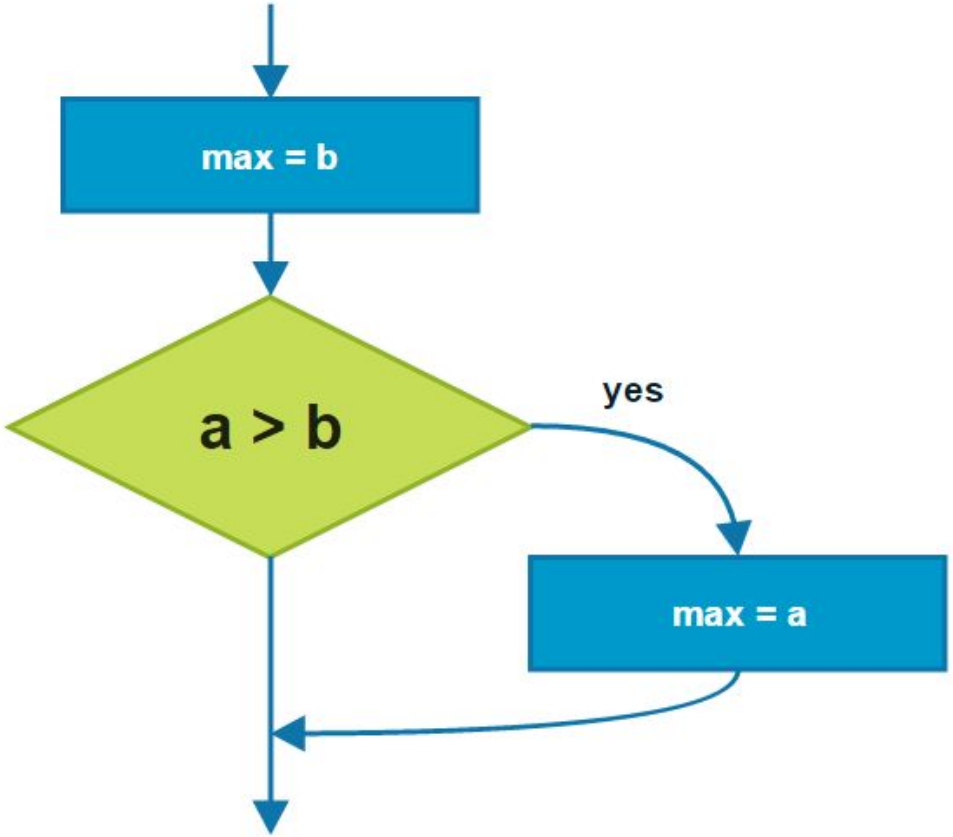
Basic structures

conditionals and loops

If-then-else statement

- needed if you want to make a decision
- executes a section of code if a condition (test) evaluates to true
- else - provides a secondary option of a particular statement execution

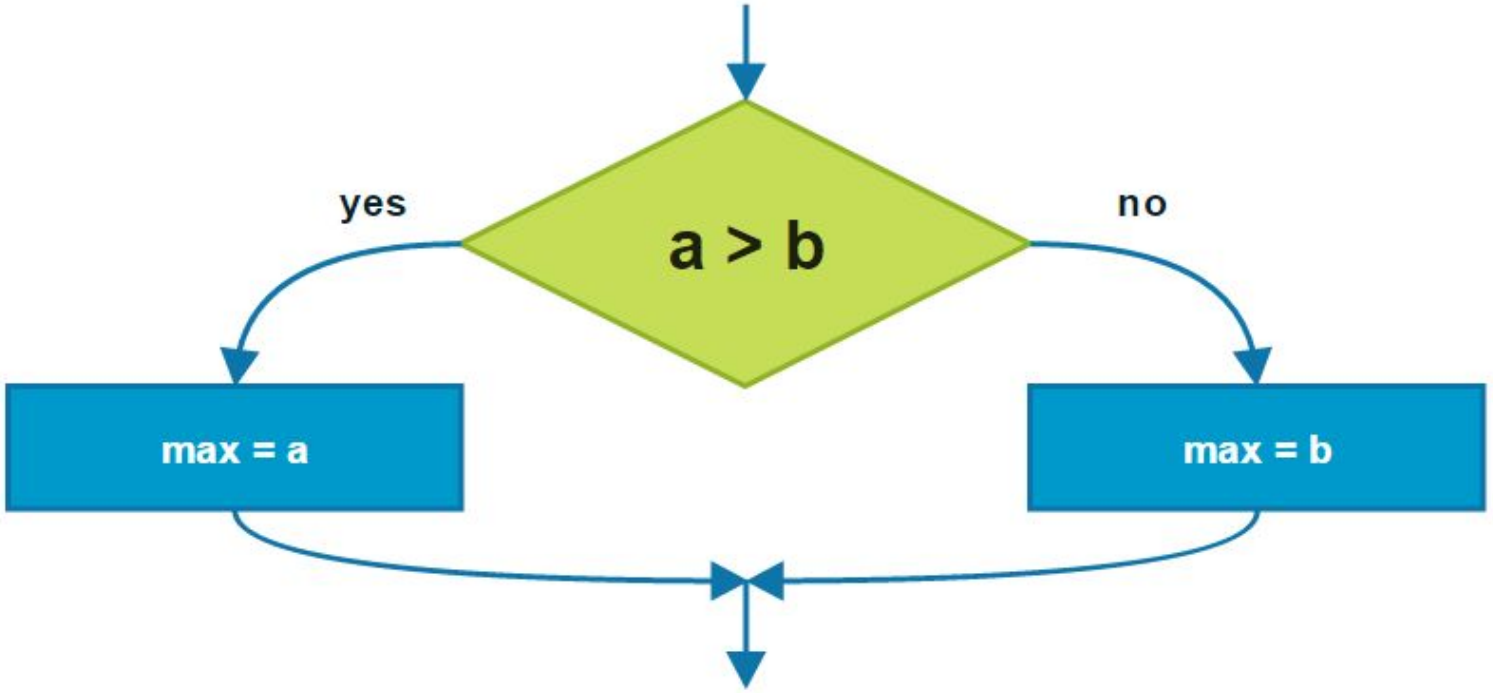
If-then flow chart



Code example

```
max = b;
if (a > b) {
    max = a;
}
```

If-then-else flow chart



Code example

```
if (a > b) {
    max = a;
} else {
    max = b;
}
```

“? :” operator

```
max = (a > b) ? a : b;
```

Relational operators

Relational operator	Meaning
==	equal to
!=	not equal to
>	greater than
>=	greater than or equal to
<	less than
<=	less than or equal to

Conditional and bitwise operators

Operator	Meaning
&	bitwise AND operation
|	bitwise OR operation
>>, <<, >>>	bit shifts
&&	conditional AND
||	conditional OR

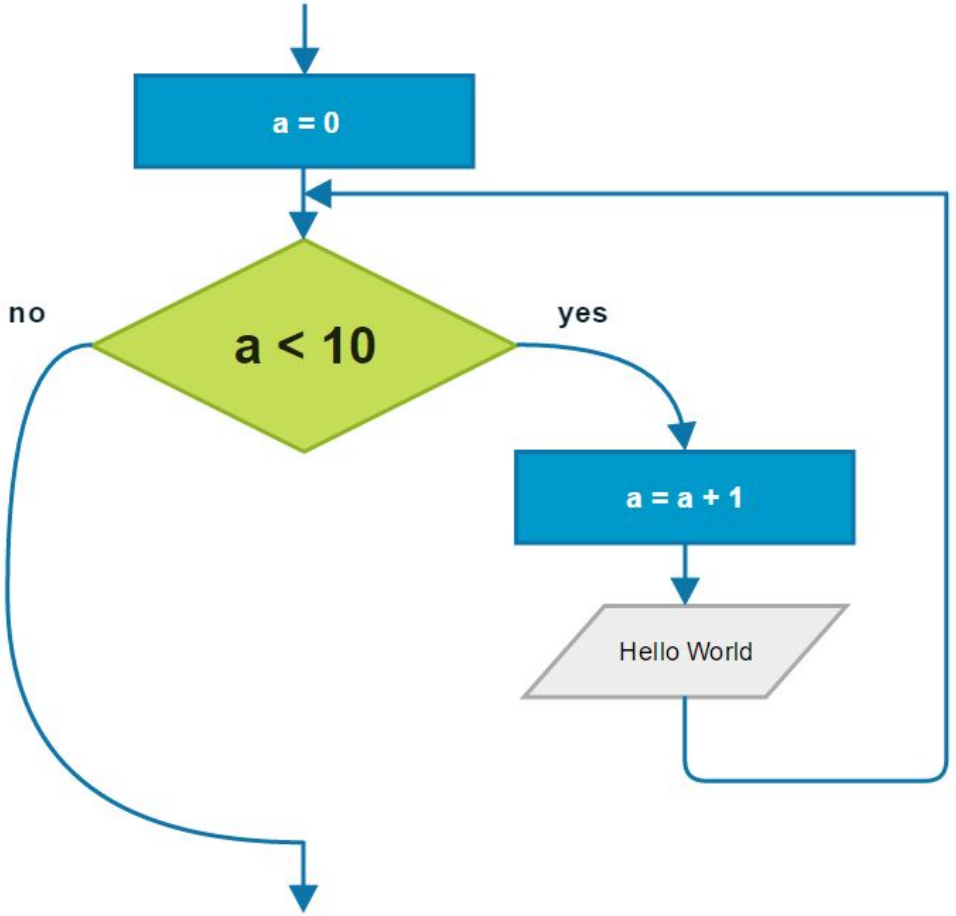
Conditional-AND (&&)

First operand	Second operand	Result (operand1 && operand2)
true	true	true
true	false	false
false	true	false
false	false	false

Conditional-OR (||)

First operand	Second operand	Result (operand1 || operand2)
true	true	true
true	false	true
false	true	true
false	false	false

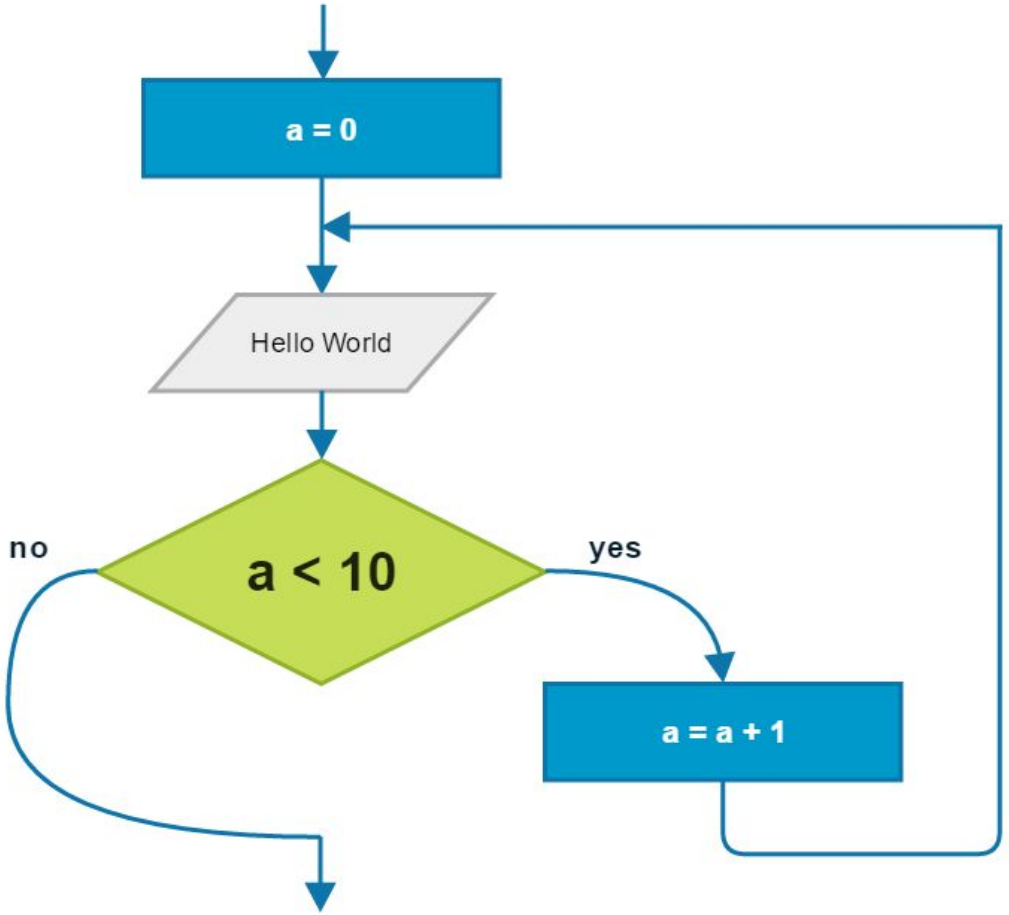
While loop



Code example

```
a = 0;
while (a < 10) {
    a++;
    System.out.println("Hello World");
}
```

Do-While loop



Code example

```
a = 0;
do {
    System.out.println("Hello World");
    a++;
} while (a < 10);
```