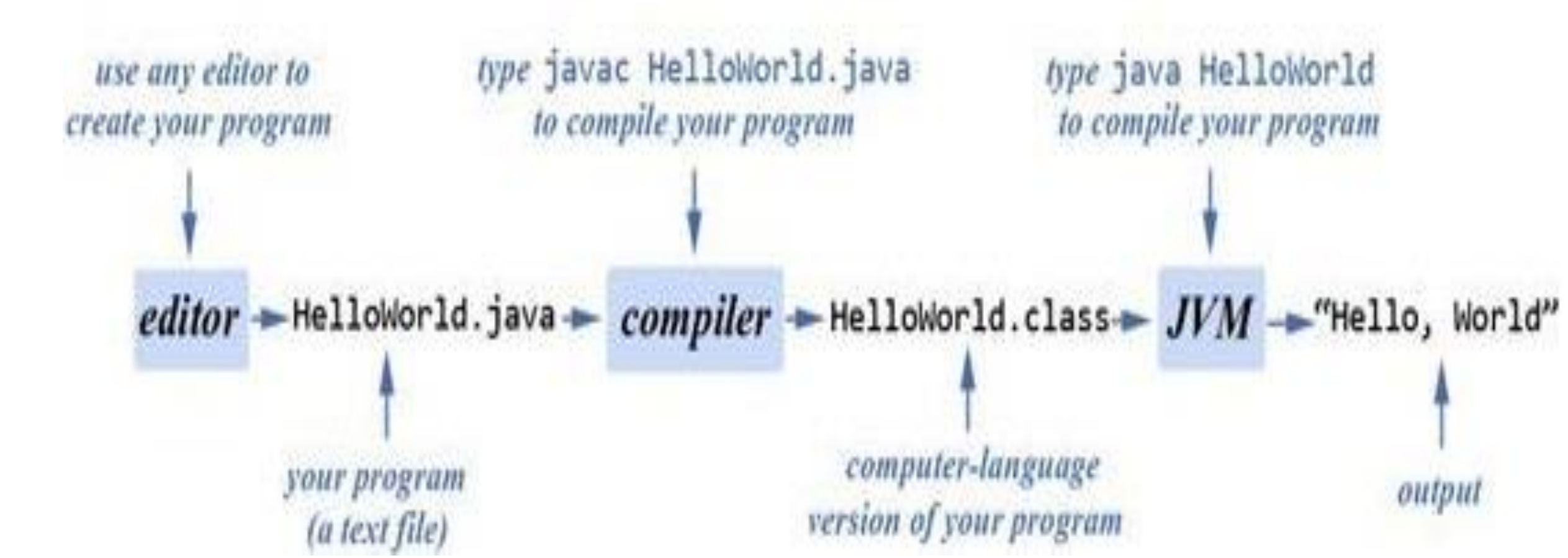


Athanasia Katsouraki - Promo P20

Input and Output



Basic concepts

- Class** Describes a particular kind of object. It can contain related methods and variables.
- Method** A function defined in a class. Methods implement the behavior for objects.
- Object** The principal building blocks of Java. Objects consist of variables (data) and methods (functionality).

| SEPARATORS | |
|------------|------------------------------------------------------------------------------------------------------------|
| () | Used to surround parameters. |
| { } | Defines a block of code for a class or method or to contain the values of automatically initialized arrays |
| [] | Declares arrays or references array values |
| : | Denotes the end of a statement |
| . | Separates variables |
| . | Separates package names from subpackages/ classes or a variable/method from a reference variable |

Primitive Data types

| Type | Description | Bits |
|----------------|---------------------------------|------|
| (Integers) | | |
| byte | Byte-length integer | 8 |
| short | Short integer | 16 |
| int | Integer | 32 |
| long | Long integer | 64 |
| (Real Numbers) | | |
| float | Single-precision floating point | 32 |
| double | Double-precision floating point | 64 |
| (Other) | | |
| char | A single character | 16 |
| boolean | A boolean value (true or false) | 1 |

Variables and Operators

Control Flow

Selection

Loop

Branch

Continue

Variable

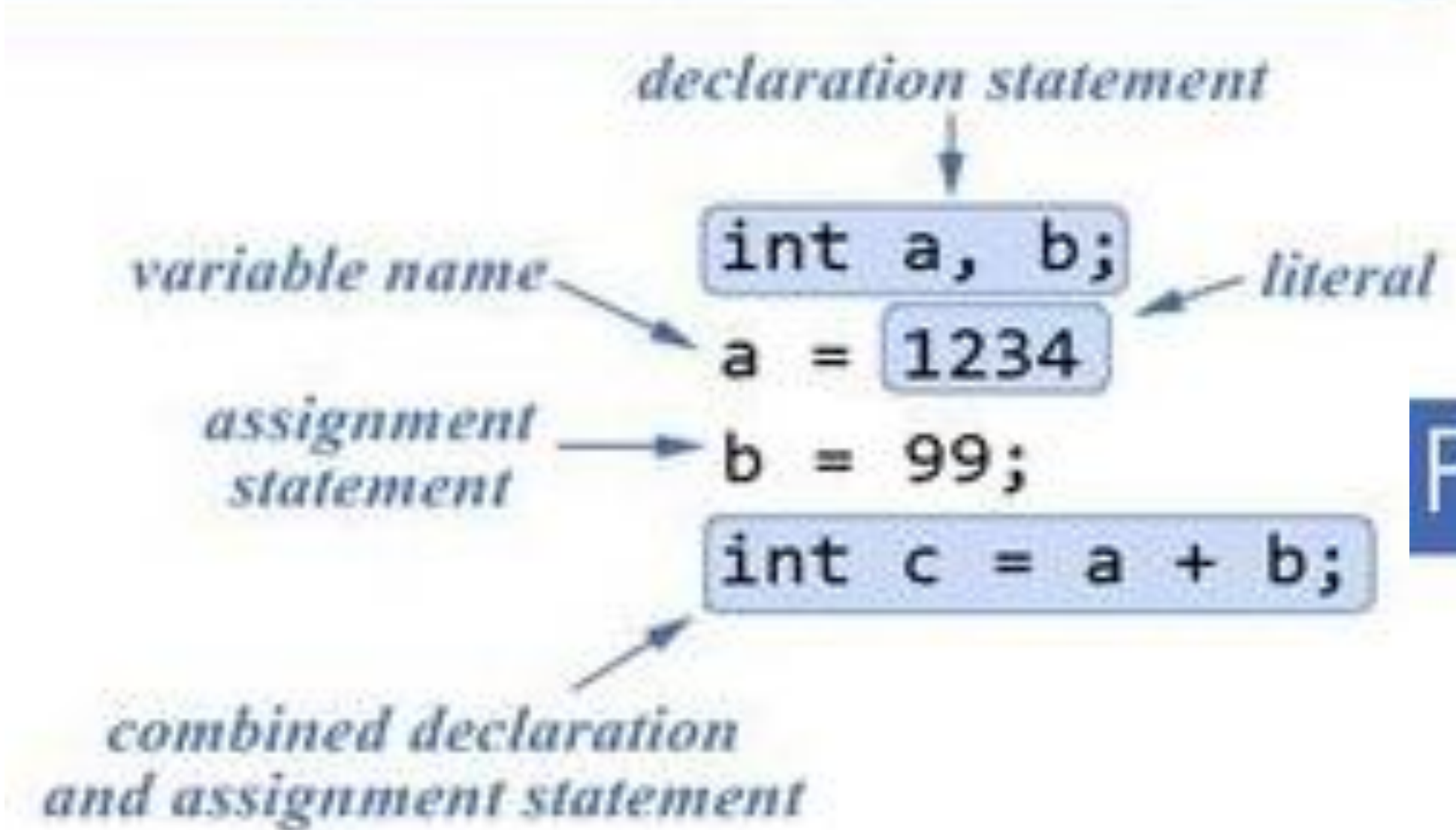
Arithmetic Operators

Relational Operators

Logical Operators

Bitwise Operators

Assignment status



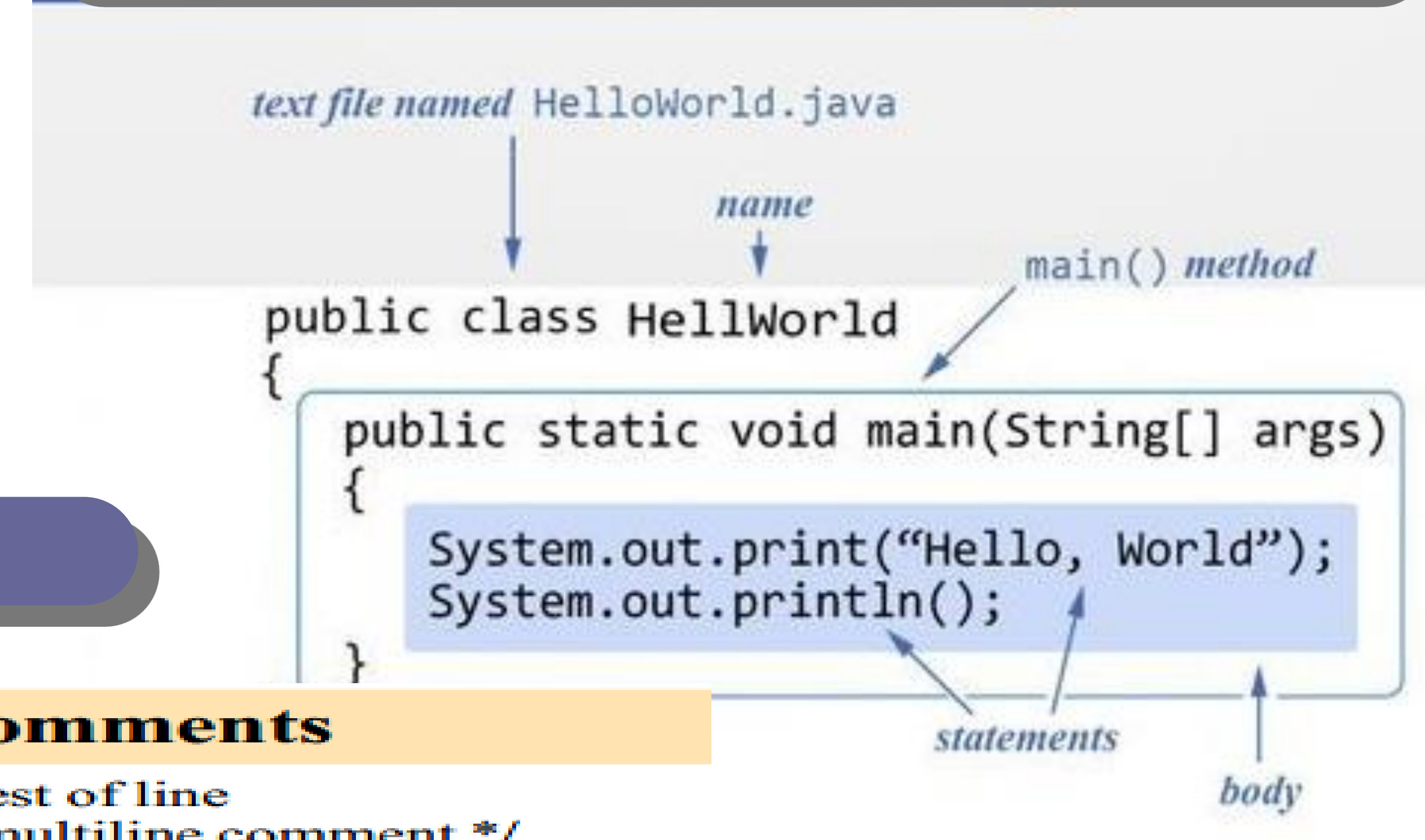
Comparison

| op | meaning | true | false |
|----|-----------------------|--------|--------|
| == | equal | 2 == 2 | 2 == 3 |
| != | not equal | 3 != 2 | 2 != 2 |
| < | less than | 2 < 13 | 2 < 2 |
| <= | less than or equal | 2 <= 2 | 3 <= 2 |
| > | greater than | 13 > 2 | 2 > 13 |
| >= | greater than or equal | 3 >= 2 | 2 >= 3 |

Arrays

- `int array_name[];` declare an integer array and allocate 100 elements of memory.
- `array_name=new int[100];` declare and allocate an integer array in one statement.
- `int array_name[] = new int[100];` declare and allocate an integer array in one statement.
- `int array_name [] = {1, 2, 3, 4};` declare and allocate an integer array in one statement.
- `int array_name[] = new int[10][20];` multidimensional array.
- `null` for reference type (class, array).

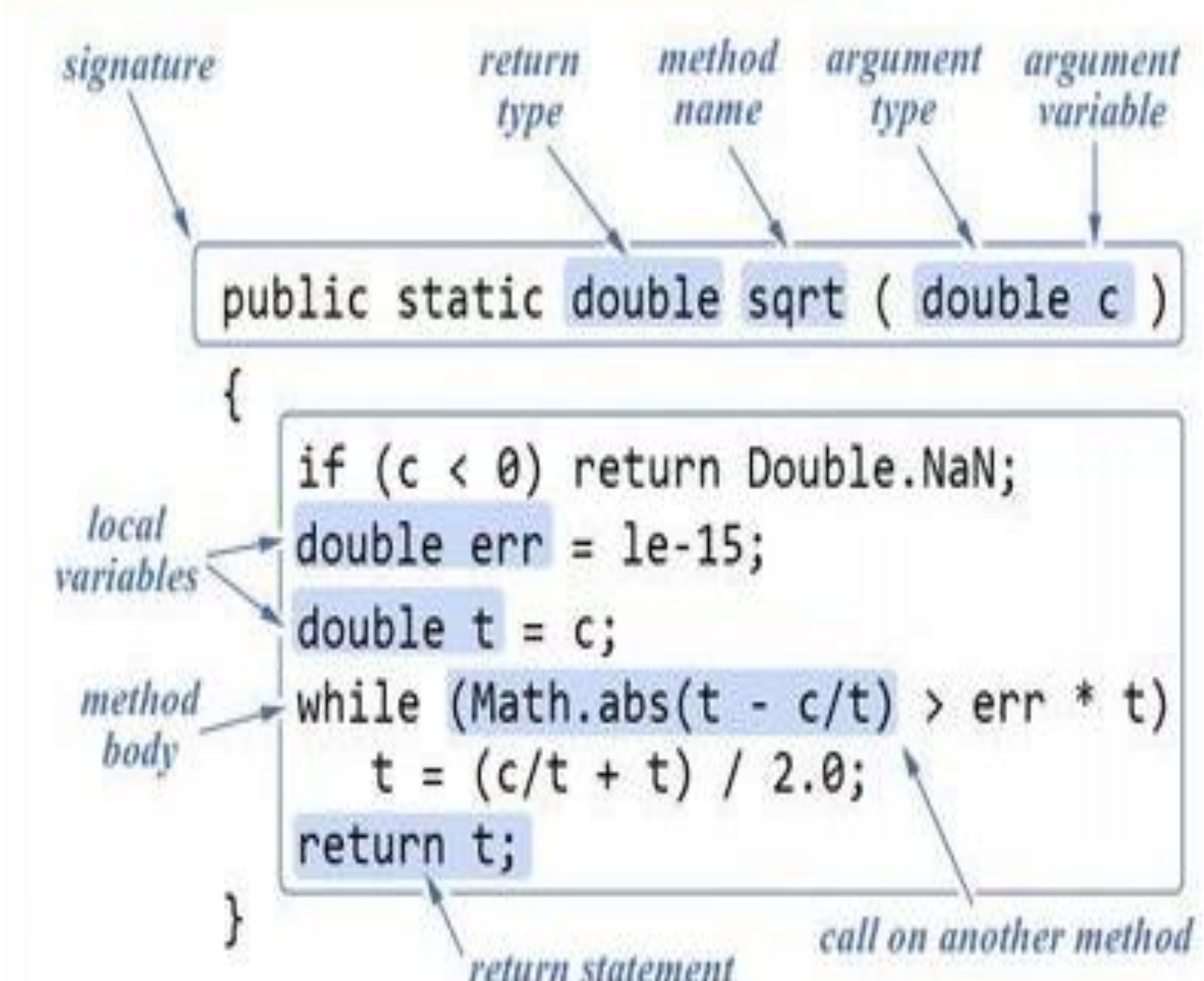
Basic Code Structure



Comments

- `// rest of line`
- `/* multiline comment */`
- `/** documentation comment */`

Functions



Class

`{public|final|abstract} class name`
`{ [class_variable_declarations]`
`public static void main{String[] args) {statements}`
`[methods]`
`}`

this, super

Key-Words

| | | | | |
|-----------|------------|-----------|--------------|------------|
| abstract | boolean | break | byte | byvalue |
| case | cast | catch | char | class |
| const | continue | default | do | double |
| else | extends | false | final | finally |
| float | for | future | generic | goto |
| if | implements | import | inner | instanceof |
| int | interface | long | native | new |
| null | operator | outer | package | private |
| protected | public | rest | return | short |
| static | super | switch | synchronized | this |
| throw | throws | transient | true | try |
| var | void | volatile | while | |

Loops

