Date:14.04.2022

**Third Year B. Tech., Sem VI 2021-22**

**4CS372 : Advanced Database System Lab**

**Assignment Submission**

**PRN No: 2019BTECS00064**

**Full name: Kunal Santosh Kadam**

**Batch: T2**

**Assignment: 11 A**

**Title of assignment: Demonstrate Neo4j Clustering**

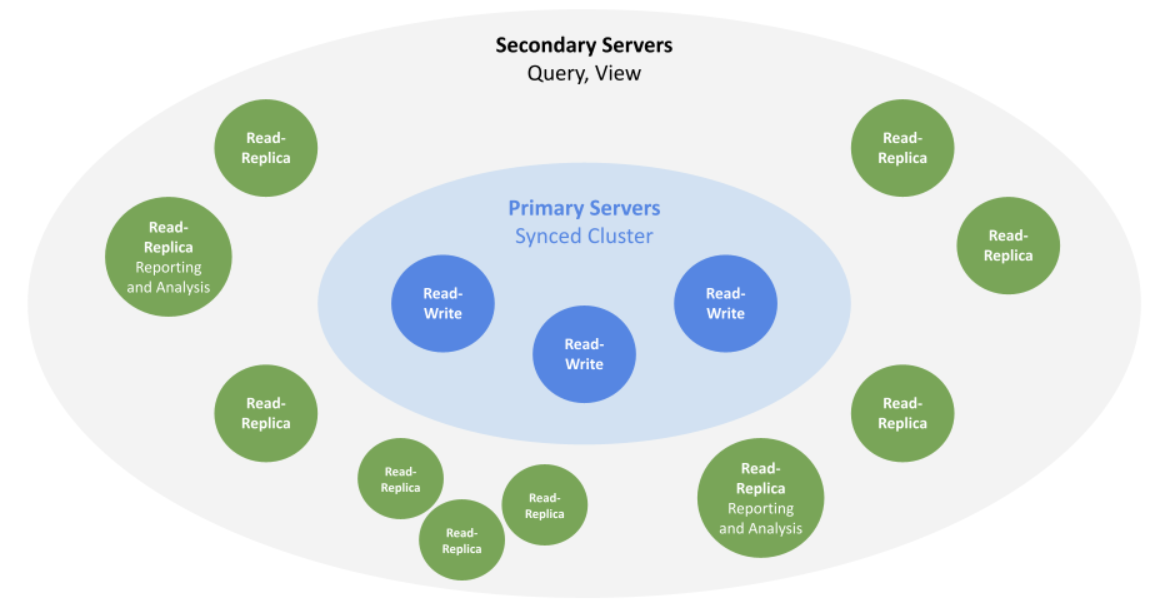
**Objective:**

1. Setup a multi-node Neo4j Cluster (Take 3 machine in lab or on single machine).
2. Deploy “Northwind” example database in above cluster.
3. Design a python GUI client to perform CRUD operations on “Customers” table of “Northwind” database. Demonstrate by connecting this client to any one node and automatically updating the data in other nodes.

**Introduction & Theory:**

Neo4j Casual Clustering provides three main features:

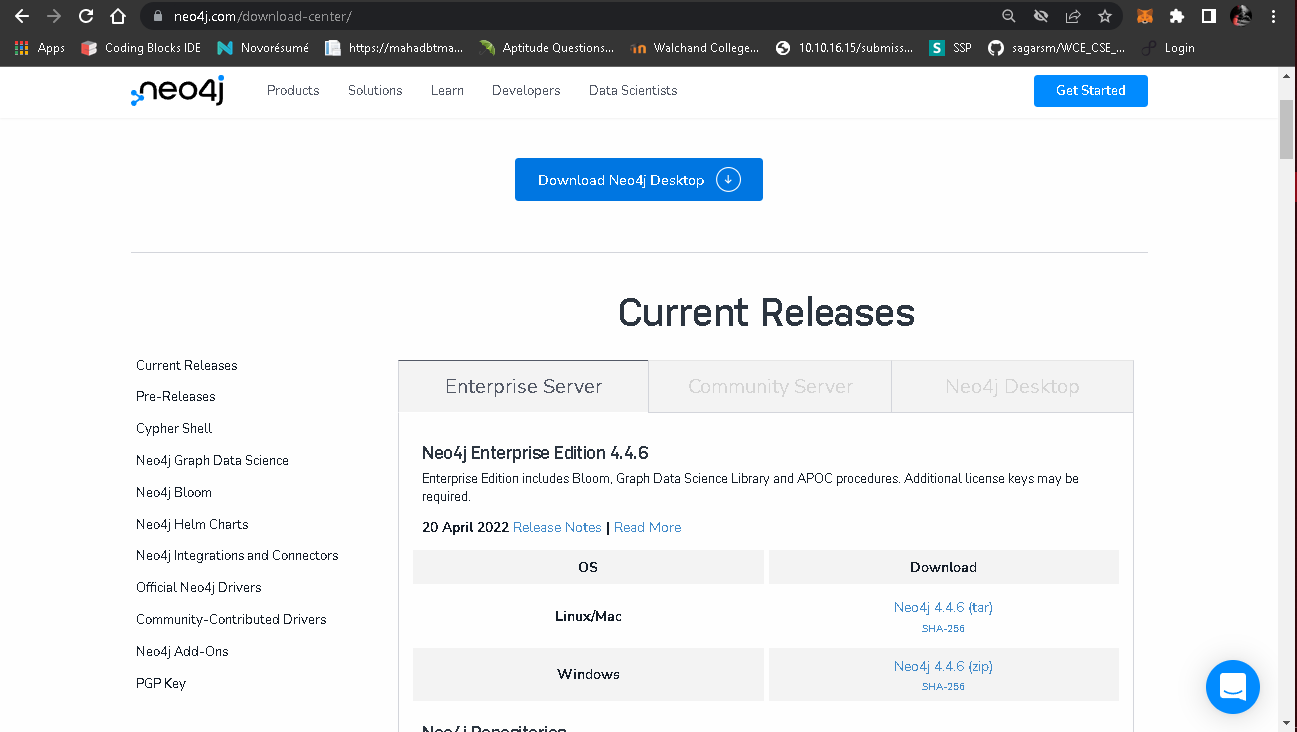
1. Safety: Core Servers provide a fault tolerant platform for transaction processing which will remain available while a simple majority of those Core Servers are functioning.
2. Scale: Read Replicas provide a massively scalable platform for graph queries that enables very large graph workloads to be executed in a widely distributed topology.
3. Casual consistency: When invoked, a client application is guaranteed to read at least its own writes. From an operational point of view, it is useful to view the cluster as being composed of servers with two different roles, referred to as Primary and Secondary servers.

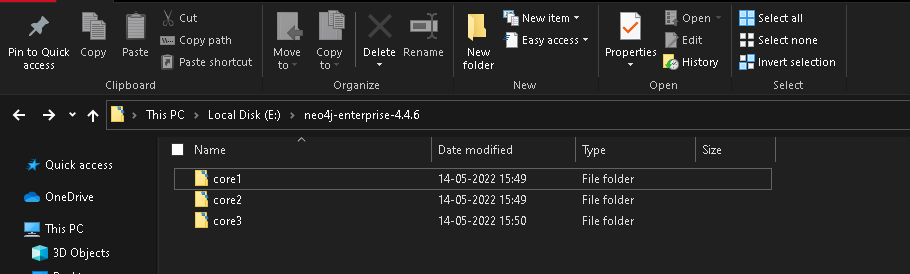


The two roles are foundational in any production deployment but are managed at different scales from one another and undertake different roles in managing the fault tolerance and scalability of the overall cluster.

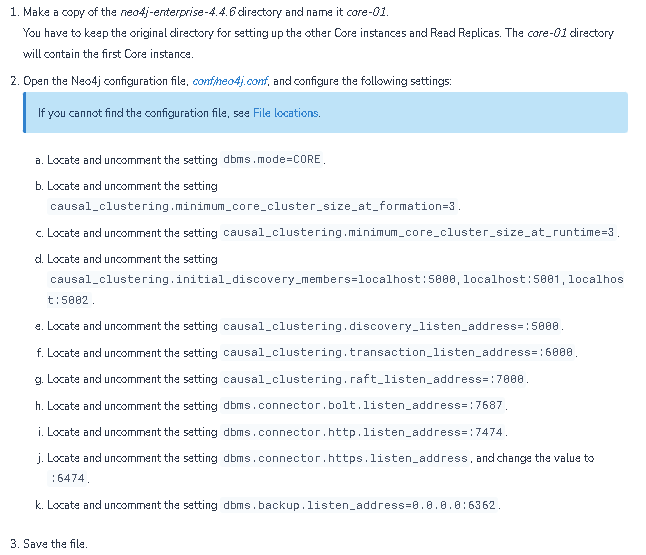
**Neo4j Local Clustering:**

First download neo4j enterprise edition.

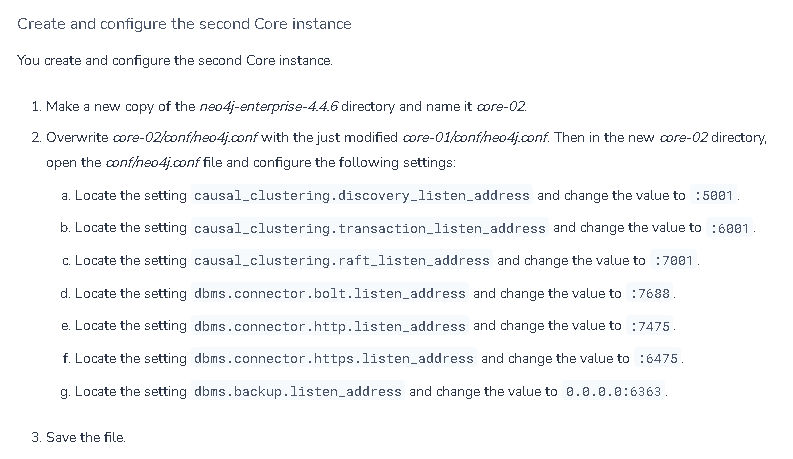




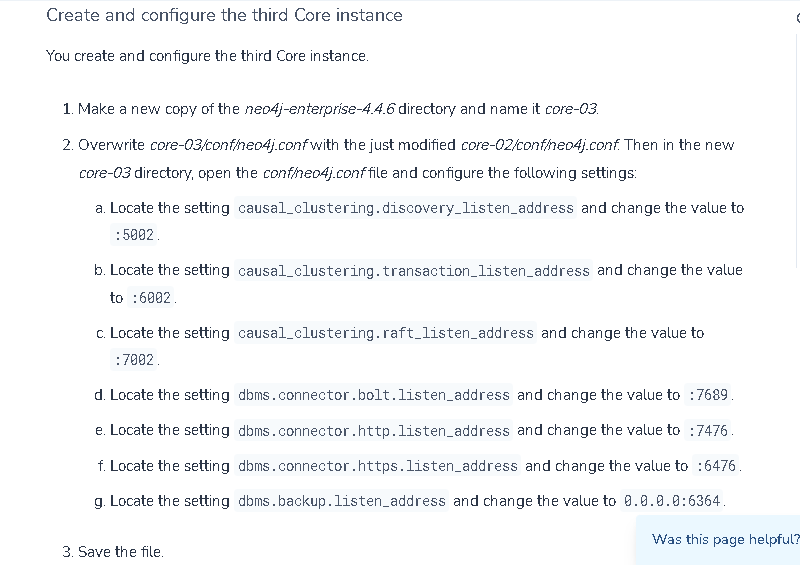
Core-01



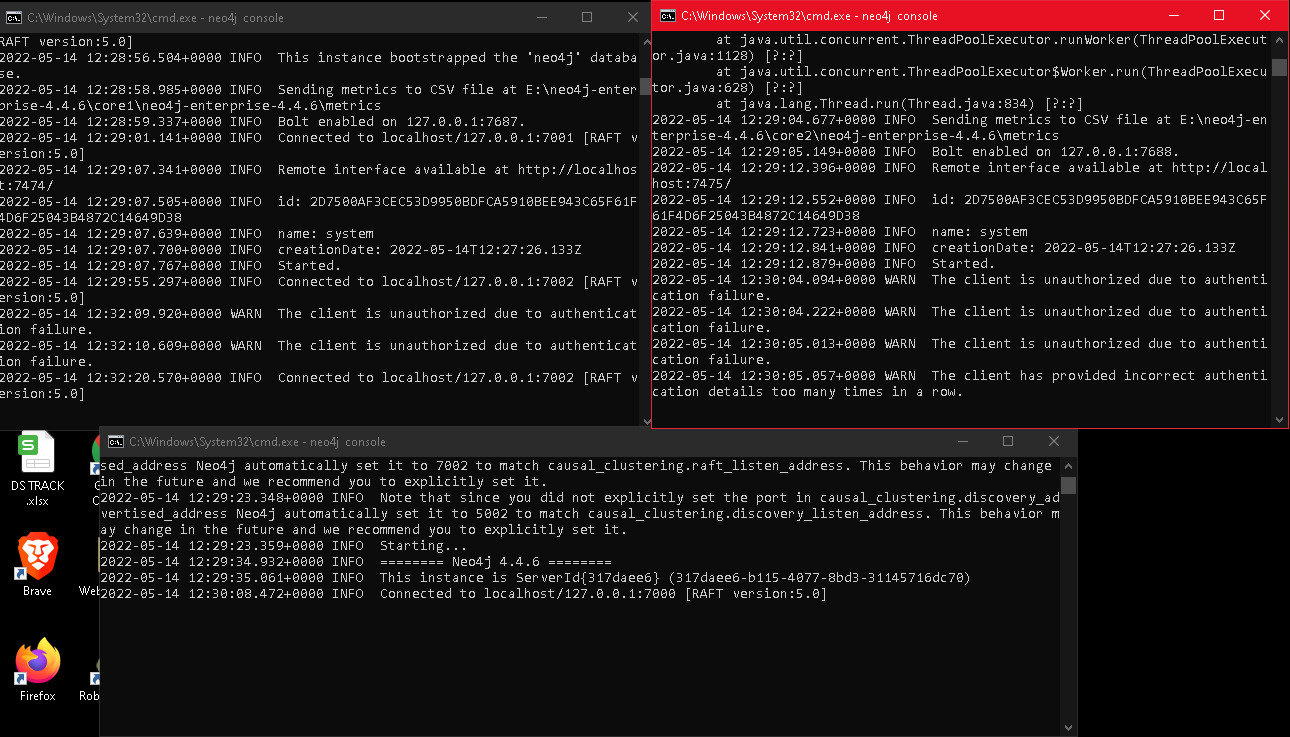
Core-02:



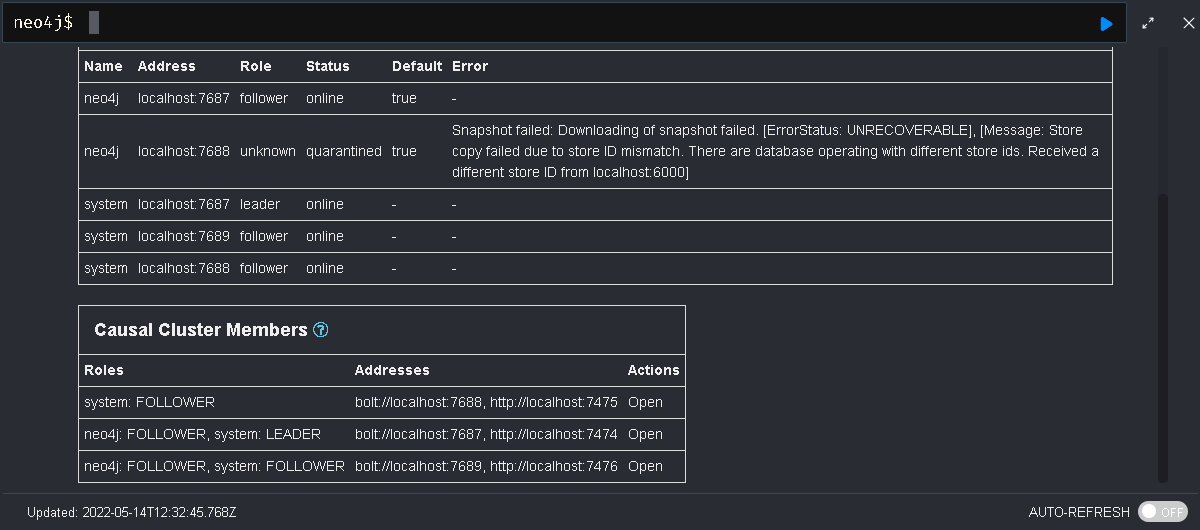
Core-03:



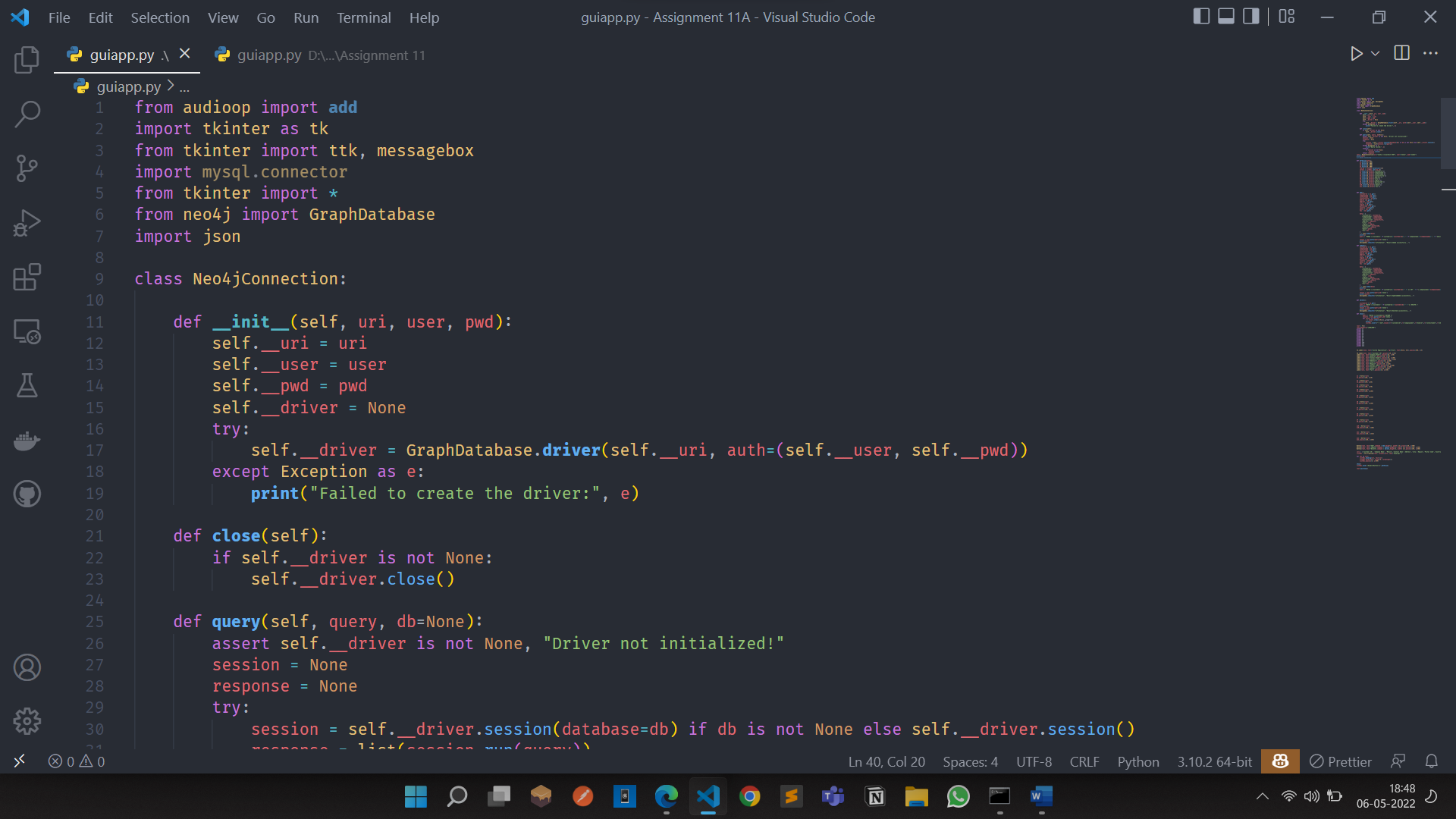
Now go to bin directory of each core and command prompt and type neo4j console.



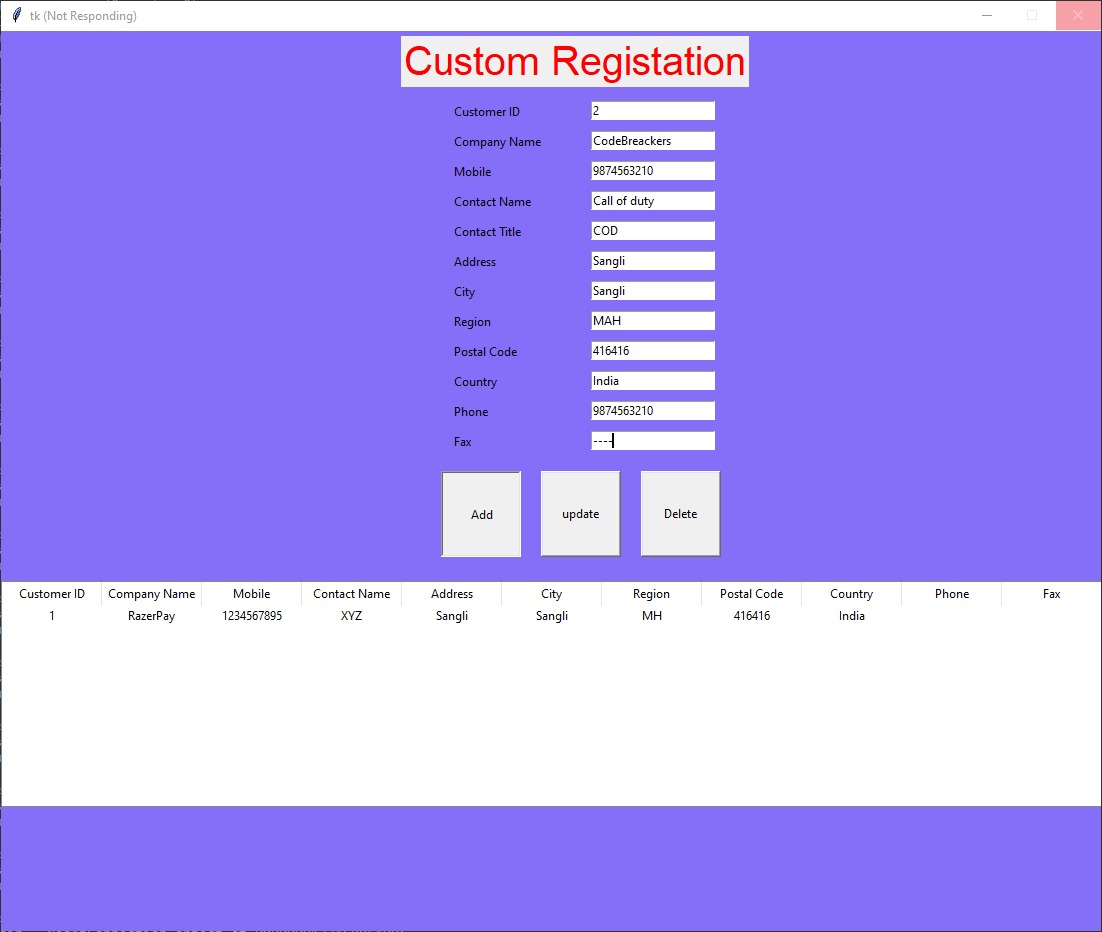
Getting cluster status:

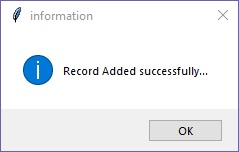


Now performing CRUD operation using python Gui Application.

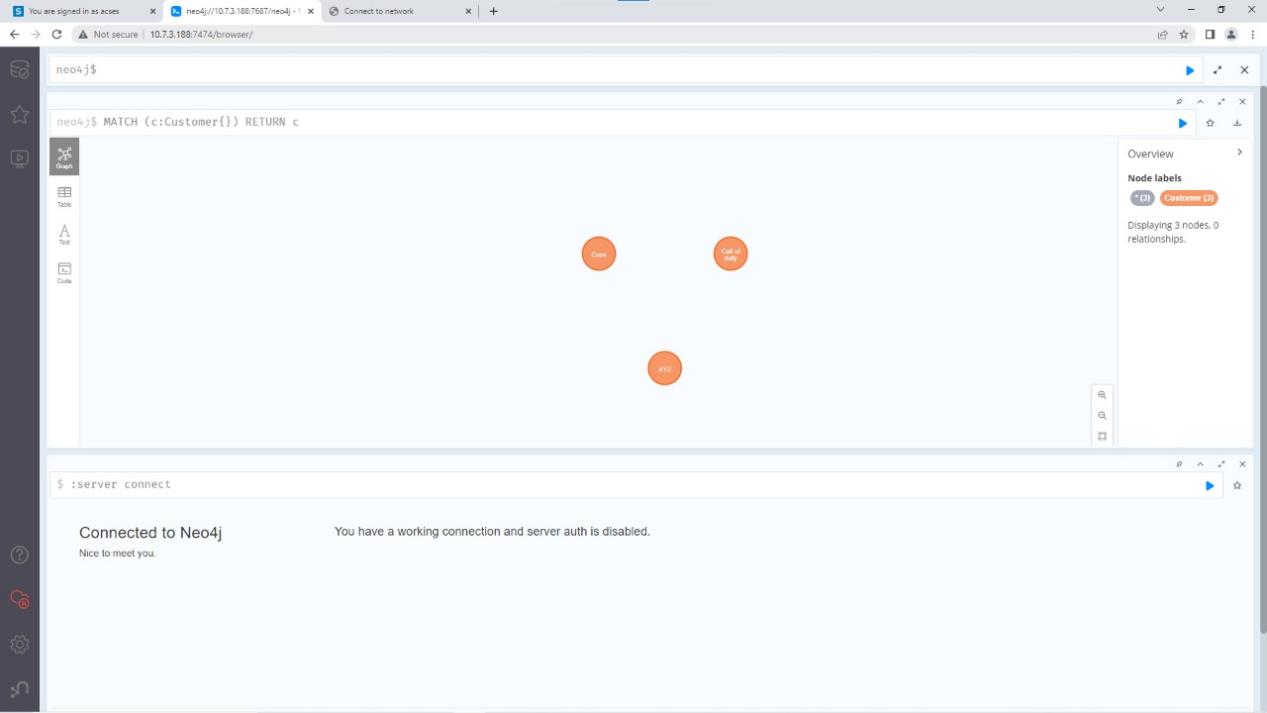


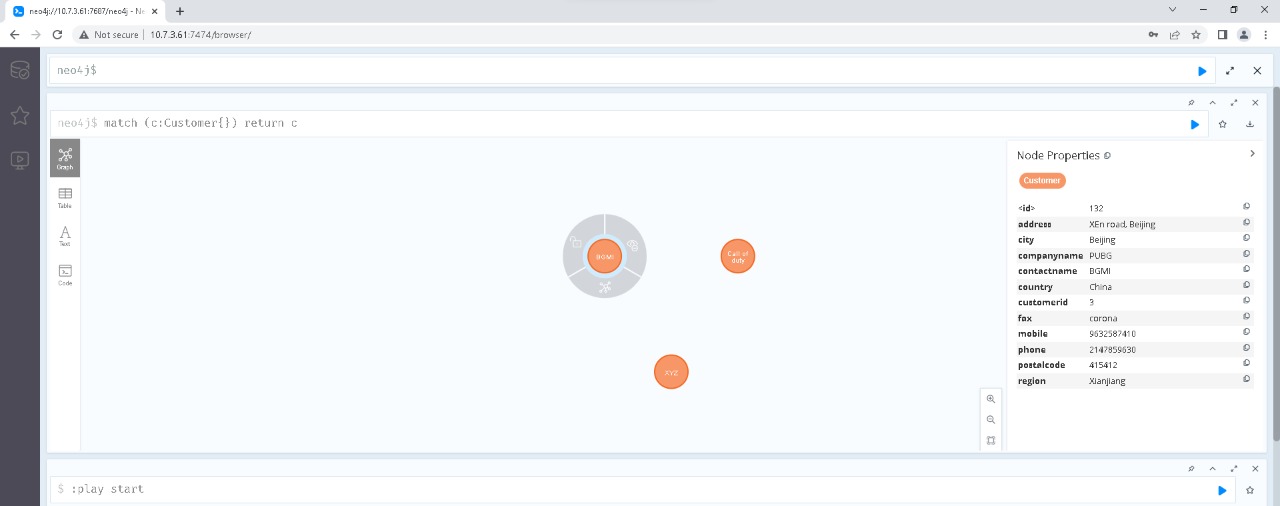
1. Add customer



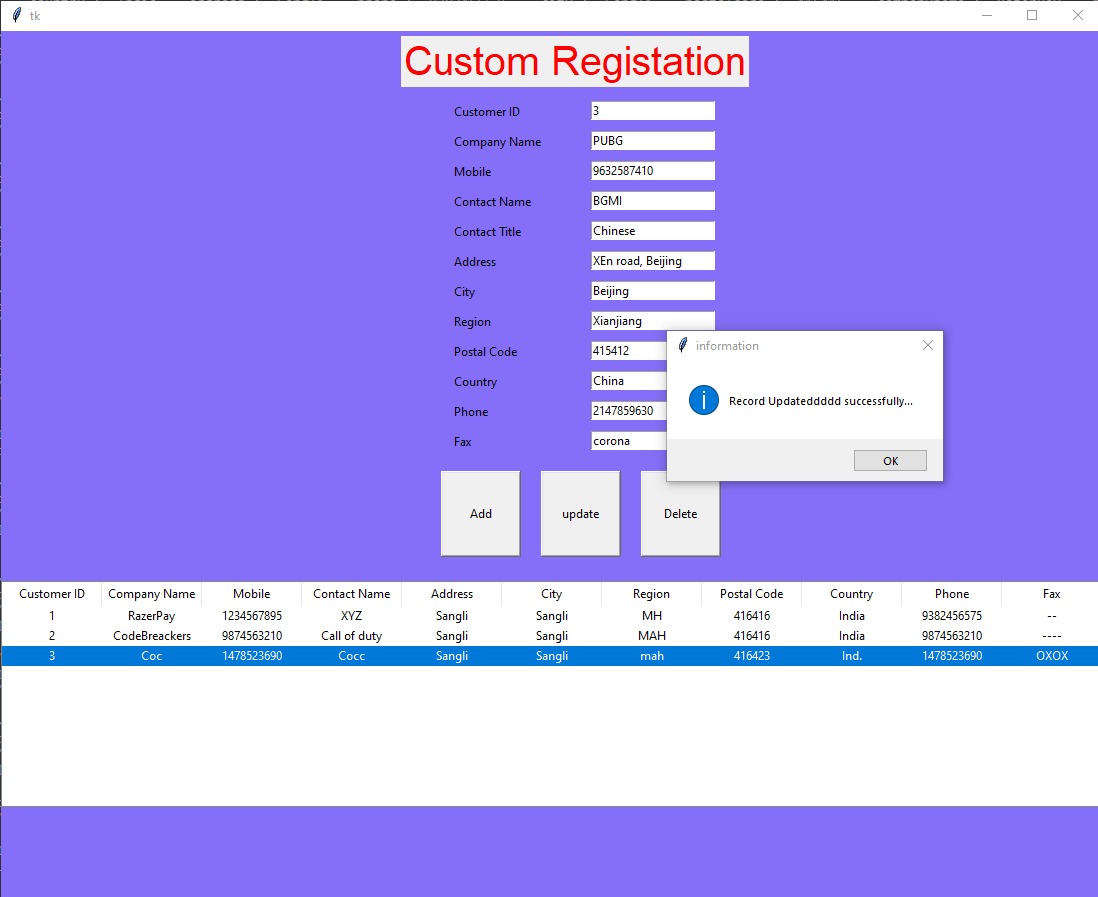


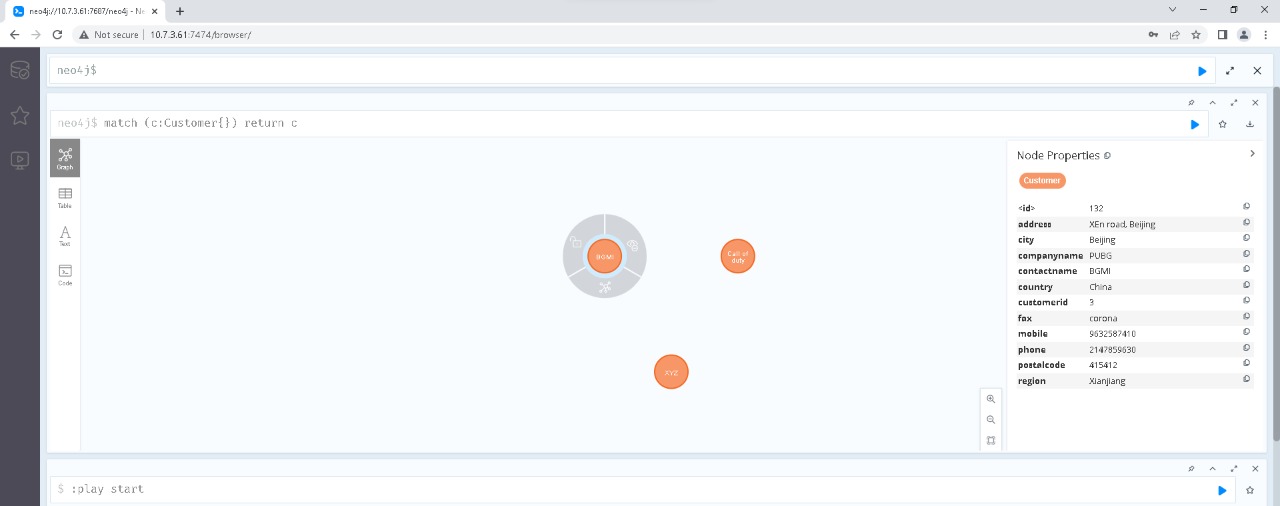
1. In followers node, nodes are added in graph



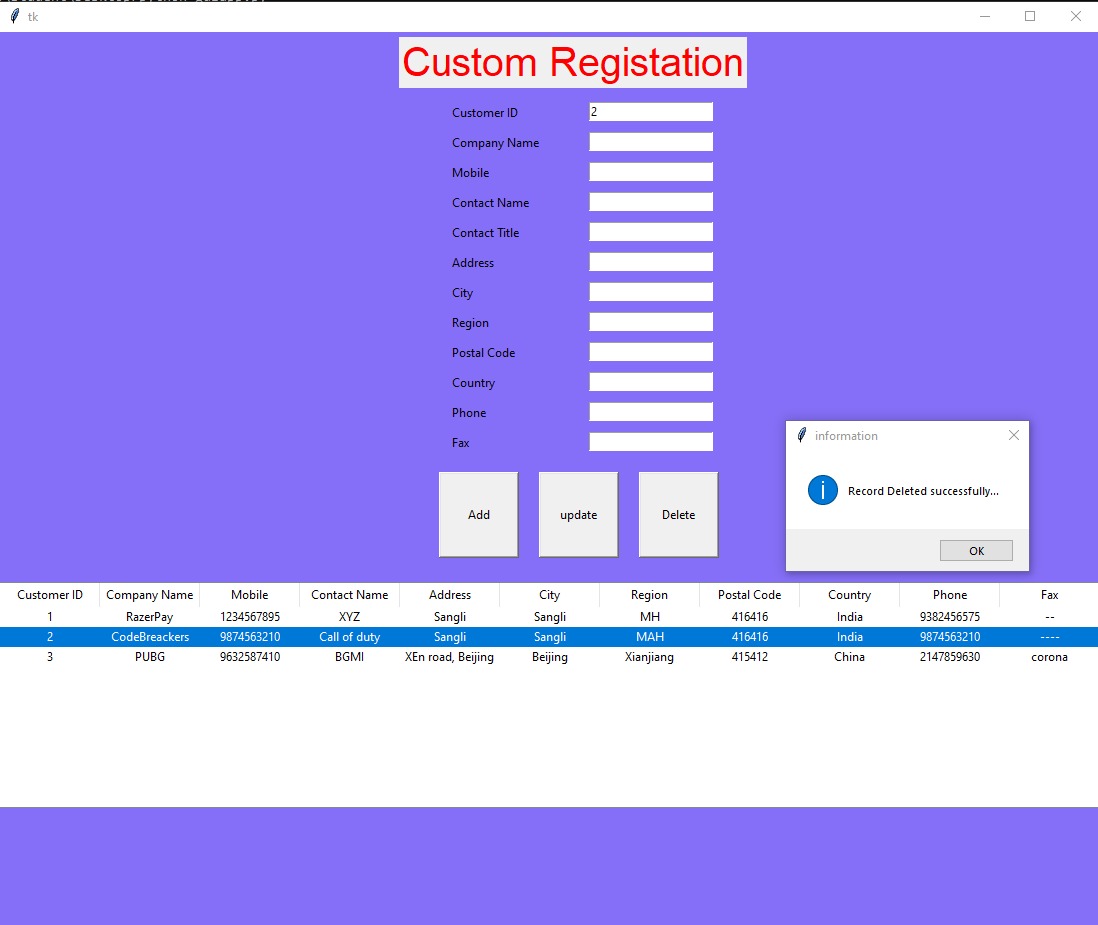


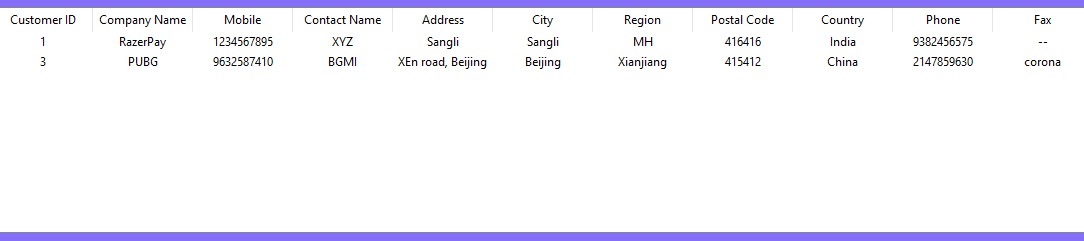
1. Read and Update Customer Record:





1. Delete a record:





1. Changes reflected in Database of followers

