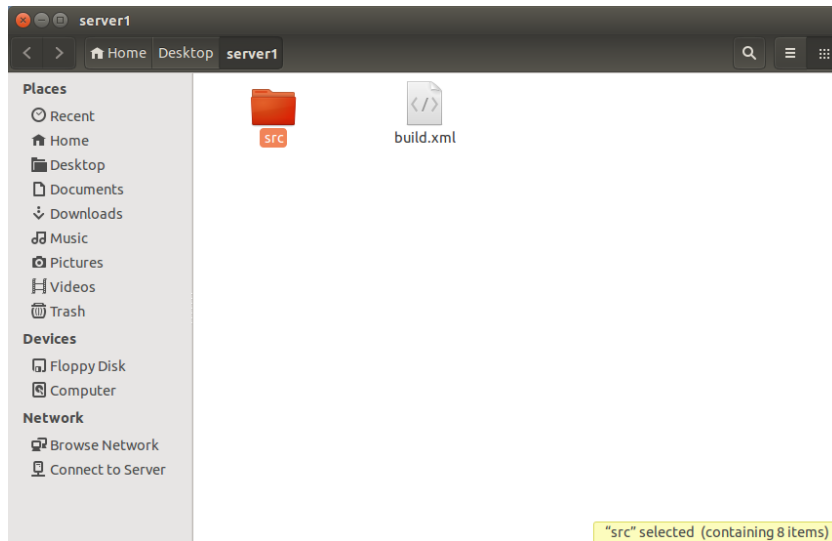


## Step 1: Config

The config file is named as 'config.txt'. The content of it is the IP address and port of indexing servers. The IP address and port is separated by "-". For example:

127.0.0.1-8000

Step 2: copy all the files in source code to a new directory (let's called server1) under the src directory create a txt called config.txt, put in 127.0.0.1-8000 and save



## Step 3 build

\$ant

```
xqy@ubuntu: ~/Desktop/server1/build
build.xml  src
xqy@ubuntu:~/Desktop/server1$ ant
Buildfile: /home/xqy/Desktop/server1/build.xml

clean:

compile:
  [mkdir] Created dir: /home/xqy/Desktop/server1/build/classes
  [javac] /home/xqy/Desktop/server1/build.xml:9: warning: 'includeantruntime'
was not set, defaulting to build.sysclasspath=last; set to false for repeatable
builds
  [javac] Warning: DeCentralizedIndexingServer.java modified in the future.
  [javac] Warning: IndexServerHelper.java modified in the future.
  [javac] Warning: Peer.java modified in the future.
  [javac] Warning: PeerClient.java modified in the future.
  [javac] Warning: PeerServer.java modified in the future.
  [javac] Compiling 10 source files to /home/xqy/Desktop/server1/build/classes

BUILD SUCCESSFUL
Total time: 12 seconds
xqy@ubuntu:~/Desktop/server1$ ls
build  build.xml  src
xqy@ubuntu:~/Desktop/server1$ cd build
xqy@ubuntu:~/Desktop/server1/build$
```

#### Step 4: Run indexing server

Under the classes directory

```
$java DeCentralizedIndexingServer 8000
```

Indexing Server is listening

```
xqy@ubuntu: ~/Desktop/server1/build/classes
builds
[javac] Warning: DeCentralizedIndexingServer.java modified in the future.
[javac] Warning: IndexServerHelper.java modified in the future.
[javac] Warning: Peer.java modified in the future.
[javac] Warning: PeerClient.java modified in the future.
[javac] Warning: PeerServer.java modified in the future.
[javac] Compiling 10 source files to /home/xqy/Desktop/server1/build/classes
BUILD SUCCESSFUL
Total time: 12 seconds
xqy@ubuntu:~/Desktop/server1$ ls
build  build.xml  src
xqy@ubuntu:~/Desktop/server1$ cd build
xqy@ubuntu:~/Desktop/server1/build$ ls
classes
xqy@ubuntu:~/Desktop/server1/build$ cd classes
xqy@ubuntu:~/Desktop/server1/build/classes$ ls
DeCentralizedIndexingServer.class  Peer.class          test
helper                             PeerClient.class
IndexServerHelper.class           PeerServer.class
xqy@ubuntu:~/Desktop/server1/build/classes$ java DeCentralizedIndexingServer 8000
0
Index server is listening 8000
```

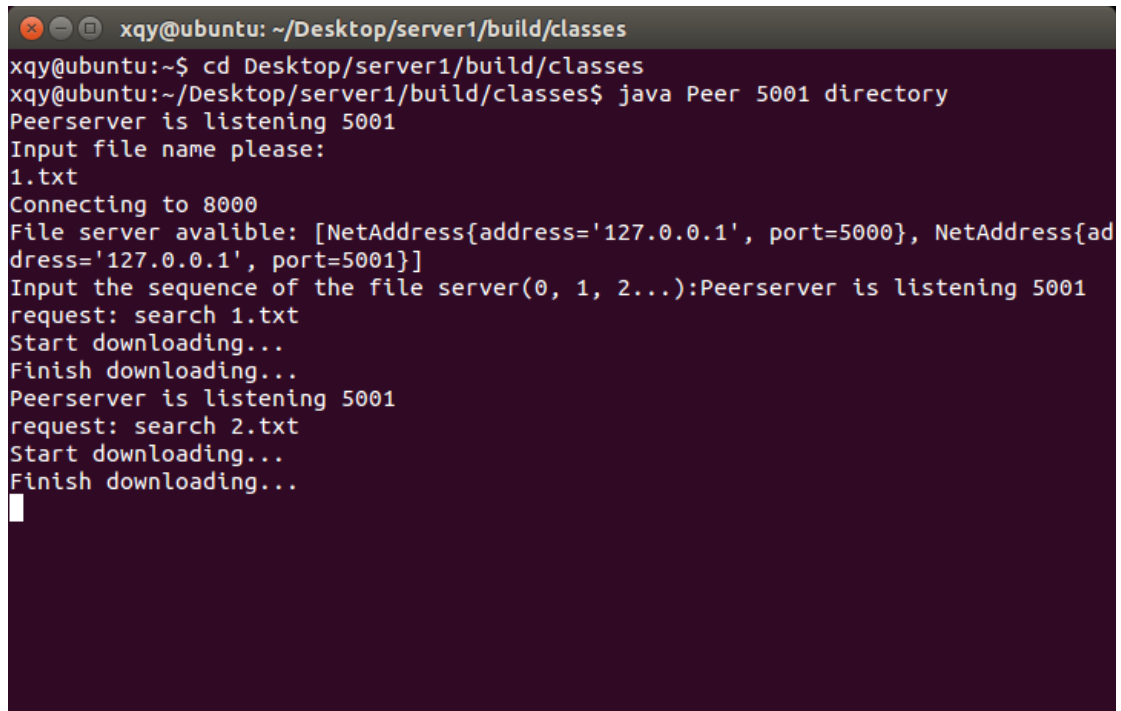
#### Step 5: create a test directory and make some test files in it. (Under build/classes)

```
xqy@ubuntu: ~/Desktop/server1/build/classes/directory
xqy@ubuntu:~/Desktop/server1/build$ cd Desktop/server1/build
xqy@ubuntu:~/Desktop/server1/build$ ls
classes
xqy@ubuntu:~/Desktop/server1/build$ cd classes
xqy@ubuntu:~/Desktop/server1/build/classes$ ls
DeCentralizedIndexingServer.class  Peer.class          test
helper                             PeerClient.class
IndexServerHelper.class           PeerServer.class
xqy@ubuntu:~/Desktop/server1/build/classes$ cd ..
xqy@ubuntu:~/Desktop/server1/build$ ls
classes
xqy@ubuntu:~/Desktop/server1/build$ cd classes
xqy@ubuntu:~/Desktop/server1/build/classes$ ls
DeCentralizedIndexingServer.class  Peer.class          test
helper                             PeerClient.class
IndexServerHelper.class           PeerServer.class
xqy@ubuntu:~/Desktop/server1/build/classes$ ls
DeCentralizedIndexingServer.class  IndexServerHelper.class  PeerServer.class
directory                          Peer.class              test
helper                             PeerClient.class
xqy@ubuntu:~/Desktop/server1/build/classes$ cd directory
xqy@ubuntu:~/Desktop/server1/build/classes/directory$ ls
1.txt  2.txt  3.txt
xqy@ubuntu:~/Desktop/server1/build/classes/directory$
```

#### Step 6: Run peer

Create a new terminal and find the classes directory (we first registry in server1)

\$ java Peer 5001 directory

A terminal window with a dark purple background and light green text. The window title is 'xqy@ubuntu: ~/Desktop/server1/build/classes'. The user enters 'cd Desktop/server1/build/classes' and then 'java Peer 5001 directory'. The program outputs 'Peerserver is listening 5001' and prompts for a file name. The user enters '1.txt'. The program connects to port 8000 and lists available file servers: '[NetAddress{address='127.0.0.1', port=5000}, NetAddress{address='127.0.0.1', port=5001}]'. It then prompts for a sequence of file servers, and the user enters '0, 1, 2...'. The program outputs 'Peerserver is listening 5001' and processes a 'search 1.txt' request, showing 'Start downloading...' and 'Finish downloading...'. It then processes a 'search 2.txt' request, also showing 'Start downloading...' and 'Finish downloading...'. A cursor is visible at the end of the last line.

```
xqy@ubuntu: ~/Desktop/server1/build/classes
xqy@ubuntu:~$ cd Desktop/server1/build/classes
xqy@ubuntu:~/Desktop/server1/build/classes$ java Peer 5001 directory
Peerserver is listening 5001
Input file name please:
1.txt
Connecting to 8000
File server available: [NetAddress{address='127.0.0.1', port=5000}, NetAddress{ad
dress='127.0.0.1', port=5001}]
Input the sequence of the file server(0, 1, 2...):Peerserver is listening 5001
request: search 1.txt
Start downloading...
Finish downloading...
Peerserver is listening 5001
request: search 2.txt
Start downloading...
Finish downloading...
█
```

Step 7: Input file name to download

Step 8: Choose an available file server

(If you want to set more peer server do the step 6 many times)

```
xqy@ubuntu: ~/Desktop/server1/build/classes
xqy@ubuntu:~/Desktop/server1/build/classes$ java DeCentralizedIndexingServer
Index server is listening 8000
Index server is listening 8000
Receive: registry 5000 1.txt 2.txt 3.txt
Registry success...
Index server is listening 8000
Receive: search 1.txt
Reply: 0 127.0.0.1-5000
Index server is listening 8000
Receive: registry 5001 1.txt 2.txt 3.txt
Registry success...
Index server is listening 8000
Receive: search 1.txt
Reply: 0 127.0.0.1-5000 127.0.0.1-5001
Index server is listening 8000
Receive: registry 5002 1.txt 2.txt 3.txt
Registry success...
Index server is listening 8000
Receive: search 1.txt
Reply: 0 127.0.0.1-5000 127.0.0.1-5001 127.0.0.1-5002
Index server is listening 8000
Receive: search 1.txt
Reply: 0 127.0.0.1-5000 127.0.0.1-5001 127.0.0.1-5002
Index server is listening 8000
```

Hint: Pictures can be seen in the output manual.