



# Java CheatSheet

---

## For Beginners

HelloWorld.java

File Name

```
public class HelloWorld {  
  
    public static void main(String[] args) {  
  
        //Print "Hello World " in the terminal window  
        System.out.println("Hello World")  
  
    }  
}
```

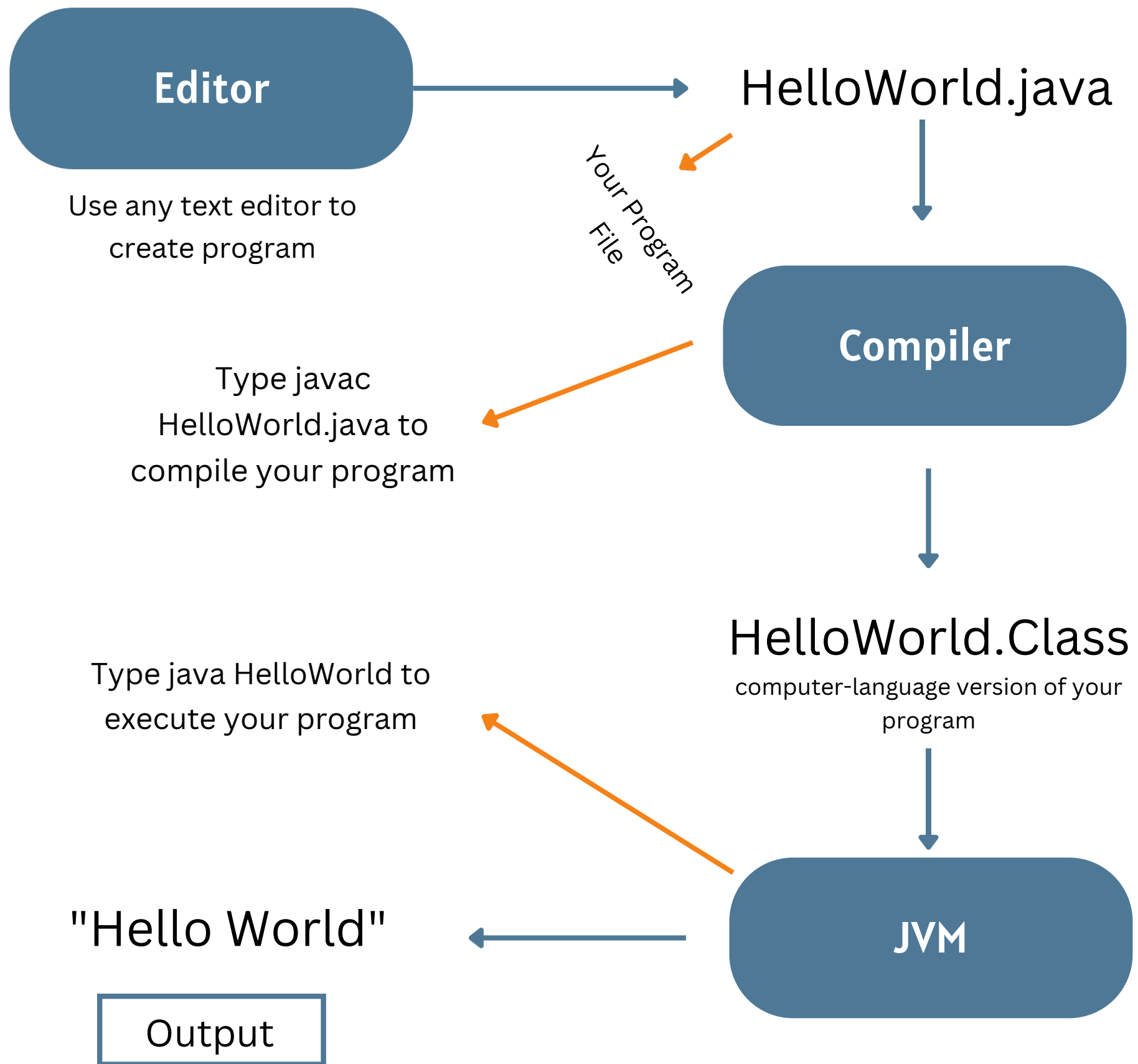
Class Name

Main Method

Statement

# Editing, compiling, and executing.

---



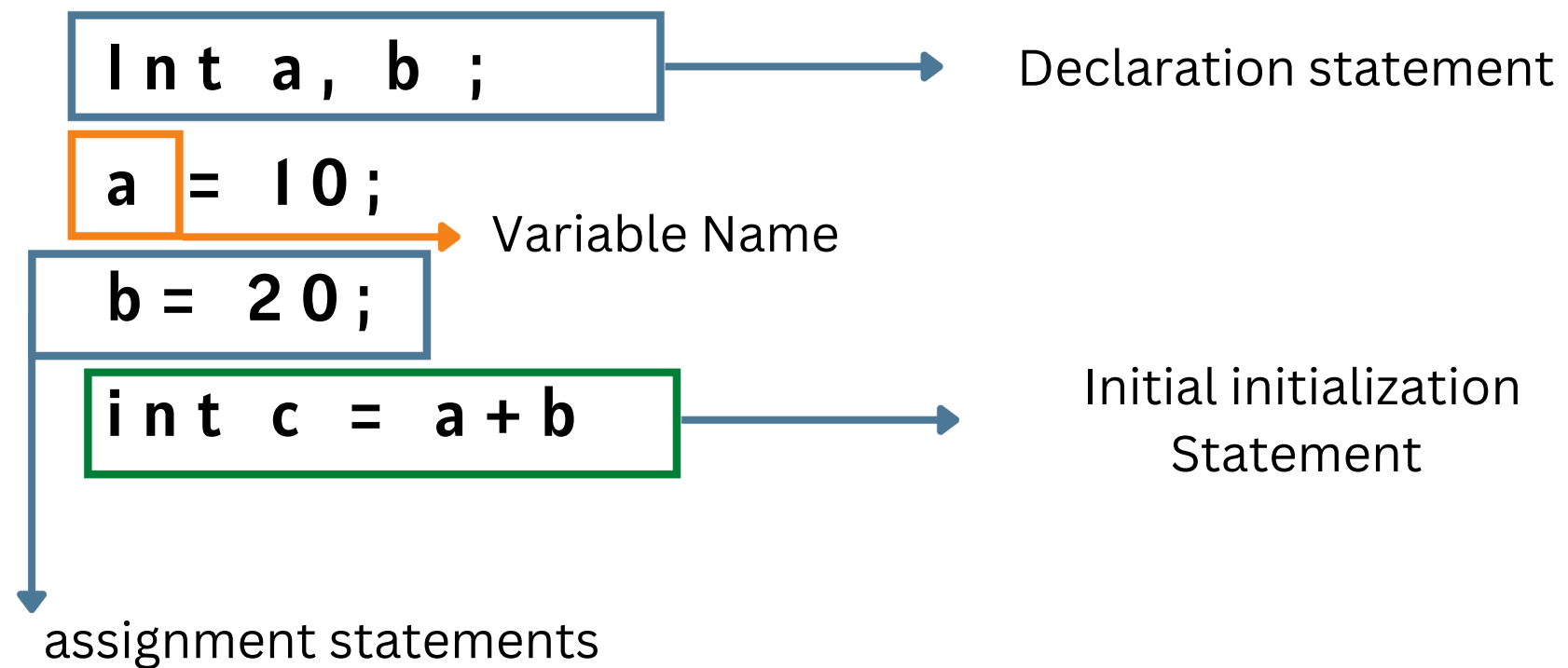
# Data Types and Examples

Type	Set of Values	Common Operators
Int	Integers	+ - * / %
double	floating point numbers	+ - * /
boolean	boolean value	&&    !
char	characters	
String	sequence of characters	+

Int	:	74, 99, 21478521
double	:	3.5, 2.55, 6.0222e8
boolean	:	true , flase
char	:	'A' , 'D' , 'F ' , '1' , '%' , '/n'
String	:	"AB" , "Hello"

# Declaration and assignment statements

---



# Integers

Values	Integers between $-2^{31}$ and $+2^{31}-1$					
Examples	1234, 24, 5, 200000					
Operations	sign	add	subtract	multiply	divide	remainder
operators	+-	+	-	*	/	%

# Floating-point numbers.

Values	Real Numbers (Specified by IEEE 754 Standard))					
Examples	3.14159, 2.0, 1.4142556568, 6.022e23					
Operations	add subtract multiply divide					
operators		+	-	*	/	

# Floating-point numbers.

Values	true or flase		
literals	True / Flase		
Operations	and	or	not
operators	&&		!

# Comparison operators.

---

op	Meaning	True	False
==	equal	2 == 2	2 == 5
!=	not equal	3 != 2	2 != 2
<	less then	2 < 13	2 < 1
<=	less then or equal	3 <= 4	3 <= 2
>	greater thenl	14 > 5	5 > 14
>=	greater then or equal	3 > = 2	2 >= 3

## Printing

---

`void System.out.print(String s)`  
`void System.out.println(String s)`

`print s`

`print s`

**Followed by new line**

`print a new line`

`void System.out.println()`

# if and if else statement

---

<i>absolute value</i>	<pre>if (x &lt; 0) x = -x;</pre>
<i>put the smaller value in x and the larger value in y</i>	<pre>if (x &gt; y) {     int t = x;     x = y;     y = t; }</pre>
<i>maximum of x and y</i>	<pre>if (x &gt; y) max = x; else      max = y;</pre>
<i>error check for division operation</i>	<pre>if (den == 0) System.out.println("Division by zero"); else          System.out.println("Quotient = " + num/den);</pre>
<i>error check for quadratic formula</i>	<pre>double discriminant = b*b - 4.0*c; if (discriminant &lt; 0.0) {     System.out.println("No real roots"); } else {     System.out.println((-b + Math.sqrt(discriminant))/2.0);     System.out.println((-b - Math.sqrt(discriminant))/2.0); }</pre>

# Nested if-else statement.

---

```
// Outer if statement.
if(condition)
{
    // Inner if statement defined in outer if else statement.
    if(condition)
        statement1;
} // Else part of outer if
statement. else {

    statement2;

}
```

For example:

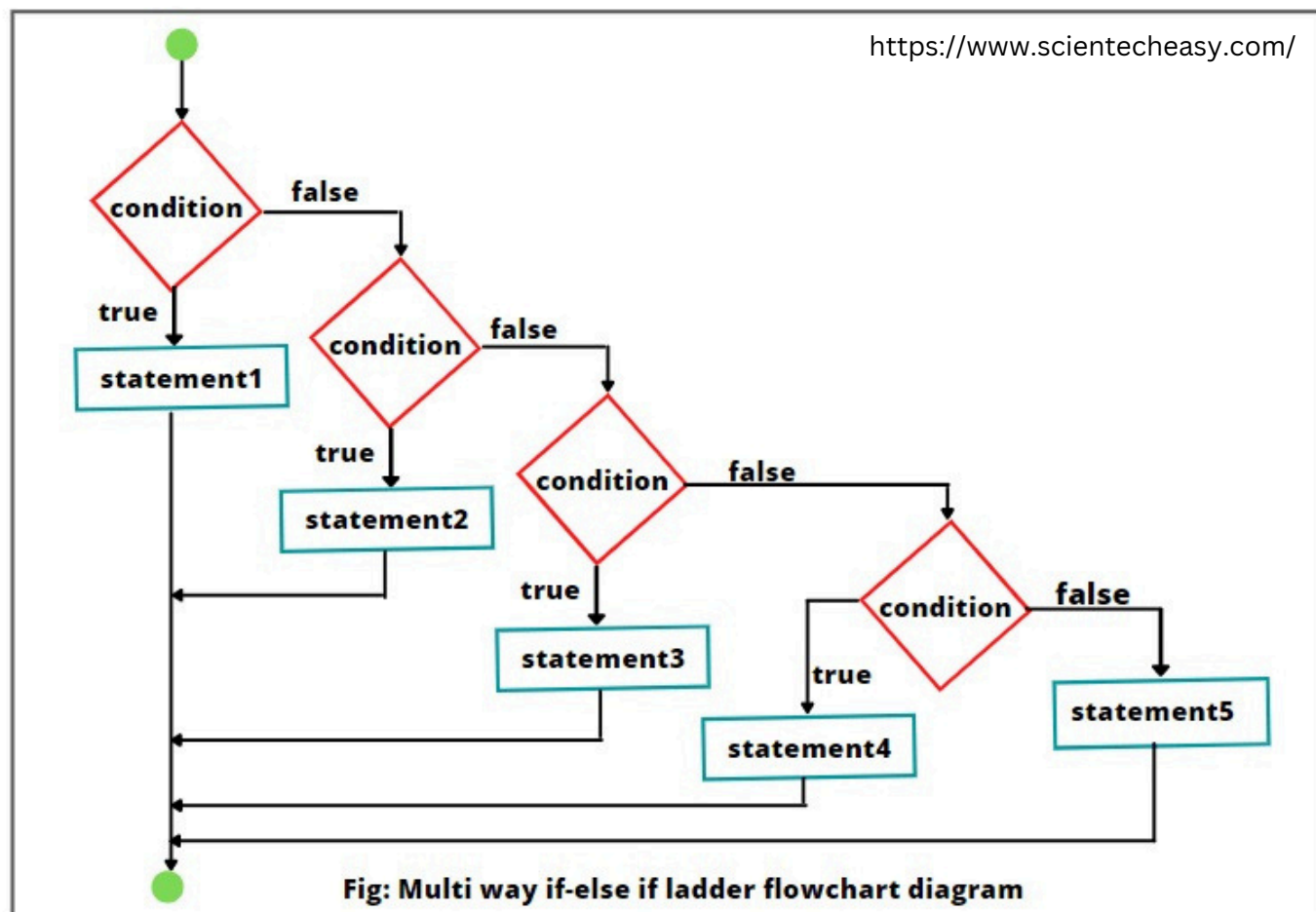
```
if (x > y)
{
    if (y > z)
        System.out.println("x is greater than y and z"); //
statement1.
}
else
System.out.println("x is less than or equal to y"); //
statement2.
```



# if-else if Ladder Statements in Java

---

```
if(condition)
    statement1;
else if(condition)
    statement2;
else if(condition)
    statement3;
...
else
    statement4;
```



# while loop.

---

```
int power = 1;  
while ( power <= n/2)  
{  
    power = 2*power;  
}
```

Loop Continuation condition

Brackets are option when body  
is a single statement

BODY

# For Loop

---

declare and initialize a loop control value

Loop Continuation condition

Increment

```
int power = 1;  
for ( int i = 0 ; i<=n ; i++)  
{  
    System.out.println( i + " " + power)  
    power = 2*power  
}
```

Body

The diagram illustrates the components of a Java for loop. It shows a code snippet with three parts of the for loop header highlighted by blue boxes: 'int i = 0', 'i<=n', and 'i++'. A blue arrow points from the text 'declare and initialize a loop control value' to the first box. Two orange arrows point from the text 'Loop Continuation condition' to the second box. Another orange arrow points from the text 'Increment' to the third box. The body of the loop, consisting of two lines of code, is enclosed in a blue box, with an orange arrow pointing from the text 'Body' to this box. The entire code snippet is presented within a dark-themed window with three colored window control buttons (red, yellow, grey) in the top-left corner.

# Loops.

---

<i>compute the largest power of 2 less than or equal to n</i>	<pre>int power = 1; while (power &lt;= n/2)     power = 2*power; System.out.println(power);</pre>
<i>compute a finite sum (1 + 2 + ... + n)</i>	<pre>int sum = 0; for (int i = 1; i &lt;= n; i++)     sum += i; System.out.println(sum);</pre>
<i>compute a finite product (n! = 1 × 2 × ... × n)</i>	<pre>int product = 1; for (int i = 1; i &lt;= n; i++)     product *= i; System.out.println(product);</pre>
<i>print a table of function values</i>	<pre>for (int i = 0; i &lt;= n; i++)     System.out.println(i + " " + 2*Math.PI*i/n);</pre>
<i>compute the ruler function (see PROGRAM 1.2.1)</i>	<pre>String ruler = "1"; for (int i = 2; i &lt;= n; i++)     ruler = ruler + " " + i + " " + ruler; System.out.println(ruler);</pre>

# Switch statement.

---

```
switch(expression){
case value1:
    //code to be executed;
    break; //optional
case value2:
    //code to be executed;
    break; //optional
.....

default:
    code to be executed if all cases are not
matched;
}
```

# Arrays

---

## Single Dimensional Array in Java

### **Syntax to Declare an Array in Java**

```
dataType[] arr; (or)  
dataType []arr; (or)  
dataType arr[];
```

### **Instantiation of an Array in Java**

```
arrayRefVar=new datatype[size];
```

# Arrays

---

## Single Dimensional Array in Java

```
int a[]=new int[5]; //declaration and  
instantiation
```

```
a[0]=27; //initialization  
a[1]=25;          a[2]=75;  
a[3]=47; a[4]=59;
```

# Arrays

---

## Multidimensional Array in Java

### **Syntax to Declare Multidimensional Array in Java**

```
dataType[][] arrayRefVar; (or)  
dataType [][]arrayRefVar; (or)  
dataType arrayRefVar[][]; (or)  
dataType []arrayRefVar[];
```

### **Example to instantiate Multidimensional Array in Java**

```
int[][] arr=new int[3][3];//3 row and 3 column
```



# Arrays

---

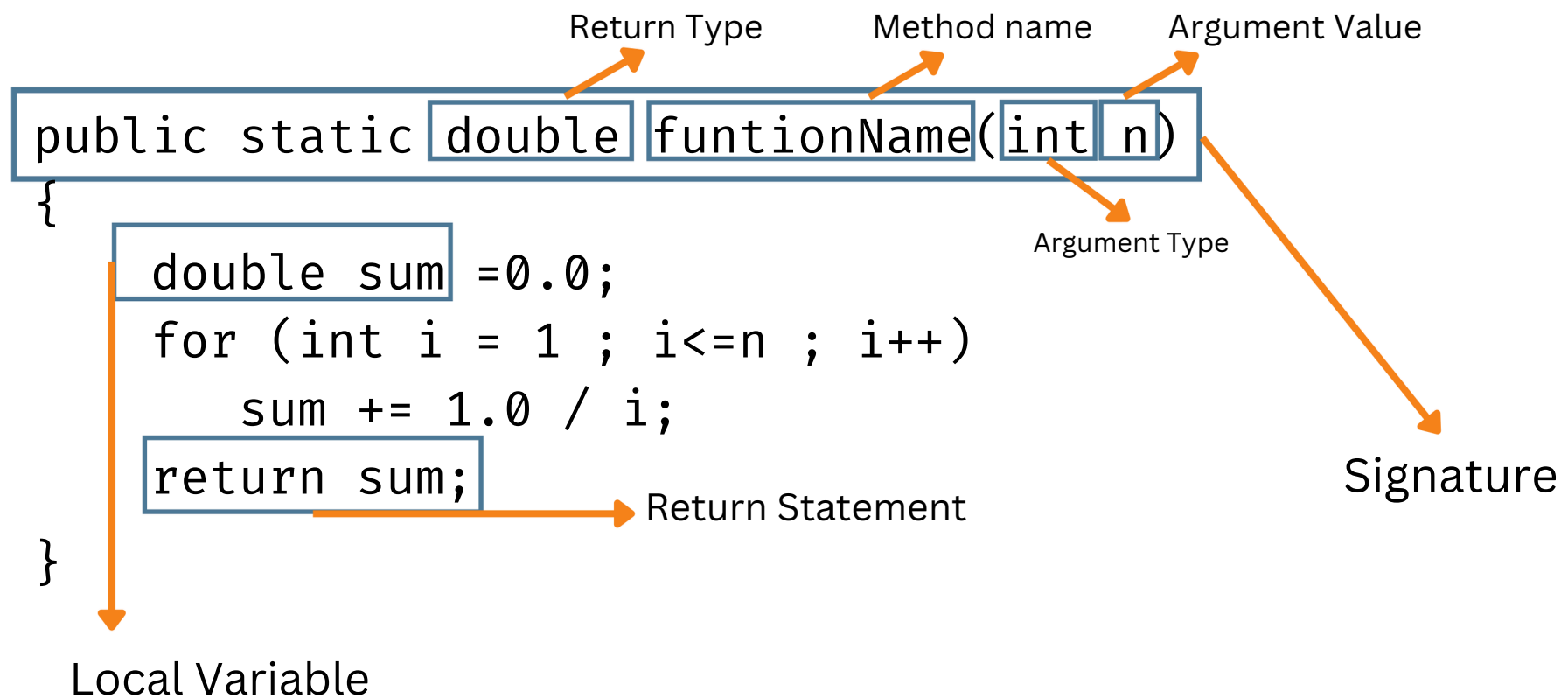
## Multidimensional Array in Java

### **Example to initialize Multidimensional Array in Java**

```
arr[0][0]=1;  
arr[0][1]=2;  
arr[0][2]=3;  
arr[1][0]=4;  
arr[1][1]=5;  
arr[1][2]=6;  
arr[2][0]=7;  
arr[2][1]=8;  
arr[2][2]=9;
```

# Functions

---



# Functions

---

<i>absolute value of an int value</i>	<pre>public static int abs(int x) {     if (x &lt; 0) return -x;     else      return x; }</pre>
<i>absolute value of a double value</i>	<pre>public static double abs(double x) {     if (x &lt; 0.0) return -x;     else        return x; }</pre>
<i>primality test</i>	<pre>public static boolean isPrime(int n) {     if (n &lt; 2) return false;     for (int i = 2; i &lt;= n/i; i++)         if (n % i == 0) return false;     return true; }</pre>
<i>hypotenuse of a right triangle</i>	<pre>public static double hypotenuse(double a, double b) { return Math.sqrt(a*a + b*b); }</pre>
<i>harmonic number</i>	<pre>public static double harmonic(int n) {     double sum = 0.0;     for (int i = 1; i &lt;= n; i++)         sum += 1.0 / i;     return sum; }</pre>
<i>uniform random integer in [0, n)</i>	<pre>public static int uniform(int n) { return (int) (Math.random() * n); }</pre>
<i>draw a triangle</i>	<pre>public static void drawTriangle(double x0, double y0,                                 double x1, double y1,                                 double x2, double y2 ) {     StdDraw.line(x0, y0, x1, y1);     StdDraw.line(x1, y1, x2, y2);     StdDraw.line(x2, y2, x0, y0); }</pre>

# Classes

