|  |  |  |  |
| --- | --- | --- | --- |
| **Data Type** | **Size** | **Range** | |
| byte | 1 byte | Stores whole numbers from  -128 to 127 | -27 and a maximum value of 27-1. |
| short | 2 bytes | Stores whole numbers from  -32,768 to 32,767 | -215 and a maximum value of 215-1. |
| int | 4 bytes | Stores whole numbers from -2,147,483,648 to 2,147,483,647 | -231 and a maximum value of 231-1. |
| long | 8 bytes | Stores whole numbers from  -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 | -263 and a maximum value of 263-1. |
| Float single precision | 4 bytes | Stores fractional numbers. Sufficient for storing 6 to 7 decimal digits | -1.4e-0.45 to 3.4e+38 |
| Double double precision | 8 bytes | Stores fractional numbers. Sufficient for storing 15 decimal digits | -4.9e-324 to 1.8e+308 |
| boolean | 1 bit | Stores true or false values |  |
| char | 2 bytes | Stores a single character/letter or ASCII values | UNICODE |

Type Conversion and Type Casting :

* when you assign a value of one primitive data type to another type.
* two types
  + **Widening Casting** (automatically) - converting a smaller type to a larger type size

int x = 9;

double y = x;

* + **Narrowing Casting** (manually) - converting a larger type to a smaller size type

double x = 9.78;

int y = (int) x;

**Type Promotion Rules:**

1. all byte, short, and char values are promoted to int

Ex: byte a=4

byte b=5

byte c=a\*b; [incompatible types]

‘a’& ‘b’ get automatically promoted to int whereas we are storing in byte so incompatabile.

So type casting has to be done

byte c= (byte) (a\*b);



* if one operand is a long, the whole expression is promoted to long.
* If one operand is a float, the entire expression is promoted to float.
* If any of the operands is double, the result is double.

**FIRST JAVA PROGRAM**

class HelloWorld {

public static void main(String[] args) {

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

}

}

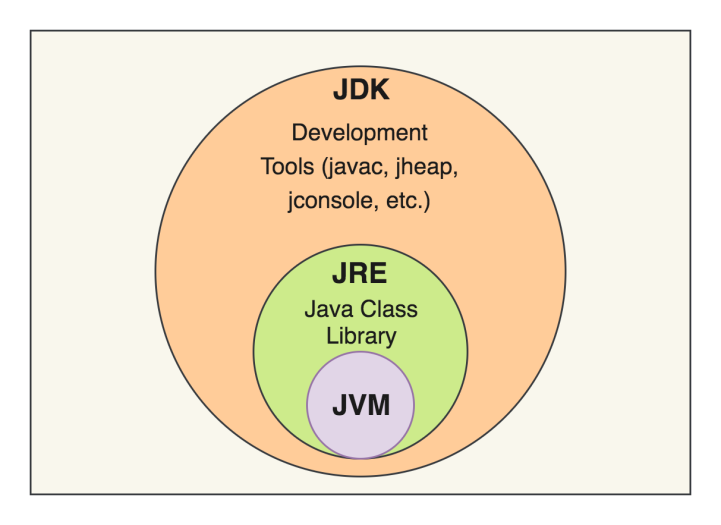
1. Save with .java extension
2. Can be saved using any name
3. Compiled as javac HelloWorld.java (if the file name is HelloWorld.java)
4. This creates a .class file called HelloWorld.class which contains the bytecode
5. The program is executed as java HelloWorld

**NOTE:** If you have saved program as First.java

Then compile as javac First.java

But run as java HelloWorld

**JDK,JRE &JVM**



JDK=JRE+DEVELOPMENT TOOLS

JRE=JVM+LIBRARY CLASSES