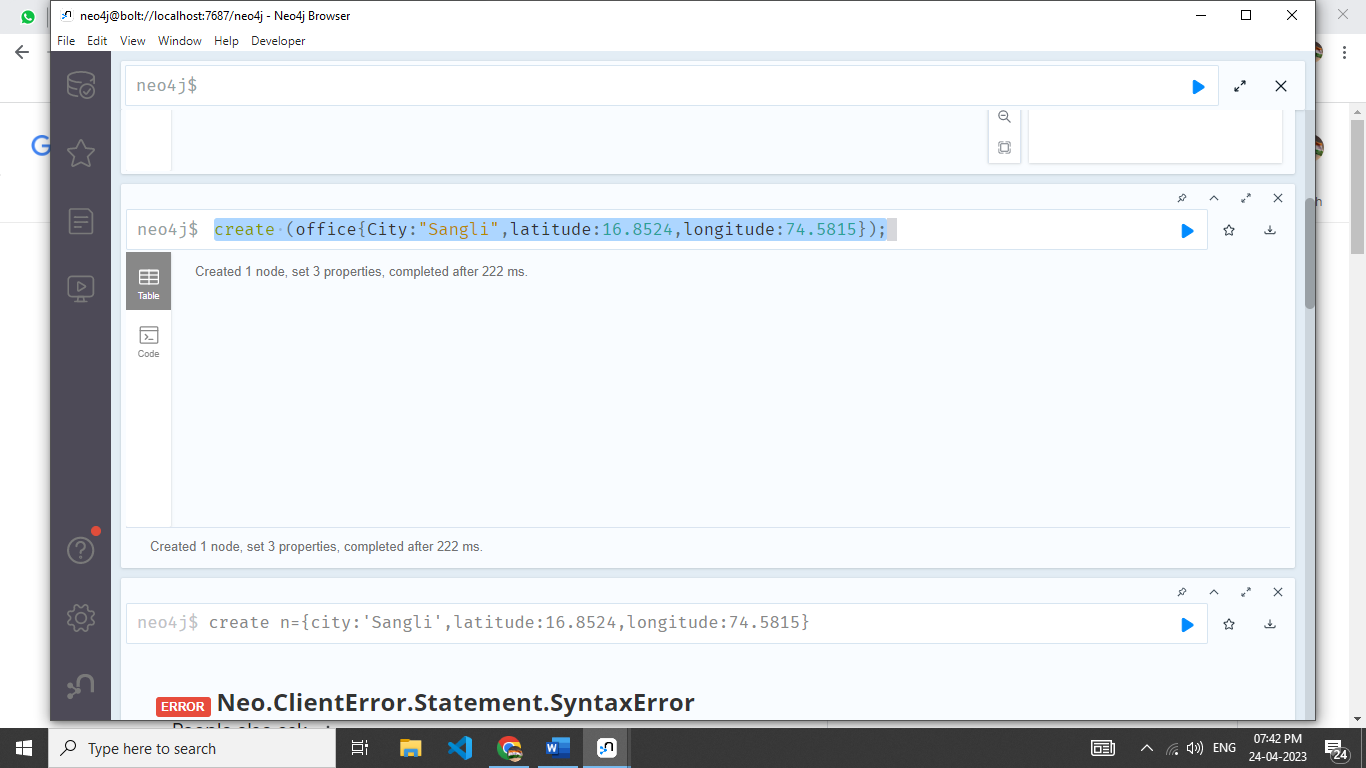
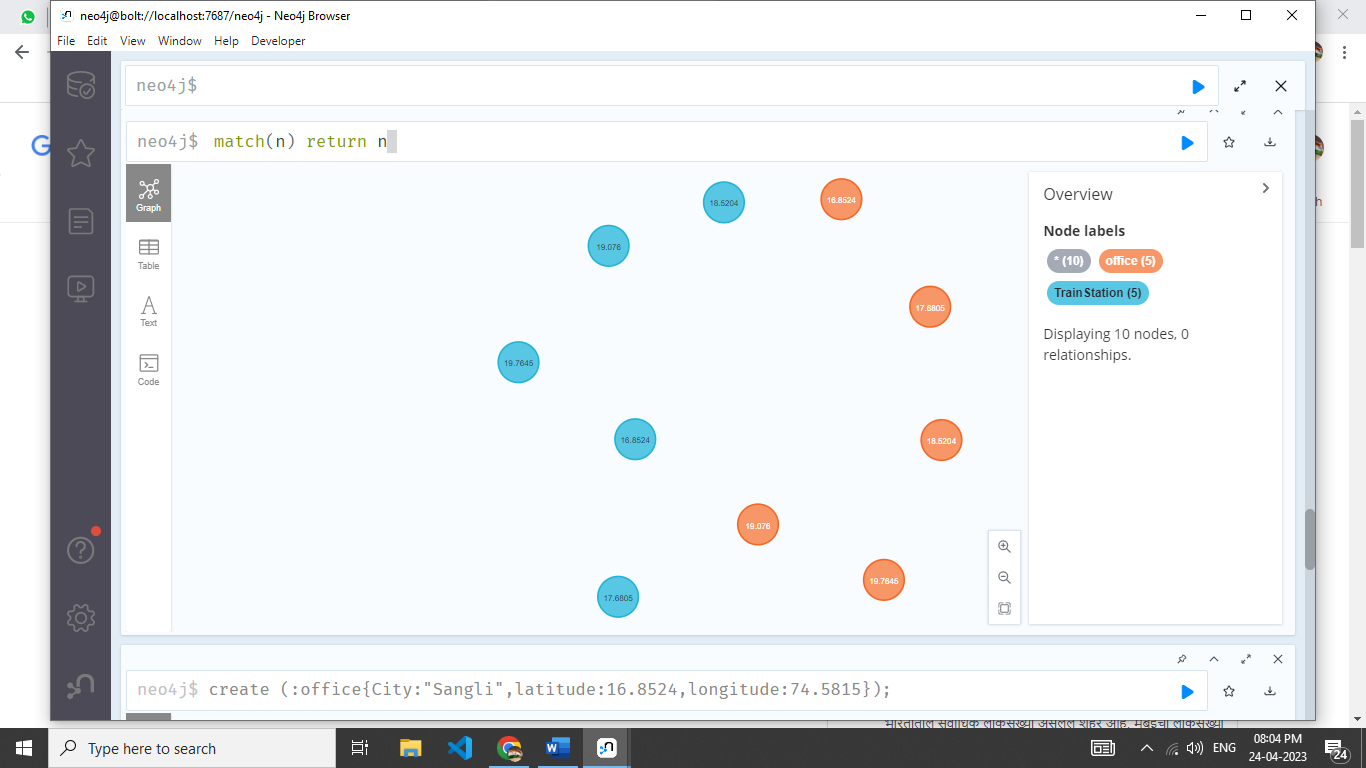
***Advanced Database System Lab***

**Assignment no. 12**

**PRN:** 2020BTECS00005

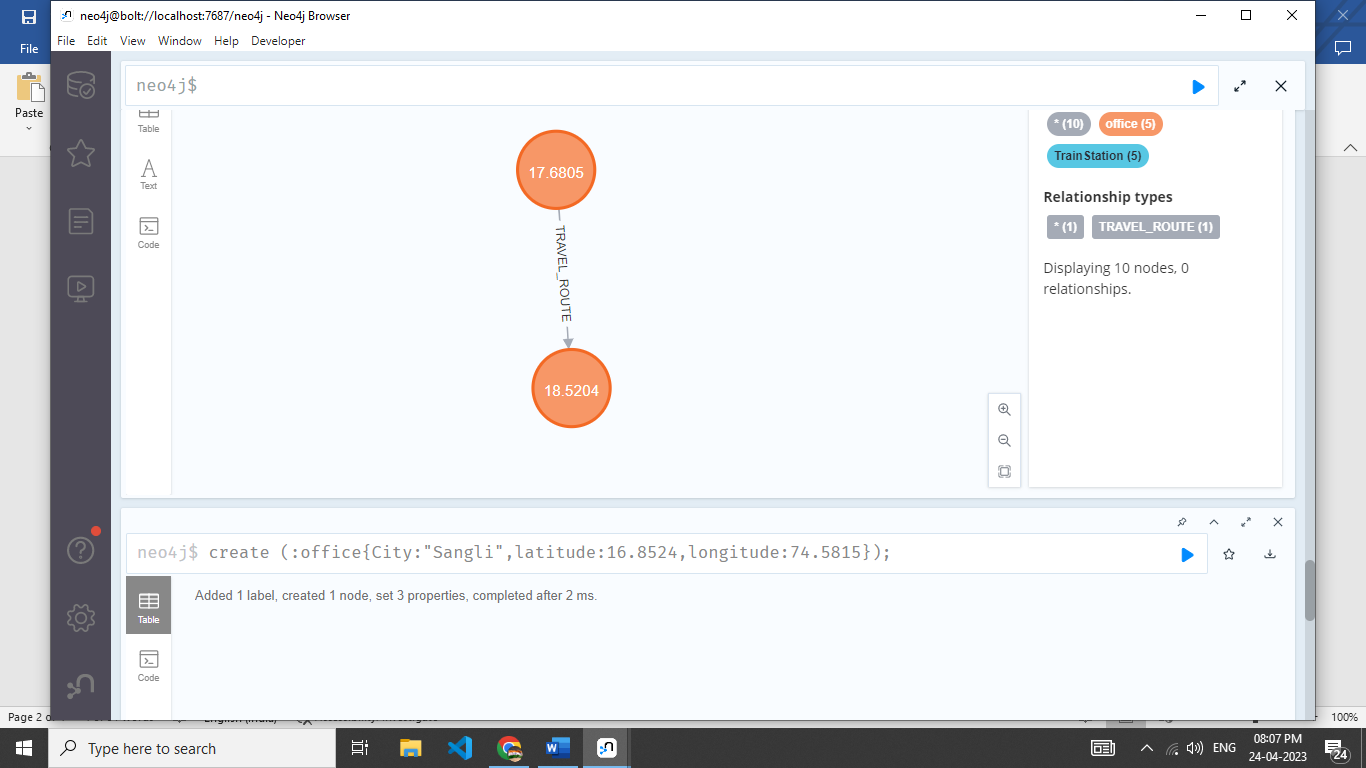
**Name:** Sanket Shivaji Jadhav

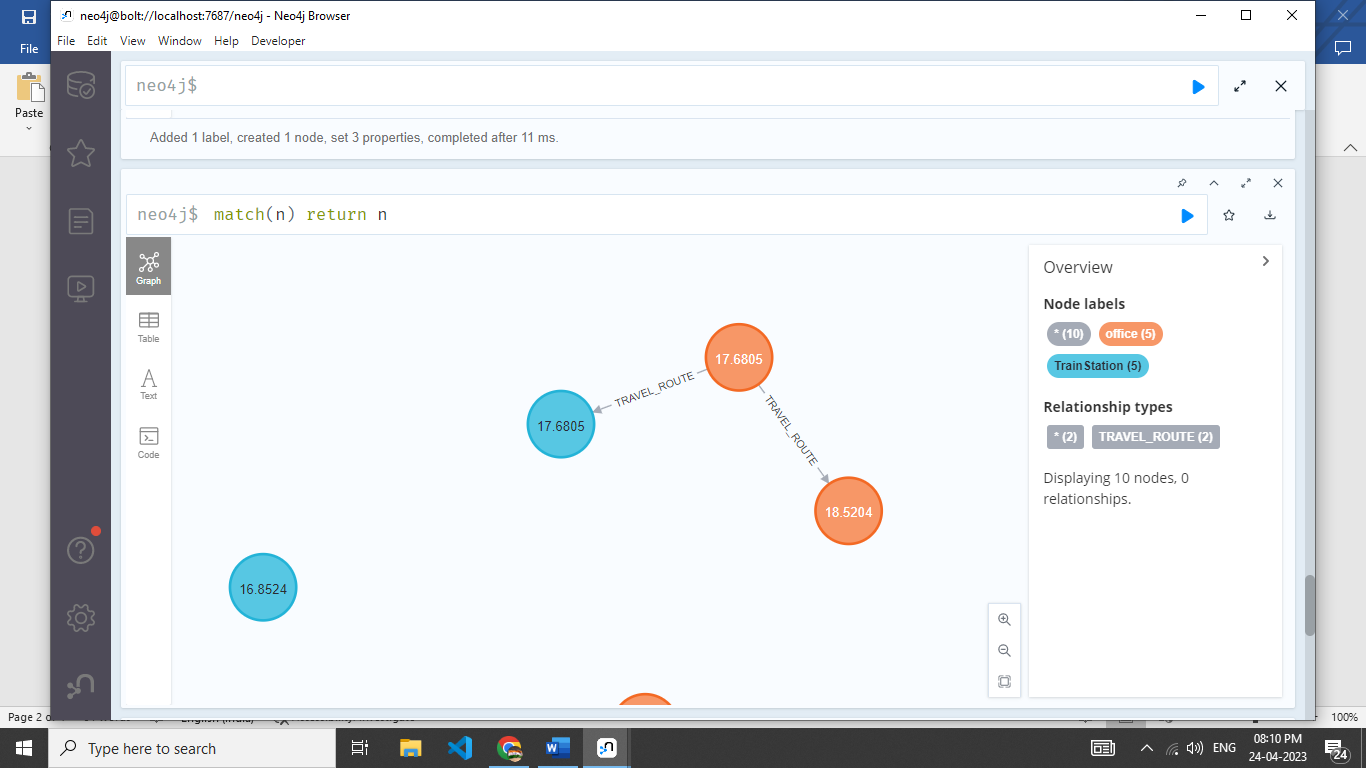




match  (l1:office {city:"Satara"}),(l2:office {city:"Pune "})

create (l1)-[:TRAVEL\_ROUTE]->(l2)



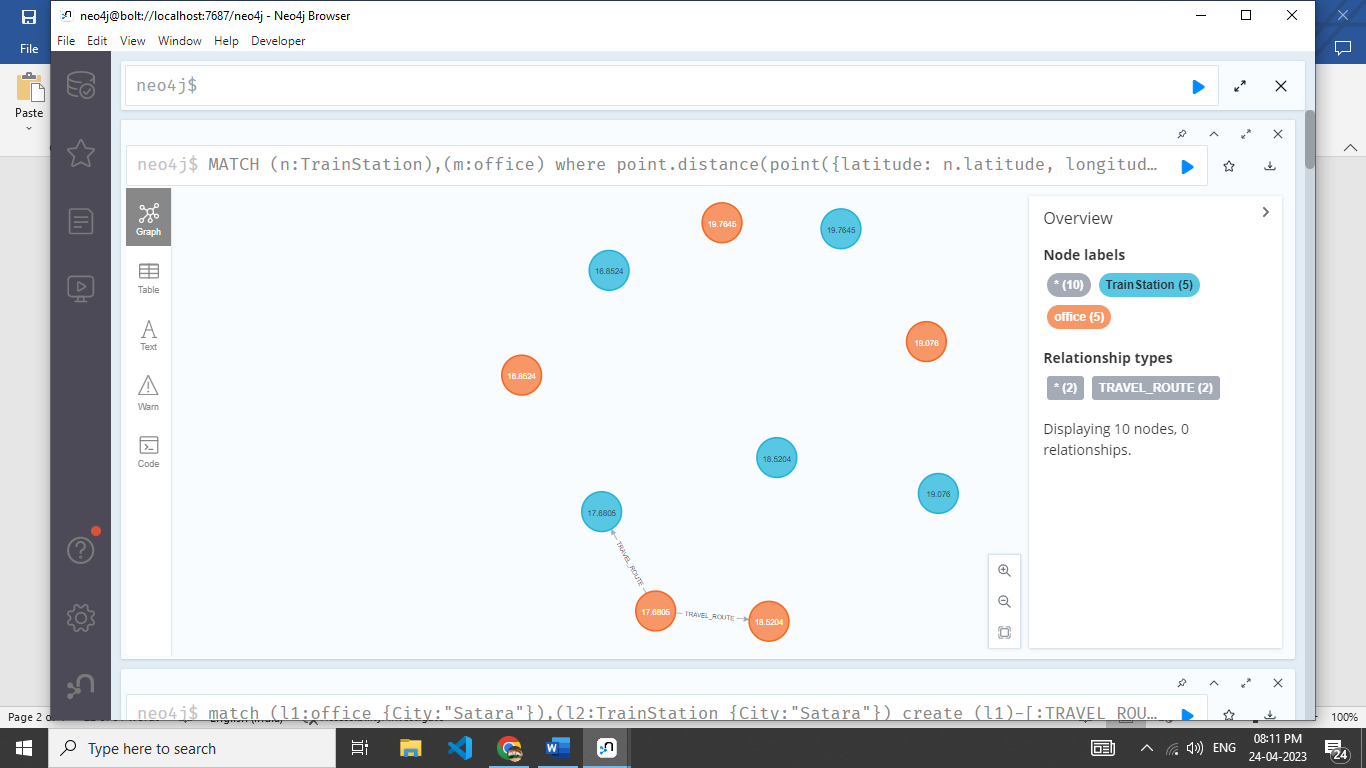


// Nearest distance with/without routes

MATCH (n:TrainStation),(m:Office)

where point.distance(point({latitude: n.latitude, longitude: n.longitude}), point({latitude: m.latitude, longitude: m.longitude})) < 1000

RETURN n,m,point.distance(point({latitude: n.latitude, longitude: n.longitude}), point({latitude: m.latitude, longitude: m.longitude}))

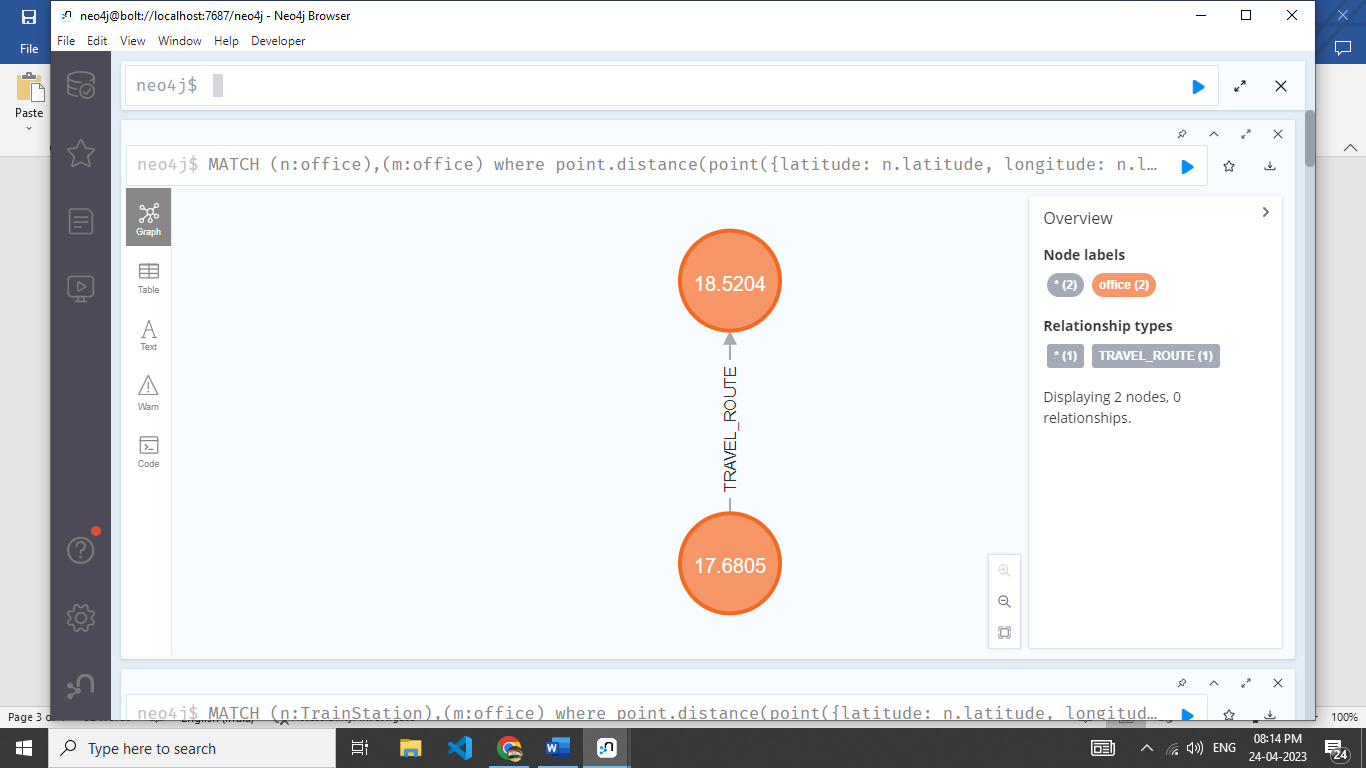


// Nearest distance with routes

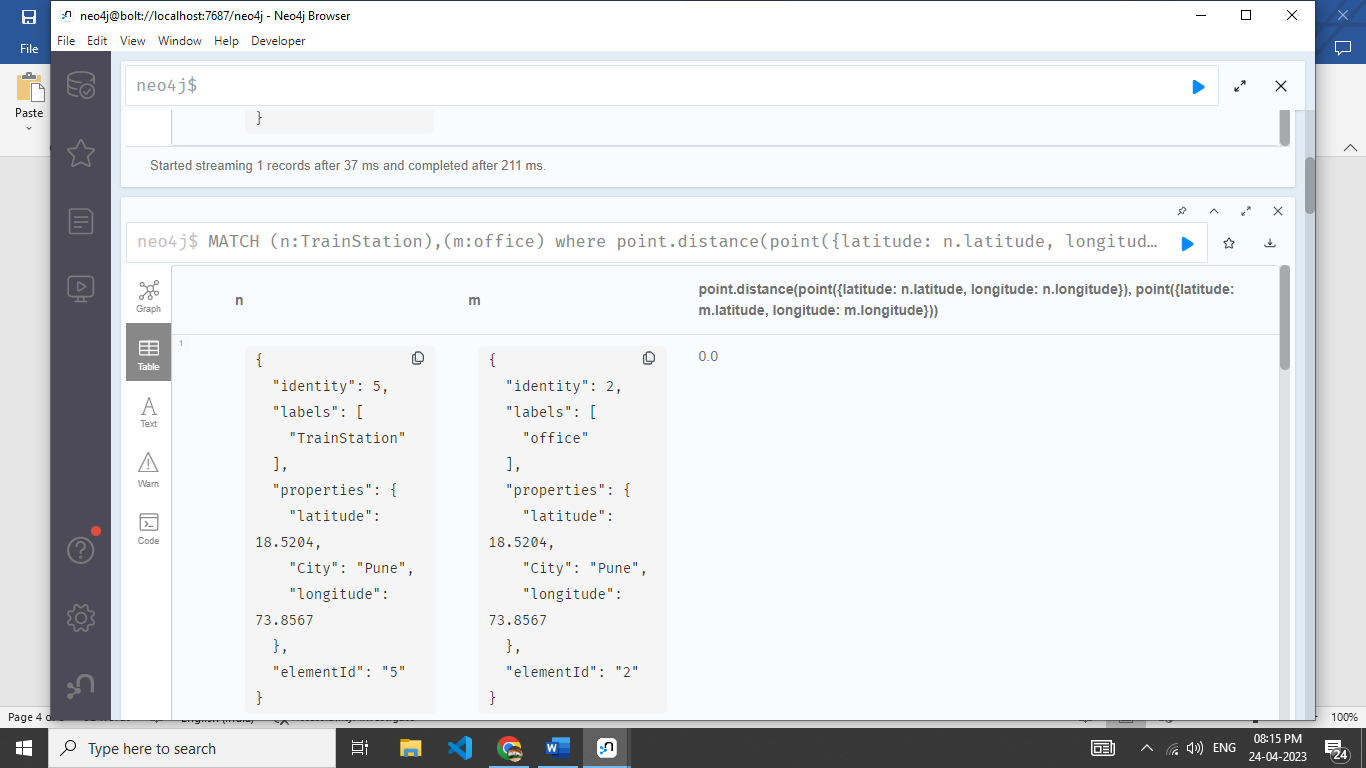
MATCH (n:Office),(m:Office)

where point.distance(point({latitude: n.latitude, longitude: n.longitude}), point({latitude: m.latitude, longitude: m.longitude})) < 1000000 and (n)-[\*]->(m)

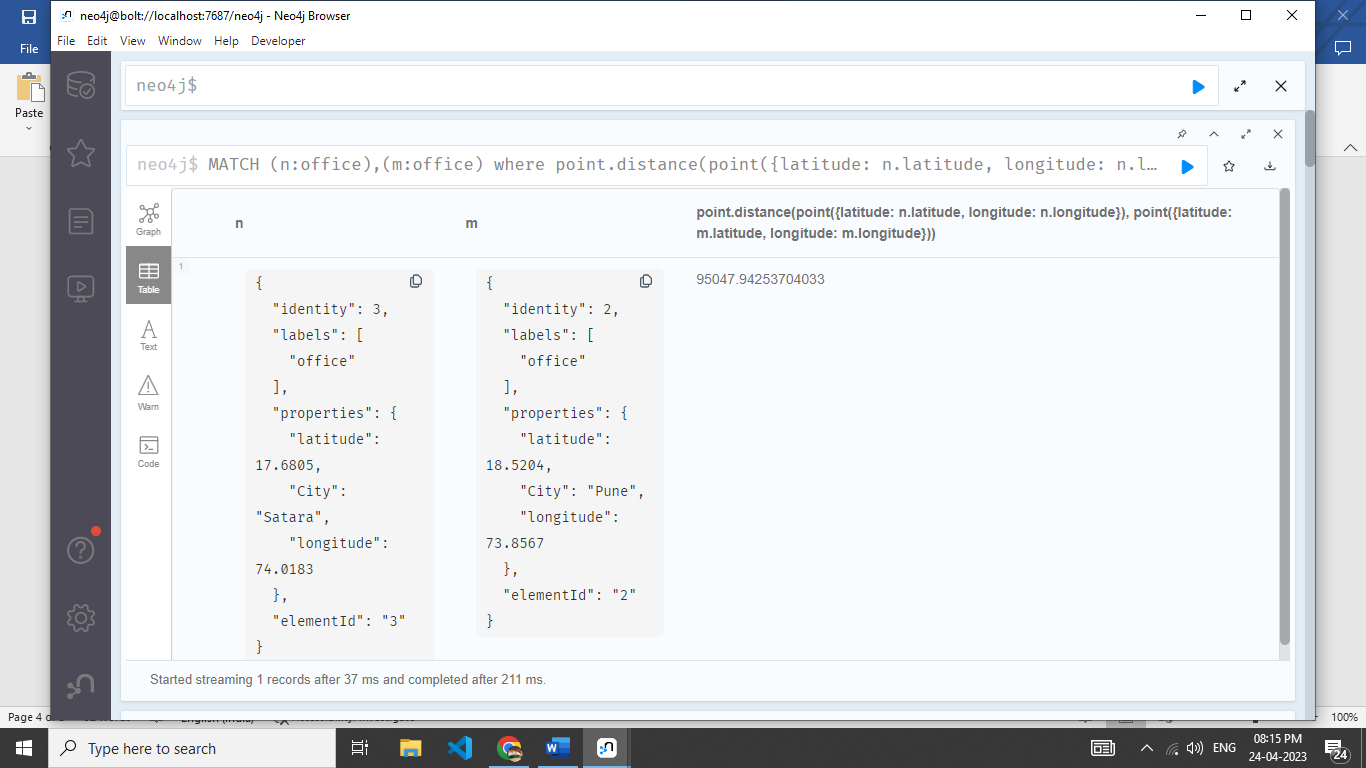
RETURN n,m,point.distance(point({latitude: n.latitude, longitude: n.longitude}), point({latitude: m.latitude, longitude: m.longitude}))



**Nearest trainstation and office with or without travel routes**

****

**Nearest office to office with travel routes**

****