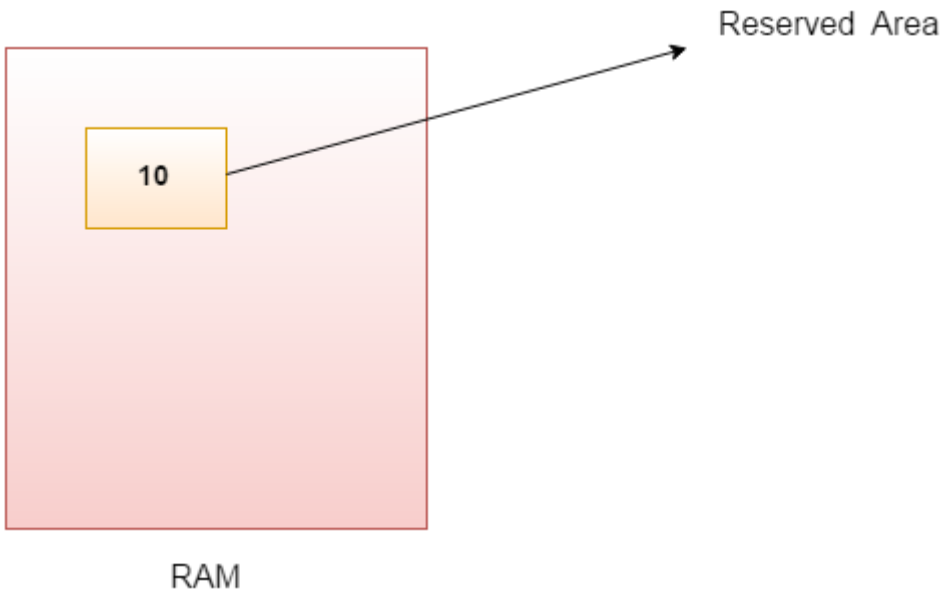
Variables and Data Types in Java

- Java variable is a name given to a memory location. It is the basic unit of storage in a program.
- Variables are the data containers that save the data values during Java program execution.
- The value stored in a variable can be changed during program execution.
- In Java, all variables must be declared before use.

Example:

int a=10;



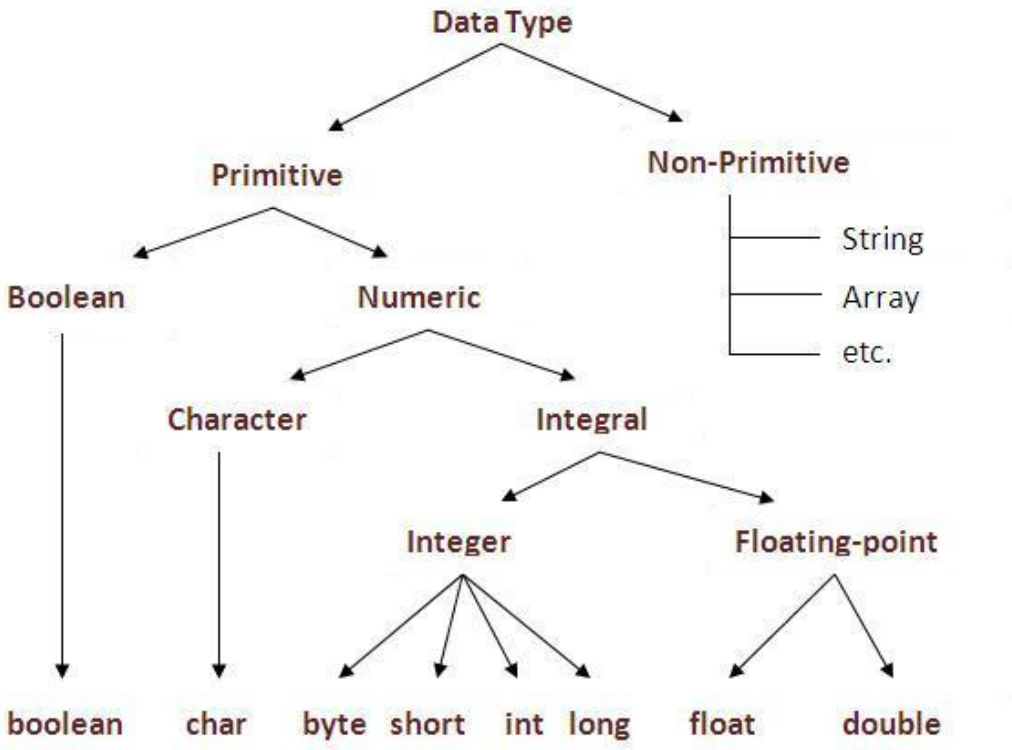
Note:

In Java, every variable and expression has some type; the compiler should check every assignment for type compatibility; because of this reason, Java language is a strongly typed programming language**.**

Data Types in Java

Data types represent the different values to be stored in the variable. In Java, there are two types of data types:

- **Primitive data types:** The primitive data types include boolean, char, byte, short, int, long, float, and double.
- **Non-primitive data types:** The non-primitive data types include Classes, Interfaces, and Arrays.



Data Type	Default Value	Default size
boolean	false	1 bit
char	'\u0000'	2 byte
byte	0	1 byte
short	0	2 byte
int	0	4 byte
long	0L	8 byte
float	0.0f	4 byte
double	0.0d	8 byte

Data Type Range

The range of values is calculated as $-(2^{n-1}) to\ (2^{n-1}) - 1$

Where n is the number of bits required.

For example, the byte data type requires 1 byte = 8 bits. Therefore, the range of values that can be stored in the byte data type is $-(2^{8-1})$ to $(2^{8-1})-1$. = -2^7 to (2^7) -1. = −128 to 127.

Data Type	Size	Description
byte	1 byte	Stores whole numbers from -128 to 127
short	2 bytes	Stores whole numbers from -32,768 to 32,767

int	4 bytes	Stores whole numbers from -2,147,483,648 to 2,147,483,647
long	8 bytes	Stores whole numbers from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
float	4 bytes	Stores fractional numbers. Sufficient for storing 6 to 7 decimal digits
double	8 bytes	Stores fractional numbers. Sufficient for storing 15 decimal digits
boolean	1 bit	Stores true or false values
char	2 bytes	Stores a single character/letter or ASCII values

Example: Add Two Numbers

AddTwoNumbers.java

```
1  class AddTwoNumbers{
2  public static void main(String[] args){
3  int a=10;
4  int b=20;
5  //int c=a+b;
6  //System.out.println("addition is:"+c);
7  System.out.println("Addition is:"+(a+b));
8  }
9  }
```

Output:

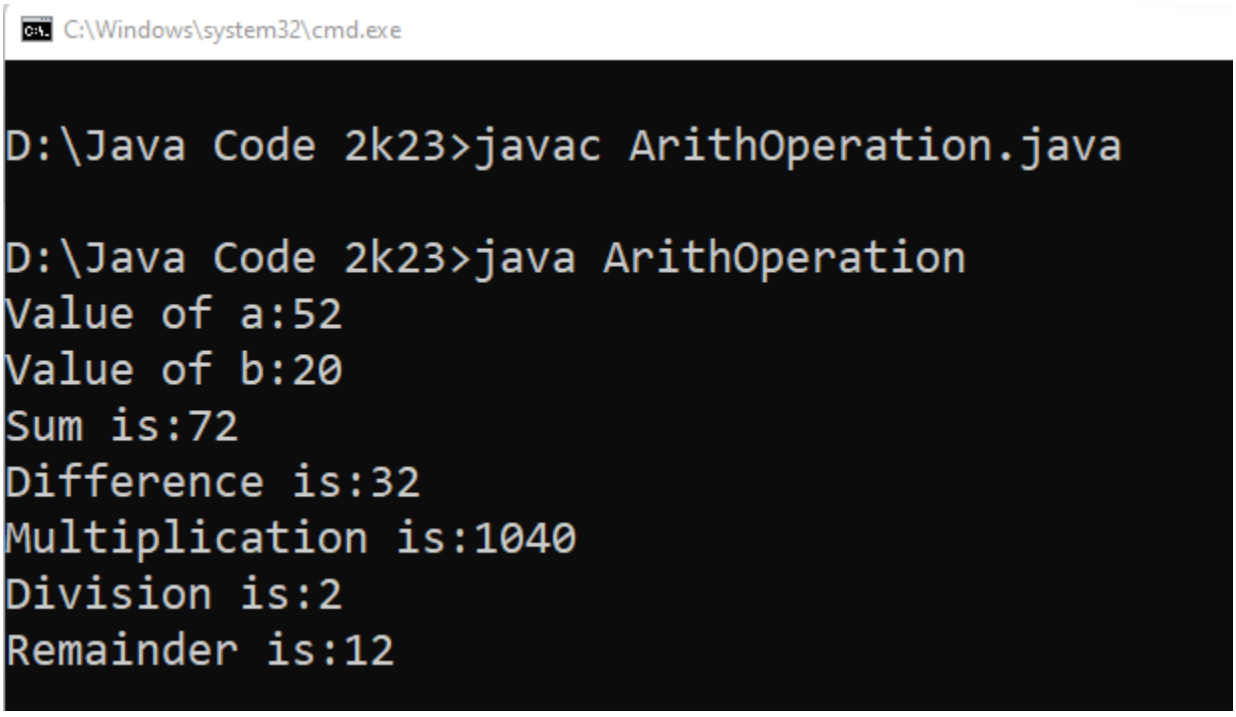
```
D:\Java Code 2k23>javac AddTwoNumbers.java

D:\Java Code 2k23>java AddTwoNumbers
Addition is:30
```

Example: Arithmetic Operations

ArithOperation.java

```
1  class ArithOperation
2  {
3      public static void main(String args[])
4      {
5          int a=52,b=20;
6          System.out.println("Value of a:"+a);
7          System.out.println("Value of b:"+b);
8          //System.out.println("Value of a is:"+a +"\nValue of b is:"+b);
9          int c=a+b;
10         System.out.println("Sum is:"+ c);
11         //System.out.println("Sum is:"+ (a+b));
12         System.out.println("Difference is:"+ (a-b));
13         System.out.println("Multiplication is:"+ (a*b));
14         System.out.println("Division is:"+ (a/b));
15         System.out.println("Remainder is:"+ (a%b));
16         }
17 }
18
```

Output:

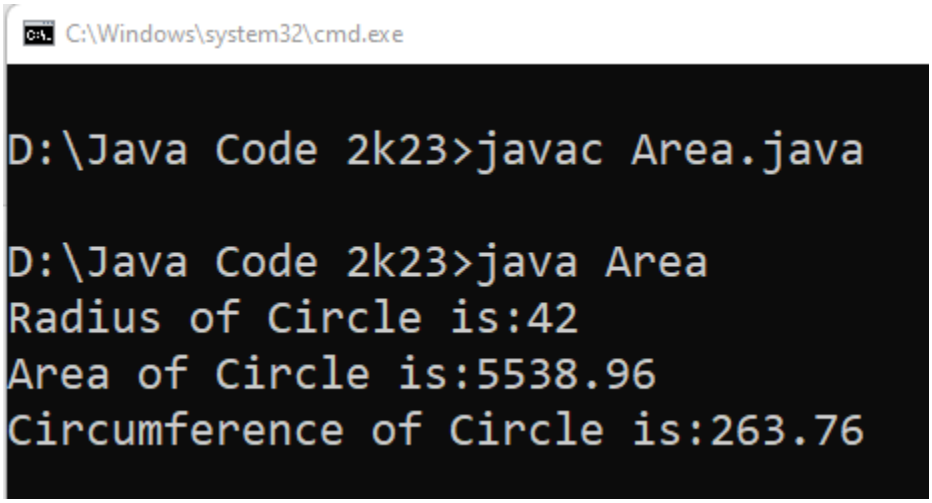


Example: Calculate the area and circumference of a circle.

Area.java

```java
class Area
{
    public static void main(String args[])
    {
        int r=42;
        double area, circum;
        area=3.14*r*r;
        //area=Math.PI*Math.pow(r, 2);
        circum=2*3.14*r;
        //circum =2*Math.PI*rad;
        System.out.println ("Radius of Circle is:"+r);
        System.out.println ("Area of Circle is:"+area);
        System.out.println ("Circumference of Circle is:"+circum);
        }
}
```

Output:



Printing default values of primitive data types.

Case 1:

DefaultValues.java

```
1  class DefaultValues
2  {
3  public static void main(String args[])
4  {
5  int i;
6  float f;
7
8  System.out.println(i);
9  System.out.println(f);
10 }
11 }
```

C:\Windows\system32\cmd.exe

```
D:\Java Code 2k23>javac DefaultValues.java
DefaultValues.java:8: error: variable i might not have bee
n initialized
System.out.println(i);
                   ^
DefaultValues.java:9: error: variable f might not have bee
n initialized
System.out.println(f);
                   ^
2 errors
```

Solution:

```java
class DefaultValues
{
public static void main(String args[])
{
int i=0;
float f=0.0f;

System.out.println(i);
System.out.println(f);
}
}
```

Output:

```
D:\Java Code 2k23>javac DefaultValues.java

D:\Java Code 2k23>java DefaultValues
0
0.0
```

Case 2

Declaring variables outside the main method.

DefaultValues.java

```
1  class DefaultValues
2  {
3  int i;
4  float f;
5  public static void main(String args[])
6  {
7  System.out.println(i);
8  System.out.println(f);
9  }
10 }
11
12
```

C:\Windows\system32\cmd.exe

```
D:\Java Code 2k23>javac DefaultValues.java
DefaultValues.java:7: error: non-static variable i cannot
be referenced from a static context
System.out.println(i);
                   ^
DefaultValues.java:8: error: non-static variable f cannot
be referenced from a static context
System.out.println(f);
                   ^
2 errors
```

Solution:

```java
class DefaultValues
{
static int i;
static float f;
public static void main(String args[])
{
System.out.println(i);
System.out.println(f);
}
}
```

```
D:\Java Code 2k23>javac DefaultValues.java

D:\Java Code 2k23>java DefaultValues
0
0.0
```

Java program that prints the range of primitive data types

DataTypeRangeExample.java

```java
public class DataTypeRangeExample {
    public static void main(String[] args) {
        System.out.println("Range of primitive data types:");

        System.out.println("byte: " + Byte.MIN_VALUE + " to " + Byte.MAX_VALUE);
        System.out.println("short: " + Short.MIN_VALUE + " to " + Short.MAX_VALUE);
        System.out.println("int: " + Integer.MIN_VALUE + " to " + Integer.MAX_VALUE);
        System.out.println("long: " + Long.MIN_VALUE + " to " + Long.MAX_VALUE);

        System.out.println("float: " + Float.MIN_VALUE + " to " + Float.MAX_VALUE);
        System.out.println("double: " + Double.MIN_VALUE + " to " + Double.MAX_VALUE);

        System.out.println("char: " + (int) Character.MIN_VALUE + " to " + (int) Character.MAX_VALUE);
        System.out.println("boolean: " + Boolean.FALSE + " to " + Boolean.TRUE);
    }
}
```

Output:

```
D:\Java Code 2k23>javac DataTypeRangeExample.java

D:\Java Code 2k23>java DataTypeRangeExample
Range of primitive data types:
byte: -128 to 127
short: -32768 to 32767
int: -2147483648 to 2147483647
long: -9223372036854775808 to 9223372036854775807
float: 1.4E-45 to 3.4028235E38
double: 4.9E-324 to 1.7976931348623157E308
char: 0 to 65535
boolean: false to true
```