Java Scope

Topics : JAVA
Written on April 07, 2023

In Java, scope refers to the area of the program where a variable, method, or object can be accessed. There are four types of scope in Java:

1. Static scope: Variables and methods declared as static are accessible throughout the class.

   Example :
   ```
   public class MyClass {
     static int x = 10; // x has static scope
     public void method() {
       // do something with x
     }
   }
   ```

2. Instance scope: Instance variables and methods are accessible throughout the class and can be accessed by any method in the class.

   Example :

   ```
   public class MyClass {
     int x = 10; // x has instance scope
     public void method() {
       // do something with x
     }
   }
   ```

3. Local scope: Variables declared inside a method or block are only accessible within that method or block

   Example :

   ```
   public void method() {
     int x = 10; // x has local scope
     // do something with x
   }
   ```

4. Block scope: Variables declared inside a loop or if statement are only accessible within that block.

Example :

```
public void method() {
  {
    int x = 10; // x has block scope
    // do something with x
  }
  // x is not accessible here
}
```

It's important to keep in mind that variables with the same name can have different scopes. For example, you can have a class-level variable and a local variable with the same name, but they refer to different things and have different scopes.

Java also has the concept of variable shadowing, which occurs when a variable declared within a certain scope has the same name as a variable declared in an outer scope. In this case, the inner variable "shadows" the outer variable, and any reference to that variable within the inner scope refers to the inner variable.