**Static Block in Java**

**Static block** is a set of statements, which will be executed by the JVM before execution of main method.

At the time of class loading if we want to perform any activity we have to define that activity inside static block because this block execute at the time of class loading.

In a class we can take any number of static block but all these blocks will be execute from top to bottom.

**Syntax**

**static**

{

........

//Set of Statements

........

}

**Note:**In real time application static block can be used whenever we want to execute any instructions or statements before execution of main method.

**Example of Static Block**

**class** StaticDemo

{

**static**

{

System.**out**.println("Hello how are u ?");

}

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

{

System.**out**.println("This is main()");

}

}

**Output**

Hello how are u ?

This is main()

**Run java program without main method**

**class** StaticDemo

{

**static**

{

System.**out**.println("Hello how are u ?");

}

}

**Output**

Output:

Hello how are u ?

Exception is thread "main" java.lang.no-suchmethodError:Main

**Note:**"Exception is thread "main" java.lang.no-suchmethodError:Main" warning is given in java 1.7 and its above versions

**More than one static block in a program**

**class** StaticDemo

{

**static**

{

System.**out**.println("First static block");

}

**static**

{

System.**out**.println("Second Static block");

}

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

{

System.**out**.println("This is main()");

}

}

**Output**

Output:

First static block

Second static block

This is main()

**Note:**"Here static block run according to there order (sequence by) from top to bottom.

**Why a static block executes before the main method ?**

A class has to be loaded in main memory before we start using it. Static block is executed during class loading. This is the reason why a static block executes before the main method.