

I/O and Simple NetWork Programming

Li Chengyao

5140219288

May 28, 2017

1 PROBLEM

1.1 DESCRIPTION

Create a class WGet that takes a URL and a filename on the command line, and attempts to retrieve the data at that URL into the specified file. For example, "java WGet http://www.cs.columbia.edu/index.html myfile.html", then you are retrieving the file at CS website and store it into your own file myfile.html.

It should fail if the filename you are trying to write to is already existing, and fail with an informative message if something goes wrong (i.e. don't just dump a stack trace).

You can assume that the data is binary, i.e. don't worry about whether to use a Reader, just use an InputStream. You can also assume the file is text only. Just notify it in your code.

2 ALGORITHM

2.1 INPUT STREAM FROM URL

In order to retrieve data from a URL we use package *java.net.HttpURLConnection* and *java.net.URL* to found a connection between local machine and web. Then we use the self-implemented method *readInputStream()* to download data to a local buffer.

2.2 OUTPUT STREAM TO LOCAL FILE

In order to save the data to a local file, file I/O stream is useful. We use class *FileOutputStream* to open an newly-created file, then use method *write()* to save data to the file. In order to avoid editing an existing file we implement a class *ExistingFileException* to deal with such circumstances.

```
F:\OS Project\URLParse\src\homework4>java WGet http://blog.csdn.net/mageshuai/article/details/3849143 myfile.html
http://blog.csdn.net/mageshuai/article/details/3849143 download success.
```

Figure 3.1: Screenshot of Command Line Window

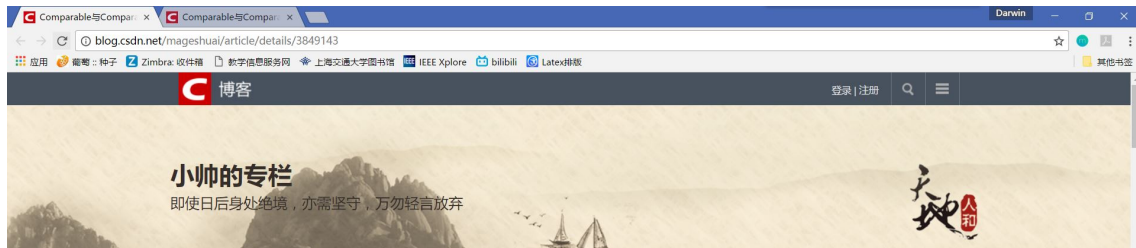


Figure 3.2: Screenshot of Original Web

3 RESULTS

3.1 ENVIRONMENT

- Windows 10
- Java Development Kit 1.8.0_131
- Eclipse
- Chrome

3.2 SCREENSHOTS OF THE RESULT

We use command line to compile and execute the program. The result is shown in Fig. 3.1, Fig. 3.2 and Fig 3.3.

3.3 THOUGHTS

This project helps us to learn basic skills about web programming in Java. It's very interesting because it combines web programming and file I/O programming.

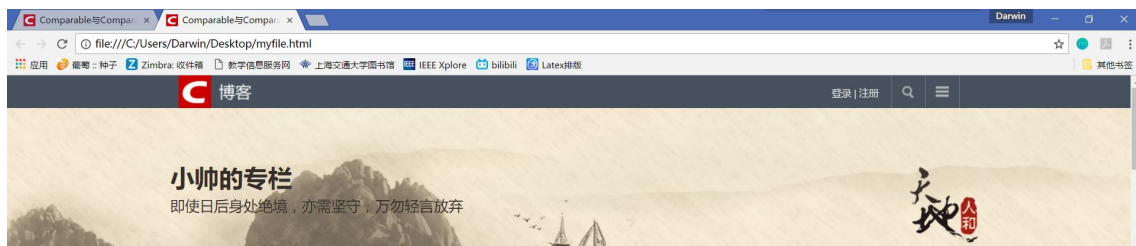


Figure 3.3: Screenshot of Local HTML File