

Manual for Programming Assignment #3

Prerequisite

1. You must have Ant installed in you system for compiling
2. Have java version 1.7 or after 1.7 for compiling and running application

Project Directory Structure

To navigating to project Directory: Open terminal then go to project directory using command “cd” in terminal.

Project Directory typically is like below

src :: contains the source code of server

conf :: contains property file and information regarding server port

dist:: contains the compiled jar file (dir created after compile)

bin :: contains compiled files (dir created after compile)

build.xml :: For compiling project using ant

Compiling Server Code

We need Ant for compiling and creating jar file of source code::

Type command “ant” inside the folder server directory named as “DecentralizedIndexServer”. It will automatically pick the build.xml file and compile the code and create a jar file named like “DecentralizedIndexServer.jar” in “dist” directory of current project folder

Your compiled jar file will have class file

Configuring Server

Once you have successfully created jar file. There is a conf directory in server folder. Inside that folder we have a properties files names “CentralIndexServer.properties” that is used for configuring the Index server

Configuration file for Server Application

server.port takes port no on which server will run.
server.port=9000

brokerServer.info takes the brokers information of other Index Server.
#Each index server acting as broker to this index server must provide its IP Address and port no
#For multiple broker index Server (IP Address and port no) of each server should be seperated by ", "
#(eg:brokerServer.info=172.31.37.67:9000,172.31.37.67:9005,172.31.37.65:9000,172.31.37.65:9005,)

brokerServer.info=

First index Server should not have any broker so brokerServer.info property is left empty

For Second index Server First index Server will be acting as broker so in second index server properties will be like

Server.port={Port no for Second Index Server}

brokerServer.info={First Index Server Ip}:{ First Index Server Port},

Running server

Once we have created jar of your project and done the required configuration for server.

There is a "dist" directory inside server folder created during compilation of code, where you will see your compiled jar file.

To run the server all you need to do is go to your server folder and type the below command on terminal and server will run.

:::::Commands :::::

```
java -jar dist/{yourjarfile}.jar
```

For example

```
java -jar dist/CentralIndexServer-20150914.jar
```

Compiling Peer Code

We need Ant for compiling and creating jar file of source code

Type command "ant" inside the peer directory named as "Assignment3Peer". It will automatically pick the build.xml file and compile the code and create a jar file named like "Assignment3Peer.jar" in "dist" directory of current project folder

Your compiled jar file will have class file

Configuring Peer

Once you have successfully created jar file. There is a conf directory in Assignment3Peer folder. Inside that folder we have a properties file that is used for configuring connection Client.

Configuration file for Peer Application

```
# Every Client should have a distinct name because on client name we are calculating
the Hash function
client.name=Client 1
```

```
#indexserver.info property takes the IP Address and port no of all index server.
#every index server Ip Address and port no are seperated by ","
indexserver.info=172.31.37.67:9000,172.31.37.67:9005,172.31.37.65:9000,172.31.37.65:90
05,172.31.37.66:9000,172.31.37.66:9005,172.31.37.72:9000,172.31.37.72:9005,172.31.37.6
9:9000,172.31.37.69:9005,172.31.37.68:9000,172.31.37.68:9005,172.31.37.71:9000,172.31.
37.71:9005,172.31.37.70:9000,172.31.37.70:9005,
```

```
#As peer is also acting as a Server to store replica and as a File download server.
#localhostserver.port will takes the port no on which server of Peer will run
localhostserver.port=8098
```

```
# replica.no takes the value for creating number of replica of peer files.
# Once the server have enough peer to replicate it will start the replication process
replica.no=1
```

```
#client.dir takes the directory location of client where file are stored
client.dir=/home/ec2-user/PeerFiles/Peer1
```

```
# this property is for running the Performance Test
performanceFile.name=/home/ec2-user/PeerFiles/Peer2/20.rtf,/home/ec2-
user/PeerFiles/Peer2/21.rtf,/home/ec2-user/PeerFiles/Peer2/22.rtf,/home/ec2-
user/PeerFiles/Peer2/23.rtf,/home/ec2-user/PeerFiles/Peer2/24.rtf,/home/ec2-
user/PeerFiles/Peer2/25.rtf,/home/ec2-user/PeerFiles/Peer2/26.rtf,/home/ec2-
user/PeerFiles/Peer2/27.rtf,/home/ec2-user/PeerFiles/Peer2/28.rtf,/home/ec2-
user/PeerFiles/Peer2/29.rtf,
```

Every Peer should have different properties files configuration

Things that needs to change in the configuration of each peer are
client.name:: As every Peer should have different name
localhost.port:: If Running on same machine
localhost.port:: User Specific
client.dir:: Every Peer should have different directory

Running peer

Once you have created jar of your project and changed the port no of server if needed.
There is a "dist" directory inside Assignment3Peer project folder that is created during compilation of code, where you will see your compiled jar file.
To run the server all you need to do is go to you Peer folder and type the below command and peer will run. It will pick the Properties file form "conf" directory of Peer folder.Run the below command on Terminal

```
::::::::::Commands :::::  
java -jar dist/{yourjarfile}.jar  
For example  
java -jar dist/ Assignment3Peer.jar
```