

Java Keyword Examples

Created by BaskarTech Academy

Date: April 21, 2025

1. public - Access Modifier

Explanation:

The public keyword is an access modifier used to specify that the method or class is accessible from other classes.

If a method or class is marked public, it means you can call or use it from any other class.

Example Program:

```
public class PublicExample {  
    public void displayMessage() {  
        System.out.println("Hello, this is a public method!");  
    }  
  
    public static void main(String[] args) {  
        PublicExample example = new PublicExample();  
        example.displayMessage();  
    }  
}
```

Output:

Hello, this is a public method!

2. static - Belongs to the Class, Not the Object

Explanation:

The static keyword means that the method or variable belongs to the class itself, rather than an instance of the class.

You can call a static method or access a static variable without creating an object of the class.

Example Program:

```
public class StaticExample {  
    static void showMessage() {  
        System.out.println("This is a static method!");  
    }  
  
    public static void main(String[] args) {  
        showMessage();  
    }  
}
```

Output:

This is a static method!

3. void - No Return Value

Explanation:

The void keyword is used in a method declaration to indicate that the method does not return any value.

If a method has void as its return type, it simply performs an operation and doesn't give anything back.

Example Program:

```
public class VoidExample {  
  
    void printMessage() {  
  
        System.out.println("This method has no return value.");  
  
    }  
  
    public static void main(String[] args) {  
  
        VoidExample example = new VoidExample();  
  
        example.printMessage();  
  
    }  
}
```

Output:

This method has no return value.

4. String[] args - Command-Line Arguments

Explanation:

The String[] args parameter in the main method is used to receive command-line arguments.

These arguments are passed when the program is executed from the command line or terminal.

Example Program:

```
public class ArgsExample {  
  
    public static void main(String[] args) {
```

```
if (args.length > 0) {  
    System.out.println("Command-line arguments received:");  
    for (String arg : args) {  
        System.out.println(arg);  
    }  
} else {  
    System.out.println("No command-line arguments received.");  
}  
}  
}
```

How to Run:

Save the program as ArgsExample.java.

Compile: javac ArgsExample.java

Run: java ArgsExample Hello World

Output:

Command-line arguments received:

Hello

World