final keyword in Java

final is a non-access modifier for Java elements. The final modifier is used for finalizing the implementations of classes, methods, and variables.

Final Variables

A final variable can be explicitly initialized only once. A reference variable declared final can never be reassigned to refer to a different object.

However, the data within the object can be changed. So, the state of the object can be changed but not the reference.

With variables, the final modifier often is used with static to make the constant a class variable.

Example

```java
public class Test {
   final int value = 10;
   // The following are examples of declaring constants:
   public static final int BOXWIDTH = 6;
   static final String TITLE = "Manager";
   public void changeValue() {
      value = 12; // will give an error
   }
}
```

Final Methods

A final method cannot be overridden by any subclasses. As mentioned previously, the final modifier prevents a method from being modified in a subclass.

The main intention of making a method final would be that the content of the method should not be changed by any outsider.

Example

You declare methods using the final modifier in the class declaration, as in the following

```java
public class Test {
   public final void changeName() {
      // body of method
   }
}
```

Final Classes

The main purpose of using a class being declared as final is to prevent the class from being subclassed. If a class is marked as final then no class can inherit any feature from the final class.

Example

```
public final class Test {
   // body of class
}
```