolt://localhost:7687/neo4j - Neo4j Browser

File Edit View Window Help Developer

neo4j\$

```
1
2 MATCH (o1:Office {city: 'Malmo'})
3 MATCH (o2:Office {city: 'Malmo'})
4 WHERE o1 < o2
5 RETURN o1, o2
6 ORDER BY point.distance(o1.location, o2.location)
7 LIMIT 50
8
```

o1	o2
(:Office {city: "Malmo",location: point((srid:4326, x:13.267689979122569, y:55.87239031172561))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.267688040111882, y:55.87240726594767))})
(:Office {city: "Malmo",location: point((srid:4326, x:13.267688040111882, y:55.87240726594767))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.267689979122569, y:55.87239031172561))})
(:Office {city: "Malmo",location: point((srid:4326, x:12.995960648606786, y:55.89952577914258))})	(:Office {city: "Malmo",location: point((srid:4326, x:12.995889612905907, y:55.89955437009714))})
(:Office {city: "Malmo",location: point((srid:4326, x:12.995889612905907, y:55.89955437009714))})	(:Office {city: "Malmo",location: point((srid:4326, x:12.995960648606786, y:55.89952577914258))})
(:Office {city: "Malmo",location: point((srid:4326, x:13.321071735155705, y:55.71544058191535))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.320983646951458, y:55.71544114334308))})
(:Office {city: "Malmo",location: point((srid:4326, x:13.320983646951458, y:55.71544114334308))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.321071735155705, y:55.71544058191535))})
(:Office {city: "Malmo",location: point((srid:4326, x:13.4903765583352152, y:55.7000265020070111))})	(:Office {city: "Malmo",location: point((srid:4326, x:13.490295977732, y:55.70001505226070111))})

MAX COLUMN WIDTH: 100

