

# CCC 计算机竞赛预习材料—Basic Data Type

---

## Primitive Types

Primitive number types are divided into two groups:

**Integer types** stores whole numbers, positive or negative (such as 123 or -456), without decimals. Valid types are byte, short, int and long. Which type you should use, depends on the numeric value.

**Floating point types** represents numbers with a fractional part, containing one or more decimals. There are two types: float and double.

## Non-Primitive Data Types

Non-primitive data types are called **reference types** because they refer to objects.

The main difference between **primitive** and **non-primitive** data types are:

- Primitive types are predefined (already defined) in Java. Non-primitive types are created by the programmer and is not defined by Java (except for String).
- Non-primitive types can be used to call methods to perform certain operations, while primitive types cannot.
- A primitive type has always a value, while non-primitive types can be null.
- A primitive type starts with a lowercase letter, while non-primitive types starts with an uppercase letter.
- The size of a primitive type depends on the data type, while non-primitive types have all the same size.

## Byte

Can store whole numbers from -128 to 127. This can be used instead of int or other integer types to save memory when you are certain that the value will be within -128 and 127.

```
byte myNum = 100;  
  
System.out.println(myNum);
```

## Short

The short data type can store whole numbers from -32768 to 32767.

```
short myNum = 5000;  
  
System.out.println(myNum);
```

## Int

The int data type can store whole numbers from -2147483648 to 2147483647. In general, and in our tutorial, the int data type is the preferred data type when we create variables with a numeric value.

```
int myNum = 100000;  
  
System.out.println(myNum);
```

## Long

The long data type can store whole numbers from -9223372036854775808 to 9223372036854775807. This is used when int is not large enough to store the value. Note that you should end the value with an "L".

```
long myNum = 15000000000L;  
  
System.out.println(myNum);
```

## Float

The float data type can store fractional numbers from  $3.4\text{e}-038$  to  $3.4\text{e}+038$ . Note that you should end the value with an "f".

```
float myNum = 5.75f;  
  
System.out.println(myNum);
```

## Double

The double data type can store fractional numbers from  $1.7\text{e}-308$  to  $1.7\text{e}+308$ . Note that you should end the value with a "d".

```
double myNum = 19.99d;  
  
System.out.println(myNum);
```

## Booleans

A boolean data type is declared with the boolean keyword and can only take the values true or false.

```
boolean isJavaFun = true;  
  
boolean isFishTasty = false;  
  
System.out.println(isJavaFun);    // Outputs true  
  
System.out.println(isFishTasty);  // Outputs false
```

## Characters

The char data type is used to store a single character. The character must be surrounded by single quotes, like 'A' or 'c':

```
char myGrade = 'B';  
  
System.out.println(myGrade);
```

## Strings

The String data type is used to store a sequence of characters (text). String values must be surrounded by double quotes.

```
String greeting = "Hello World";  
System.out.println(greeting);
```

Reference : W3C School [https://www.w3schools.com/java/java\\_data\\_types.asp](https://www.w3schools.com/java/java_data_types.asp)