



Java Scope

[< Previous](#)[Next >](#)

Java Scope

In Java, variables are only accessible inside the region they are created. This is called **scope**.

Method Scope

Variables declared directly inside a method are available anywhere in the method following the line of code in which they were declared:

Example

```
public class Main {  
    public static void main(String[] args) {  
  
        // Code here CANNOT use x  
  
        int x = 100;  
  
        // Code here can use x  
        System.out.println(x);  
    }  
}
```

☐ Dark mode

[Try it Yourself »](#)

Block Scope

A block of code refers to all of the code between curly braces `{}`.

Variables declared inside blocks of code are only accessible by the code between the curly braces, which follows the line in which the variable was declared:

Example

```
public class Main {  
    public static void main(String[] args) {  
  
        // Code here CANNOT use x  
  
        { // This is a block  
  
            // Code here CANNOT use x  
  
            int x = 100;  
  
            // Code here CAN use x  
            System.out.println(x);  
  
        } // The block ends here  
  
        // Code here CANNOT use x  
  
    }  
}
```

[Try it Yourself »](#)☐ Dark mode

A block of code may exist on its own or it can belong to an `if`, `while` or `for` statement. In the case of `for` statements, variables declared in the statement itself are also available inside the block's scope.

[< Previous](#)[Next >](#)

ADVERTISEMENT