# Module 3 String Handling

In java, every string that we create is actually an object of type **String**. One important thing to notice about string object is that string objects are **immutable** that means once a string object is created it cannot be changed.

In Java, string is basically an object that represents sequence of char values. An arrayof characters works same as Java string. For example:

# How to create a string object?

There are two ways to create String object:

1. By string literal
2. By new keyword

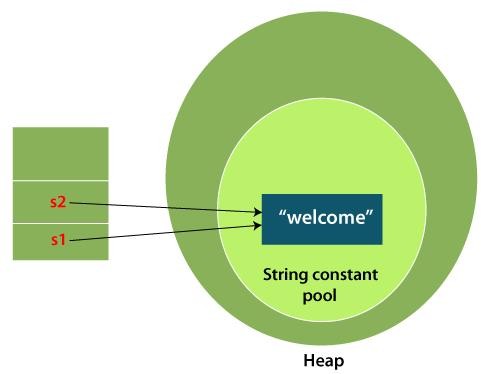
# String Literal

Java String literal is created by using double quotes. For Example:

1. String s="welcome";

Each time you create a string literal, the JVM checks the "string constant pool" first. If the string already exists in the pool, a reference to the pooled instance is returned. If the string doesn't exist in the pool, a new string instance is created and placed in the pool. For example:

1. String s1="Welcome";
2. String s2="Welcome";//It doesn't create a new instance



In the above example, only one object will be created. Firstly, JVM will not find any string object with the value "Welcome" in string constant pool that is why it will create a new object. After that it will find the string with the value "Welcome" in the pool, it will not create a new object but will return the reference to the same instance.

# By new keyword

1. String s=**new** String("Welcome");//creates two objects and one reference variable In such case, JVM

will create a new string object in normal (non-pool) heap memory, and the literal "Welcome" will be placed in the string constant pool. The variable s will refer to the object in a heap (non-pool).

# Java String Example StringExample.java

**public class** StringExample{

**public static void** main(String args[]){

String s1="java";//creating string by Java string literal



Only two pages were converted.

Please **Sign Up** to convert the full document.

[**www.freepdfconvert.com/membership**](https://www.freepdfconvert.com/membership)