



First Java Program: Hello World

Printing to Console

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

Printing to Console

```
public class HelloWorld
```



Class Header

```
{ }
```



Opening and Closing Braces

```
public static void main(String[] args)
```



Main Method

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



Displays a message on the screen

```
;
```



Semicolon

main() method

- Every Java application must contain a main method.
- When the Java interpreter executes an application, it starts by calling the class's main method

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

println() method

- Used for printing data on the console
- Appends a newline at the end of the data output

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

print() method

- Used for printing data on the console
- Does **not** Append a newline at the end of the data output

```
System.out.print("Hello World!");
```

println() vs print()

- `System.out.println()`: Appends a new line at the end
- `System.out.print()`: Does not append new line

Example	Output
<code>System.out.print("one");</code> <code>System.out.print("two");</code> <code>System.out.println("three");</code>	onetwothree
<code>System.out.println("one");</code> <code>System.out.println("two");</code> <code>System.out.println("three");</code>	one two three
<code>System.out.println();</code>	[new line]

Task 1

1. Create a class called JavaComponents.java
2. Print the full form of each of the followings:
 1. JDK
 2. JRE
 3. JVM

Task 2

1. Create a class called PrintExecution.java
2. Write a code to print following outputs in the console:

```
Duygu  
Oleksandr  
Zoia
```

```
Ahmet  
Yuliya  
Cassandra
```

```
Cydeo
```

Task 3

1. Create a class called MyClassmates.java
2. Print five of your classmates' names on the console

Single Line & Multi Line Comments

Comments

- Java comments can be used to make the code more readable.
- Single line comments are often used for short descriptions of what the code is doing.
- Multi line comments are often used for descriptions of how the script works, or to prevent a section of the script from running when testing it.

Single Line Comment

- Anything that follows two forward slash characters `//` on that line will not be processed.

```
//Java program to show single line comments
class Comment
{
    public static void main(String args[])
    {
        // Single line comment here
        System.out.println("Single line comment above");
    }
}
```

Multi Line Comment

- To write a comment that stretches over more than one line, you use a multi-line comment, starting with the `/*` characters and ending with the `*/` characters

```
//Java program to show multi line comments
class Scomment
{
    public static void main(String args[])
    {
        System.out.println("Multi line comments below");
        /*Comment line 1
           Comment line 2
           Comment line 3*/
    }
}
```