Lab 1: Introduction to Java Basics

Objectives of Lab1

In this lab, you will practice the foundational concepts of Java covered in your first lecture. The focus will be on:

- Writing your first Java program.
- Understanding and using variables and data types.
- Working with basic operators and expressions.
- By the end of this lab, you will have a clear understanding of how to write basic Java code, declare variables, use data types, and manipulate data using operators.

Exercise 1: Setting Up Your First Java Program

1. **Task**: Create a new project in IntelliJ, and write a Java program that prints "Hello, World!".
    - **Hint**: Create a new class and include the `public static void main(String[] args)` method.
    - **Expected Output**:

      ```
      Hello, World!
      ```

Exercise 2: Declaring a Class and Main Method

2. **Task**: Declare a new class named `FirstClass` and within that, declare the `main` method as discussed in the lecture.
    - **Expected Output**: No specific output required yet, but the code should compile without errors.

Exercise 3: Understanding Code Blocks and Statements

3. **Task**: Within the `main` method, write a statement that outputs your name to the console using `System.out.println()`.
    - **Expected Output**:

      ```
      My name is [Your Name]
      ```

Exercise 4: Declaring Variables and Printing Their Values

4. **Task**: Declare an integer variable `age` and set its value to your age. Then, print it using `System.out.println()`.
    - **Expected Output**:

```
My age is: [Your Age]
```

Exercise 5: Working with Primitive Data Types

5. **Task**: Declare variables for different data types: `byte`, `short`, `int`, `long`, `float`, `double`, `char`, and `boolean`. Assign values to each and print them.
   - **Expected Output**:

```
Byte value: [Your Byte Value]
Short value: [Your Short Value]
Int value: [Your Int Value]
Long value: [Your Long Value]
Float value: [Your Float Value]
Double value: [Your Double Value]
Char value: [Your Char Value]
Boolean value: [Your Boolean Value]
```

Exercise 6: Wrapper Classes and MIN/MAX Values

6. **Task**: Use the wrapper class for `int` (Integer) to print the minimum and maximum values.
   - **Hint**: Use `Integer.MIN_VALUE` and `Integer.MAX_VALUE`.
   - **Expected Output**:

```
Min int value: -2147483648
Max int value: 2147483647
```

Exercise 7: Demonstrating Overflow and Underflow

7. **Task**: Assign the value `2147483647` to an `int` variable and add 1. Print the result to observe overflow. Repeat for a `byte` variable.
   - **Expected Output**:

```
Maximum value of int: [originally assigned value]
Maximum value of byte: [originally assigned value]
Overflow example with int: [Overflowed Value]
Overflow example with byte: [Overflowed Value]
Underflow example with int: [Underflowed Value]
Underflow example with byte: [Underflowed Value]
```

Exercise 8: Working with Long and Double

8. **Task**: Declare a `long` variable with the value `10000000000L` and a `double` variable with a value of your choice. Print both values. Demonstrate why adding `L` is important for long data types.
   - **Expected Output**:

```
Long value: 10000000000
Double value: [Your Double Value]
```

Exercise 9: Conversion between Types (Casting)

9. **Task**: Convert a `double` value (e.g., 5.99) to an `int` and print the result.
   - **Expected Output**:

```
Double value: 5.99
After casting to int: 5
```

Exercise 10: Converting Pounds to Kilograms

10. **Task**: Write a program to convert pounds to kilograms (1 pound = 0.45359237 kg). Declare a `double` variable for pounds and calculate the equivalent kilograms. Print both values.
    - **Expected Output**:

```
Pounds: [Your Pounds Value]
Kilograms: [Calculated Kilograms]
```

Exercise 11: Working with Char and Boolean

11. **Task**: Declare a `char` variable and assign a Unicode character. Also, declare a `boolean` variable and assign it `true`. Print both values.
    - **Expected Output**:

```
Char value: [Your Unicode Character]
Boolean value: true
```

Look online for the Unicode table. Try to assign different Unicode values to the char datatype and print the result. Match the values for correct output.

Exercise 12: Basic Conditional Logic (if-else)

12. **Task**: Write an `if-else` statement that checks whether a number is greater than 10. If it is, print "Number is greater than 10"; otherwise, print "Number is less than or equal to 10."
    - **Expected Output**:

```
Number is [Condition Result]
```

Exercise 13: Using Ternary Operator

13. **Task**: Use a ternary operator to check whether a boolean value `isAvailable` is `true` or `false`, and print the result.
    - **Expected Output**:

```
Availability: [true/false]
```

Exercise 14: Exploring Strings

14. **Task**: Declare a `String` variable and assign a value to it. Print the string, and then reassign a new value and print again to observe the immutability of strings.
    - **Expected Output**:

```
Initial String: [Your String]
Modified String: [New String]
```

Exercise 15: Conclusion and Cleanup

15. **Task**: Go through all your code, ensure it compiles and runs correctly, and print any final messages or observations you have made during the lab session.

Additional Descriptive Exercises for Lab 1

Exercise 16: Simple Calculator

1. **Task**: Write a program that acts as a basic calculator. It should take two integers and perform addition, subtraction, multiplication, and division on them. Use `int` data types for inputs and the result.
    - **Hint**: You can hardcode the values for this exercise.
    - **Expected Output**:

```
Number 1: [Your First Number]
Number 2: [Your Second Number]
Addition: [Result of Addition]
Subtraction: [Result of Subtraction]
Multiplication: [Result of Multiplication]
Division: [Result of Division]
```

Exercise 17: Area and Perimeter of a Rectangle

2. **Task**: Write a program that calculates the area and perimeter of a rectangle. Use two `int` variables for length and width, and calculate the area (`length * width`) and perimeter (`2 * (length + width)`).
    - **Expected Output**:

```
Length: [Your Length Value]
Width: [Your Width Value]
Area: [Calculated Area]
Perimeter: [Calculated Perimeter]
```

Exercise 18: Temperature Converter (Celsius to Fahrenheit)

3. **Task**: Create a program that converts temperature from Celsius to Fahrenheit using the formula:
   `Fahrenheit = (Celsius * 9/5) + 32.`
   - **Hint**: Use a `double` variable for the Celsius input and the Fahrenheit result.
   - **Expected Output**:

     ```
     Celsius: [Your Celsius Value]
     Fahrenheit: [Calculated Fahrenheit]
     ```

Exercise 19: Sum of Digits

4. **Task**: Write a program that takes a three-digit number and calculates the sum of its digits. For example, if the input is `123`, the sum will be `1 + 2 + 3 = 6`.
   - **Hint**: Use arithmetic operators like division (`/`) and modulus (`%`) to extract digits.
   - **Expected Output**:

     ```
     Input number: [Your Input Number]
     Sum of digits: [Calculated Sum]
     ```

Exercise 20: Grade Calculator

5. **Task**: Write a program that takes a student's marks out of 100 and calculates the grade based on the following conditions:
   - Marks >= 90: Grade A
   - Marks >= 80: Grade B
   - Marks >= 70: Grade C
   - Marks >= 60: Grade D
   - Marks < 60: Grade F
   - **Hint**: Use `if-else` statements.
   - **Expected Output**:

     ```
     Marks: [Your Marks]
     Grade: [Calculated Grade]
     ```