



# **SNS COLLEGE OF TECHNOLOGY**

Coimbatore-35  
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade  
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai



## **DEPARTMENT OF INFORMATION TECHNOLOGY**

# **OBJECT ORIENTED PROGRAMMING**

II YEAR - III SEM

## **INTRODUCTION TO OOP**



# Java Quickstart

- In Java, every application begins with a class name, and that class must match the filename.
- Let's create our first Java file, called Main.java, which can be done in any text editor (like Notepad).
- The file should contain a "Hello World" message, which is written with the following code:



# Java Quickstart

## Main.java

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



# Java Quickstart

- Save the code in Notepad as "Main.java".
- Open Command Prompt (cmd.exe), navigate to the directory where you saved your file, and type "javac Main.java":

```
C:\Users\Your Name>javac Main.java
```

- This will compile your code.
- If there are no errors in the code, the command prompt will take you to the next line.
- Now, type "java Main" to run the file:

```
C:\Users\Your Name>java Main
```

The output should read:

```
Hello World
```



# Java Syntax

In the previous Slide, we created a Java file called **Main.java**, and we used the following code to print "Hello World" to the screen:

Main.java

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



# Example explained

- Every line of code that runs in Java must be inside a *class*.
- In our example, we named the class *Main*.
- A class should always start with an uppercase first letter.
- Note: Java is case-sensitive: "MyClass" and "myclass" has different meaning.
- The name of the java file must match the class name.
- When saving the file, save it using the class name and add ".java" to the end of the filename.



# The main Method

## The main Method

The `main()` method is required and you will see it in every Java program:

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

Any code inside the `main()` method will be executed. Don't worry about the keywords before and after main. You will get to know them bit by bit while reading this tutorial.

For now, just remember that every Java program has a `class` name which must match the filename, and that every program must contain the `main()` method.



# The main Method

## System.out.println()

Inside the `main()` method, we can use the `println()` method to print a line of text to the screen:

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





# The main Method

**Note:** The curly braces `{}` marks the beginning and the end of a block of code.

`System` is a built-in Java class that contains useful members, such as `out`, which is short for "output". The `println()` method, short for "print line", is used to print a value to the screen (or a file).

Don't worry too much about `System`, `out` and `println()`. Just know that you need them together to print stuff to the screen.

You should also note that each code statement must end with a semicolon (`;`).