JAVA CLASS METHODS

A class method is **a method that can be invoked without reference to any object instance**; these are called static methods in other languages. The term method usually refers to an instance method. The more specific phrase class method is used to refer to class methods.

method - is a block of code which only runs when it is called.

You can pass data, known as **parameters**, into a **method**.

Functions - Methods are used to perform certain actions.

Why use methods? To **reuse code**: define the code once, and use it many times.

Create a Method

A method must be declared within a class.

It is defined with the name of the method, followed by parentheses **()**.

Java provides some pre-defined methods, such as **System.out.println(),** but you can also create your own methods to perform certain actions:

Example:

Create a method inside Main:

```
public class Main {

  static void myMethod() {

    // code to be executed

  }

}
```

Example Explained

myMethod() - is the name of the method.

static - means that the method belongs to the Main class and not an object of the Main class.

void - means that this method does not have a return value.

Call a Method

To call a method in Java, write the method's name followed by **two parentheses ()** and a **semicolon (;)**

In the following example, **myMethod()** is used to print a text (the action), when it is called:

Example:

Inside main, call the myMethod() method:

```
public class Main {
  static void myMethod() {
    System.out.println("I just got executed!");
  }

  public static void main(String[] args) {
    myMethod();
  }
}
```

Outputs: I just got executed!

A method can also be called multiple times:

Example:

```
public class Main {
  static void myMethod() {
    System.out.println("I just got executed!");
  }

  public static void main(String[] args) {
    myMethod();
    myMethod();
    myMethod();
  }
}
```

Output: I just got executed!

I just got executed!

I just got executed!

myMethod() prints a text (the action), when it is **called**.

The **dot (.)** is used to access the object's attributes and methods.

To call a method in Java, write the method name followed by a set of parentheses **()**, followed by a semicolon (;).

A class must have a matching filename (Main and **Main.java**).

Static vs. Public

Static - which means that it can be accessed without creating an object of the class.

Public - which can only be accessed by objects:

Example to demonstrate the differences between static and public methods:

```
public class Main {
  // Static method
  static void myStaticMethod() {
    System.out.println("Static methods can be called without creating objects");
  }

  // Public method
  public void myPublicMethod() {
    System.out.println("Public methods must be called by creating objects");
  }

  // Main method
  public static void main(String[] args) {
    myStaticMethod(); // Call the static method
    // myPublicMethod(); This would compile an error

    Main myObj = new Main(); // Create an object of Main
    myObj.myPublicMethod(); // Call the public method on the object
  }
}
```

Output: Static methods can be called without creating objects

Public methods must be called by creating objects

Reference:

https://www.w3schools.com/java/java_class_methods.asp