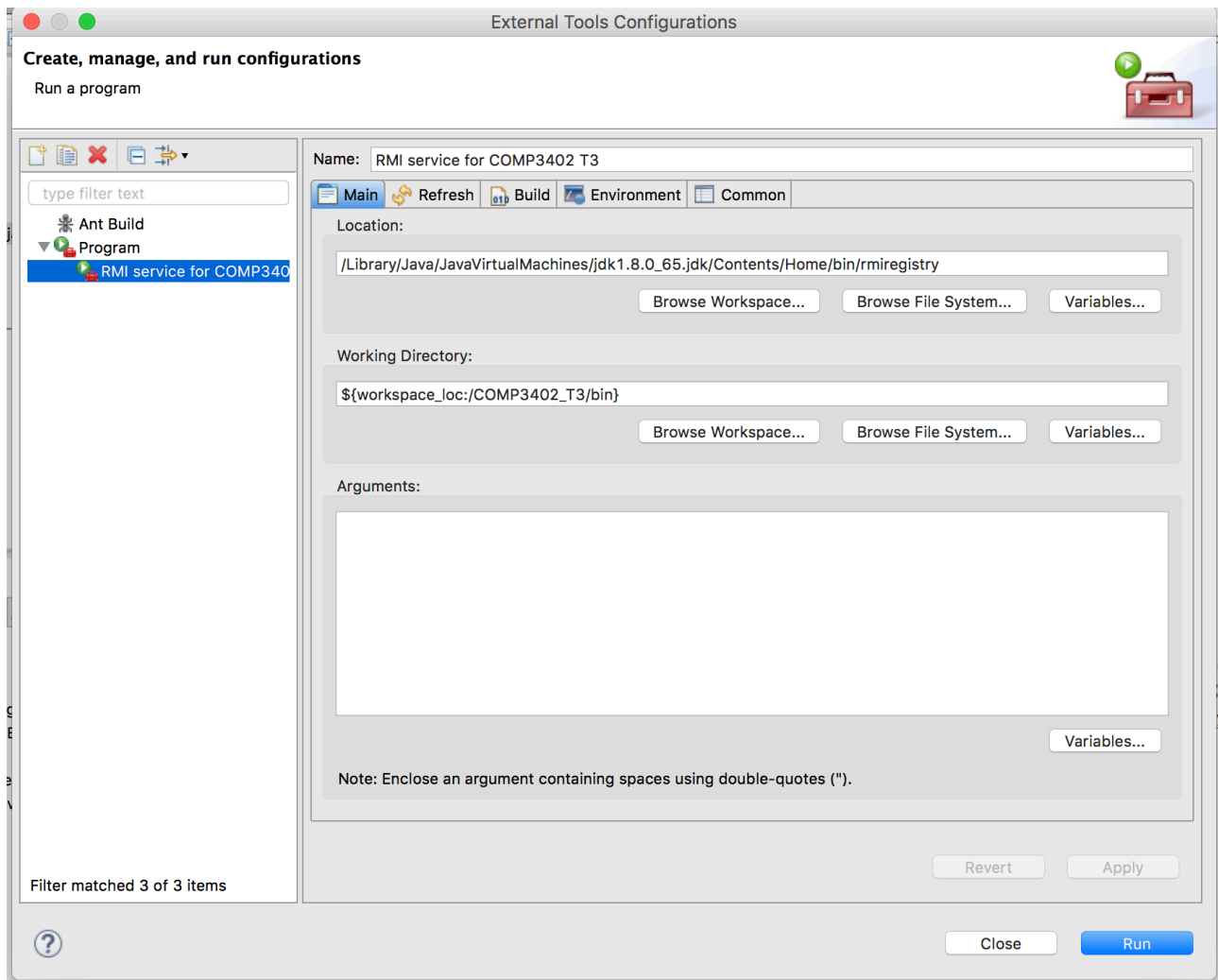
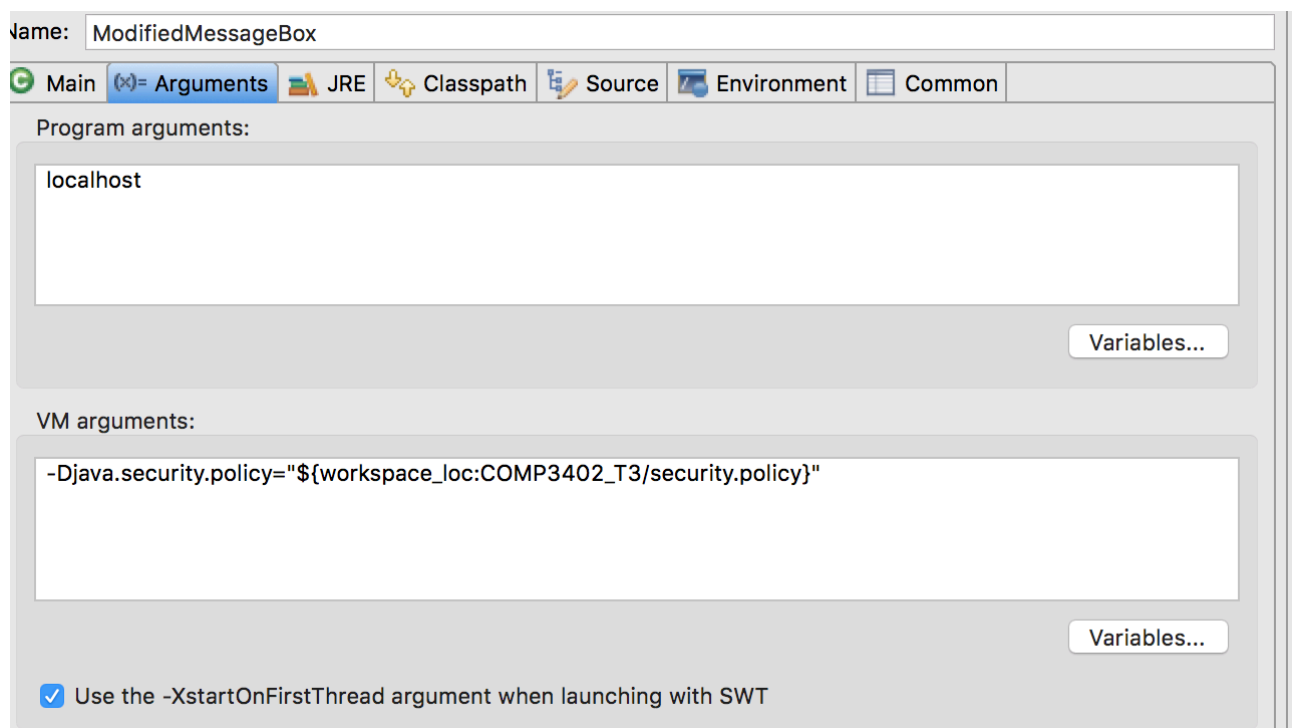
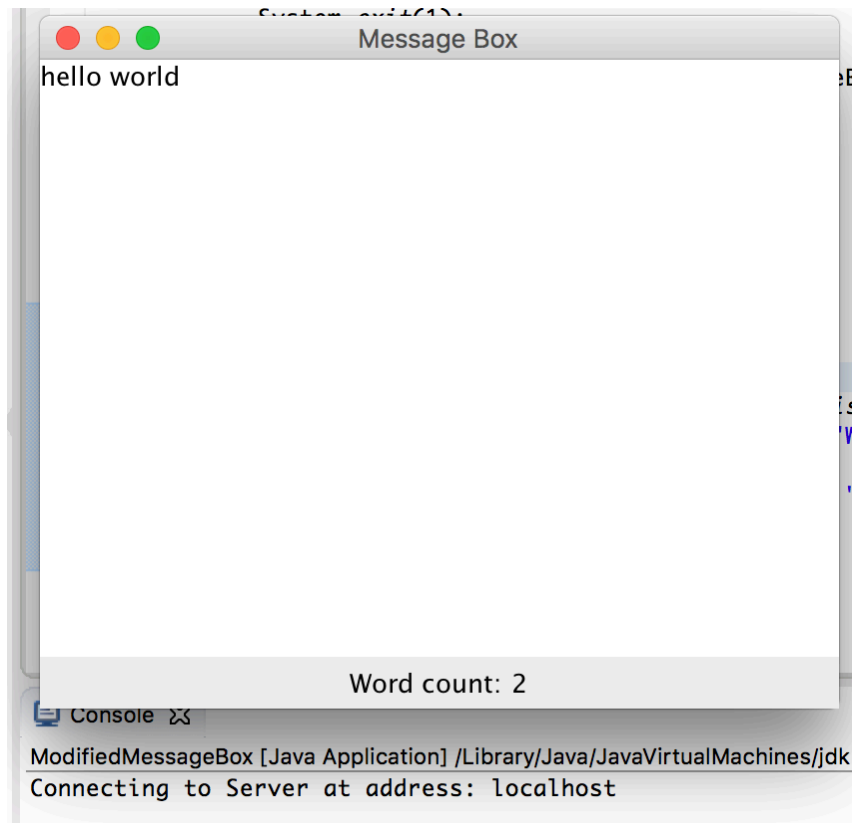


# COMP3402 Tutorial 03 Topic: RMI (Remote Method Invocation)

## 1. [Step 1: Setup RMI registry in Eclipse]



2. [Step 2: Implement WordCount client] : See attached file **ModifiedMessageBox.java** for detail



### 3. [Step 3: Implement WordCount Server] : See attached file **WordCounter.java** for detail

```
import java.rmi.*;
import java.rmi.server.*;

public class WordCounter extends UnicastRemoteObject implements WordCount
{
    public static void main(String[] args)
    {
        try {
            WordCounter app = new WordCounter();
            System.setSecurityManager(new SecurityManager());
            Naming.rebind("WordCounter", app);
            System.out.println("Service registered");

            // int count = app.count("The quick brown fox jumps over a lazy dog");
            // System.out.println("There are "+count+" words");
        } catch (Exception e) {
            System.err.println("Exception thrown: "+e);
        }
    }

    public WordCounter() throws RemoteException
    {
    }

    public int count(String message) throws RemoteException
    {
        return message.split(" ").length;
    }
}
```

Console [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0\_65.jdk/Contents/Home/bin/java (Feb 25, 2017, 5:07:53 PM)  
Service registered

Name: WordCounter

Main Arguments JRE Classpath Source Environment Common

Program arguments:

Variables...

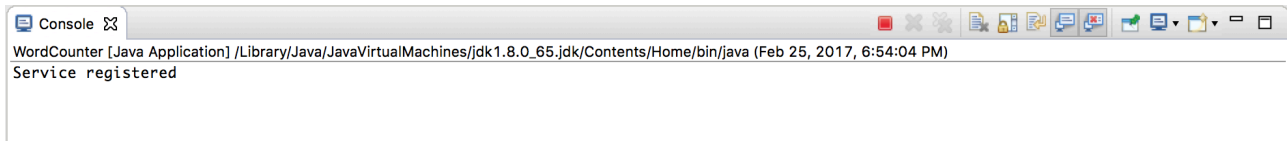
VM arguments:

-Djava.security.policy="\${workspace\_loc:COMP3402\_T3/security.policy}"

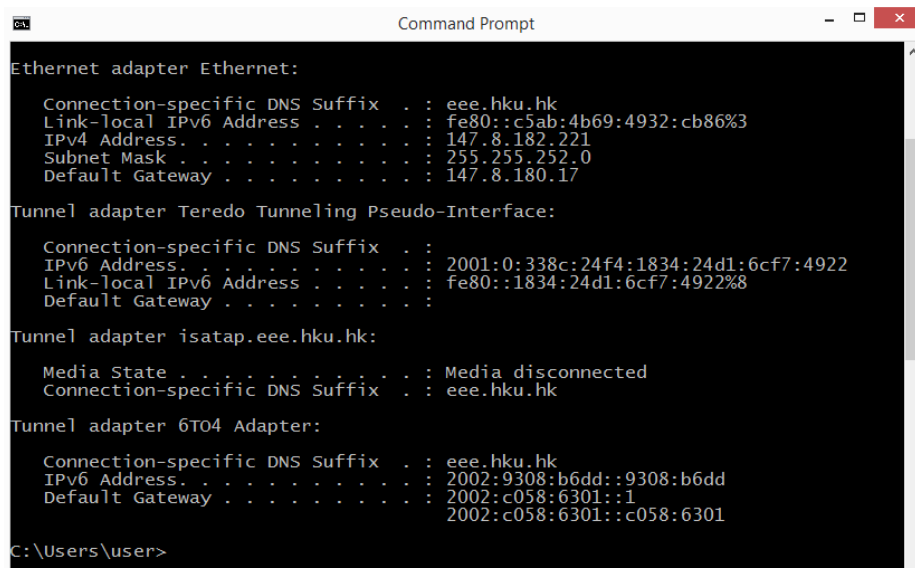
Variables...

☒ Use the -XstartOnFirstThread argument when launching with SWT

4. [Exercise: Running WordCount server on different machine]:
  1. Server machine: MAC, Client machine: Windows
  2. Run rmiregistry service on server machine [IP address: 147.8.115.109 ]
  3. Run WordCounter program on server



4. Run client application on a Windows machine [IP address: 147.8.182.221]



5. The WordCounter service runs successfully on client side.

