Installing Java Development Kit (JDK) and Introducing Basics of Elementary Programming in Java

Lab Objective

Familiarize students with JDK installation and basic program structure in Java.

Lab Outcomes

After completing this lab successfully, students will be able to:
1. **Install** JDK and **compile** and **execute** a Java program.
2. **Understand** basic program structure in Java and **solve** a few simple problems in Java.

Psychomotor Learning Levels

This lab involves activities that encompass the following learning levels in psychomotor domain.

Level	Category	Meaning	Keywords
P1	Imitation	Copy action of another; observe and replicate.	Relate, Repeat, Choose, Copy, Follow, Show, Identify, Isolate.
P2	Manipulation	Reproduce activity from instruction or memory	Copy, response, trace, Show, Start, Perform, Execute, Recreate.

Lab Activities

A. Downloading and Installing JDK (Java Development Kit)

- JDK is required to be installed in your machine to compile and execute a Java code.
- JRE (Java Runtime Environment) is needed to be installed in your system to run a Java program.
- We need to install JDK since we are going to write, compile and execute Java programs.
- To download JDK 1.8 go to the following link:
  https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
- After downloading, run the *exe* file. It will start the installation procedure.
- The procedure may take a few minutes depending on the configuration of your machine.
- Once the installation procedure is finished, we need to configure the PATH system variables so that Java code can be compiled and executed from anywhere in the hard disk.
- To do that, right click on This PC or My Computer icon on the desktop and then click properties. You will see the following window opens as shown in Figure 1.
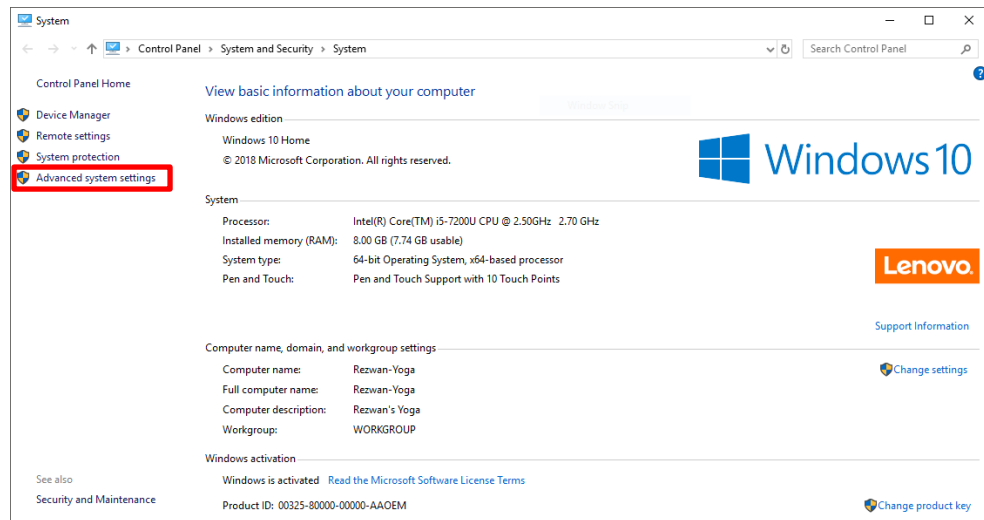
Figure 1: System Window

- Click on Advanced system setting as highlighted. System Properties window pops up as shown in Figure 2.
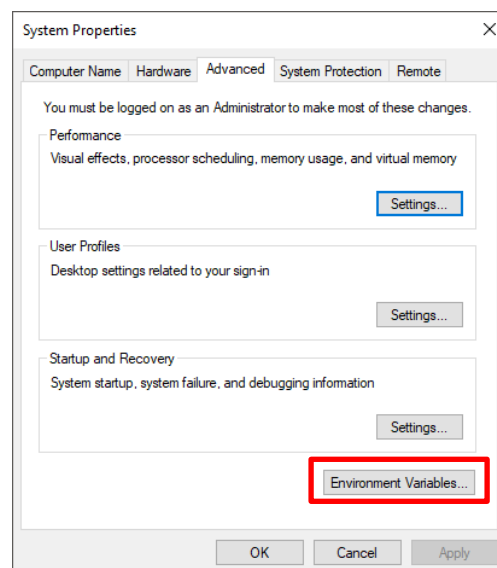

Figure 2: System Properties Window

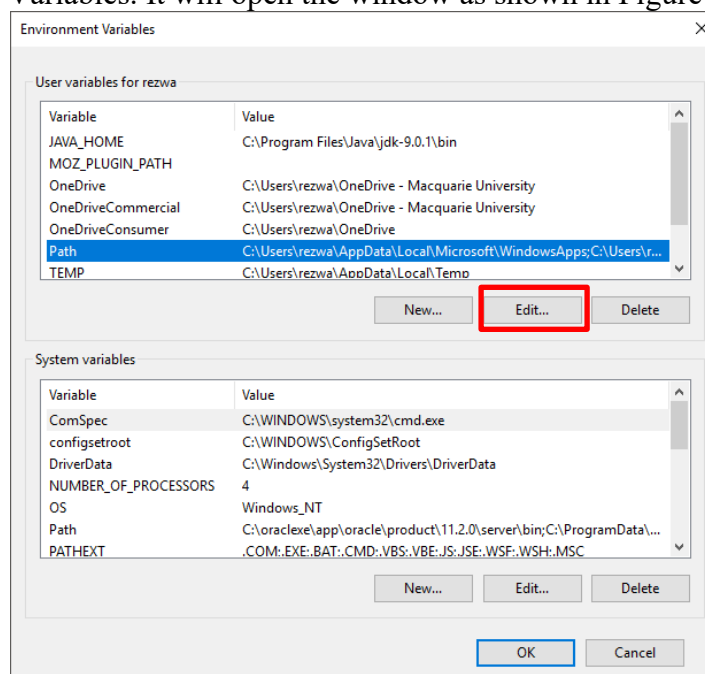- Click on Environment Variables. It will open the window as shown in Figure 3.

Figure 3: Environment Variables Window

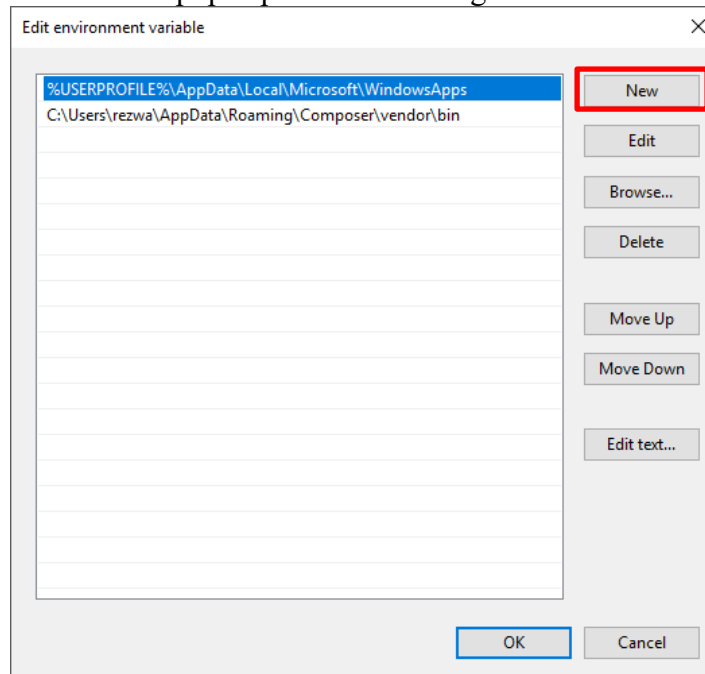- Now, click on Edit. A new window pops up as shown in Figure 4.

Figure 4: Edit environment variable Window

- Press New and then paste the path where the JDK has been installed. Usually, the path is - C:\Program Files\Java\jdk-XXX\bin
- We are done and now it is time to compile and execute our first program.

B. Writing, Compiling and Executing the First Program
- Open notepad or notepad++ or any text editor of your choice.
- Write the following as it is shown below:

```
class Sample{
    public static void main (String[] args){
        System.out.println("Hello World");
    } // main method ends
} // Main class ends
```

- Save the code as Sample.java inside a folder/directory.
- Now, open a command prompt. Suppose, your program is saved in the location – D:\DMRH. Then, write the followings in the command prompt and press enter after every line as shown in Figure 5.

Figure 5: Compiling and Executing a Java Program

- *cd* (change directory) is used to change the directory from the current location.
- After coming to the location where the Sample.java is saved, compile the program using *javac*.
- Then execute the program using *java* command as shown in Figure 5.
- **Write a Java program that prints your name, age, cgpa and department, each in a single line.**

C. Reading inputs from user

- Everything in Java comes in form of a class.
- To read inputs from a user, we need to use *Scanner* class.
- The following program reads name, age and department name of a student and print them accordingly.

```java
import java.util.Scanner;
class SampleReadInput{
   public static void main (String[] args){
       Scanner input = new Scanner (System.in);
        System.out.println("Enter your name: ");
        String name = input.next();
        System.out.println("Enter your age: ");
        int age = input.nextInt();
        System.out.println("Enter your CGPA: ");
        double cgpa = input.nextDouble();
        System.out.println("Enter your department: ");
        String department = input.nextLine();
        System.out.printf("Your Name: %s\n", name);
        System.out.printf("Your Age: %d\n", age);
        System.out.printf("Your CGPA: %f\n", cgpa);
        System.out.printf("Your Deparmtent: %s\n", department);
    } // main method ends
} // Main class ends
```

- **Does the program execute as we have wanted? What is the problem? How can you solve it?**

D. Revising Conditional Statements and Looping

if-else	*switch*

`if(condition){`
// do something
`}`
`else{`
// do something else
`}`
	`switch(case){`
	case 1:
	// do things for case 1
	break;
	case 2:
	// do things for case 2
	break;
	default:
	// do things for default case
	`}`

for	*while*

`for(int i = 0; i < 10; i++){`
// do something
`}`
	`while(condition){`
	// do something
	`}`

Programming Problems

Lab01_Problem01: Write a program that displays **Welcome to Java**, **Welcome to Computer Science**, and **Programming is fun** at three different lines.

Lab01_Problem02: Write a program that displays **Welcome to Java** five times in five lines.

Lab01_Problem03: Write a program that displays the area and perimeter of a circle that has a radius of **5.5** using the following formula:

$$perimeter = 2 * \pi * radius$$
$$area = \pi * radius * radius$$

Lab01_Problem04: Change the program in the previous problem (Lab01_Problem03) in such a way so that the user gives the value of the radius of the circle. Now compute the perimeter and area.

Lab01_Problem05: Assume a runner runs **14** kilometers in **45** minutes and **30** seconds. Write a program that displays the average speed in miles per hour. (Note that **1** mile is **1.6** kilometers.)

Lab01_Problem06: Change the program in the previous problem (Lab01_Problem05) in such a way so that user gives three values: distance the runner completed in km, minutes and seconds spent during the run. Now compute the average speed in miles per hour.

Lab01_Problem07: Write a program that reads a Celsius degree in a **double** value from the console, then converts it to Fahrenheit and displays the result. The formula for the conversion is as follows:

$$fahrenheit = (9 / 5) * celsius + 32$$

Lab01_Problem08: Write a program that reads an integer from the console and determines whether the given number is divisible by either 2 or 3 (but not both). Then the program should print TRUE, otherwise, the program should print FALSE.

Lab01_Problem09: Write a program that prompts the user to enter the minutes (e.g., 1 billion), and displays the number of years and days for the minutes. For simplicity, assume a year has **365** days. Here is a sample run:

```
Enter the number of minutes: 1000000000 ↵Enter
1000000000 minutes is approximately 1902 years and 214 days
```

Lab01_Problem10: Write a program that reads an integer and adds all the digits in the integer. For example, if an integer is **932**, the sum of all its digits is **14**.

```
Enter a number between 0 and 1000: 999 ↵Enter
The sum of the digits is 27
```