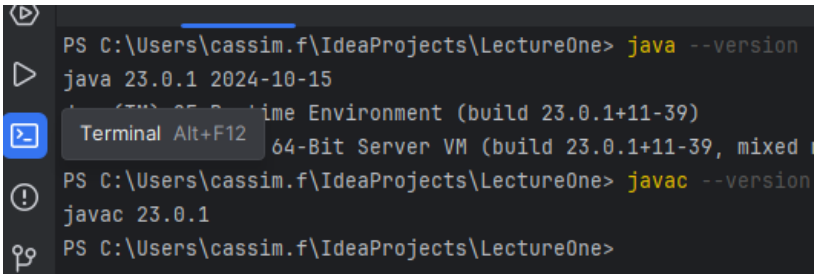
Tutorial Semester Two – 01	
CM1601: Programming Fundamentals	
Module Leader	Cassim Farook
Stage	1
Academic Year	2025/26
Prepared by	Cassim Farook (28.01.2026)
You will gain knowledge about the following concepts at the end of the tutorial: • Creating a Project using Intelli J to compile a java program to executing it. • Creating variables in Java within the main method. • Using String and Integer variables for Output, Input, arithmetic operations in java • Convert a design to a java program	

Lab Sheet

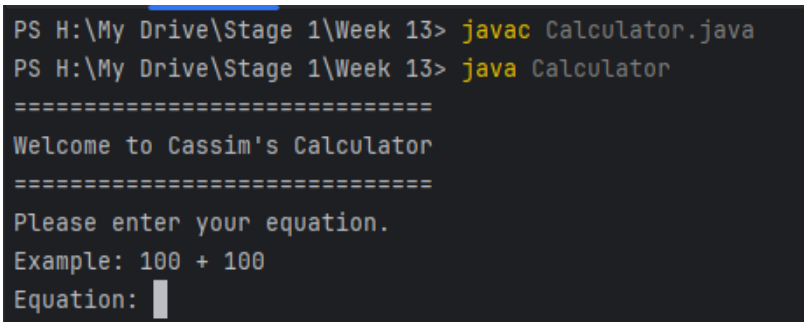
Doing (Assisted): Find the java virtual machine and the java compiler versions. You can run the following commands on a windows or mac os terminal. You can also open a terminal within Intelli J.

```
java --version
```

```
javac --version
```



Doing (Assisted): Download the "Calculator.java" , compile and run the program using the terminal. You will notice that it's a two step process. Compile the Calculator.java to a Calculator.class file. You will see this new fill getting created with a ".class" in the same directory. Then tell the JVM to execute the compiled class file. Your tutor will demonstrate, and you will follow. (Note: avoid file paths with spaces, even though the image shows otherwise)



Analysing (compare and contrast):

Using the Calculator Java version - the tutor will take you through the structure of a Java code program and will explain to you the following.

a. Code comments - Single line comments – "//" used instead of "#". Then Multi-line /*** **/
b. Import statement
c. Semicolon – used to tell the end of a statement.
d. Main method – entry point to the program.
e. Public keyword – allows to expose a function to be used outside of the class.
f. Static keyword - allows to call a function without creating an object of its class, simulating a global function.
g. System.out.print
h. System.out.println
i. System.out.printf – allows formatted output and variable place holders (later tutorials will use it)

Doing (Assisted): Create a project as "TutorialSemTwoOne". Attempt the rest of the tutorial inside the main method. You already have the python knowledge. It's a matter of converting and learning the Java syntax – Your tutor will help you. You can attempt this in groups of three.

1. Output string literal values using **System.out.print()**

a) Print "Hello World". Notice in java you need to use double quotes.

b) Print your name.

c) Print more details about yourself: Frist Name, Last Name, Age.

d) Print the following to the screen

```
I save my files to C:\Documents in a folder called 'Music'.
The details of the folder is as follows.
Name        : Music
Files       : 20
My favourite music is "Titanic".
```

You should use the following escape sequences. (surprise ! same as python !! this is so easy !)

Sequence	Meaning
\\	literal backslash
\'	single quote
\"	double quote
\n	newline
\t	tab

2. **Creating variables and assigning values**. Refer lecture slide deck.

   a) Select suitable primitive variables to hold age, bank balance. Then create code inside the main method.
   b) Create String (non-primitive) variables to hold name, telephone and email. Initialize them with suitable literal values.
   c) Create a **String** variable named "country", assign value "Sri Lanka".
   d) Create a **Integer** variable named "amount", assign value 10.
   e) Create a **int** variable named "expense", assign float value 5.
   f) Try the following code statements:

```
int pounds = 3;
Integer pence = pounds * 100 ;
System.out.println(pence);              //line1
System.out.println(pounds * 100);      //line2
```

3. **Arithmetic Operators with type conversion between different numeric data types**

   a) Try the following code statements of arithmetic operators.

```
System.out.println(6 + 8);        // Output: 14
System.out.println(7 - 2);        // Output: 5
System.out.println(2.5 + 4.8);    // Output: 7.3
System.out.println(5.0 - 4.2);    // Output: 0.8
System.out.println(6 * 3);        // Output: 18
System.out.println(6 / 3);        // Output: 2
System.out.println(6 % 3);        // Output: 0 modulus
System.out.println(6 / 3.0);      // Output: 2.0 (floating point division)
System.out.println(Math.pow(3, 2)); // Output: 9.0 (3 to power of 2)
```

   b) Implicit Type Conversion.

```
int intVal = 10;
double doubleVal = 5.5;
double result = intVal + doubleVal;
// intVal is implicitly converted to double
System.out.println("Result: " + result);  // Output: Result: 15.5
```

   What is the data type of the output of the following code?

```
int x = 8;
int y = 3;
System.out.println(x/y);
```

   c) Explicit Type Conversion

```
Integer a = 10; String b = "3";
```

   What is the output of the following code statement to observe the error it would generate.

```
System.out.print(a + b);
```

   Modify the code to the following to fix the error.

```
System.out.print( a + Integer.parseInt(b));  //Explicit casting using
```

4. **Concatenating string values and variables**

a) Try the following code.

```
String a = "Hello out there!" ;  System.out.print(a);
String b = "Hello again.";       System.out.print(b);
```

Modify the program so that it concatenates (joins) variables a and b and prints the result.

5. **Capture keyboard inputs – many methods exists , import and use the Scanner class.**

a) Try the following code:

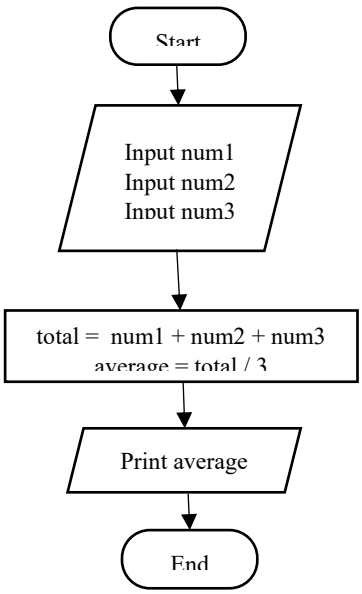
```
Scanner scanner = new Scanner(System.in);
System.out.print("Please enter your name: \n");
String myname = scanner.nextLine();
System.out.print("Hello " + myname);
```

- The "`\n`" within quote marks forces a new line to be printed. Alternatively you can use "println"
- When you run the program, you should see the message "Please enter your name:" in the output window. Type in your name, followed by the ENTER key. The program will greet you.

b) Extend the program to capture the values for variables defined in Question 2.

6. **Convert Design Scenario to a Program**

a) Write a Java program using the flow chart below.



b) Write the Java program using the pseudocode below.

INPUT num_1
INPUT num_2
total <- num_1 + num_2
PRINT total