



CSIS 3101 Resources - Java

Defining a "main" method in Java:

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

Java Classes and Objects:

Considering Java is an Object-Oriented programming language, it is essential to learn how to define new classes and instantiate objects for that class. As demonstrated in the following example:

```
public class Main {
    int x = 5;

    public static void main(String[] args) {
        Main myObj = new Main();
        System.out.println(myObj.x);
    }
}
```

In this case, *myObj* is a new instance of the main class, and contains all parameters, attributes, and methods the class definition has.

Java Class Constructors:

A constructor in Java is a special method that is used to initialize objects. The constructor is called when an object of a class is created. It can be used to set initial values for object attributes. Example:

```
// Create a Main class
public class Main {
    int x; // Create a class attribute

    // Create a class constructor for the Main class
    public Main() {
        x = 5; // Set the initial value for the class attribute x
    }

    public static void main(String[] args) {
        Main myObj = new Main(); // Create an object of class Main (This will call the constructor)
        System.out.println(myObj.x); // Print the value of x
    }
}
```

Declaring variable types in Java:

In java, variables are required to have a declared variable type upon definition, unlike Python which does so dynamically. Declaring types is done as follows:

`type variableName = value;`

In Java, there are different types of variables, such as:

- + String - stores text, such as "Hello". String values are surrounded by double quotes
- + int - stores integers (whole numbers), without decimals, such as 123 or -123
- + float - stores floating point numbers, with decimals, such as 19.99 or -19.99
- + char - stores single characters, such as 'a' or 'B'. Char values are surrounded by single quotes
- + boolean - stores values with two states: true or false

Example: `String name = "John";`
`System.out.println(name);`

Essential String manipulation methods in Java:

<code>replace()</code>	Searches a string for a specified value, and returns a new string where the specified values are replaced
<code>replaceFirst()</code>	Replaces the first occurrence of a substring that matches the given regular expression with the given replacement
<code>replaceAll()</code>	Replaces each substring of this string that matches the given regular expression with the given replacement
<code>split()</code>	Splits a string into an array of substrings
<code>startsWith()</code>	Checks whether a string starts with specified characters
<code>subSequence()</code>	Returns a new character sequence that is a subsequence of this sequence
<code>substring()</code>	Returns a new string which is the substring of a specified string
<code>toCharArray()</code>	Converts this string to a new character array
<code>toLowerCase()</code>	Converts a string to lower case letters
<code>toString()</code>	Returns the value of a String object
<code>toUpperCase()</code>	Converts a string to upper case letters
<code>trim()</code>	Removes whitespace from both ends of a string
<code>valueOf()</code>	Returns the string representation of the specified value
<code>charAt()</code>	Returns the character at the specified index (position)
<code>indexOf()</code>	Returns the position of the first found occurrence of specified characters in a string
<code>lastIndexOf()</code>	Returns the position of the last found occurrence of specified characters in a string
<code>length()</code>	Returns the length of a specified string

Contact us via:

📍 Student Affairs Building,
2nd floor
☎️ (954) 262-8350
📧 @nsu_ttc @nsu_si