

In Java programming, tokens are the smallest individual units of a program that are meaningful to the compiler.

They represent the various building blocks of the language, such as keywords, identifiers, literals, operators, separators, and comments.

Here are the common types of tokens in Java:

1. Keywords:

Keywords are reserved words that have predefined meanings in Java and cannot be used as identifiers. Examples include `public`, `class`, `if`, `while`, `int`, `void`, `static`, and so on.

2. Identifiers:

Identifiers are names used to identify classes, variables, methods, and other program elements. They must follow certain rules, such as starting with a letter, underscore, or dollar sign, and can consist of letters, digits, underscores, or dollar signs. Examples of identifiers are `myVariable`, `calculateSum`, `Employee`, and `MAX_SIZE`.

3. Literals:

Literals represent fixed values in a program. They can be of different types, including:

- **Numeric Literals:** These represent numbers and can be integers, floating-point numbers, or hexadecimal numbers. Examples include `5`, `3.14`, `10L`, `0b1010`, and `0xFF`.
- **Character Literals:** These represent individual characters enclosed in single quotes. Examples include `'A'`, `'5'`, and `'\n'`.

- String Literals: These represent sequences of characters enclosed in double quotes. Examples include "Hello", "Java", and "123".

4. Operators:

Operators are symbols that perform operations on operands. They can be arithmetic operators (+, -, *, /, %), assignment operators (=, +=, -=), comparison operators (==, !=, <, >, <=, >=), logical operators (&&, ||, !), and more.

5. Separators:

Separators are symbols used to separate or group elements in a program. Examples include parentheses (and), braces { and }, square brackets [and], comma ,, semicolon ;, period ., and colon :.

6. Comments:

Comments are used to add explanatory notes to the code and are ignored by the compiler. There are two types of comments in Java:

- Single-line comments: These start with // and extend until the end of the line.
- Multi-line comments: These start with /* and end with */ and can span multiple lines.

Practice problems on Java tokens:

Q1. Find number of token in the program ?

Java 

```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello, World!");  
    }  
}
```

A. Token Count: 13 tokens

Explanation:

- public, class, static, and void are keywords (4 tokens).
- HelloWorld, main, and String are identifiers (3 tokens).
- [], (), and {} are separators (4 tokens).
- "Hello, World!" is a string literal (1 token).
- The statement ends with a semicolon (;) (1 token).

Q2. Find number of token in the program ?

Java 

```
public class Calculator {  
    public static void main(String[] args) {  
        int a = 5;  
        int b = 10;  
    }  
}
```

```
int sum = a + b;  
System.out.println("The sum is: " + sum);  
}  
}
```

Token Count: 25 tokens

Explanation:

- public, class, static, void, int, and System.out.println are keywords (6 tokens).
- Calculator, main, String, a, b, and sum are identifiers (6 tokens).
- =, +, ,, (, {, }, and "" are separators (7 tokens).
- 5 and 10 are integer literals (2 tokens).
- "The sum is: " is a string literal (1 token).

Q3. Find number of token in the program ?

Java

```
public class LoopExample {  
    public static void main(String[] args) {  
        int n = 5;  
        for (int i = 1; i <= n; i++) {  
            System.out.println("Current number: " + i);  
        }  
    }  
}
```

Token Count: 30 tokens

Explanation:

- public, class, static, void, int, for, and System.out.println are keywords (7 tokens).
- LoopExample, main, String, n, and i are identifiers (5 tokens).
- =, ,, (), {}, <=, <, ++, and "" are separators (8 tokens).
- 5 is an integer literal (1 token).
- "Current number: " is a string literal (1 token).

Q4. Find number of token in the program ?

Java

```
public class NumberChecker {  
    public static void main(String[] args) {  
        int number = 15;  
        if (number % 2 == 0) {  
            System.out.println("The number is even.");  
        } else {  
            System.out.println("The number is odd.");  
        }  
    }  
}
```

Token Count: 26 tokens

Explanation:

- public, class, static, void, int, if, else, and System.out.println are keywords (8 tokens).
- NumberChecker, main, String, and number are identifiers (4 tokens).
- =, ,, (), {}, %, ==, "", and . are separators (8 tokens).

- 15 is an integer literal (1 token).
- "The number is even." and "The number is odd." are string literals (2 tokens).

Related Posts:

1. Can Java have same name variable
2. Types of variables in Java programming
3. JAVA and its Support Systems
4. JAVA environment
5. JAVA program structure
6. Java statements
7. Java virtual machine
8. C++ Versus JAVA
9. Constants and Variables in Java
10. Data types JAVA
11. Defining a class
12. Constructor in JAVA
13. Array in Java
14. Applet
15. Applets Vs Applications
16. Writing applets
17. Applets life cycle
18. Creating an Executable Applet
19. Graphics in Applet
20. Applet image display
21. Applet digital clock
22. Applet mouse event handling
23. JDBC

24. Execute an SQL Statement
25. Process the result
26. CLOSE THE DATABASE CONNECTION
27. File handling
28. Define a class to declare an integer array of size n and accept the elements into the array.
29. Define a class to declare an array of size 20 of the double datatype, accept the elements into the array and perform the following: Calculate and print the sum of all the elements.
30. Java program for String, to uppercase, to equal, length of string
31. Write a Java program for Bubble sort.
32. Write a Java program String to uppercase and count words starting with 'A'
33. How to set path in Java
34. Understanding public static void main (String args[]){ } in Java
35. Difference between static and non static methods in Java