# A. Section: Introduction

The document is to instruct how to deploy the endpoint service and use command line program to verity the service. *WordCountService* will be used to represent the ending point service.

*WordCountService*'s URI will be
http://localhost:8080/WordCountService/wordcountservice/query?word=abc
abc can be replaced by a word being queried but must comply with URI form rule.

*WordCountService* s response will be pure text file and the output is the three-column format [Query_word query_count word_count], a space as delimiter.

# B. Section : Preparation

WordCountService_servelet_ubuntu_java16.tar.gz

This services can be deployed to Ubuntu 11.10 version.

Requirement packages

Ubuntu(11.10 to high)

Eclipse IDE(Kepler or higher)

Jdk 1.6

After unzip either one, you will see file structure like
Note: here only lists directory you should use during deployment)

**WordCountService_servelet_ubuntu_java16**

**apache-tomcat-7.0.47**

Tomcat to deploy service

**CommandLineTestProgram**

Testing cli program for the service.

Please refer to section D: command Line Test Program

**WordCountService**

The existing maven project should be imported by eclipse.

Please refer to section C: Eclipse deployment process

Please copy above three directories to the directory where you want eclipse to import the project. In the following sections, *INSTALL_DIRECTORY* will be the notation for this directory.

# C. Section: Eclipse deployment process

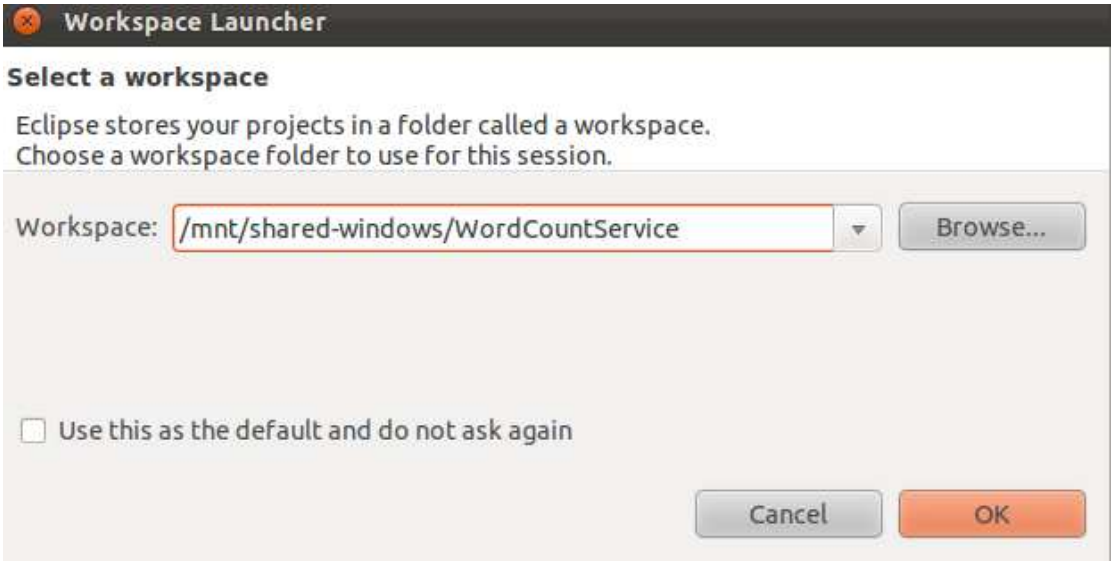
Note: using Ubuntu as an example

- Download eclipse
  http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/marsr
  please choose JAVA EE according to your machine. For Ubuntu, please choose linux 32-bit or 64-bit

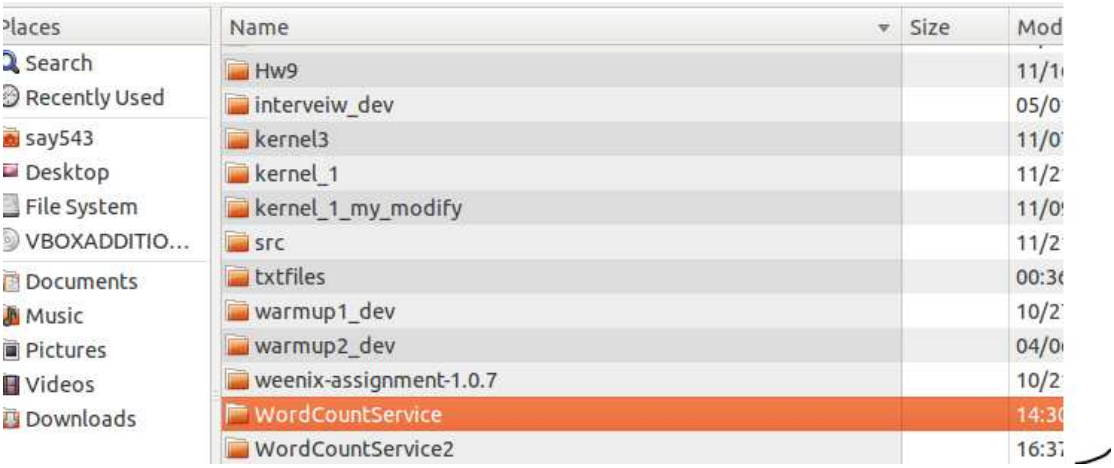
- unzip eclipse to a directory
  sudo tar -xzvf eclipse-jee-kepler-SR2-linux-gtk-x86_64.tar.gz -C /opt
  Note: using /opt as the extracted location

- activate eclipse
  cd /opt/eclipse/
  ./eclipse

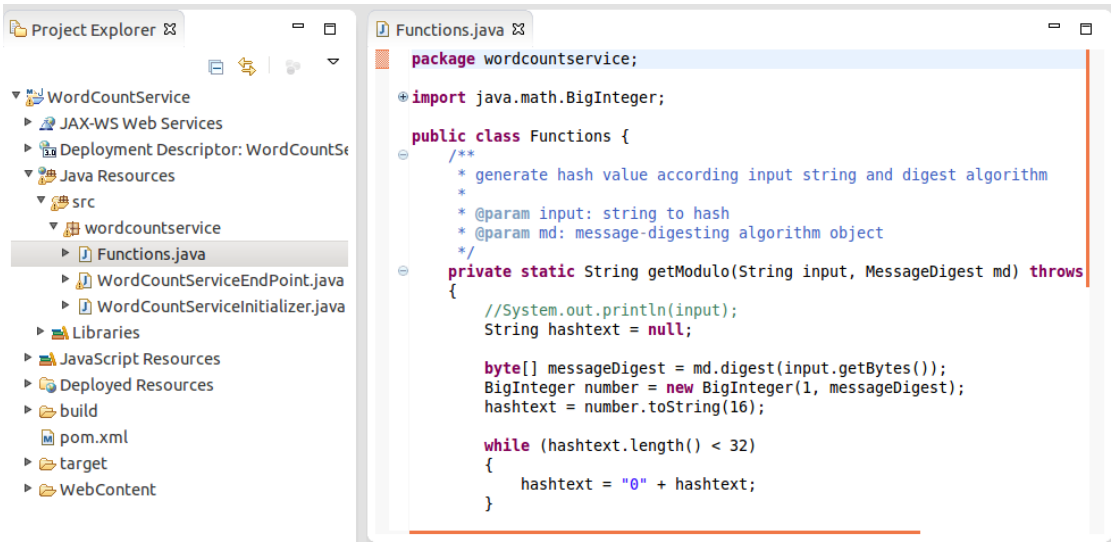
- Choose the working directory by yourself




- Choose File tab -> import tab-> select Maven in drop-down menu-> choose exising Maven projects-> select directory you want to import.
  According to file structure(referring to section B: preparation), you should select INSTALL_DIRECTORY/WordCountService if deploying on Ubuntu.

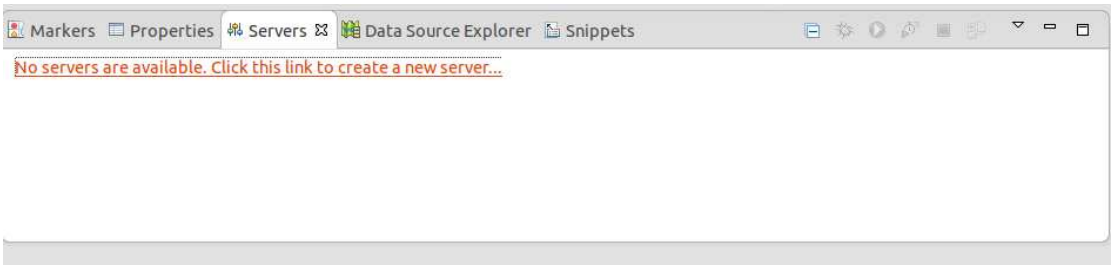



- You will see project as the following

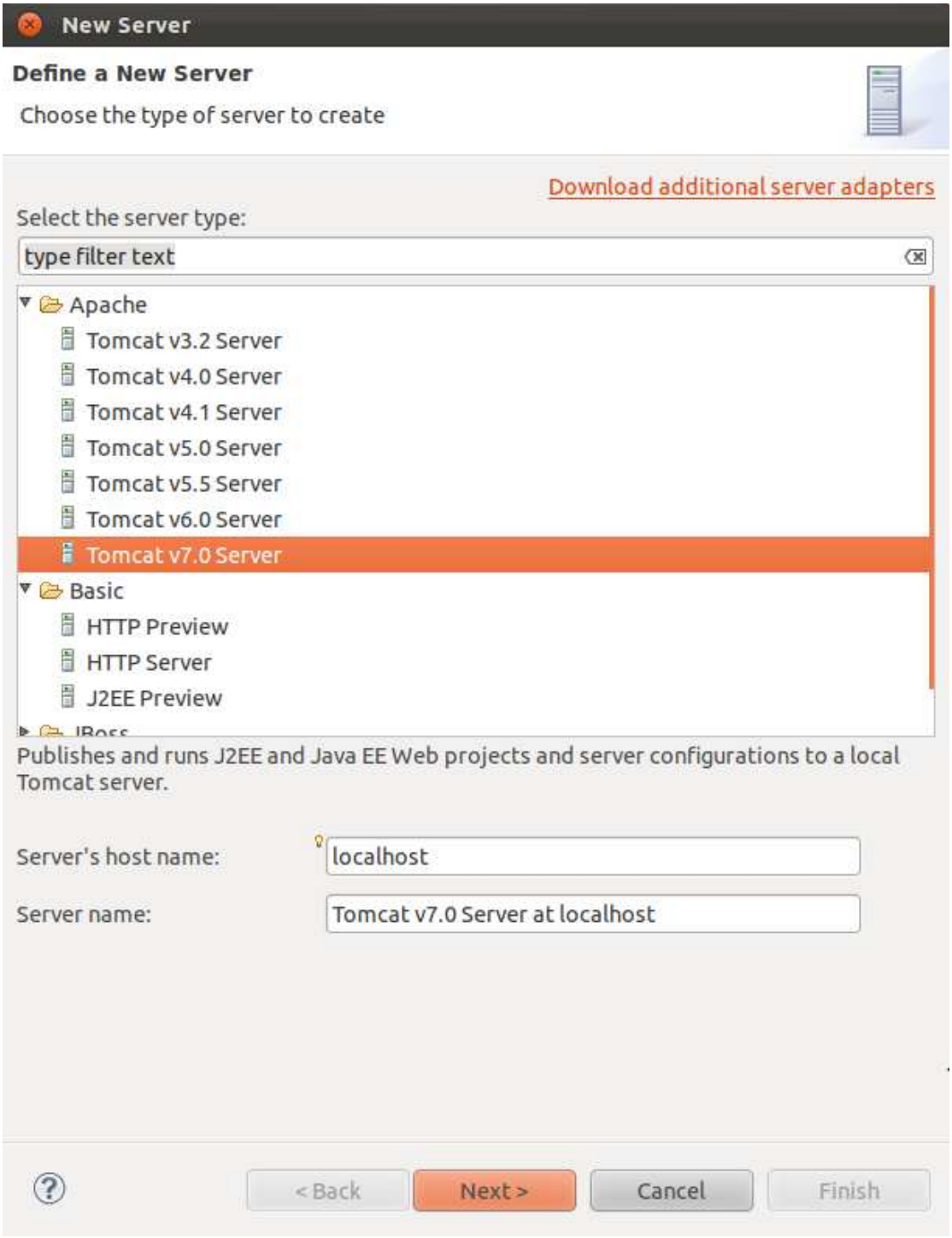



- Select tomcat server

  In the bottom console, select Servers tab and click to create a new server



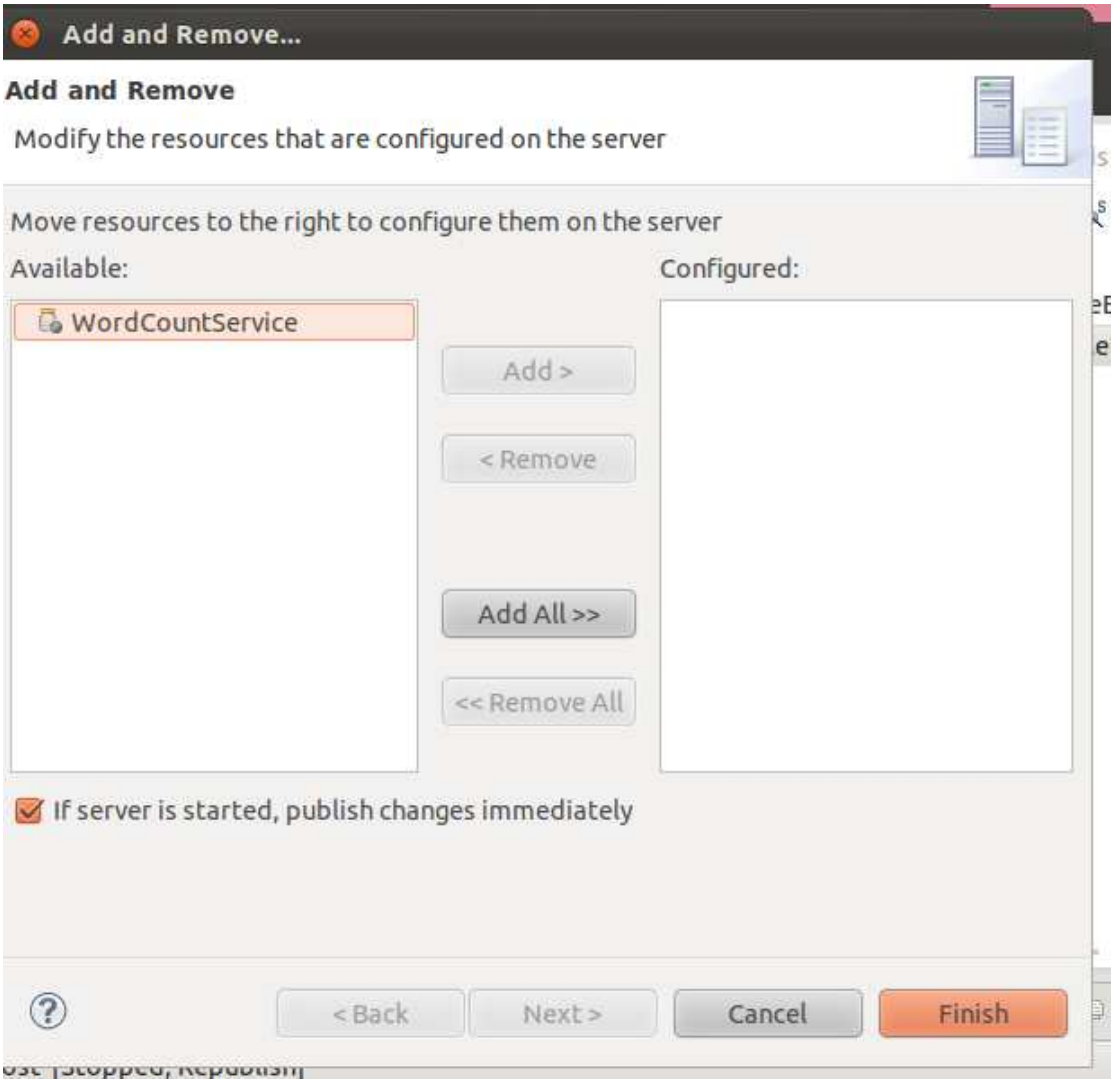

- Select tomcat v7.0 Server -> click Next



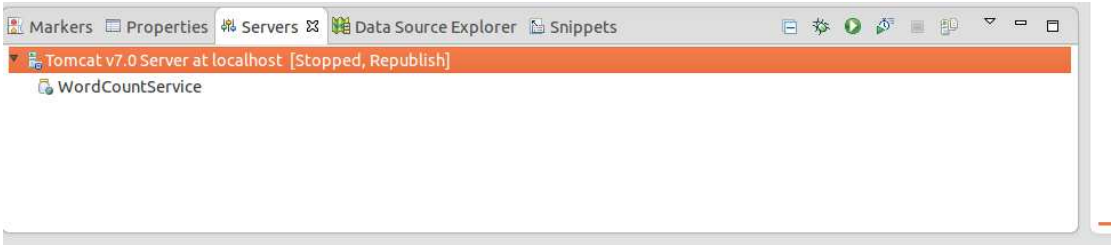

- Select browse tab -> Select where tomcat being install-> click finish
  Note: According to file structure (referring to section B: preparation), you should select INSTALL_DIRECTORY/apache-tomcat-7.0.47 if deploying on Ubuntu.

- You will see tomcat is available.

- Right click Tomcat at the bottom panel ->choose add and remove from pop up window -> add WordCountService to the right -> click Finish



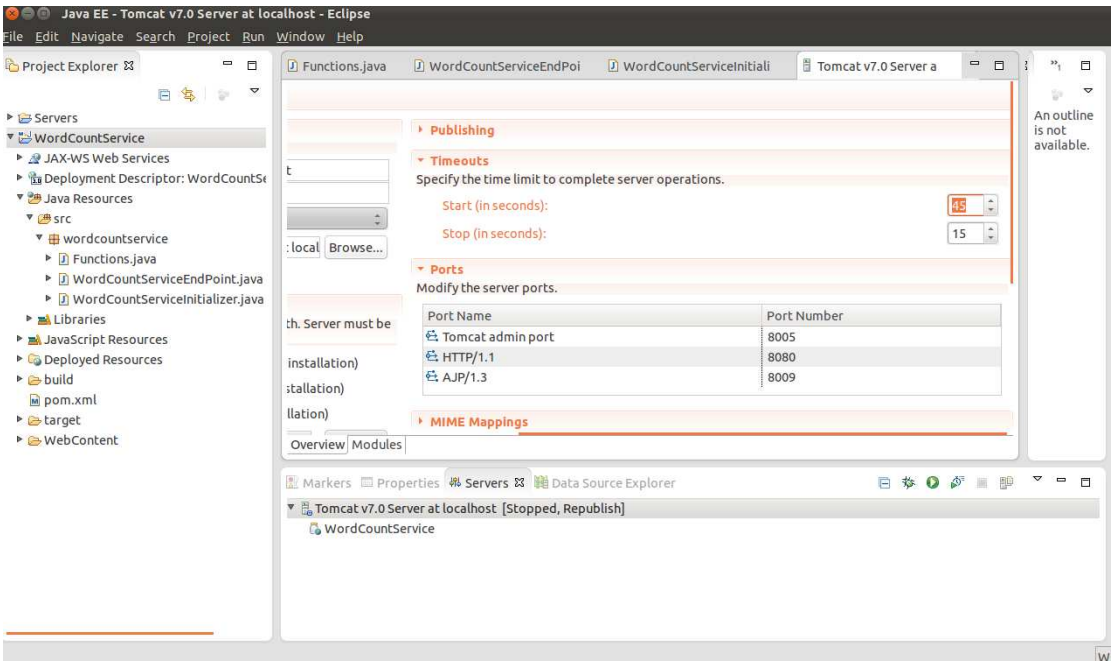

- You will see available service as the following.



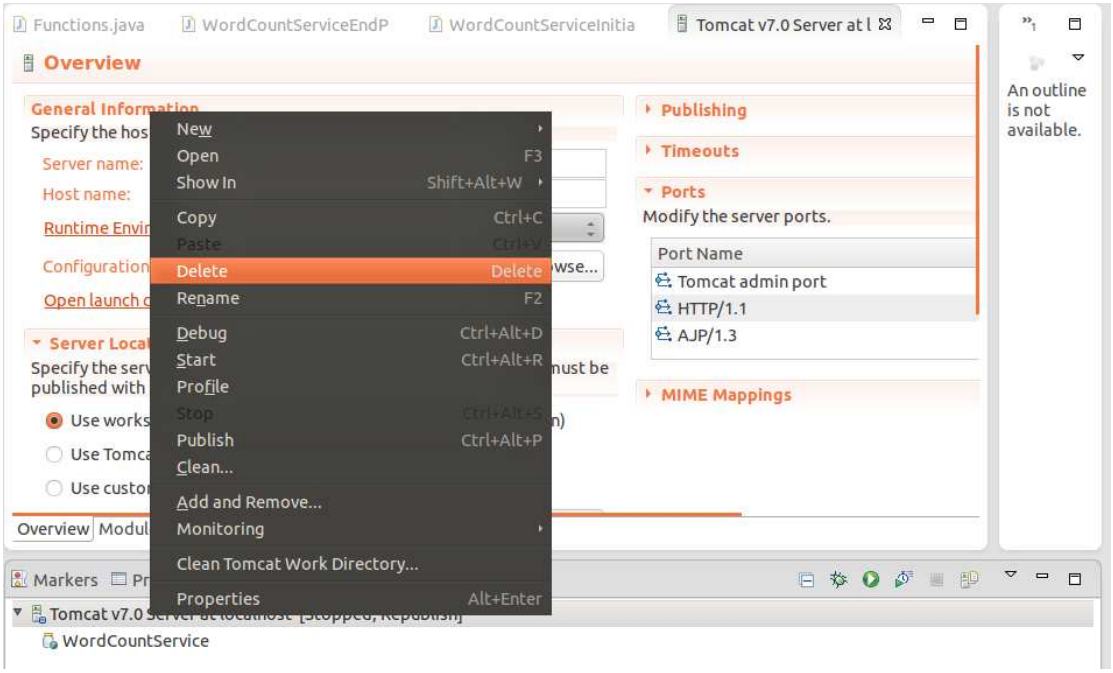

- Double click Tomcat at the bottom panel -> adjust timeouts for server start time.

Note: Because word count service needs to fetch all txt files before making the endpoint available, you need to fill in proper value to prevent from time out. Timeout might depends on how many words being fetched through all txt files.

When you change start time, remember to save



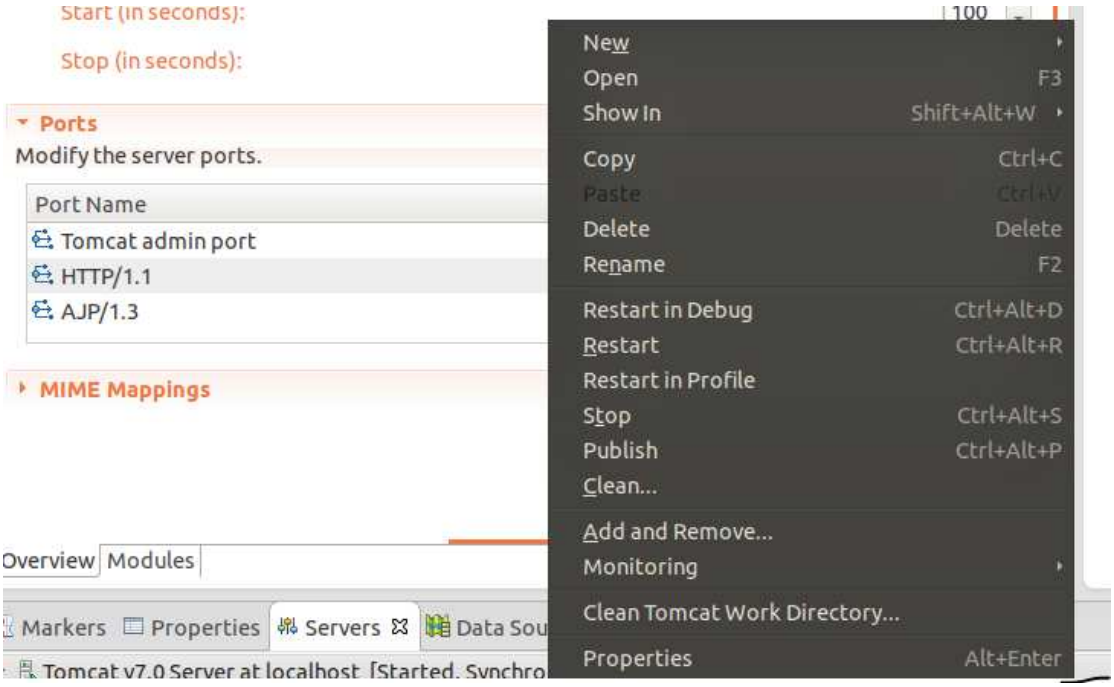

- Right click Tomcat at the bottom panel -> choose start.




- Checking information from console. If you see “context initialization is done”, it means *WordCountService* is activated.

```
ContextListener: context initialization...
Sep 10, 2016 8:44:26 PM org.apache.catalina.core.ApplicationContext log
INFO: ContextListener: context initialization done
Sep 10, 2016 8:44:26 PM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-bio-8080"]
Sep 10, 2016 8:44:26 PM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["ajp-bio-8009"]
Sep 10, 2016 8:44:26 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 37739 ms
```

- Right click Tomcat at the bottom panel -> choose stop. It will terminate *WordCountService* and let you prepare txt files



- Preparing txtfiles for service

  Please make a new directory and copy all related txt files to the following directory. INSTALL_DIRECTORY/.metadata/.plugins/org.eclipse.wst.server.core/tmp0/wtpwebapps/WordCountService/txtfiles. PLEASE DO NOT RENAME THIS DIRECTORY.

  An example of list of files is the following

```
say543@say543-VirtualBox:/mnt/shared-windows/WordCountService/.metadata/.plugins/org.eclipse.wst.server.core/tmp0/wtpwebapps/WordCountService/txtfiles$
 ls
0paine.txt   7gwss11.txt  7spur11.txt  altif10.txt               dywrk10.txt                 machiavelli-prince-123.txt  smith-inquiry-161.txt
2almy10.txt  7hckb10.txt  7untr10.txt  bacon-new-98.txt          franklin-boston-245.txt     marx-manifesto-213.txt      teddy10.txt
2cahe10.txt  7herm10.txt  7wdvn10.txt  bfree10.txt               franklin-london-246.txt     mill-representative-216.txt thoreau-walden-186.txt
2lcky10.txt  7isbl10.txt  7wml112.txt  bgopr10.txt               franklin-philadelphia-248.txt mklmt10.txt               tl41510.txt
3sdms10.txt  7itr110.txt  7wml210.txt  btowe10.txt               gcoaz10.txt                 nilet10.txt                 ttalk10.txt
4sdms10.txt  7itr210.txt  7wml310.txt  cbtls12.txt               gjagd10.txt                 paine-american-397.txt      tvsrv10.txt
71agt10.txt  7lgnb11.txt  7wml410.txt  chrnv10.txt               hcath10.txt                 pg1342.txt                  ungst10.txt
7efgm10.txt  7ljw111.txt  7wml510.txt  clotl10a.txt              hume-enquiry-65.txt         remus11.txt                 utrkj10.txt
7gesw10.txt  7ljw211.txt  7wml610.txt  dcart10.txt               hume-essays-733.txt         repub13.txt                 vgmld10.txt
7getz10.txt  7myrt10.txt  7wml710.txt  deliv10.txt               king-i-150.txt              sbmaa10.txt                 wldsp10.txt
7gmnt10.txt  7nvll10.txt  7wml810.txt  descartes-discourse-124.txt lucretius-on-395.txt     sbmfa10.txt
```

- Restart tomcat as the previous step. If you see “context initialization is done”, it means *WordCountService* is activated again with txtfiles.

```
ContextListener: context initialization...
Sep 10, 2016 8:44:26 PM org.apache.catalina.core.ApplicationContext log
INFO: ContextListener: context initialization done
Sep 10, 2016 8:44:26 PM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-bio-8080"]
Sep 10, 2016 8:44:26 PM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["ajp-bio-8009"]
Sep 10, 2016 8:44:26 PM org.apache.catalina.startup.Catalina start
INFO: Server startup in 37739 ms
```

## D. Section: command Line Test Program

Note: using Ubuntu as an example

Command line program will send out requests and verify the result based on the following rules

- ✓ The number of requests.
- ✓ The word count for a query string
- ✓ None of two-line results are the same.

Please refer to the below running process and the following example.

- Go to INSTALL_DIRECTORY/CommandLineTestProgram and compile command line program

  javac WordCountServiceMultiThreadTest.java

- Command line program MUST have four options provided

-w people:
User-defined word to query word count service

-t 10000
User-defined the number of request the word will query. It only supports positive integers ranging from 2^32-1 to 1

-f test.txt
User-defined file to store query history. Each query as one line by the format [Query_word query_count word_count]

-d ../.metadata/.plugins/org.eclipse.wst.server.core/tmp0/wtpwebapps/WordCountService/txtfiles
PLEASE DO NOT CHANGE ANYTHING OF THIS PATH
The txt files to collect word count. It is shared with command line program and the *WordCountService*

- This is the successful example to query “books” with 30000 times, output history to “books.txt”

```
say543@say543-VirtualBox:/mnt/shared-windows/WordCountService2/CommandLineTestProgram$ java WordCountServiceMultiThreadTest -w books -t 30000 -f books.txt -d ../.m
etadata/.plugins/org.eclipse.wst.server.core/tmp0/wtpwebapps/WordCountService/txtfiles
test initizalize and setup...
all preprocessing and word restful fetch done!!
verfiy expected result and restufl fetch result...
verify result success!!
```

- Each query as one line by the three-column format [Query_word query_count word_count]
- The result from the successful example presents the following truth if success.
  - ✓ 30000 lines are presented and none of two lines are exactly the same.
  - ✓ For query_word column, each line’s query_word should be “books”
  - ✓ For query_count column, Books minimum Query_count is x and maximum Query_count is x+30000. X is the starting number whenever *WordCountService* being query. X is a variable. If *WordCountService* just starts, x equals to zero. Otherwise, x depends on the previous ending query_count.
  - ✓ For word_count column, Books’ word count should all be the same all lines

- The possible failed results might be due to the following exceptions:
  Note : using query_word, people, to quiry 10000 time as an example

  QueryKeyMisMatchException
  unmatched query word: People v.s Alison
  QueryCountDupException
  queryCount duplication: 2000

two lines have the same in column query_count

QueryCountMissException

numOfReq miss certain of requests: 10000

10000 is the number of total requests but some query_counts miss in column query_count