Lab2 – Vectors, Exceptions & I/O Streams

CSC 245 – Objects & Data Abstraction – Dr Danielle Azar

Prepared by: Tatiana Abou Assi
2/11/2015

Concepts:

How to use Vectors:

Vector proves to be very useful if you don't know the size of the array in advance or you just need one that can change sizes over the lifetime of a program.

1. The Vector class has a constructor which creates a vector whose initial capacity is specified by size:

```
import java.util.Vector;
Vector v = new Vector(int size);
```

2. To get the size of the vector, we use:

```
v.Size();
```

3. To add elements to the Vector, we use:

```
v.add("Paul");
```

4. To remove elements from the Vector, we use:

```
v.remove("Paul");
or
v.remove(int index);
```

5. To get an element from the Vector, we use:

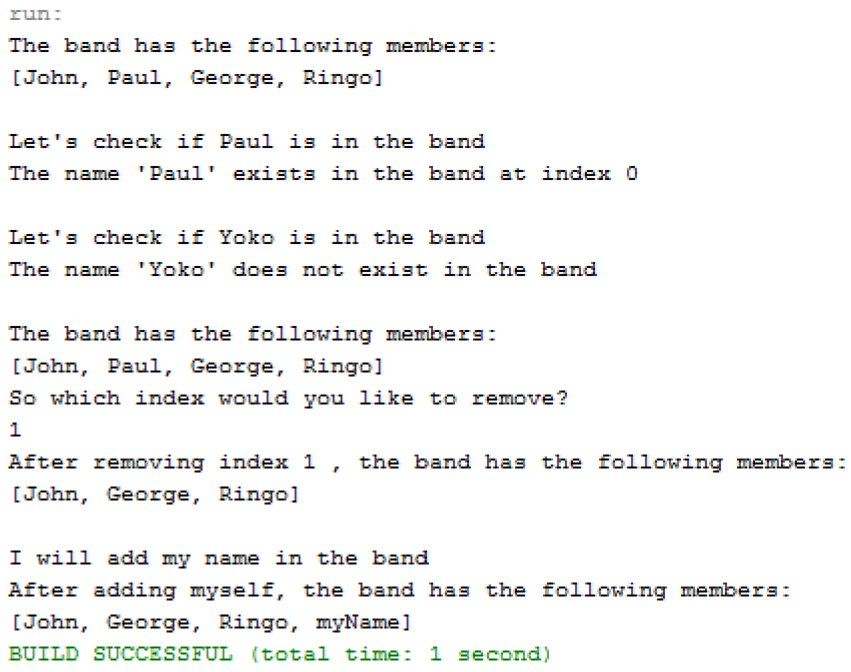
```
v.get(int index);
```

6. To find out more functions for the Vector, simply type "v." (v and dot) and wait for the functions to be listed.

Write a program which has one Vector which has size of 5.
Your program should do the following:

1. Create a vector having four names
2. Print the vector
3. Check whether a name exists in the vector: if the name is found, then print “the name ... exists in the vector at index ...” otherwise print “the name ... does not exist”
4. Print the vector
5. Ask the user to input a valid index to remove
   Hint: import java.util.Scanner;
6. Print the vector after removing this index
7. Add your name
8. Print the vector

The program should run as follows:



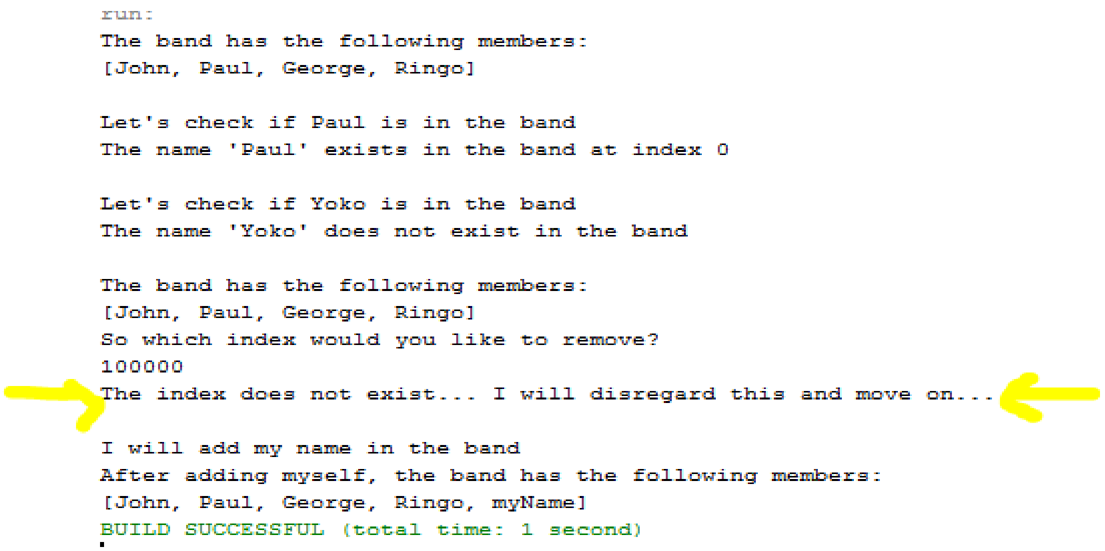
```
run:
The band has the following members:
[John, Paul, George, Ringo]

Let's check if Paul is in the band
The name 'Paul' exists in the band at index 0

Let's check if Yoko is in the band
The name 'Yoko' does not exist in the band

The band has the following members:
[John, Paul, George, Ringo]
So which index would you like to remove?
1
After removing index 1 , the band has the following members:
[John, George, Ringo]

I will add my name in the band
After adding myself, the band has the following members:
[John, George, Ringo, myName]
BUILD SUCCESSFUL (total time: 1 second)
```

How to deal with Exceptions:

To demonstrate, let us use the example of the program you just wrote. What if in our program, in step5, where the user is asked to give a valid index, he/she give a non-valid index for example 100. In such a case, does your program break, or does it handle such exceptions? Should the program continue and skip this step?

Let us consider that in this case we want the program to skip the removal step and continue normally. **Add the try/catch block** in such a way to simply output "The index does not exist… Let us move on…" in case ArrayIndexOutOfBoundsException is encountered.

The program should run as follows:



```
run:
The band has the following members:
[John, Paul, George, Ringo]

Let's check if Paul is in the band
The name 'Paul' exists in the band at index 0

Let's check if Yoko is in the band
The name 'Yoko' does not exist in the band

The band has the following members:
[John, Paul, George, Ringo]
So which index would you like to remove?
100000
The index does not exist... I will disregard this and move on...

I will add my name in the band
After adding myself, the band has the following members:
[John, Paul, George, Ringo, myName]
BUILD SUCCESSFUL (total time: 1 second)
```

Notes:

1. One try can have several catch blocks

```
try
{
   //code to test on
}catch(ExceptionType1 e1)
{
   //Catch block 1
}catch(ExceptionType2 e2)
{
   //Catch block 2
}catch(ExceptionType3 e3)
{
   //Catch block 3
}
```

2. If needed, we can define our own Exception class as below:

```
class MyException extends Exception{
}
```

How to use I/O Streams:

There are many ways of working with files in Java. File streams are perhaps the easiest streams to understand. The file streams-- FileReader, FileWriter, FileInputStream, and FileOutputStream--each read or write from a file on the native file system. You can create a file stream from a file name in the form of a string, a File object, or a FileDescriptor object.
The following Copy program uses FileReader and FileWriter to copy the contents of a file named farrago.txt into a file called outagain.txt:

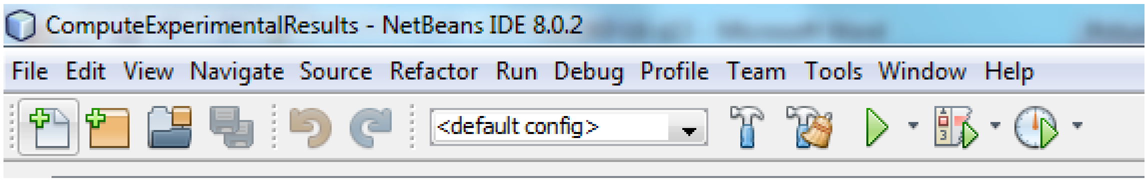
```
import java.io.*;
public class Copy {
   public static void main(String[] args) throws IOException {
       File inputFile = new File("farrago.txt");
       File outputFile = new File("outagain.txt");

      FileReader in = new FileReader(inputFile);
      FileWriter out = new FileWriter(outputFile);
      int c;

      while ((c = in.read()) != -1){
         out.write(c);
       }

      in.close();
      out.close();
   }
}
```

This program is very simple. It opens a FileReader on farrago.txt and opens a FileWriter on outagain.txt. The program reads characters from the reader as long as there's more input in the input file and writes those characters to the writer. When the input runs out, the program closes both the reader and the writer.

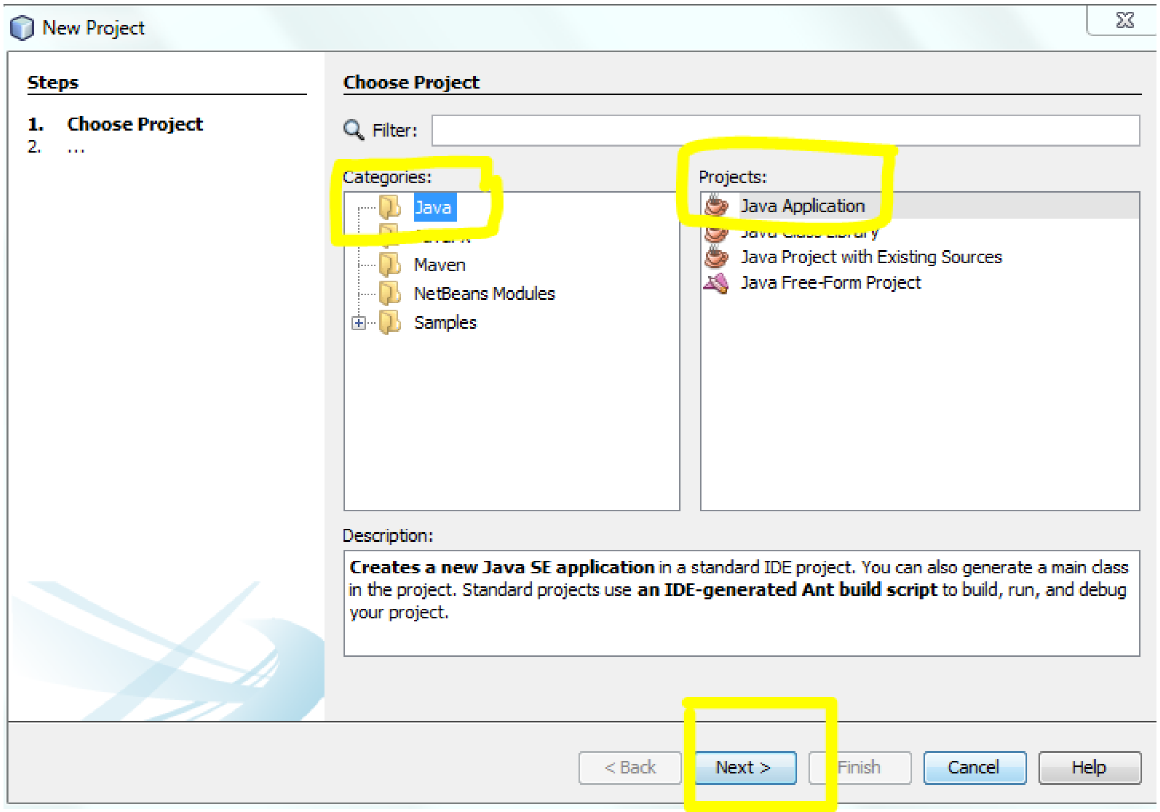
Write a java programs that reads from a text file and creates another file were all lower case characters are transformed to upper case. It is recommended for this assignment to use the BufferdReader class as it includes a readline() method.

Reminder: How to create a NetBeans Project

1. Click on "File" in the top of NetbBeans window.
   Then, click on "New Project"

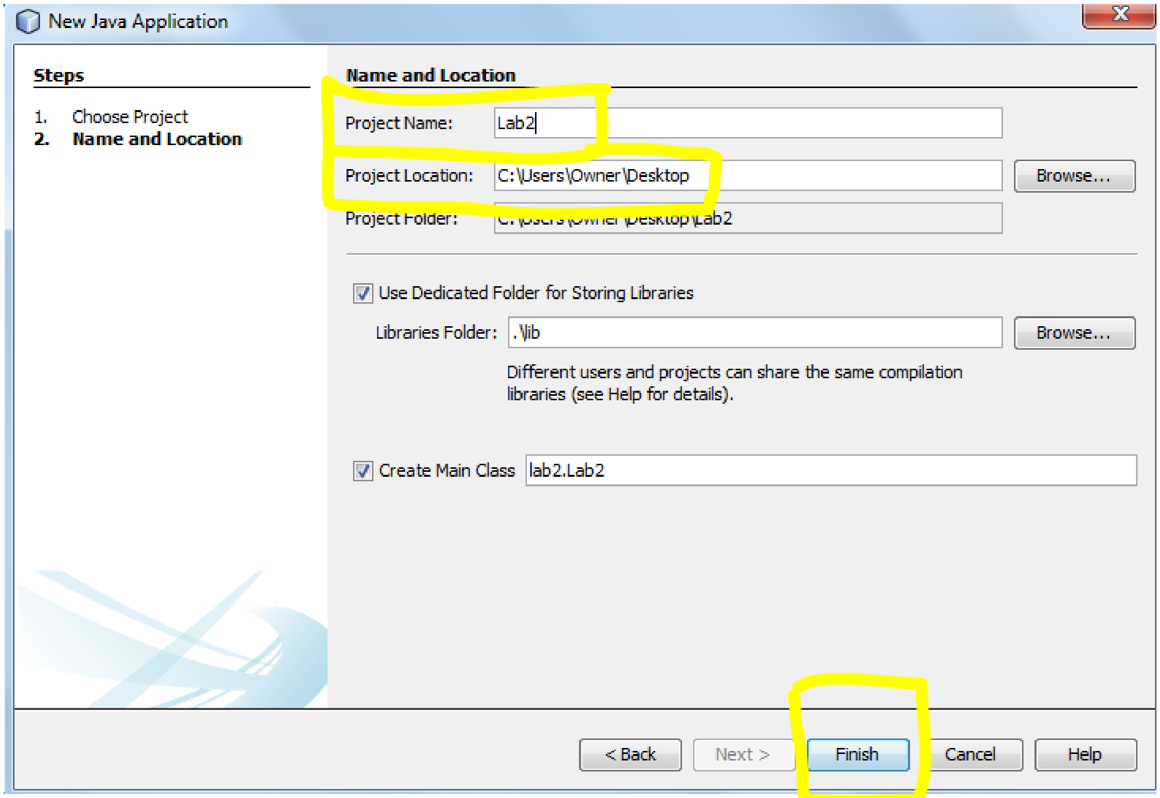

ComputeExperimentalResults - NetBeans IDE 8.0.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>

2. Under "Categories", click on "Java"
   Under "Projects", click on "Java application"
   Click "Next"



Creates a new Java SE application in a standard IDE project. You can also generate a main class in the project. Standard projects use **an IDE-generated Ant build script** to build, run, and debug your project.

3. Name your project under "Project Name"
   Check where the project is going to be located under "Project Location". You need to know the location in case you need to submit your project.
   Then, click "Finish"



New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name: Lab2

Project Location: C:\Users\Owner\Desktop

Project Folder: C:\Users\Owner\Desktop\Lab2

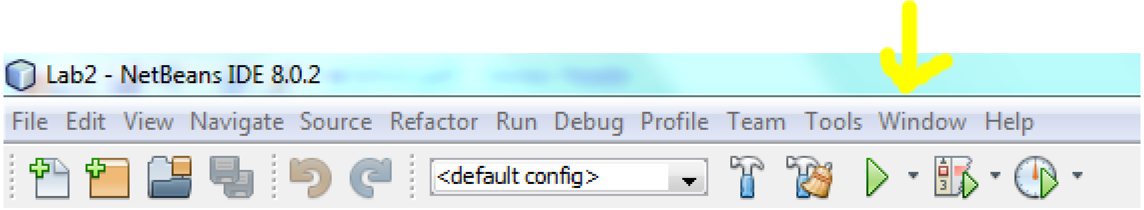
☑ Use Dedicated Folder for Storing Libraries

Libraries Folder: .\lib

Different users and projects can share the same compilation libraries (see Help for details).

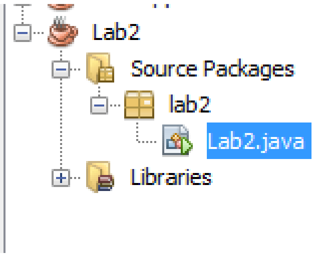
☑ Create Main Class lab2.Lab2

4. Optional: Click on "Windows", then "Project" to view the project window on the left



Lab2 - NetBeans IDE 8.0.2

Now, you are able to view something similar to the below image.



Lab2
Source Packages
lab2
Lab2.java
Libraries