

Java strings

Strings

In Java, a string is a sequence of characters. For example, "hello" is a string containing a sequence of characters 'h', 'e', 'l', 'l', and 'o'. We use double quotes to represent a string in Java.

For example,

Syntax: datatype variable_name="value";

Example: String a="Skyline";

Programming example:

Public class JavaStrings

```
{  
    public static void main(String args[])  
    {  
        //creating a string  
        String a="Hello Java";  
        String b="Programming";  
  
        System.out.println(a); // printing a string  
        System.out.println(b); // printing a string  
    }  
}
```

Java String Operations

Java String provides various methods to perform different operations on strings. We will look into some of the commonly used string operations.

1. Get length of a string

To find the length of a string, we use the `length()` method of the String. For example,

```
public class GetLength {  
    public static void main(String[] args) {  
  
        // create a string  
        String greet = "Hello! World";  
        System.out.println("String: " + greet);  
  
        // get the length of greet  
        int length = greet.length();  
        System.out.println("Length: " + length);  
    }  
}
```

2. Join two strings

We can join two strings in Java using the `concat()` method. For example,

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

        // create first string
        String first = "Java ";
        System.out.println("First String: " + first);

        // create second
        String second = "Programming";
        System.out.println("Second String: " + second);

        // join two strings
        String joinedString = first.concat(second);
        System.out.println("Joined String: " + joinedString);
    }
}
```

3. Compare two Strings

In Java, we can make comparisons between two strings using the `equals()` method. For example,

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

        // create 3 strings
        String first = "java programming";
        String second = "java programming";
        String third = "python programming";

        // compare first and second strings
        boolean result1 = first.equals(second);
        System.out.println("Strings first and second are equal: " + result1);

        // compare first and third strings
        boolean result2 = first.equals(third);
        System.out.println("Strings first and third are equal: " + result2);
    }
}
```

Create Java Strings using the new keyword

Since strings in Java are objects, we can create strings using the `new` keyword as well. For example,

```
public class NewStrings {  
    public static void main(String[] args) {  
  
        // create a string using new  
        String name = new String("Java String");  
  
        System.out.println(name); // print Java String  
    }  
}
```