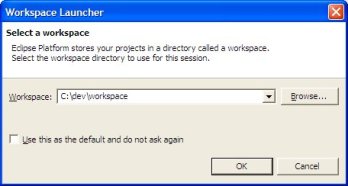
Lab 1.

**Part 1: Eclipse – How to?**

**Step 1**: Download the file Demo.java from the class site in D2L. In this file, two arrays of a random size are created for array1 and array2. array1 is filled with random values between 0 and MAX\_SIZE / 2 . These same values are copied into array2. Here the two arrays differ: array2 is sorted.

**Step 2**: Start eclipse by clicking on the icon of **eclipse.**

First start up Eclipse and supply a path to a new folder which will serve as your workspace. The workspace is a folder which Eclipse uses to store your source code.  
  


### This figure demonstrates how to start up eclipse. Avoid checking the "Use this as the default..." box, which will always start up Eclipse with the same workspace. You'll probably find it useful to be able to switch between different workspaces on startup. When Eclipse starts, you'll see the Welcome page. Close the Welcome page. Right-click select File-> New->Project. Select "Java Project" and then "Next". Enter the name of your project: Lab. We will use this project to contain the all labs that we have in this course. Click "Finish". If you get prompted about switching to the Java perspective, select "Yes".

### Step 3: Create a new folder named Lab1 under Lab project by selecting Lab project and clicking right mouse. Choose New -> Folder. Enter the folder name: Lab1.

### Step 4: Selecting Lab1 and clicking right mouse. Choose Import -> and select File System in the Import windows. You will need to specify the directory of the source file. Browse it to the folder where you just saved your Demo.java file. In the left hand side you will see the directory and in the right hand side you will see the list of \*.java and \*.class files, as well as other files in this folder. Choose Demo.java file and make sure you choose the option: Create Selected folders only. Then click on Finish.

### Step 5: Now you have successfully load your Demo.java file into eclipse. You will see something similar to this picture. Please cut and paste your eclipse screen at this point and attach it to the final report. (You can use PrintSCrn button on your keyboard to capture the whole screen)

### 

### Step 6: Click on Demo.java in eclipse. Choose Project -> Clean. Make sure you choose: Clean selected projects below, and make sure that Lab project is chosen and the option “Start a build immediately” checked. Click on OK button to clean and rebuild/compile this project.

### Step 7: Choose Run -> Run As and Choose Java Application. You will be able to run this demo.java class.

**Step 8**: While you run Demo.java class, carry out 10 searches for values, and pay attention to the results of the searches, both successful and unsuccessful. Please note down the searches in the table 1.

For array size:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Array 1** | | | **Array 2** | | |
| Search key | Found/Not Found? | Numbers of comparisons made? | Search key | Found/Not Found? | Numbers of comparisons made? |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

### Step 9: The final report will contain your screenshot of your eclipse (after you import Demo.java) and the fully filled table 1. Please submit the final report in D2L, Lab1 dropbox by Sunday midnight at the latest.

### Part 2: (DO IT YOURSELF) How to use Eclipse for development a Java program

***Step 10:*** *Task description: Forty students were asked to rate the quality of the food in the student cafeteria on a scale of 1 to 10 (1 meaning awful and 10 meaning excellent). Place the 40 responses in an integer array and summarize the results of the poll.*

Here are the 40 responses

1,2,6,4,8,5,9,7,8,10, 1,6,3,8,5,10,3,8,2,7,6,5,7,8,1,2,4,5,6,7, 10,4,3,2,5,6,7,8,9,3

In Eclipse, develop a Java program to solve this problem. You can create this class under the same Lab1 package that you have created in Part 1. This program contains one class named SurveyStat in which, you should have two data members

An array named *responses* to contains these 40 responses

An array named *frequency.* and assign all elements of these array to be zero

Four methods:

Constructor: where you initialize all elements of frequency array to be zero

Compute: in which the computation is done. For each response in the *responses* array, update the corresponding frequency element in *frequency* array.

Print: in which you print out the content of the frequency array

main method: in which you call the Compute and then Print methods.

Note that these are just the main guidelines for you to finish this task, you will need to fill in all the details to make this work.

**Step 11.** Submit your SurveyStat.java to Lab1 folder in D2L by midnight of the due date.