JAVAGURU INTRODUCTION TO JAVA

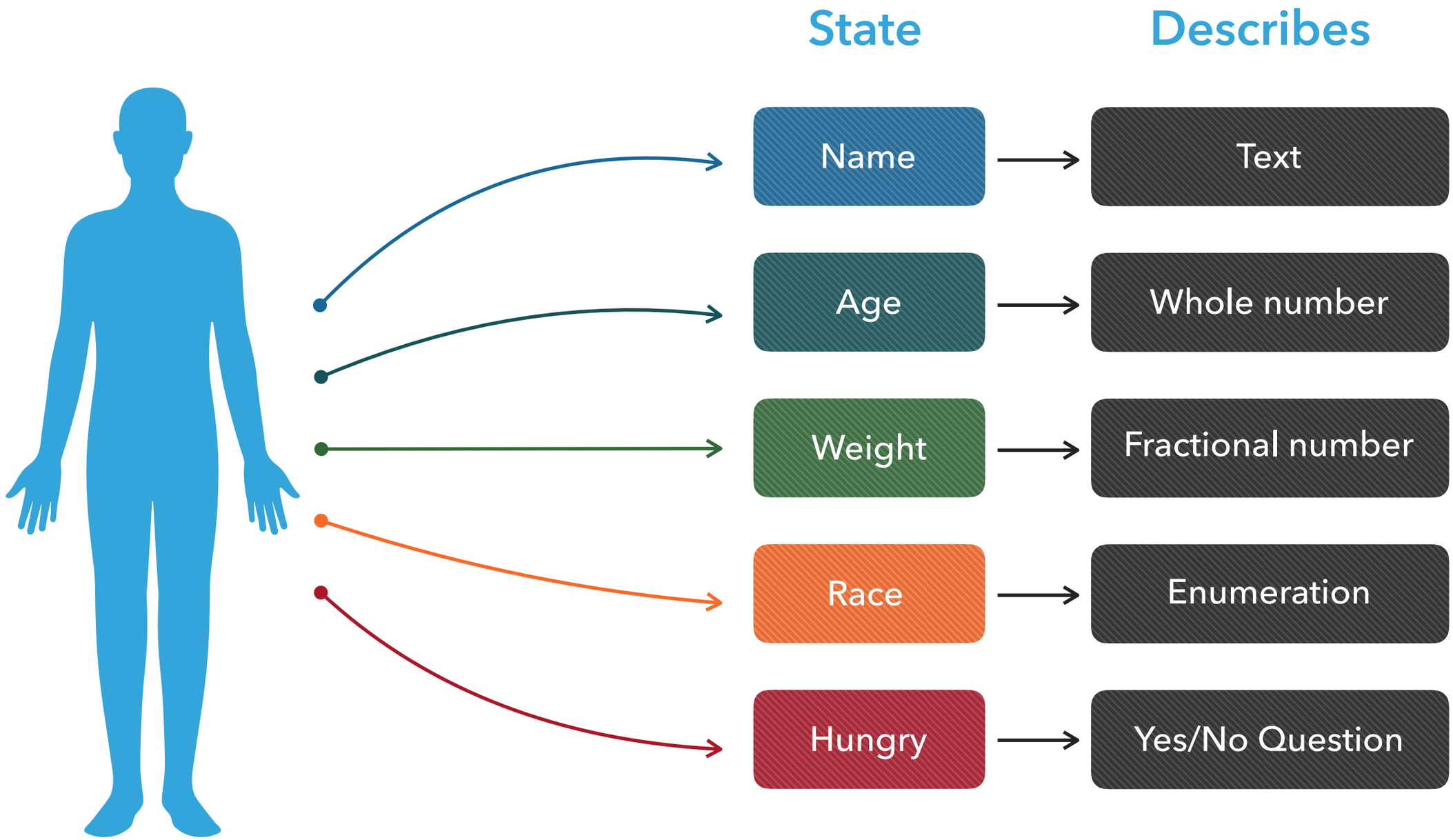
LESSON 2

VARIABLES
OVERVIEW

VARIABLES DEFINITION

- Variable is a named placeholder that
  - Stores data
  - Describes what type of data you can store
  - Describes size or amount of data it can store

VARIABLES RELATIONSHIP WITH OBJECTS



DATA TYPE CATEGORIES

- **Primitive** values
  - Integer: *byte, short, int, long* (e.g. 3, 7, 42, 2018)
  - Fractional: *float, double* (e.g. 3.1415, 2.7, 19.0)
  - Logical: *boolean* (true or false)
  - Textual: *char* (e.g. a, b, c, x, y, z)
- **Reference** values
  - Everything else

PRIMITIVE DATA TYPES IN DEPTH: INTEGER

Name	Assignable Values	Space
byte	-128 ... 127	1 byte
short	-32,768 ... 32,767	2 bytes
int	$-2^{31} \ldots 2^{31}-1$	4 bytes
long	$-2^{63} \ldots 2^{63}-1$	8 bytes

PRIMITIVE DATA TYPES IN DEPTH: FLOATING POINT

Name	Precision	Space
float	Single	4 bytes
double	Double	8 bytes

PRIMITIVE DATA TYPES IN DEPTH: LOGICAL

Name	Assignable Values	Space
boolean	true / false	1 byte

PRIMITIVE DATA TYPES IN DEPTH: TEXTUAL

Name	Assignable Values	Space
char (unicode)	0 ('\u0000') ... 65535 ('\uffff')	2 bytes

DEFINING VARIABLES IN JAVA CODE

VARIABLE DECLARATION IN JAVA: SYNTAX

- Variable declaration without value assignment

```
type name;
```

- Variable declaration **with** value assignment

```
type name = value;
```

VARIABLE DECLARATION IN JAVA: EXAMPLE

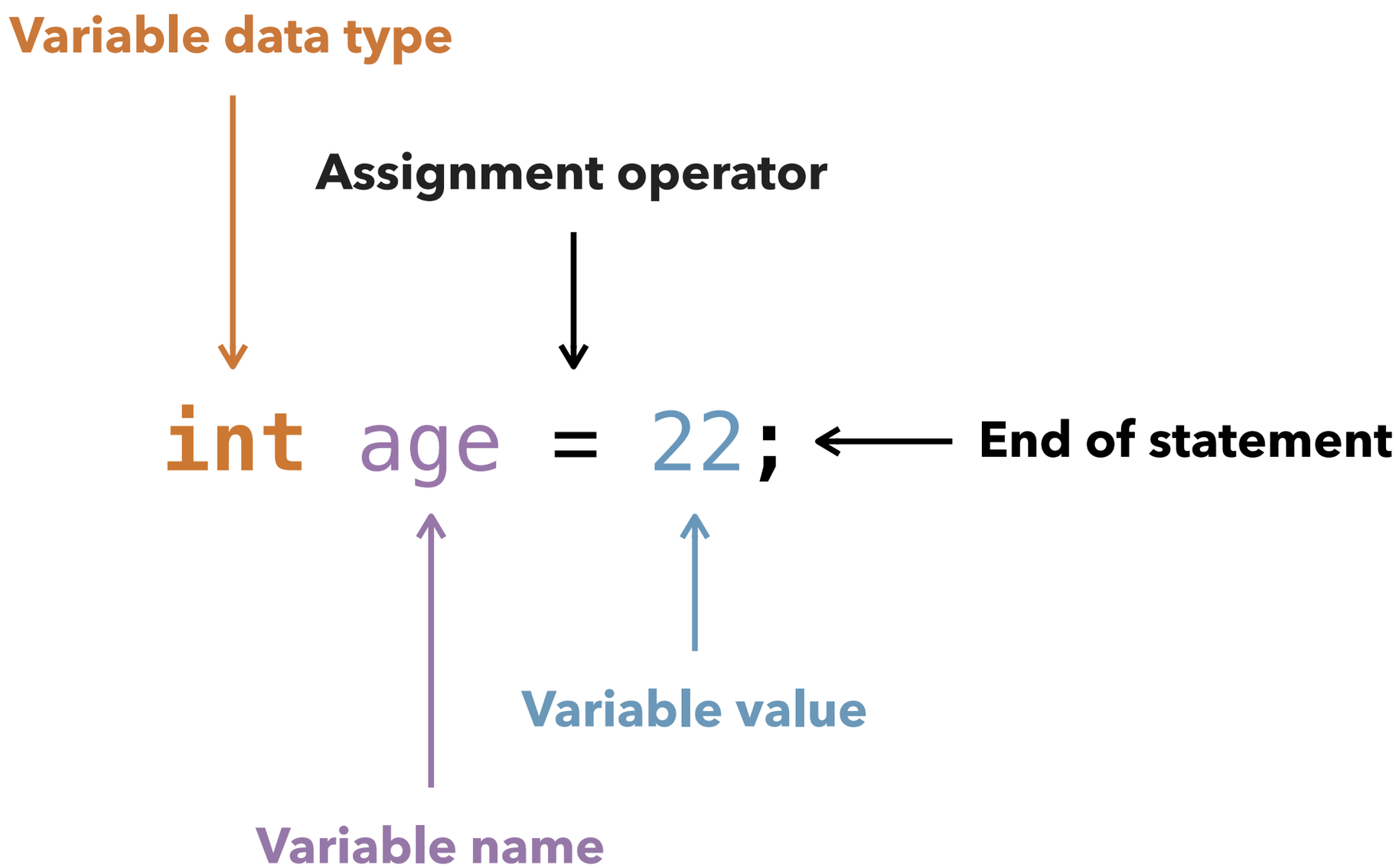
- Variable declaration without value assignment

```
int age;
```

- Variable declaration **with** value assignment

```
int age = 22;
```

VARIABLE DECLARATION BREAKDOWN



MORE EXAMPLES

```
byte numberOfWheels = 4;
short selfEsteem = -1;
int studentsGraduated = 1001;
long height = 80;
float pie = 3.14f;
double weight = 70.5;
boolean hungry = true;
char lastLetterOfTheAlphabet = 'Z';
```

NAMING RULES

- Any variable is allowed to **start** with
  - Letters (A-Z)
  - Special characters ('$' - dollar, '_' - underscore)
- Any variable name is allowed to **contain**
  - Alphanumeric characters (A-Z, 0-9)
  - Special characters ('$' - dollar, '_' - underscore)
- Variable name is **case-sensitive**
- Java language keywords[1] or reserved words **cannot** be used as variable name

[1] List of keywords can be found at https://docs.oracle.com/javase/tutorial/java/nutsandbolts/_keywords.html

NAMING DOS

- Single-worded name should be **lowercase**
- Multi-worded name should
  - First word **lowercase**
  - **Subsequent words start with** capital letters
  - No intervening spaces or punctuation
- Explains the purpose of variable

NAMING DON'TS

- Starting variable name with $ or _ is highly discouraged
- Avoid using $ anywhere in the variable name

NAMING DOS AND DON'TS EXAMPLES

- Please, do
  - size, xCoordinate, skinColor, currentDayOfTheWeek
- Please, don't
  - _counter, $bankBalance, Timestamp, 7daysOfTheWeek, !variableName, *notPointer

ARITHMETIC OPERATORS

ARITHMETIC OPERATORS OVERVIEW

Operator	Operation
+	Addition
-	Subtraction
/	Division
*	Multiplication
%	Remainder

OPERATORS BREAKDOWN: ADDITION

Integer numbers

```
int a = 10;
int b = 30;
int result = a + b;

result == 40
```

Fractional numbers

```
double x = 1.5;
double y = 2.7;
double result = x + y;

result == 4.2
```

OPERATORS BREAKDOWN: SUBTRACTION

Integer numbers

```
int a = 30;
int b = 20;
int result = a - b;

result == 10
```

Fractional numbers

```
double x = 5.4;
double y = 1.6;
double result = x - y;

result == 3.8
```

OPERATORS BREAKDOWN: MULTIPLICATION

Integer numbers

```
int a = 2;
int b = 4;
int result = a * b;

result == 8
```

Fractional numbers

```
double x = 2.5;
double y = 6.4;
double result = x * y;

result == 16.0
```

OPERATORS BREAKDOWN: DIVISION

Integer numbers

```
int a = 10;
int b = 5;
int result = a / b;

result == 2
```

Fractional numbers

```
double x = 18.0;
double y = 4.8;
double result = x / y;

result == 3.75
```

OPERATORS BREAKDOWN: REMAINDER

Integer numbers

```
int a = 9;
int b = 6;
int result = a % b;

result == 3
```

Fractional numbers

```
double x = 10.0;
double y = 4.5;
double result = x % y;

result == 1.0
```

TRICKY QUESTIONS

Type for division result is integer?

```
int a = 10;
int b = 4;
int result = a / b;

result == ?
```

Type for division result is double?

```
int x = 10;
int y = 4;
double result = x / y;

result == ?
```

TYPE CONVERSION: CASTING

- Operations with widening result require explicit type conversion (cast)

```
int x = 10;
int y = 4;
double result = x / (double) y;

result == ?
```

WRITING OUTPUT TO THE CONSOLE

WRITING OUTPUT TO THE CONSOLE: SYNTAX

- Write to the console content of the *"args"* variable

```
System.out.println(args);
```

- Write to the console directly without variable

```
System.out.println("Hello World");
```

WRITING OUTPUT TO THE CONSOLE: EXAMPLE

Printing Variable

```
int a = 10;
String b = "Hi"
System.out.println(a);
System.out.println(b);
```

```
10
Hi

Process finished with exit
code 0
```

Printing directly

```
System.out.println(374);
System.out.println("A");
```

```
374
A

Process finished with exit
code 0
```

WRITING OUTPUT STATEMENT BREAKDOWN

Class with standard output, input and error output streams

Function that writes argument to the specified stream

```
System.out.println("Hello World");
```

Inner class that helps to specify stream flow direction

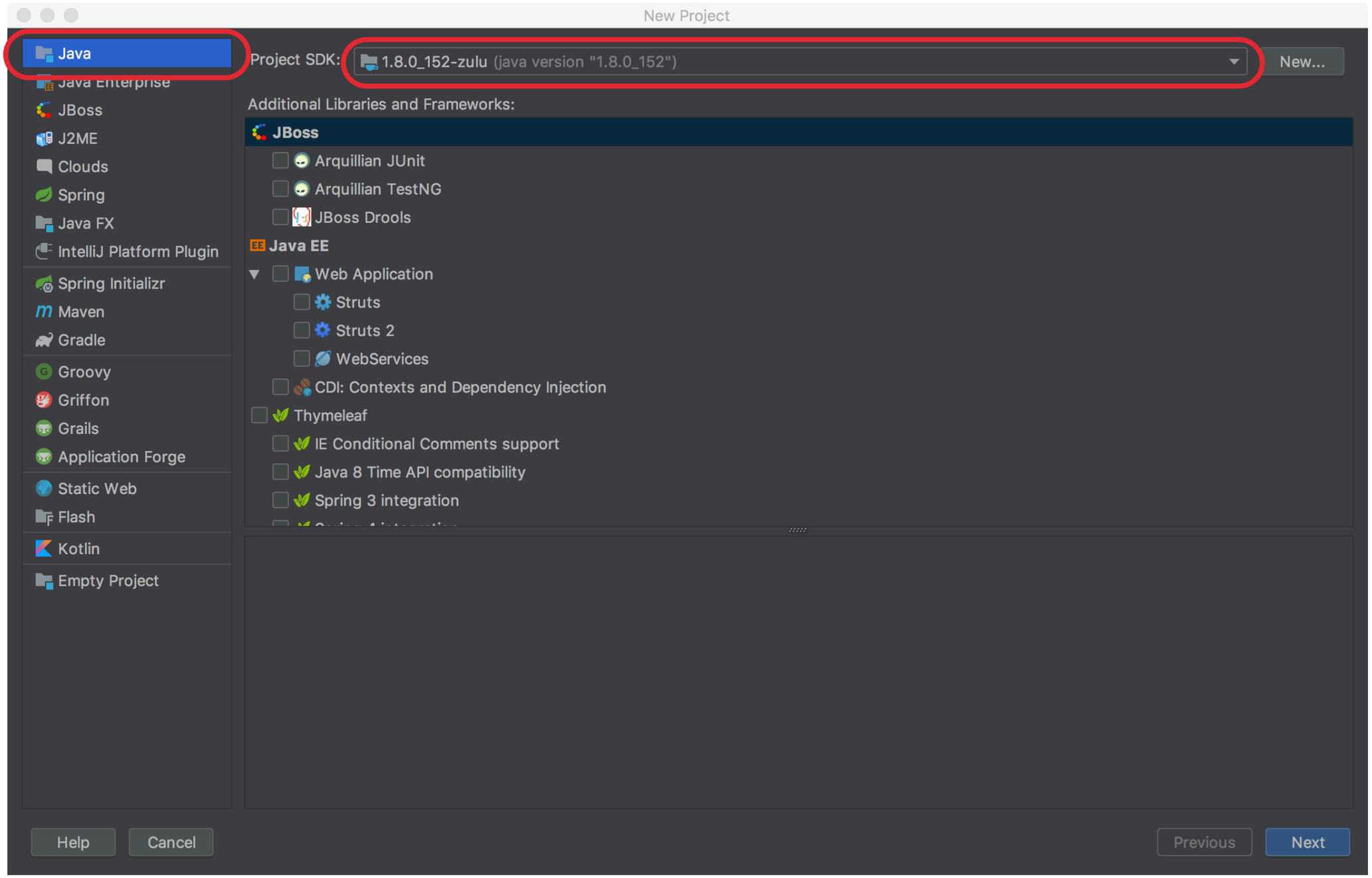
Argument to be written to the stream

CREATING FIRST JAVA PROJECT

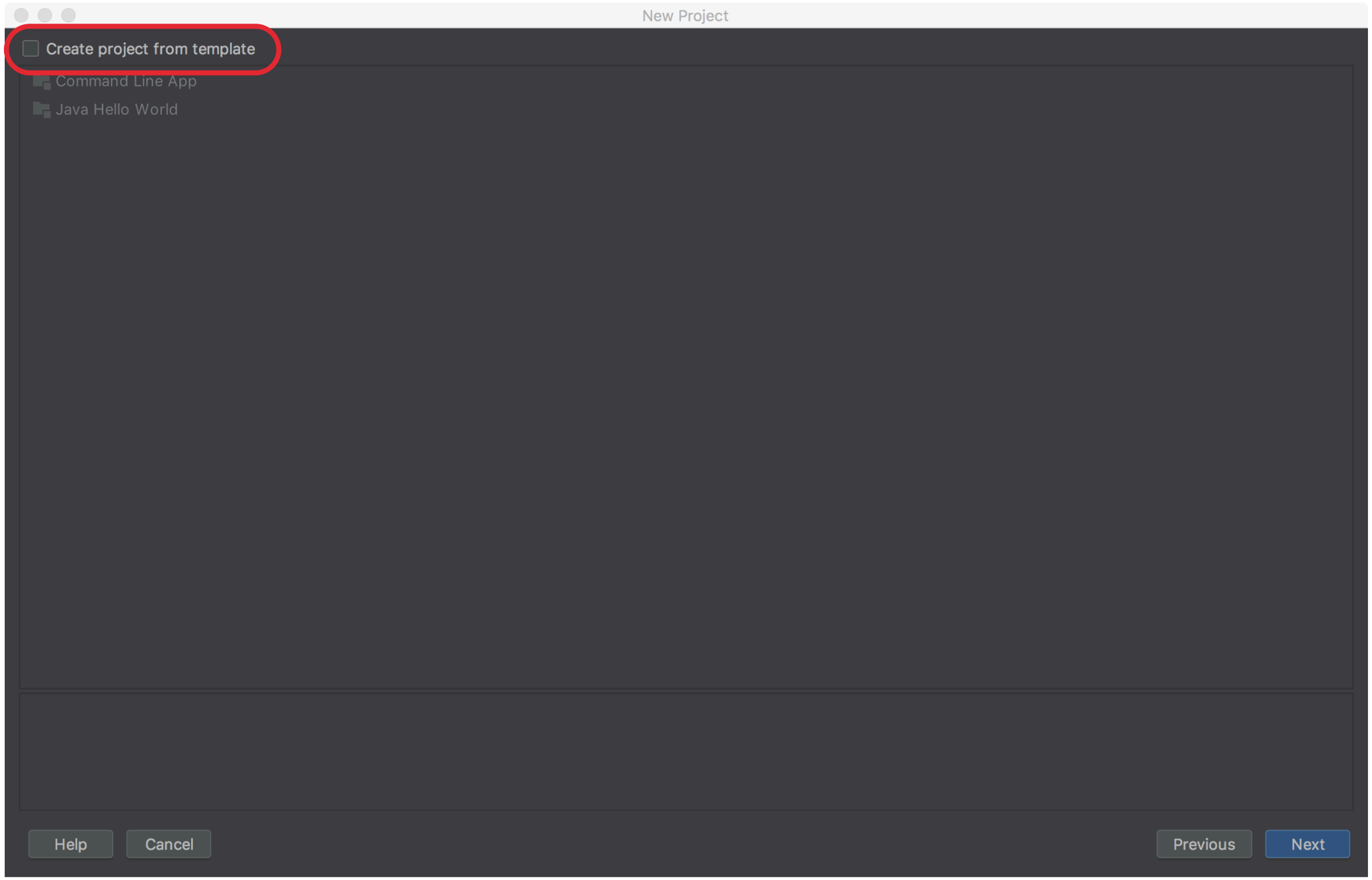
CREATING PROJECT OVERVIEW: RUN INTELLIJ IDEA

CREATING PROJECT OVERVIEW: WELCOME SCREEN

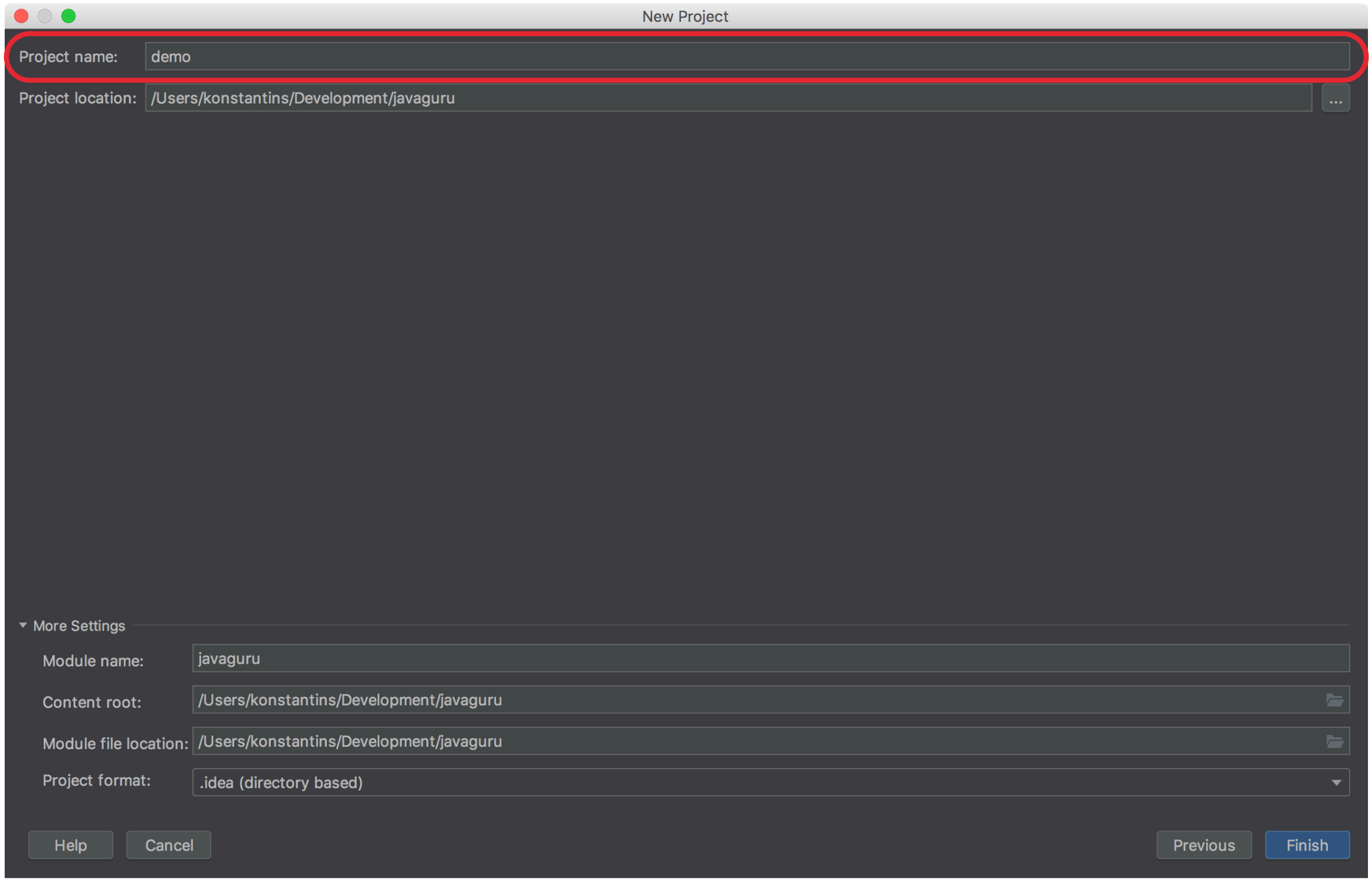
CREATING PROJECT OVERVIEW: PLATFORM SELECTION



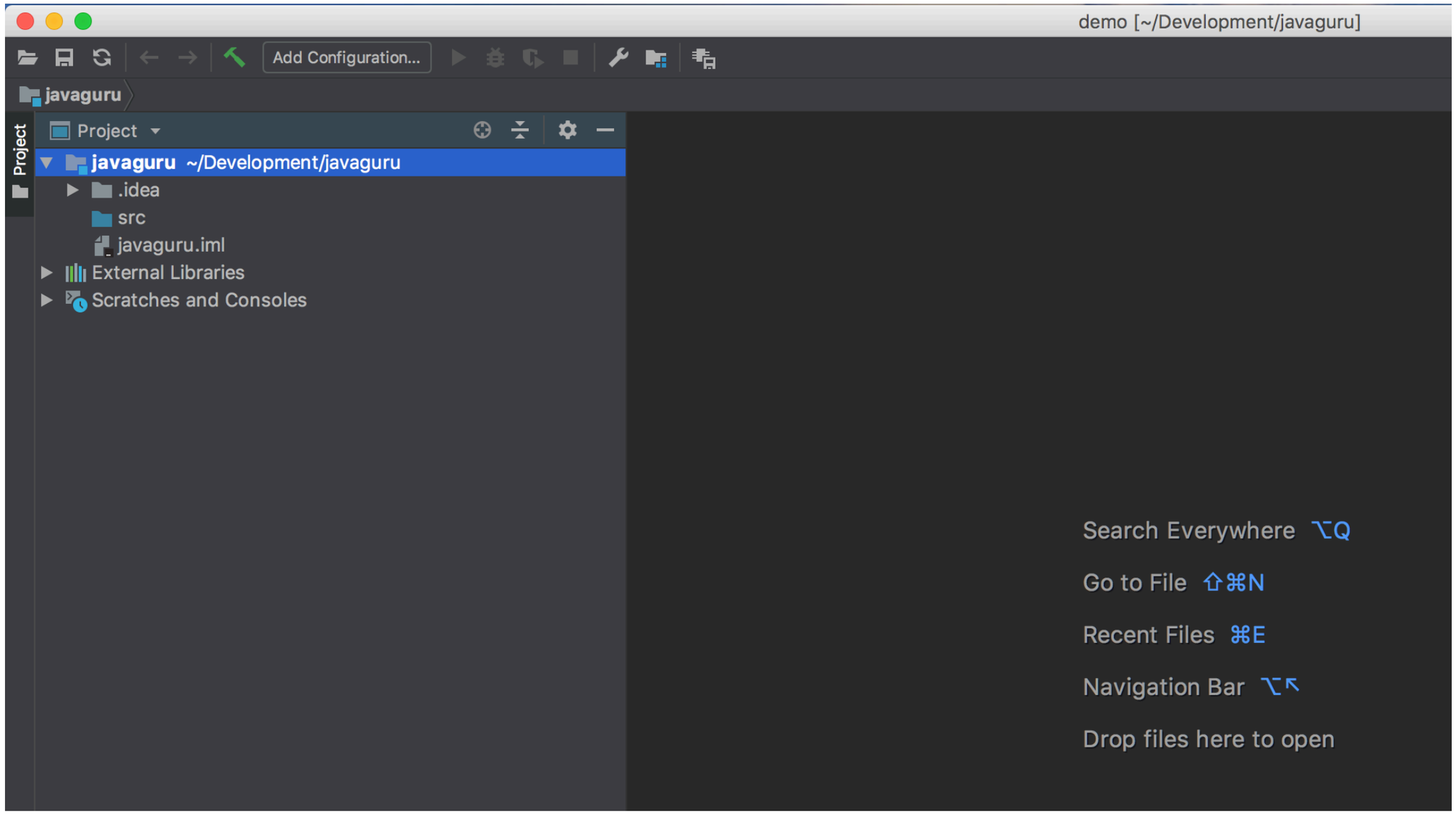
CREATING PROJECT OVERVIEW: TEMPLATE SELECTION



CREATING PROJECT OVERVIEW: NAMING



CREATING PROJECT OVERVIEW: GREAT SUCCESS

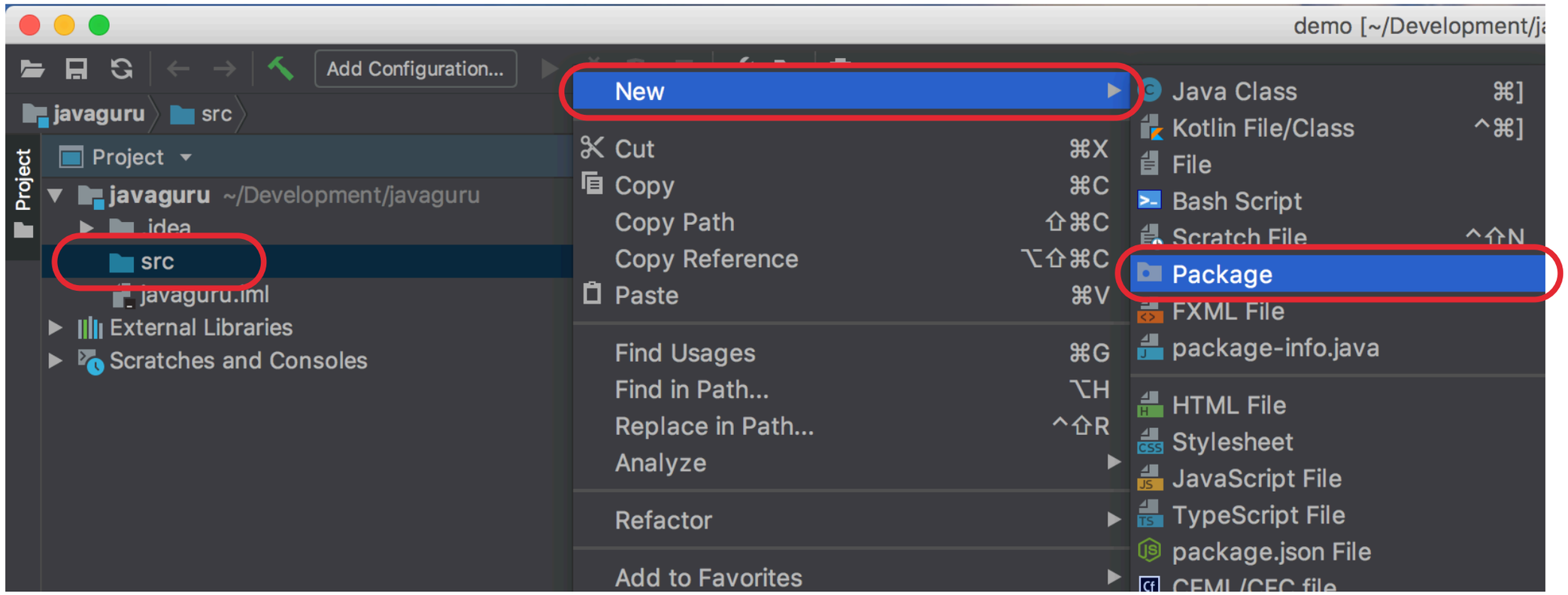


WRITING FIRST PROGRAM

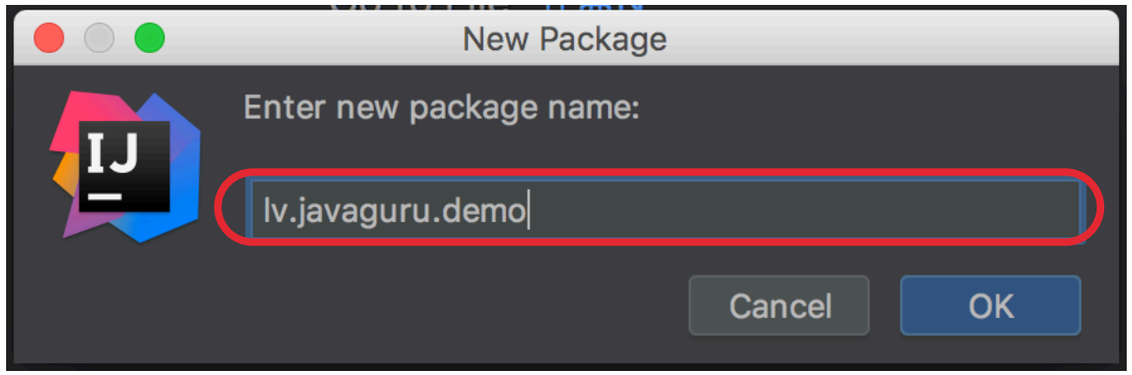
TASK 0: OBJECTIVES

1. Print *"Hello, JavaGuru World"* text to the console

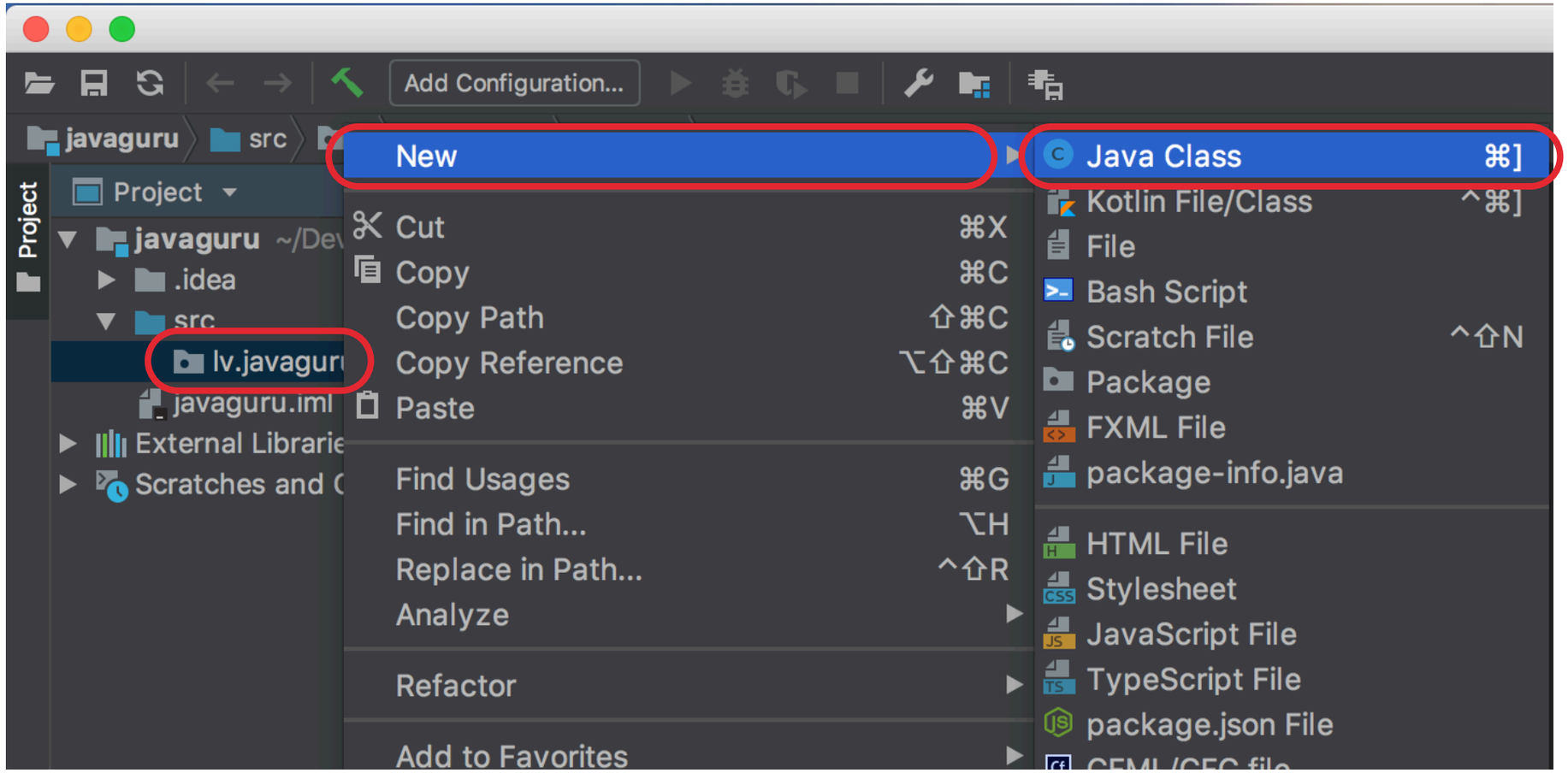
TASK 0: CREATE NEW PACKAGE



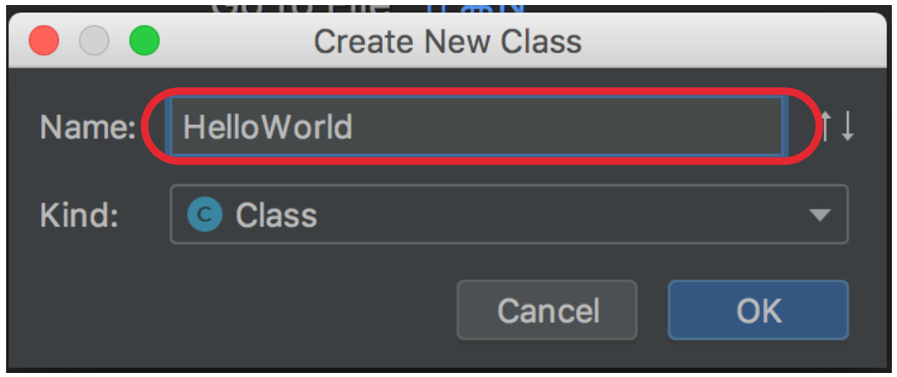
TASK 0: NAME YOUR PACKAGE



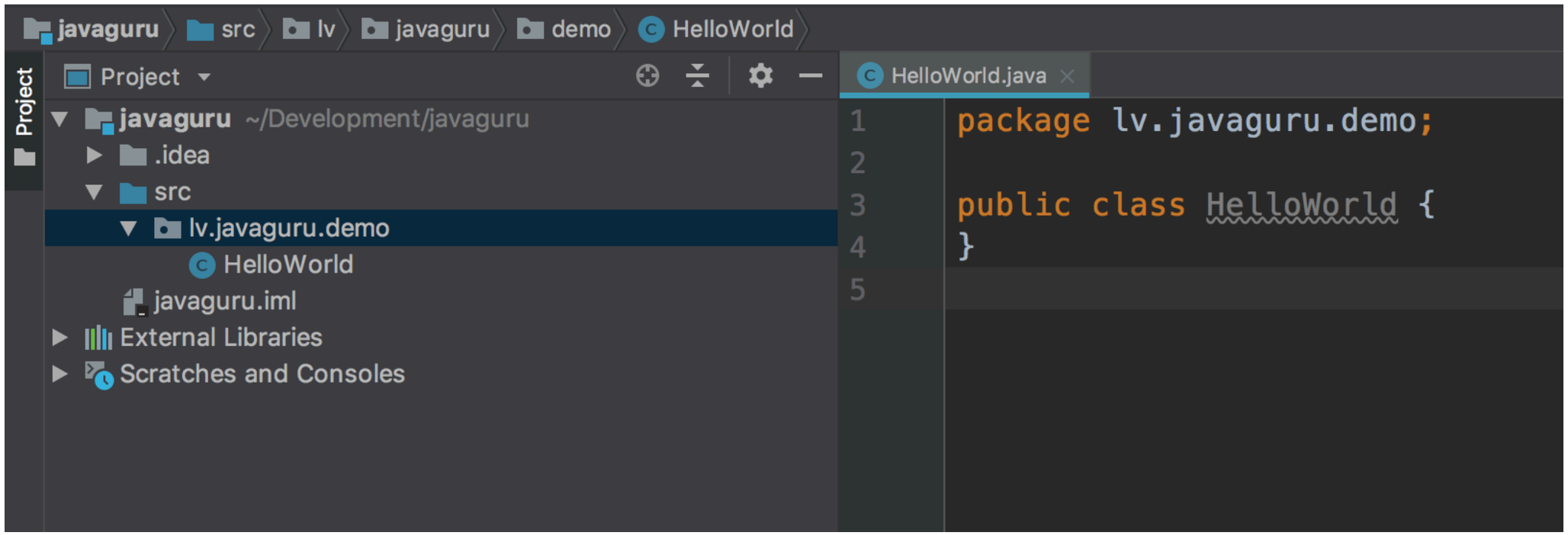
TASK 0: CREATE NEW JAVA CLASS



TASK 0: NAME YOUR JAVA CLASS



TASK 0: EXPECTED RESULT



TASK 0: WRITE MAIN METHOD – POINT OF START

HelloWorld.java

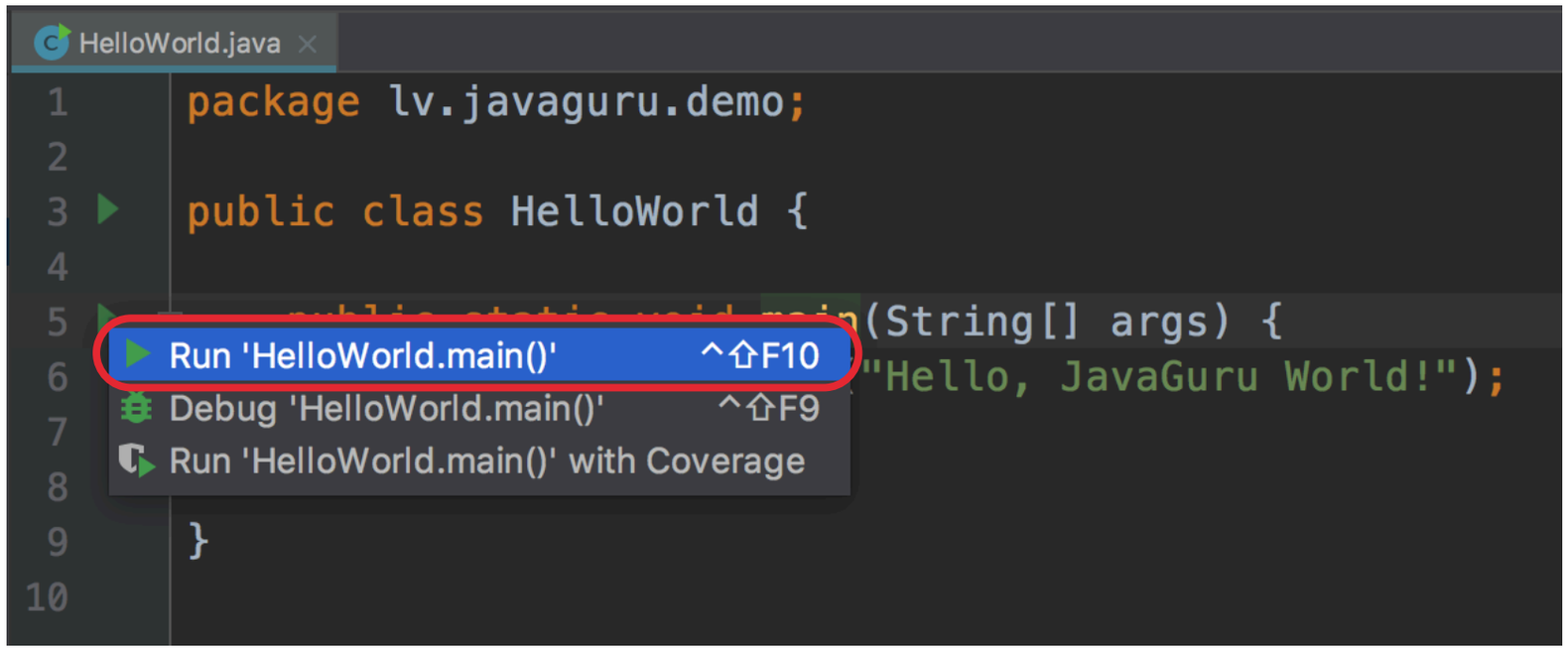
```
1   package lv.javaguru.demo;
2
3   public class HelloWorld {
4
5       public static void main(String[] args) {
6
7       }
8
9   }
10
```

TASK 0: WRITE CODE THAT GREETS THE WORLD

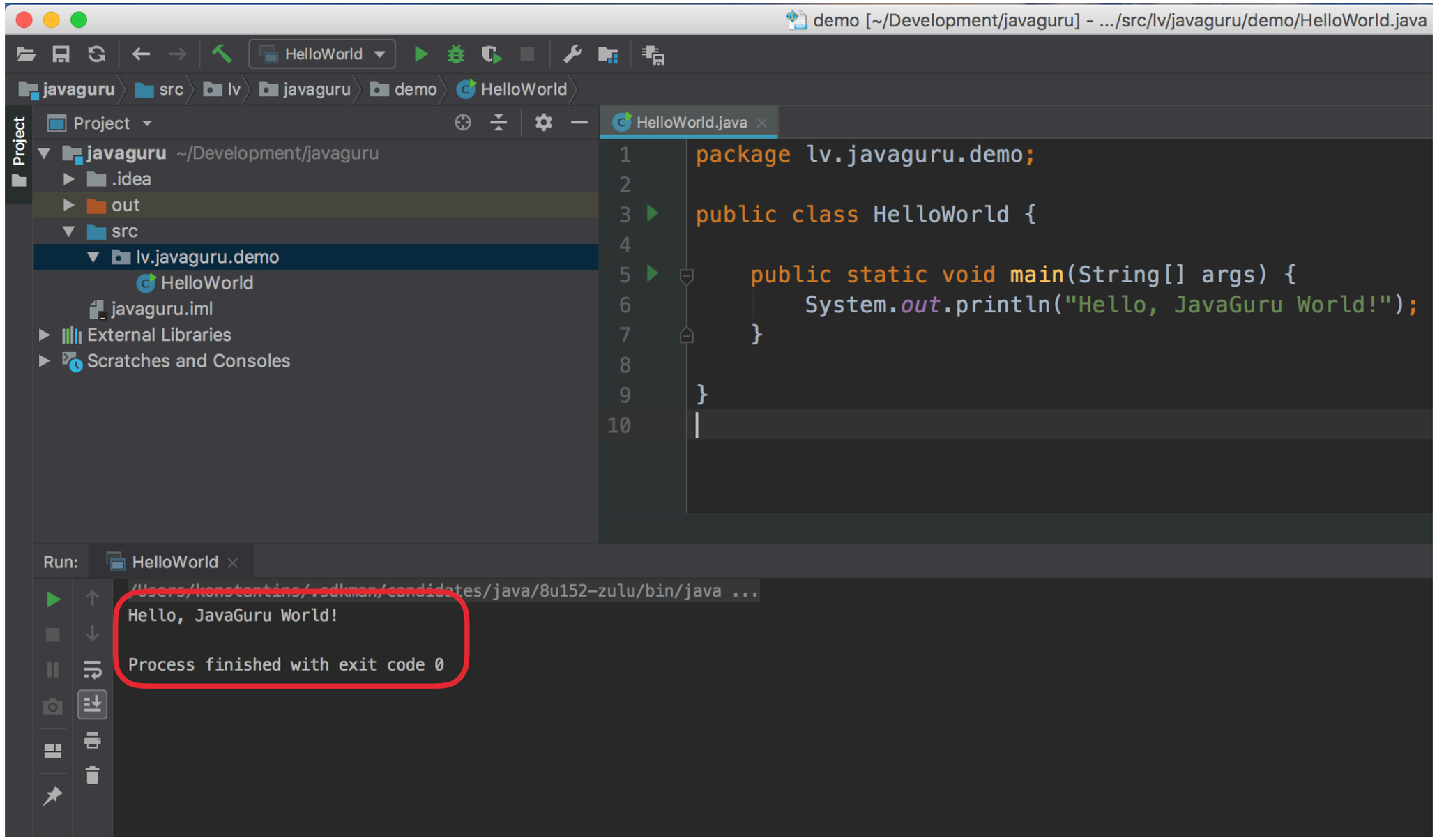
HelloWorld.java

```
1   package lv.javaguru.demo;
2
3   public class HelloWorld {
4
5       public static void main(String[] args) {
6           System.out.println("Hello, JavaGuru World!");
7       }
8
9   }
10
```

TASK 0: RUN YOUR CODE!



TASK 0: FINAL RESULT



TASK 1: OBJECTIVES

1. The program should define **two** integer variables
2. **Calculate the sum** of these variables
3. **Print** result to the console

TASK 1: CODE LISTING

1. Create 1st variable and assign it value of 10
2. Create 2nd variable and assign it value of 20
3. Assign 3rd variable result of computed sum
4. Print 3rd variable to the console

```
package lv.javaguru.demo;

public class CalculatorSumTest {

    public static void main(String[] args) {

      1 int firstNumber = 10;
      2 int secondNumber = 20;
      3 int sumResult = firstNumber + secondNumber;
      4 System.out.println(sumResult);

    }

}
```

TASK 2: OBJECTIVES

1. The program should create random number generator
2. Generate random number within 0 - 100 range inclusive
3. Print result to the console

TASK 2: CODE LISTING

1. Create random generator and assign it to the variable
2. Generate random number and assign result to the variable
3. Print *randomNumber* variable to the console

```
package lv.javaguru.demo;

import java.util.Random;

public class RandomNumberGeneratorTest {

    public static void main(String[] args) {
      1 Random randomGenerator = new Random();
      2 int randomNumber =
randomGenerator.nextInt(101);
      3 System.out.println(randomNumber);
    }

}
```

REFERENCES

- https://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html
- http://tutorials.jenkov.com/java/variables.html
- http://tutorials.jenkov.com/java/data-types.html
- https://javapapers.com/core-java/system-out-println/