

## Lab Practice Week 1

---

Students should install NetBeans IDE 8.2 software (this software package includes the appropriate Java Development Kit). Unless you wish to use the command line interface for your work, just installing NetBeans IDE 8.2 will install the required Java functionality. See the extra notes available on LMS for more information.

**What to submit:** your solutions to exercises 3, 4, 5, 6, and 8 (below).

You should submit your Lab Practice work on a weekly basis to your tutor. Your tutor will expect to see them by your next session. Do not fall behind or leave it to end of the term as the workload will be very heavy towards the end of the term.

*Note: even though you only need to submit those exercises mentioned above, you should attempt all exercises in each lab practice to help broaden your understanding; this may also help with your assignment work.*

---

## Exercises Assignment Project Exam Help

Before starting the exercises, make sure you have read the relevant material.

As many of the exercises involve modifying previous answers, it is strongly suggested that (in such cases) you copy your project from a previous exercise, then paste and rename it for your current exercise. You can then modify as needed.

1. Create a directory in which to do your work. Download the program *SmallIO.java* from the unit LMS website and save the file into your working directory. Note that this application uses the Scanner class from the Java.util package, which you would need to import into your program (refer lecture notes examples).
2. Using the NetBeans IDE development environment, create and build a project using the source file that you just downloaded (*SmallIO.java*). Execute the program.
3. Using the IDE make a new project to modify the previous program so that it now repeats the loop five times and then stops.
4. Using the IDE make a new project to modify the previous program so that integers instead of Strings are entered by the user.
5. Using the IDE make a new project to modify the previous program so that it only loops until the number -999 is entered by the user.
6. Using the IDE make a new project to modify the previous program so that it:
  - a) Keeps and displays a running total and the average so far of the entered numbers.
  - b) Displays the largest and the smallest numbers so far of the entered numbers.
7. Download the program *TwoDPs.java* from the unit LMS website and save the file in your working directory. Use the NetBeans IDE development

environment, create and build a project using the source file that you just downloaded. Execute the program.

8. Create a new project called *Trunc.java* either by copying and modifying *TwoDPs.java* or by typing that source code from scratch. *Trunc.java* should loop around getting floating point (i.e. type double) numbers and displaying both the number and the running total to the nearest whole number. The program will finish when a number is entered outside of the range -100 to 100. The program should use methods to check whether the number is in this range, and then rounded to the nearest whole number. (NOTE: You are **not** to use the JAVA Round method, write your own method by modifying the *calcTwoDPs* method in *TwoDPs.java*).

**Assignment Project Exam Help**

**<https://powcoder.com>**

**Add WeChat powcoder**