Java Loop

Update til today’s content

For loop

One complete java statement

One conditional statement

One complete java statement
Or a java operation
Run after the loop operation


```
for ( [initial control variable declare]; [looping condition check] ; [condition change]) {

    // do something
}
```

While loop

One conditional statement

```
while ( [changeable condition]) {
        // keep doing something
}
```

Do while loop

```
do {

        // keep doing something

} while ( [changeable condition]);
```

One conditional statement

You can take the initiative and break the loop,
but you need to know what you are doing,
break should be used in side a if condition to be break properly

```
int myNumber = 1024;

while (myNumber > 0) {
    myNumber--;
    if (myNumber == 512) {
        break;
    }
}
System.out.println(myNumber);
```

Prints: 512

Nested for loop example

```
for (int i = 1; i <= 10 ; i++) {
    / / code here will be executed for 10 times (i times)
    for (int j = 1; j <= 5 ; j++)  {
    //code here will be executed for 50 times (i*j times)
    / / and yes, there are no restrict on how may nested for loops here
    }
}
```

For loop tips

- Do not modify the control variable (e.g. i or j) inside the loop body, it is very easy to mess up the loop logic

- Double check the control variable before you start the loop body writing, it is very easy to make a infinite loop

- Write down detailed steps if you are confused with the looping logic.

Java Variable Scope

Java Variable Scope

- Defines the variable accessibility level

- The deeper the variable get created, the less accessible it gets

```
public class MyProgram {
        public static void main(String[] args) {

            int myVariable = 6;

            if (myVariable >= 3) {
                // you can access myVariable here
                int innerVariable = myVariable;
                // you can only access innerVariable here inside if block
            }

             // you CANNOT access innerVariable here!!! EVER!!!!

            for (int i = 1; i <= 10 ; i++) {
                  // you can access i here
                  // you can access myVariable here
                  for (int j = 1; j <= 5 ; j++)  {
                      // you can access i here
                      // you can access j here
                      // you can access myVariable here
                  }
                  // you CANNOT access j here!!!!!
                  // you can access i here
            }

            // you CANNOT access  i  and j  here!!!!!
            // you can access myVariable here

        }  // nothing get access here
}
```