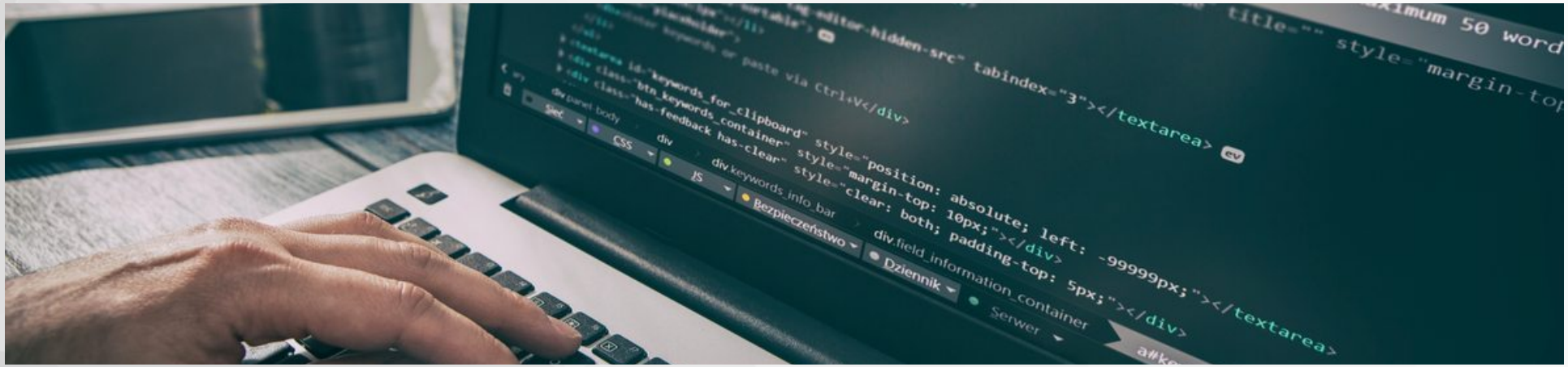
CCS0023/L

Object Oriented Programming (Java)

# Fundamentals of Java Language



02

LESSON #2

# Fundamentals of Java Language

## BASIC STRUCTURE OF A JAVA CLASS

```
class HelloWorld{
    public static void main(String args[ ]){
        System.out.println("Hello World");
    }
}
```

***class HelloWorld*** – defines a class, a template for an object of derived type HelloWorld

## BASIC STRUCTURE OF A JAVA CLASS

```
class HelloWorld{
    public static void main(String args[ ]){
        System.out.println("Hello World");
    }
}
```

***public*** – access specifier/modifier, the main method is declared public so that it is accessible as part of the public interface of the program.

# BASIC STRUCTURE OF A JAVA CLASS

```
class HelloWorld{
    public static void main(String args[ ]){
        System.out.println("Hello World");
    }
}
```

***static*** – state of the method, it is static because it must be called before the class that hosts the method is instantiated

## BASIC STRUCTURE OF A JAVA CLASS

```
class HelloWorld{
    public static void main(String args[ ]){
        System.out.println("Hello World");
    }
}
```

***void*** – It returns void because the Java interpreter does not expect to receive or process any output from the class.

## BASIC STRUCTURE OF A JAVA CLASS

```
class HelloWorld{
    public static void main(String args[ ]){
        System.out.println("Hello World");
    }
}
```

***System.out.println( )*** – The println method is one of dozens of methods in the System class. The System class is a part of the core Java language package of the Application Programming Interface (API)

# JAVA LANGUAGE FUNDAMENTALS

- Identifiers
- Keywords
- Comments
- Data types

**Identifiers** are names that are given by the programmer as name of variables, methods or functions, classes etc. The name used as an identifier must follow the following rules in Java™ technology:

- Each character is either a digit, letter, underscore or currency symbol.
- First character cannot be a digit.
- The identifier name must not be a reserved word.

Java keywords are reserved and cannot be used as identifiers.

| | | | | |
|---|---|---|---|---|
| abstract | do | import | return | void |
| boolean | double | instance of | short | volatile |
| break | else | int | static | while |
| byte | extends | interface | super | |
| case | final | long | switch | |
| catch | finally | native | synchronized | |
| char | float | new | this | |
| class | for | package | throws | |
| const | goto | private | transient | |
| continue | if | protected | try | |
| default | public | implements | | |

**Java Comments** are of three (3) types:

1. A single-line comment starting with //
2. A multi-line comment enclosed within /* */
3. A documentation or javadoc comment is enclosed between /** and */. These comments can be used to generate HTML documents using the javadoc utility, which is part of Java language comment.

# JAVA DATA TYPES

## Primitive Data Types

| Type | Width (in bytes) | Min Value | Max Value |
|---|---|---|---|
| **byte** | 1 | -128 | 127 |
| **short** | 2 | -32768 | 32767 |
| **int** | 4 | -2147483648 | 21474833647 |
| **long** | 8 | Long.MIN_VALUE | Long.MAX_VALUE |
| **float** | 4 | Float.MIN_VALUE | Float.MAX_VALUE |
| **double** | 8 | Double.MIN_VALUE | Double.MAX_VALUE |
| **boolean** | (1 bit) | true | false |
| **char** | 2 (unsigned) | ‘\u0000’ | ‘\uFFFF’ |

Derived Data Types

- String
- Date
- Integer
- Double
- Long
- Float
- …

## VARIABLES

- Variable Declaration
  Syntax: <datatype> <varName>; [= value;]
  Example: String name;
  int age;
  double price = 55.66;
- Assigning a value
  Syntax: <varName> = value;
  Example: name = “Maria Blanco”;
  age = 22;
  price = 200.50;

## WRAPPER CLASSES

***Java Wrapper Classes*** are used in converting one data type (such as a String) into another data type (such as int or double). It is also used in wrapping a primitive value into an object.

| Wrapper Class | Primitive Type |
|---|---|
| Integer | int |
| Float | float |
| Double | double |
| Long | long |
| Byte | byte |
| Short | short |
| Character | char |
| Boolean | boolean |

# THANK YOU