

OOP Using JAVA

06 – Loops

Dr. Mostafa A. El-Hosseini

melhosseini@eng.kfs.edu.eg

www.melhosseini.co.cc

Recap

- » Data Input using Keyin
- » Relational Operators
- » Logical Operators
- » If statements
- » If – else statements
- » Else – if statements
- » Switch statements
- » Examples

AGENDA

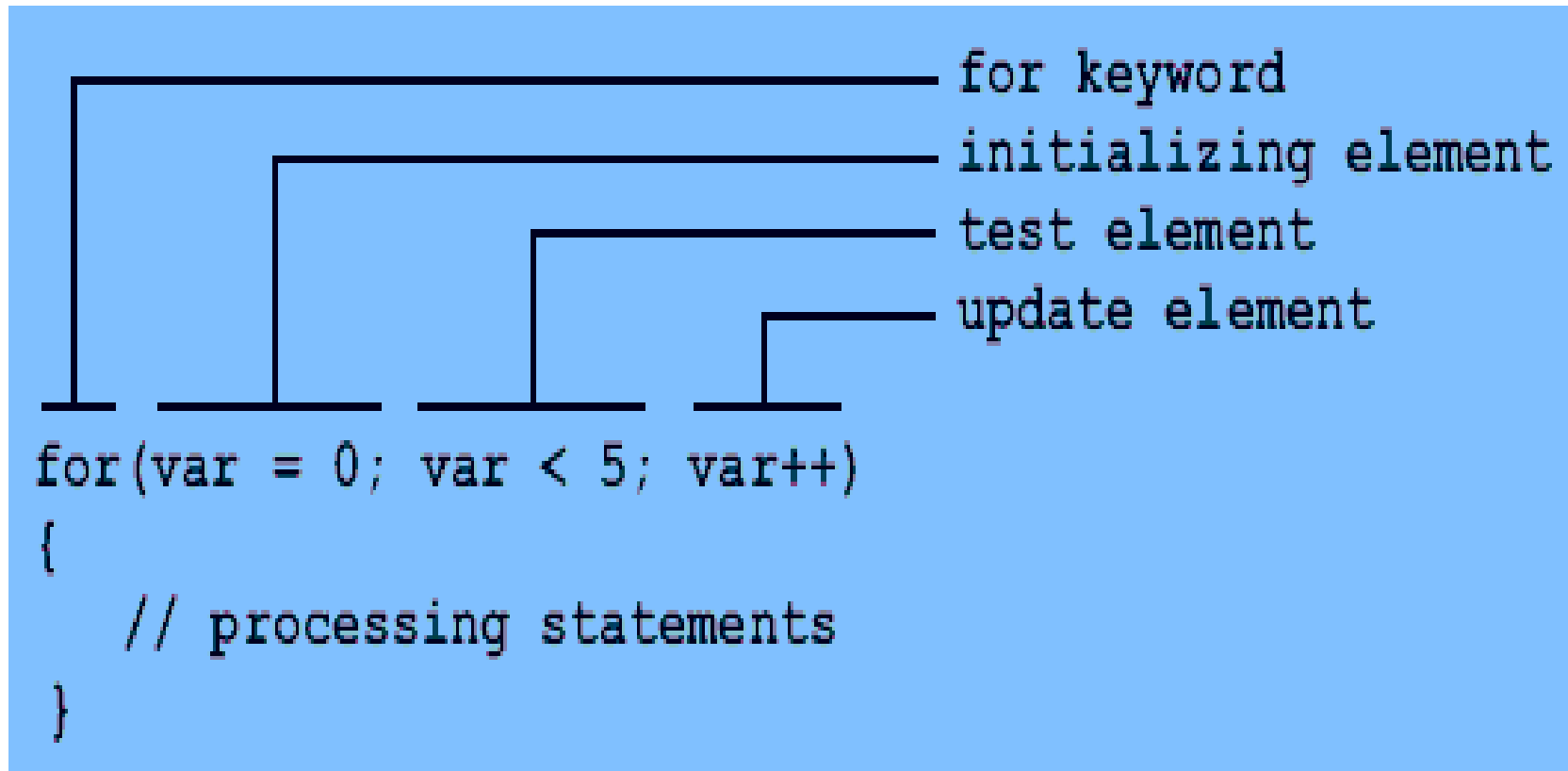
- » For loop
- » While loop
- » Do-while loop
- » Nested loops
- » Examples
- » Break
- » Continue

For loop

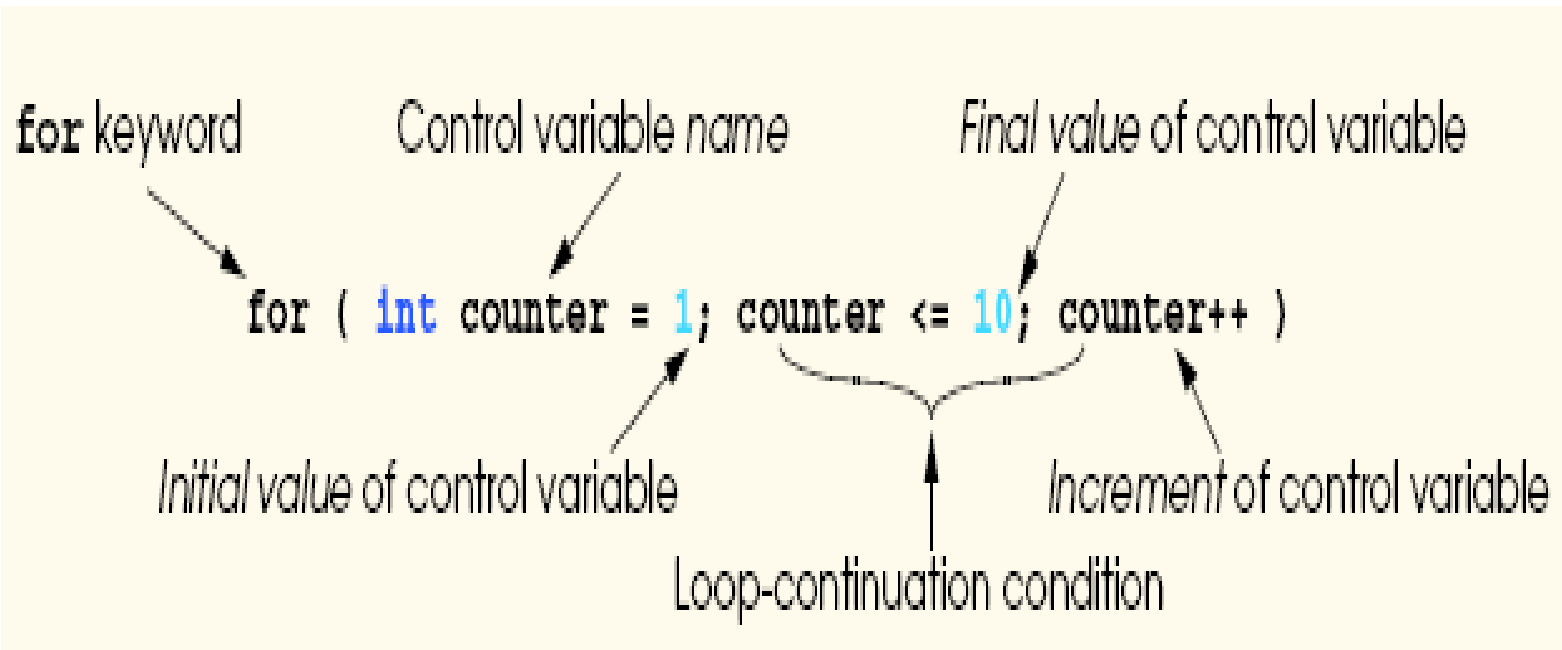
The general format of the **for** structure is

```
for ( expression1; expression2; expression3 )  
    statement
```

For loop



For loop



HINTS

- » **for (int i = 1; i <= 100; i++)**
- » **for (int i = 100; i >= 1; i--)**
- » **for (int i = 7; i <= 77; i += 7)**
- » **for (int i = 20; i >= 2; i -= 2)**
- » **for (int j = 2; j <= 20; j += 3)**
- » **for (int j = 99; j >= 0; j -= 11)**

Common Programming Error

- » When the control variable of a for structure is initially defined in the initialization section of the header of the for structure, using the control variable after the body of the structure is a **syntax error**.

While loop

```
expression1;  
while ( expression2 ) {  
    statement  
    expression3;  
}
```

While loop

```
loopVar = 0; ← External initialization
while(loopVar != 10) ← Loop continuation test
{
    // Processing statements
    loopVar++; ← Loop variable update
}
```

Do-While Loop

- » A characteristic of the **while loop** is that if the **test condition is initially false, the loop never executes**. For example, the while loop in the following code fragment will not execute the statement body because the variable **x** evaluates to 0 before the first iteration

```
int x = 0;
```

```
...
```

```
while (x != 0)
```

```
    System.out.println(x);
```

Do-While

- » In the **do-while** loop, the test expression is evaluated after the loop executes. This ensures that the **loop executes at least once**, as in the following example:

```
int x = 0;
```

```
...
```

```
do
```

```
    System.out.println(x);
```

```
while (x != 0);
```

Do-While

```
loopVar = 0; ← External initialization
do ← Start of loop
{
    // Processing statements
    loopVar++; ← Loop variable update
}
while(loopVar != 10); ← Loop continuation test
```

Factorial example

```
public class myFact
{
    public static void main(String[] args)
    {
        int number,fact,num;
        fact = 1;
        num= Keyin.inInt("Number = ");
        for (int i = 2; i<=num;i++)
            fact *= i;
        System.out.println("Factorial = "+fact);
    }
}
```

Running code

```
D:\>javac myFact.java
```

```
D:\>java myFact
```

```
Number = 3
```

```
Factorial = 6
```

```
D:\>java myFact
```

```
Number = 5
```

```
Factorial = 120
```

```
D:\>
```

Quiz

- Write down a program to calculate the sum of even numbers between 0 and 49.

Ans.

```
public class addeven
{

    public static void main(String[] args)
    {
        int sum = 0;
        for (int i = 0;i<49;i+=2)
            sum +=i;
        System.out.println("sum = "+sum);
    }
}
```

Quiz

- » A class of ten students took a quiz. The grades (integers in the range 0 to 100) for this quiz are available to you. Determine the class average on the quiz. The class average is equal to the sum of the grades divided by the number of students.

Break

- » The **break** statement, when executed in a while, for, do/while or switch structure, **causes immediate exit from that structure.**
- » **Execution continues with the first statement after the structure.** Common uses of the break statement are to escape early from a loop or skip the remainder of a switch structure.

Continue

- » The **continue** statement, when executed in a while, for or do/while structure, **skips** the remaining statements in the loop body and **proceeds** with the **next iteration** of the loop.
- » In while and do/while structures, the program evaluates the loop-continuation test immediately after the continue statement executes.

Homework

- Write down a program to calculate the roots of a **number** of quadratic equations in the form of **$ax^2 + bx + c = 0$**