

Data Types in Java

BY CHAITANYA SINGH | FILED UNDER: [LEARN JAVA](#)

Data type defines the values that a variable can take, for example if a variable has int data type, it can only take integer values. In java we have two categories of data type: 1) Primitive data types 2) Non-primitive data types – Arrays and Strings are non-primitive data types, we will discuss them later in the coming tutorials. Here we will discuss primitive data types and literals in Java.

Java is a statically typed language. A language is statically typed, if the data type of a variable is known at compile time. This means that you must specify the type of the variable (Declare the variable) before you can use it.

In the last tutorial about [Java Variables](#), we learned how to declare a variable, let's recall it:

```
int num;
```

So in order to use the variable num in our program, we must declare it first as shown above. It is a good programming practice to declare all the variables (that you are going to use) in the beginning of the program.

1) Primitive data types

In Java, we have eight primitive data types: boolean, char, byte, short, int, long, float and double. Java developers included these data types to maintain the portability of java as the size of these primitive data types do not change from one operating system to another.

byte, **short**, **int** and **long** data types are used for storing whole numbers.

float and **double** are used for fractional numbers.

char is used for storing characters(letters).

boolean data type is used for variables that holds either true or false.

byte:

This can hold whole number between -128 and 127. Mostly used to save memory and when you are certain that the numbers would be in the limit specified by byte data type.

Default size of this data type: 1 byte.

Default value: 0

Example:

```
class JavaExample {  
    public static void main(String[] args) {  
  
        byte num;  
  
        num = 113;  
        System.out.println(num);  
    }  
}
```

Output:

113

Try the same program by assigning value assigning 150 value to variable num, you would get **type mismatch** error because the value 150 is out of the range of byte data type. The range of byte as I mentioned above is -128 to 127.

short:

This is greater than byte in terms of size and less than integer. Its range is -32,768 to 32767.

Default size of this data type: 2 byte

```
short num = 45678;
```

int: Used when short is not large enough to hold the number, it has a wider range: -2,147,483,648 to 2,147,483,647

Default size: 4 byte

Default value: 0

Example:

```
class JavaExample {  
    public static void main(String[] args) {  
  
        short num;  
  
        num = 150;  
        System.out.println(num);  
    }  
}
```

Output:

150

The byte data type couldn't hold the value 150 but a short data type can because it has a wider range.

long:

Used when int is not large enough to hold the value, it has wider range than int data type, ranging from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807.

size: 8 bytes

Default value: 0

Example:

```
class JavaExample {  
    public static void main(String[] args) {  
  
        long num = -12332252626L;  
        System.out.println(num);  
    }  
}
```

Output:

```
-12332252626
```

double: Sufficient for holding 15 decimal digits

size: 8 bytes

Example:

```
class JavaExample {  
    public static void main(String[] args) {  
  
        double num = -42937737.9d;  
        System.out.println(num);  
    }  
}
```

Output:

```
-4.29377379E7
```

float: Sufficient for holding 6 to 7 decimal digits

size: 4 bytes

```
class JavaExample {  
    public static void main(String[] args) {  
  
        float num = 19.98f;  
        System.out.println(num);  
    }  
}
```

Output:

```
19.98
```

boolean: holds either true or false.

```
class JavaExample {
```

```
public static void main(String[] args) {  
  
    boolean b = false;  
    System.out.println(b);  
}  
}
```

Output:

```
false
```

char: holds characters.

size: 2 bytes

```
class JavaExample {  
    public static void main(String[] args) {  
  
        char ch = 'Z';  
        System.out.println(ch);  
    }  
}
```

Output:

```
Z
```

Literals in Java

A literal is a fixed value that we assign to a variable in a Program.

```
int num=10;
```

Here value 10 is a Integer literal.

```
char ch = 'A';
```

Here A is a char literal

Integer Literal

Integer literals are assigned to the variables of data type `byte`, `short`, `int` and `long`.

```
byte b = 100;  
short s = 200;  
int num = 13313131;  
long l = 928389283L;
```

Float Literals

Used for data type `float` and `double`.

```
double num1 = 22.4;  
float num2 = 22.4f;
```

Note: Always suffix float value with the “f” else compiler will consider it as double.

Char and String Literal

Used for char and String type.

```
char ch = 'Z';  
String str = "BeginnersBook";
```

Check out these [basic java programs](#) before proceeding to the next topic:

1. [Java Program to Add two Numbers](#)
2. [Java Program to Multiply two numbers](#)
3. [Java Program to read a number \(entered by user\)](#)