Programming Assignment 1:

Prerequisite:

- The working environment should have Java-1.8 installed.
- Apache Ant-1.9.3 and above.

Steps to Execute:

- Extract PROG1_KALE_ADITYA.tar to a particular location in your Linux environment. Go to above extracted location from terminal. Go to project NapsterPeerToPeer.
- Execute **command** /NapsterPeerToPeer$ **ant**
- Build.xml gets execute and generates a jar. A **dist** folder is generated which contains **NapsterPeerToPeer.jar.**
- If you want to add some more files to register to an Indexing Server, then copy those files to folder **ClientData**. These files are the files which will be available for other peers to download.
- The project NapsterPeerToPeer **must contain** a folder named **Downloaded.** All the files which will get download from another peer will be stored in that folder location.

Steps to start Client and Server.

- Go to folder resources and open **config.properties**. Update serverIp variable value with Ipaddress of that machine where Indexing Server will be started.

  Ex: **serverIp** = 192.168.239.131

- Update serverPort and clientPort variable value with port numbers which are available for communication.

  Ex: **srerverPort** = 9999

  clientPort = 8888

- **All client must have same port number** for communication.
- Go to above project location from terminal and execute following commands
- For Linux Environment
  - Execute **command** /NapsterPeerToPeer$ **./runServer.sh**
    - Server gets start and ready to listen other peer request.
  - Execute **command** /NapsterPeerToPeer$ **./runClient.sh**
    - Clients gets Start and all the files from folder ClientData will be registered in an Indexing Server.
  - Now you can give input from client.
    1. **Get all available file names to download** → Displays all file names which are registered by all different clients.
    2. **Download a file** → Will ask for another input as file name, which you want to download. And if the Peer is up, will download the file in **Downloaded** folder.
    3. **Start Client to send files** → Will make current Client as server. All the files which are registered to Indexing Server will become available to download for other peers.
    4. **Stop Client** → Will terminate the program.
    5. Also can Use **ctrl + C** to terminate the program.

Steps to run multiple Clients.

- Copy the same project structure and configuration to multiple Linux machine which are connected in a single network.
- And run multiple clients as multiple Peers with above given commands. And you should be able to communicate among different clients and an Indexing Server.

For Windows Environment

- Execute **command** \NapsterPeerToPeer> **java –cp dist\NapsterPeerToPeer.jar edu.indexingserver.IndexingServer**
  - Server gets start and ready to listen other peer request.
- Execute **command** \NapsterPeerToPeer> **java –cp dist\NapsterPeerToPeer.jar edu.peer.Client**
  - Clients gets Start and all the files from folder ClientData will be registered in an Indexing Server.