



# Java loops

# Hello!

**I am Sayyid Mohammad  
Reza Ayazi**


**B.Sc. Computer Engineering  
M.A Industrial Management  
Supply Chain  
smrayazi@gmail.com**




# Java While Loop

- ◎ The while loop loops through a block of code as long as a specified condition is **true**:

```
while (condition) {  
    // code block to be executed  
}
```




```
int i = 0;
while (i < 5) {
    System.out.println(i);
    i++;
}
```




## The Do/While Loop

- © The do/while loop is a variant of the while loop. This loop will execute the code block once, before checking if the condition is true, then it will repeat the loop as long as the condition is true.

```
do {  
    // code block to be executed  
}  
while (condition);
```




```
int i = 0;
do {
    System.out.println(i);
    i++;
}
while (i < 5);
```



## Java For Loop

- ◎ When you know exactly how many times you want to loop through a block of code, use the **for** loop instead of a **while** loop:
  - **Statement 1** is executed (one time) before the execution of the code block.
  - **Statement 2** defines the condition for executing the code block.
  - **Statement 3** is executed (every time) after the code block has been executed.

```
for (statement 1; statement 2; statement 3) {  
    // code block to be executed  
}
```



```
for (int i = 0; i < 5; i++) {  
    System.out.println(i);  
}
```

```
for (int i = 0; i <= 10; i = i + 2) {  
    System.out.println(i);  
}
```



## For-Each Loop

- © There is also a "for-each" loop, which is used exclusively to loop through elements in an array

```
for (type variableName : arrayName) {  
    // code block to be executed  
}
```

```
String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};  
for (String i : cars) {  
    System.out.println(i);  
}
```

## Java Break

- © The `break` statement can also be used to jump out of a loop

```
for (int i = 0; i < 10; i++) {  
    if (i == 4) {  
        break;  
    }  
    System.out.println(i);  
}
```

## Java Continue

- © The `continue` statement breaks one iteration (in the loop), if a specified condition occurs, and continues with the next iteration in the loop.

```
for (int i = 0; i < 10; i++) {  
    if (i == 4) {  
        continue;  
    }  
    System.out.println(i);  
}
```

A decorative network diagram in the top-left corner, featuring a complex web of interconnected nodes and lines. Some nodes are highlighted with blue circles, and others with blue dots. The lines are thin and gray, creating a mesh-like structure.

# Anylogic

A decorative network diagram in the bottom-right corner, similar to the one in the top-left. It shows a cluster of nodes connected by lines, with several nodes highlighted in blue.

# Sample

- Calculate factorial

- Calculate  $\binom{n}{k} = \frac{n!}{k! \times (n-k)!}$



# Thanks!

## Any questions?

You can find me at:  
[smrayazi@gmail.com](mailto:smrayazi@gmail.com)