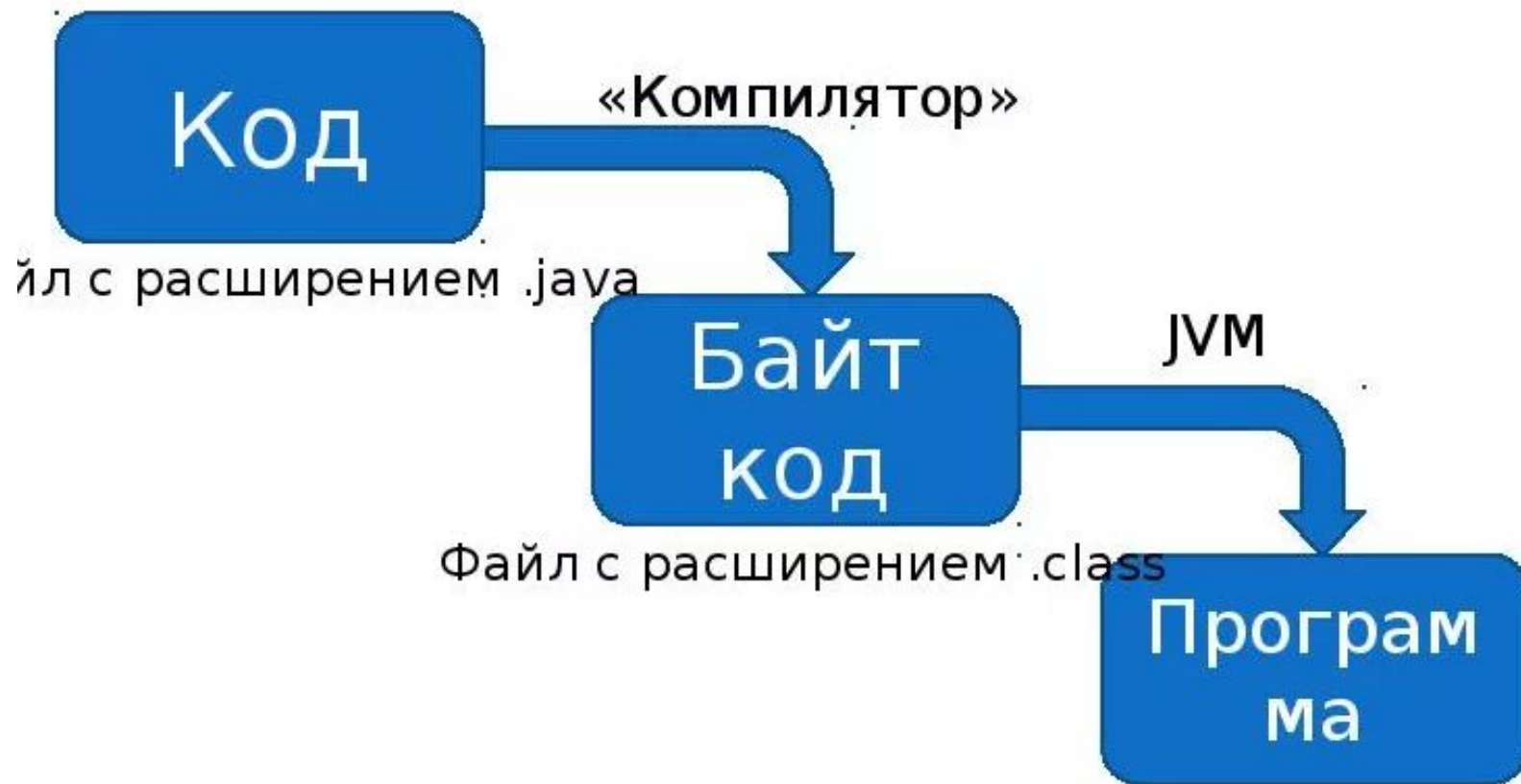




# Java Programming



# Типы данных

| Тип     | Размер  | Диапазон  | Значение по умолчанию | Пример  |
|---------|---------|---|-----------------------|---|
| boolean | 1 bit   | NA  | false                 | boolean bool = true;  |
| char    | 16 bits | Символы Unicode                                       | '\u0000' или 0        | char c = 'A';<br>char c = '\u0041';<br>char c = 65;<br>char c = '\t'; |
| byte    | 8 bits  | [-128,127] или [-2 <sup>7</sup> to 2 <sup>7</sup> -1] | 0                     | byte b = 10;<br>byte b = 0b010;                                       |
| short   | 16 bits | [-32768,32767]  | 0                     | short s = 32;<br>short s = 'A';                                       |
| int     | 32 bits | [-2147483648,2147483647]                              | 0                     | int i = 10;<br>int i = 'A';   |
| long    | 64 bits | [-2 <sup>63</sup> ,2 <sup>63</sup> -1]                | 0                     | long l = 3200L;<br>long l = 3200;                                     |
| float   | 32 bits | [-3.4E38, 3.4E38]                                     | 0.0f                  | float f = (float) 12.34;<br>float f = 12.34f;                         |
| double  | 64 bits | [-1.7E308, 1.7E308]                                   | 0.0                   | double d = 12.34;   |

## Арифметические бинарные операции

В Java определены следующие **арифметические бинарные операции**:

- сложение "+";
- вычитание "-";
- умножение "\*";
- деление "/";
- вычисление остатка от деления целых чисел "%" (возвращает остаток от деления первого числа на второе, причем результат будет иметь тот же знак, что и делимое), например, результат операции `5%3` будет равен `2`, а результат операции `(-7)%(-4)` будет равен `-3`. В Java операция может использоваться и для вещественных переменных (типа `float` или `double`).

## Унарные операции

В Java определены следующие унарные операции:

- унарный минус "-" – меняет знак числа или выражения на противоположный;
- унарный плюс "+" – не выполняет никаких действий над числом или выражением;
- побитовое дополнение "~" (только для целых) – инвертирует все биты поля числа (меняет 0 на 1 и 1 на 0);
- инкремент "++" (только для целых) – увеличивает значение переменной на 1;
- декремент "--" (только для целых) – уменьшает значение переменной на 1.

## Операции сравнения

В Java определены следующие операции сравнения:

- "`==`" (равно), "`!=`" (не равно),
- "`>`" (больше), "`>=`" (больше или равно),
- "`<`" (меньше) "`<=`" (меньше или равно)

## Булевские операции

**Булевские операции** выполняются над булевскими переменными и их результатом также является значение типа **boolean**. В

Java определены следующие булевские операции:

- отрицание `!"` – замена `false` на `true`, или наоборот;
- операция **и** `"&"` – результат равен `true`, только, если оба операнда равны `true`, иначе результат – `false`;
- операция **или** `"|"` – результат равен `true`, только, если хотя бы один из операндов равен `true`, иначе результат – `false`.
- операция **исключающее или** `"^"` – результат равен `true`, только, если операнды не равны друг другу, иначе результат – `false`.

Операции `"&"`, `"|"` и `"^"` можно, также как и соответствующие побитовые операции использовать в составных операциях присваивания: `"&="`, `"|="` и `"^="`

# Scanner

```
1 public class Main {
2
3     public static void main(String[] args) {
4
5         Scanner sc = new Scanner(System.in);
6         System.out.println("Введите число:");
7
8         int number = sc.nextInt();
9
10        System.out.println("Спасибо! Вы ввели число " + number);
11
12    }
13 }
```

```
1 public class Main {
2
3     public static void main(String[] args) {
4
5         Scanner sc = new Scanner(System.in);
6         System.out.println("Введите число:");
7
8         if (sc.hasNextInt()) {
9             int number = sc.nextInt();
10            System.out.println("Спасибо! Вы ввели число " + number);
11        } else {
12            System.out.println("Извините, но это явно не число. Перезапустите программу и попробуйте сн
13        }
14
15    }
16 }
```