GeeksforGeeks
A computer science portal for geeks

Custom Search	

Write an Article	Login

For Loop in Java | Important points

Prerequisite: Loops in Java

The structure of basic for loop:

```
for(initialization; boolean expression; update statement)
{
    //Body
}
```

Let's look at some basic examples of using for loop and the common pitfalls in using for loop

1. **Providing expression in for loop is must :** For loop must consist a valid expression in the loop statement failing which can lead to an infinite loop. The statement

```
for ( ; ; )
is similar to
while(true)
```

```java
// Java program to illustrate
// infinite loop
public class Example1
{
    public static void main(String[] args)
    {
        for( ; ; )
        {
            System.out.println("This is an infinite loop");
        }
    }

}
```

Output: This code prints the statement "This is an infinite loop" repeatedly.

2. **Initializing multiple variables :** In Java, multiple variables can be initialized in initialization block of for loop regardless of whether you use it in the loop or not.

```java
// Java program to illustrate
// Initializing multiple variables
// in initialization block
public class Example2
{
```

```java
    public static void main(String[] args)
    {
        int x = 2;
        for(long y = 0, z = 4; x < 10 && y < 10; x++, y++)
        {
            System.out.println(y + " ");
        }

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

Run on IDE

In the above code, there is simple variation in the for loop. Two variables are declared and initialized in the initialization block. The variable 'z' is not being used. Also, the other two components contain extra variable. So, it can be seen that the blocks may include extra variables which may not be referenced by each other.

3. **Redeclaration of a variable in initialization block :** Suppose, an initialization vaiable is already declared as integer. Can we re-declare it in for loop with other data type? No, See the example:

```java
// Java program to illustrate
// redeclaring a variable
// in initialization block
public class Example3
{
    public static void main(String[] args)
    {
        // x is integer
        int x = 0;

        // redeclaring x as long will not work
        for(long y = 0, x = 1; x < 5; x++)
        {
            System.out.print(x + " ");
        }

    }
}
```

Run on IDE

```
Example3.java:12: error: variable x is already defined in method main(String[])
        for(long y = 0, x = 1; x < 5; x++)
```

Here, x was already initialized to zero as integer and is being re-declared in the loop with data type long.

But this problem can be fixed by slightly modifying the code. Here, the variables x and y are declared in a different way.

```java
// Java program to illustrate
// redeclaring a variable
// in initialization block
public class Example3
{
    public static void main(String[] args)
    {
        // x is integer
        int x = 0;
        long y = 10;

        for (y = 0, x = 1; x < 5; x++)
        {
            System.out.print(x + " ");
        }

    }
}
```

Run on IDE

Output:

```
1 2 3 4
```

4. **Variables declared in the initialization block must be of same type :** It is just a common sense that when we declare a variable as

```
int x, y;
```

both variables are of same type. Its just the same in for loop initialization block too.

```java
// Java program to illustrate
// declaring a variable
// in initialization block
public class Example4
{
    public static void main(String[] args)
    {
        // This will cause error;
        // int x;

        // redeclaring x as long will not work
        for (long y = 0, x = 1; x < 5; x++)
        {
            System.out.print(x + " ");
        }

    }
}
```

Run on IDE

5. **Variables in the loop are accessible only within:** The variables that are declared in the initialization block can be accessed only within the loop. For more on scope of variables, Refer here

```java
// Java program to illustrate
// scope of Initializing variables
// within the loop
public class Example5
{
    public static void main(String[] args)
    {
        // x and y scope is only
        // within for loop
        for(int x = 0, y = 0; x < 3 && y < 3; x++, y++)
        {
            System.out.println(y + " ");
        }

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

Run on IDE

Error

```
Example5.java:13: error: cannot find symbol
        System.out.println(x);
```

In the above example, variable x is not accessible outside the loop. The statement which is commented gives compiler error.

This article is contributed by **Preeti Pardeshi**. If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Java   Loops & Control Structure   Login to Improve this Article

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

Recommended Posts:

Decision Making in Java (if, if-else, switch, break, continue, jump)

For-each loop in Java

Switch Statement in Java

String in Switch Case in Java

Do we need forward declarations in Java?

Java Math copySign() method with Examples

Java | Converting an Image into Grayscale using cvtColor()

Java | Removing whitespaces using Regex

Java System.nanoTime() vs System.currentTimeMillis

Java toDegrees() method with Example

(Login to Rate)

1   Average Difficulty : **1/5.0**
Based on **12** vote(s)

| Basic | Easy | Medium | Hard | Expert |

☐ Add to TODO List

☐ Mark as DONE

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

| Load Comments | | Share this post! |

A computer science portal for geeks

710-B, Advant Navis Business Park,
Sector-142, Noida, Uttar Pradesh - 201305
feedback@geeksforgeeks.org

COMPANY

About Us

Careers

Privacy Policy

Contact Us

LEARN

Algorithms

Data Structures

Languages

CS Subjects

Video Tutorials

PRACTICE

Company-wise

Topic-wise

Contests

Subjective Questions

CONTRIBUTE

Write an Article

GBlog

Videos