Java String Interview Questions and Answers

APRIL 3, 2018 BY PANKAJ — 167 COMMENTS

String is one of the most widely used Java Class. Here I am listing some important **Java String Interview Questions and Answers**.

This will be very helpful to get complete knowledge of String and tackle any questions asked related to String in interview.

> Quizzes are fun, aren't they! I recently published java String quiz of 21 questions. It has been taken by thousands of java enthusiasts with an average score of 42.55%. You should take that and try to beat the average score and get your name into leaderboard.
>
> Here is the link that opens in a new tab: Java String Quiz

Java String Interview Questions

1. What is String in Java? String is a data type?
2. What are different ways to create String Object?
3. Write a method to check if input String is Palindrome?

What is String in Java? String is a data type?

String is a Class in java and defined in java.lang package. It's not a primitive data type like int and long. String class represents character Strings. String is used in almost all the Java applications and there are some interesting facts we should know about String. String in immutable and final in Java and JVM uses String Pool to store all the String objects.
Some other interesting things about String is the way we can instantiate a String object using double quotes and overloading of "+" operator for concatenation.

What are different ways to create String Object?

We can create String object using `new` operator like any normal java class or we can use double quotes to create a String object. There are several constructors available in String class to get String from char array, byte array, StringBuffer and StringBuilder.

```
String str = new String("abc");
String str1 = "abc";
```

When we create a String using double quotes, JVM looks in the String pool to find if any other String is stored with same value. If found, it just returns the reference to that String object else it creates a new String object with given value and stores it in the String pool.
When we use new operator, JVM creates the String object but don't store it into the String Pool. We can use `intern()` method to store the String object into String pool or return the reference if there is already a String with equal value present in the pool.

Write a method to check if input String is Palindrome?

A String is said to be Palindrome if it's value is same when reversed. For example "aba" is a Palindrome String.
String class doesn't provide any method to reverse the String but `StringBuffer` and `StringBuilder` class has reverse method that we can use to check if String is palindrome or not.

```
private static boolean isPalindrome(String str) {
    if (str == null)
        return false;
    StringBuilder strBuilder = new StringBuilder(str);
    strBuilder.reverse();
    return strBuilder.toString().equals(str);
}
```

Sometimes interviewer asks not to use any other class to check this, in that case we can compare characters in the String from both ends to find out if it's palindrome or not.


```
private static boolean isPalindromeString(String str) {
    if (str == null)
        return false;
    int length = str.length();
    System.out.println(length / 2);
    for (int i = 0; i < length / 2; i++) {

        if (str.charAt(i) != str.charAt(length - i - 1))
            return false;
    }
    return true;
}
```

Write a method that will remove given character from the String?

We can use `replaceAll` method to replace all the occurance of a String with another String. The important point to note is that it accepts String as argument, so we will use `Character` class to create String and use it to replace all the characters with empty String.

```
private static String removeChar(String str, char c) {
    if (str == null)
        return null;
    return str.replaceAll(Character.toString(c), "");
}
```

How can we make String upper case or lower case?

We can use String class `toUpperCase` and `toLowerCase` methods to get the String in all upper case or lower case. These methods have a variant that accepts Locale argument and use that locale rules to convert String to upper or lower case.

What is String subSequence method?

Java 1.4 introduced CharSequence interface and String implements this interface, this is the only reason for the implementation of subSequence method in String class. Internally it invokes the String substring method.
Check this post for String subSequence example.

How to compare two Strings in java program?

Java String implements `Comparable` interface and it has two variants of `compareTo()` methods.

`compareTo(String anotherString)` method compares the String object with the String argument passed lexicographically. If String object precedes the argument passed, it returns negative integer and if String object follows the argument String passed, it returns positive integer. It returns zero when both the String have same value, in this case `equals(String str)` method will also return true.

compareToIgnoreCase(String str): This method is similar to the first one, except that it ignores the case. It uses String CASE_INSENSITIVE_ORDER Comparator for case insensitive comparison. If the value is zero then `equalsIgnoreCase(String str)` will also return true.
Check this post for String compareTo example.

How to convert String to char and vice versa?

This is a tricky question because String is a sequence of characters, so we can't convert it to a single character. We can use use `charAt` method to get the character at given index or we can use `toCharArray()` method to convert String to character array.
Check this post for sample program on converting String to character array to String.

How to convert String to byte array and vice versa?

We can use String `getBytes()` method to convert String to byte array and we can use String constructor `new String(byte[] arr)` to convert byte array to String.

Check this post for String to byte array example.

Can we use String in switch case?

This is a tricky question used to check your knowledge of current Java developments. Java 7 extended the capability of switch case to use Strings also, earlier java versions doesn't support this.
If you are implementing conditional flow for Strings, you can use if-else conditions and you can use switch case if you are using Java 7 or higher versions.

Check this post for Java Switch Case String example.

Write a program to print all permutations of String?

This is a tricky question and we need to use recursion to find all the permutations of a String, for example "AAB" permutations will be "AAB", "ABA" and "BAA".
We also need to use Set to make sure there are no duplicate values.
Check this post for complete program to find all permutations of String.

Write a function to find out longest palindrome in a given string?

A String can contain palindrome strings in it and to find longest palindrome in given String is a programming question.
Check this post for complete program to find longest palindrome in a String.

Difference between String, StringBuffer and StringBuilder?

String is immutable and final in java, so whenever we do String manipulation, it creates a new String. String manipulations are resource consuming, so java provides two utility classes for String manipulations – StringBuffer and StringBuilder.
StringBuffer and StringBuilder are mutable classes. StringBuffer operations are thread-safe and synchronized where StringBuilder operations are not thread-safe. So when multiple threads are working on same String, we should use StringBuffer but in single threaded environment we should use StringBuilder. StringBuilder performance is fast than StringBuffer because of no overhead of synchronization.

Check this post for extensive details about String vs StringBuffer vs StringBuilder.
Read this post for benchmarking of StringBuffer vs StringBuilder.

Why String is immutable or final in Java

There are several benefits of String because it's immutable and final.

- String Pool is possible because String is immutable in java.
- It increases security because any hacker can't change its value and it's used for storing sensitive information such as database username, password etc.
- Since String is immutable, it's safe to use in multi-threading and we don't need any synchronization.

- Strings are used in java classloader and immutability provides security that correct class is getting loaded by Classloader.

Check this post to get more details why String is immutable in java.

How to Split String in java?

We can use `split(String regex)` to split the String into String array based on the provided regular expression.
Learn more at java String split.

Why Char array is preferred over String for storing password?

String is immutable in java and stored in String pool. Once it's created it stays in the pool until unless garbage collected, so even though we are done with password it's available in memory for longer duration and there is no way to avoid it. It's a security risk because anyone having access to memory dump can find the password as clear text.
If we use char array to store password, we can set it to blank once we are done with it. So we can control for how long it's available in memory that avoids the security threat with String.

How do you check if two Strings are equal in Java?

There are two ways to check if two Strings are equal or not – using "==" operator or using `equals` method. When we use "==" operator, it checks for value of String as well as reference but in our programming, most of the time we are checking equality of String for value only. So we should use equals method to check if two Strings are equal or not.
There is another function `equalsIgnoreCase` that we can use to ignore case.

```
String s1 = "abc";
String s2 = "abc";
String s3= new String("abc");
System.out.println("s1 == s2 ? "+(s1==s2)); //true
System.out.println("s1 == s3 ? "+(s1==s3)); //false
System.out.println("s1 equals s3 ? "+(s1.equals(s3))); //true
```

What is String Pool?

As the name suggests, String Pool is a pool of Strings stored in Java heap memory. We know that String is special class in java and we can create String object using new operator as well as providing values in double quotes.
Check this post for more details about String Pool.

What does String intern() method do?

When the intern method is invoked, if the pool already contains a string equal to this String object as determined by the equals(Object) method, then the string from the pool is returned. Otherwise, this String object is added to the pool and a reference to this String object is returned.
This method always return a String that has the same contents as this string, but is guaranteed to be from a pool of unique strings.

Does String is thread-safe in Java?

Strings are immutable, so we can't change it's value in program. Hence it's thread-safe and can be safely used in multi-threaded environment.
Check this post for Thread Safety in Java.

Why String is popular HashMap key in Java?

Since String is immutable, its hashcode is cached at the time of creation and it doesn't need to be calculated again. This makes it a great candidate for key in a Map and it's processing is fast than other HashMap key objects. This is why String is mostly used Object as HashMap keys.

String Programming Questions

1. What is the output of below program?

```
package com.journaldev.strings;

public class StringTest {

	public static void main(String[] args) {
		String s1 = new String("pankaj");
		String s2 = new String("PANKAJ");
		System.out.println(s1 = s2);
	}

}
```

   It's a simple yet tricky program, it will print "PANKAJ" because we are assigning s2 String to s1. Don't get confused with == comparison operator.

2. What is the output of below program?

```
package com.journaldev.strings;

public class Test {

	 public void foo(String s) {
```

```
            System.out.println("String");
            }

            public void foo(StringBuffer sb){
            System.out.println("StringBuffer");
            }

            public static void main(String[] args) {
                   new Test().foo(null);
           }

    }
```

The above program will not compile with error as "The method foo(String) is ambiguous for the type Test". For complete clarification read Understanding the method X is ambiguous for the type Y error.

3. What is the output of below code snippet?

```
String s1 = new String("abc");
String s2 = new String("abc");
System.out.println(s1 == s2);
```

It will print **false** because we are using *new* operator to create String, so it will be created in the heap memory and both s1, s2 will have different reference. If we create them using double quotes, then they will be part of string pool and it will print true.

4. What will be output of below code snippet?

```
String s1 = "abc";
StringBuffer s2 = new StringBuffer(s1);
System.out.println(s1.equals(s2));
```

It will print false because s2 is not of type String. If you will look at the equals method implementation in the String class, you will find a check using **instanceof** operator to check if the type of passed object is String? If not, then return false.

5. What will be output of below program?

```
String s1 = "abc";
String s2 = new String("abc");
s2.intern();
System.out.println(s1 ==s2);
```

It's a tricky question and output will be **false**. We know that intern() method will return the String object reference from the string pool, but since we didn't assigned it back to s2, there is no change in s2 and hence both s1 and s2 are having different reference. If we change the code in line 3 to `s2 = s2.intern();` then output will be true.

6. How many String objects got created in below code snippet?

```
String s1 = new String("Hello");
String s2 = new String("Hello");
```

Answer is 3.
First – line 1, "Hello" object in the string pool.
Second – line 1, new String with value "Hello" in the heap memory.
Third – line 2, new String with value "Hello" in the heap memory. Here "Hello" string from string pool is reused.

I hope that the questions listed here will help you in java interviews, please let me know if I have missed anything.

Make sure to check out Java Programming Questions

« PREVIOUS
How to generate XSD from Java Class

NEXT »
Composition in Java Example

About Pankaj

If you have come this far, it means that you liked what you are reading. Why not reach little more and connect with me directly on **Google Plus**, **Facebook** or **Twitter**. I would love to hear your thoughts and opinions on my articles directly.

Recently I started creating video tutorials too, so do check out my videos on **Youtube**.

FILED UNDER: INTERVIEW QUESTIONS, JAVA

Comments

Fake says

JULY 10, 2018 AT 11:47 PM

'As the name suggests, String Pool is a pool of Strings stored in Java heap memory. ', is it correct on the following page?

https://www.journaldev.com/1321/java-string-interview-questions-and-answers

Reply

Vedant Kumar says

APRIL 17, 2018 AT 8:47 PM

Question 6 answer will be 2 instead of 3. We can test it whether new String("Hello") creates Hello object in string pool or not.

String s = new String("Hello"); //only creates in Heap Memory not in String Pool

System.out.println(s=="Hello"); //prints False

Reply

Pankaj says

APRIL 17, 2018 AT 9:36 PM

The argument in String constructor is first created in Pool, you missed to count this one.

Reply

pintu says
MARCH 12, 2018 AT 12:18 PM
How many objects is created in following code?
String a1=”hello”;
String a2=”hello”;
String a3=new String(“hello”);
Reply

Pankaj says
MARCH 13, 2018 AT 12:14 AM
Two, first a1 in string pool, then a3 in heap.
Reply

pintu says
MARCH 13, 2018 AT 9:22 PM
thanks ,
Reply

SAI SUNDEEP says
JUNE 8, 2018 AT 8:49 AM
In Heap Area 1 Object will be created ie.,
a3=hello; will be created and
1 object is created in String Constant Pool And Referenced by a1 & a2.
Reply

Packiaraj Thusianthan says
FEBRUARY 3, 2018 AT 6:47 AM
It feels good after go thru these Q&A. Thanks bro.
Reply

gundamaiah says
JANUARY 4, 2018 AT 2:44 AM

String s1=new String("abc").intern();
String s2="abc";
How many objects will be created here ?
Reply

Nas says
JANUARY 10, 2018 AT 12:46 AM
2 objects
Reply

Pravesh Kharb says
MARCH 7, 2018 AT 4:47 AM
only one object will be created here in this case.
As here if you check s1==s2, it will return true, as only one object is there
Reply

palvi says
MARCH 22, 2018 AT 3:16 PM
1,
When the intern method is invoked, if the pool already contains a string equal to this String object as determined by the equals(Object) method, then the string from the pool is returned. Otherwise, this String object is added to the pool and a reference to this String object is returned.
Reply

Sandeep Patil says
MARCH 23, 2018 AT 11:35 PM
2 objects because S2 is created on second line thus s1 wont find any thing to equal thus create an object in heap. intern returns value of string if found in pool. s2 will b created in the pool whereas s1 ll b in heap memory
Reply

Dotan Mal says
MAY 7, 2018 AT 6:21 AM
2 objects.
first is argument in String constructor "abc" will be created in string pool – new String("abc")

second will be created in the Heap by the 'new' operator
the .intern() method invokes after this 2 objects created, then checks in string pool for "abc" (which is there) and return a reference to it
– in this case we have no reference to the object created in the heap by the new operator
– second line doesn't change number of objects created
Reply

bhupendra patidar says
NOVEMBER 29, 2017 AT 9:08 AM
Awesome collection of question keeps it Up Good work.
Reply

Anurag Singh says
JULY 21, 2017 AT 3:55 AM
for interview point of view String is a most important topic.
thank you to share this knowlge.
Reply

Mayank Tyagi says
JUNE 30, 2017 AT 9:35 AM
Hi Pankaj,
I read all the comments and got more confused. Can you please explain the pointers below.
In "What are different ways to create String Object?" you answered:
When we use new operator, JVM creates the String object but don't store it into the String Pool. We can use intern() method to store the String object into String pool or return the reference if there is already a String with equal value present in the pool.
and then in last 6th question "How many String objects got created in below code snippet?" you answered.
String s1 = new String("Hello");
String s2 = new String("Hello");
Answer is 3.
First – line 1, "Hello" object in the string pool.
Second – line 1, new String with value "Hello" in the heap memory.
Third – line 2, new String with value "Hello" in the heap memory. Here "Hello" string from string pool is reused.
So will it store the object in String pool or not. I hope this will clear things to all

Reply

waqas_kamran says
FEBRUARY 15, 2018 AT 11:04 PM
it will store or create new object in heap memory but literal “hello” will be in string pool .
Reply

Ajeet Kumar says
MARCH 5, 2018 AT 10:55 AM
Yes, Line 1 :- Checks if “Hello” is in Pool , if Not Then Creates in Pool. else nothing.(count = 1)
Then it creates String in Heap which will be referenced by s1. (count = 2)
Line 2 :- Checks if “Hello” is in Pool , if Not Then Creates in Pool. but it is already there so nothing to do.
Then it creates String in Heap which will be referenced by s2. (count = 3)
Reply

bharat jha says
JUNE 22, 2018 AT 10:33 PM
when we create string object with the new operator , at that time “HELLO” first get in string pool, if it is already in string pool then it will not go in pool, after that the new line will execute and create one object in heap.
Reply

Sever says
JUNE 10, 2017 AT 2:28 PM
Im totally confused.
in one article you write
“When we use new operator, JVM creates the String object but don’t store it into the String Pool. ”
in another
“String str = new String(“Cat”);
In above statement, either 1 or 2 string will be created. If there is already a string literal “Cat” in the pool, then only one string “str” will be created in the pool. If there is no string literal “Cat” in the pool, then it will be first created in the pool and then in the heap space, so total 2 string objects will be created.”
So which one is right ?
Reply

Pankaj says
JUNE 10, 2017 AT 10:21 PM
Both are correct. See when we use new operator with String, we also use String literal. So depending on whether the string literal is present in the string pool or not, either 1 or 2 string object will be created.
First statement just specify that String created will not be stored in the pool, it doesn't say how many string objects will be created. Both the statements are correct in their own area.
Reply

Sidagouda Patil says
MAY 23, 2017 AT 6:22 AM
When we create a string object by using new keyword.. 2 objects created one in heap.. and another in SCP.
Please confirm and reply back thank you.
Reply

jayashree says
APRIL 16, 2017 AT 9:19 PM
doesn't your question 3 and 6 have ambiguity??
String s =new ("Hello"); object is created and also stored in string pool as per q6.
String s2 =new ("Hello"); object is created and reference of s is passed .
when checked with if (s==s2) ,shouldn't it return true? but your q 3 says it as false. 2 different objects are created in heap. please clarify
Reply

Pankaj says
APRIL 17, 2017 AT 1:58 AM
First statement, 2 objects will be created, one in pool and then next one in heap.
Second statement, since "Hello" is already part of pool only one object will be created in heap.
Total 3 objects. I hope it's clear now.
Reply

Tirupathi Reddy says
NOVEMBER 9, 2017 AT 6:40 AM
Hi ,
According to your answer just think if it will create two objects one in heap and one in pool then

what is the use of intern() method in String.

Reply

Ajeet Kumar says

MARCH 5, 2018 AT 11:26 AM

String s1 = "Rakesh";
String s2 = "Rakesh"; //s1 and s2 referencing to same object in Pool
String s3 = "Rakesh".intern(); //s1 , s2 and s3 referencing to same object in Pool
String s4 = new String("Rakesh"); //s4 creates in Heap and referencing to same in Heap
String s5 = new String("Rakesh").intern(); //again s5 referencing to Literal in Pool so s1, s2, s3 and s5 are referncing to same object in Pool
if (s1 == s2){
System.out.println("s1 and s2 are same"); // 1.
}
if (s1 == s3){
System.out.println("s1 and s3 are same"); // 2.
}
if (s1 == s4){
System.out.println("s1 and s4 are same"); // 3.
}
if (s1 == s5){
System.out.println("s1 and s5 are same"); // 4.
}
will return:
s1 and s2 are same
s1 and s3 are same
s1 and s5 are same

Reply

Javed Khan Siddiqui says

JUNE 5, 2017 AT 10:02 PM

Double Quote work progress:-
————————————
1) By using double Quote The String object(Value :- Hello)
2) is created in SCP (If and only if there is no Value in SCP like Hello earlier in SCP)
new operator work progress:-
————————————
3) By using 'new' operator it will store String object(Value :- Hello)
4)is created in Heap Memory and

5)is created SCP as well (If and only if there is no Value in SCP like Hello earlier in SCP , otherwise no use of SCP it is to provide less space in memory)********
Now coming to example
(A)
String s1 = new String("Hello"); //1
String s2 = new String("Hello");//2
//1 will create 1 in heap memory and as well as 1 in SCP because he found no such value in SCP
//2 will create 1 in heap memory and when he also going to create 1 in SCP as per process , it stopped because he found value(Hello) already there after compilation ,
So total = 3
(B)
String s1 = new String("Hello"); //1
String s2 = new String("Hello");//2
String s3 = '"Hello"';//3
//1 is creating 1 in heap and 1 in SCP
//2 is creating 1 in heap and 0 in SCP (because already there after executing //1)
//3 is creating 0 in SCP (because already there after executing //1)
So total 1+1+1 = 3
(C)
String s1 = '"Hello"';//1
String s2 = new String("Hello"); //2
String s3 = new String("Hello");//3
String s4 = '"Hello"';//4
String s5 = 's4 //5
//1 is creating 1 in SCP(because he didn't find any thing like this)
//2 is creating 1 in heap memory but 0 in SCP(Because already created in //1)
//3 is creating 1 in heap memorybut 0 in SCP(Because already created in //1)
//4 is creating 0 in SCP (Because already created in //1)
//5 this is important … passing reference…. again 0 means "Hello' passing to both reference s1 and s5
So total again 1+1+1 = 3
Reply

bhargav says
JULY 16, 2017 AT 5:05 AM
IF it will create 3 object that means..
String s1=new String("Hello");
String s2=new String("Hello");
String s3="Hello";
if total 3 object get created consider s1 will stored in SCP and if i compared s1==s3 it should be true. if not then check q5. Only intern method is responsible to stored in SCP.
correct me if i am wrong.
Reply

Tirupathi Reddy says
NOVEMBER 9, 2017 AT 6:45 AM
yes it's correct.
we are going to create object by using new operator it will be created in Heap memory not in SCP. if you are going to place the same object in SCP then we can use intern() method.
Reply

Ketki Gawande says
APRIL 14, 2017 AT 11:01 AM
good
Reply

arun says
FEBRUARY 18, 2017 AT 10:03 AM
sir i have doubt string objects are created in 4 ways
1. literal method ————–> String s1="arun"
2. new operator ————–> String s2=new String("arun");
3. satic factory method———->String s3=.String.valueOf("arun");
4.instance factory method——> String s4=s1.concat(s2);
it is correct or not
Reply

sahil mudgal says
MARCH 23, 2017 AT 11:51 PM
Hello arun,
I think the first two methods for creating string are optimum .
The 3rd method i.e String.valueOf (arg a) returns the string representation of the passed argument for char,boolean,int and double. whereas 4th method only concats two string and creates another object of String classs.
please correct me if i am wrong.
Reply

Szanowne says
DECEMBER 30, 2016 AT 8:27 AM
Hi Pankaji,
thanks for the tutorial, is very good.
I have doubts about last exercise:
How many String objects got created in below code snippet?
String s1 = new String("Hello");
String s2 = new String("Hello");
You say that the object created will be 3. I would answered 2, as new operator creates new object in the heap memory (not in the string pool) regardless of wheter there is same string stored in the string pool.
Why you say 3?
Thanks in advice.
Reply

Pankaj says
DECEMBER 31, 2016 AT 11:17 AM
Because "Hello" will be first created in String Pool, then s1 and s2.
Reply

Hareesh says
JANUARY 4, 2017 AT 10:30 AM
We have created string objects using the new operator.
String s1 = new String("Hello");
String s2 = new String("Hello");
I am thinking this objects will store only in Heap. If I am wrong correct me how exactly String objects store?.
Reply

Hareesh says
JANUARY 4, 2017 AT 7:52 PM
No need answer. I got it. Thanks
Reply

Harshal says
MARCH 11, 2017 AT 11:47 PM
By using new operator alwaz 2 objects are created , one in string constant and other in heap
Reply

sachin says
APRIL 12, 2017 AT 7:27 AM
i think you are not know that when we use a string literal than it create a new string object in the string pool area the object which is created in string pool can be reused
Reply

Szanowne says
JANUARY 6, 2017 AT 1:14 PM
I am confused, as far as I leant, string created with new operator goes into heap memory, NOT in the String pool. So why in this case an object is added in String pool?
I thought that in this case, the only way to add one of this objects in the string pool is using intern(), so smtg like this:
String s1 = new String("Hello").intern();
Reply

Dushyant Sheoran says
FEBRUARY 17, 2017 AT 7:38 AM
Using the intern function does not explicitly add a String in the String pool. What intern() does is that it assigns the reference from the String pool if it is available. So it would point the object to the same reference. See question 5 to clear all your doubts
Reply

arun says
FEBRUARY 18, 2017 AT 9:27 AM
(String constant pool(scp)) in the name only meaning is there constant values are stored in scp.
example String s1=new String("hello");;
hello is constant one object is created in scp and new keyword create another object in heap area.
if you give String s1="arun" ————->arun is constant object created in scp only
String s1="arun"
String s2="arun"_____>in this case only 1 object is created for s1 and s2.
in scp already same content exists no object created
Reply

Javed Khan Siddiqui says
JUNE 5, 2017 AT 10:16 PM
the new operation only creates one object.
<<<<<<<<<<>>>>>>>>>>
Any time you use a String literal, you implicitly create an object (unless the same literal was already used elsewhere). This isn't skipped just because you then use the object as a parameter for a new String() operation.
FYI…
String literal internally call intern method
So new operator work as 2 object ,
Think logically if new operator create only 1 then SCP will be empty So literals has to be there to provide in SCP ,
that's why we mostly not using new operator to avoid memory optimization
Reply

Sunitha Kristipati says
NOVEMBER 17, 2016 AT 1:46 AM
Hi Pankaj,
can u please look on the following,
String s1=new String("sun");
String s="sun";
System.out.println(s1.hashCode());
System.out.println(s.hashCode());
System.out.println(s1==s);
O/P:
-1856818256
-1856818256
false.
here 1st i am creating s1 with new keyword .so as you said , its creates object in heap memory and in pool,next i created using literal which will creates object only in pool. As the s1 and s are present in the pool so they are having same hashcode, but when i am comparing them why its giving false.?????
Reply

Smruti says
NOVEMBER 21, 2016 AT 9:26 PM
s1 and s are referencing the same object "sun"
Reply

ilavarasan says
NOVEMBER 28, 2016 AT 1:13 AM
Eventhough it points the same objects , it returns false because you are using the == operator to compare their references but not content.
Reply

Raja says
DECEMBER 13, 2016 AT 6:10 AM
If You use == Operator that is Identical Comparsion so.New Object Will be genearted so that is Why it return False But They are Stored in String Constant Pool only.
Reply

anshu says
JANUARY 1, 2017 AT 11:44 AM
In first object , your object in string constant pool is anonymous , it means object which have not any reference . next time when you creating object with same content then it will not create new object only it refers earlier object to new reference a reference and for making "s1′ to the reference of sun then you should have to write code like – s1=s1.intern(); then it returns true
Reply

arun says
FEBRUARY 18, 2017 AT 9:33 AM
if u r useing compareing new key word objects it will compare hash code
String s1=new String("arun") String s2=new String("arun");
new means it create object in heap area ———- if you use string literal method object is created in scp if same contenet is there new object not created only reference will store so when u use == the object ref both same thats why it will give true
new key word means different objects will created so different ref == compares so it gives false
Reply

Sunitha Kristipati says
NOVEMBER 17, 2016 AT 1:35 AM

hi pankaj,
can u please explain the following doubt….
String have two methods for creating one is literal,another way is new keyword ,if literal is the way which is memory efficient ,then why we also using ‘new’ ,please explain the scenarios where we can use them and main differentiating point to use them.
Reply

Vimal says
OCTOBER 16, 2016 AT 9:14 PM
Hi,
I have one doubts for below code.
String str = “test”;
str = str+”test2″;
str = str+”test3″;
str = str + ”test4″;
str = str + “test5”;
Now many objects will be created and how many objects will be available for garbage collection?
Can you please explain this?
Reply

rajesh says
NOVEMBER 18, 2016 AT 6:44 AM
one object, whc is stored in constant pool area,zero obj avai;able for garbage collector
Reply

AtSvm says
JANUARY 15, 2017 AT 11:40 AM
I think 9 String literals will be created in given below sequence:-
“test” , “test2” , “testtest2” , “test3”, “testtest2test3” , “test4” , “testtest2test3test4” , “test5” , “testtest2test3test4test5”
and str will point to testtest2test3test4test5 at the end .
So remaining all 8 literals will be eligible for Garbage Collection.
Reply

arun says
FEBRUARY 18, 2017 AT 9:38 AM
9 objects will create 4 objects for garbage collection

in scp 5 objects will create
in heap area 4 objects will create
one very imp point is scp area all object are not eligible for garbage collection
scp area object are distroyed when jvm shuddown
so during schudle maintaince objects will distoryed
scp area objects are not eligible for garbage collection
Reply

Javed Khan Siddiqui says
JUNE 5, 2017 AT 10:29 PM
String str = “test”;//1
str = str+”test2″;//2
str = str+”test3″;//3
str = str + ”test4″;//4
str = str + “test5”;//5
//1 1 in SCP = test
total object till now one with value —- “test”
and now str is refering to “test”
//2 1 in SCP = testtest2
total object till now two with value —- “test” , “testtest2”
and now str which was refering to “test” start referering to “testtest2”
//3 1 in SCP = testtest2test3
total object till now three with value —- “test” , “testtest2” , “testtest2test3”
and now str which was refering to “testtest2” start referering to “testtest2test3”
//4 1 in SCP = testtest2test3test4
total object till now four with value —- “test” , “testtest2” , “testtest2test3” , “testtest2test3test4”
and now str which was refering to “testtest2test3” start referering to “testtest2test3test4”
//5 1 in SCP = testtest2test3test4test5
total object till now four with value —- “test” , “testtest2” , “testtest2test3” , “testtest2test3test4” , “testtest2test3test4test5”
and now str which was refering to “testtest2test3test4” start referering to “testtest2test3test4test5”
So if you put S.O.P(str) = testtest2test3test4test5
and check how many “” object are there
test
testtest2
testtest2test3
testtest2test3test4
testtest2test3test4test5
=== Fiveeeeeeeeeeeee
Reply

shailesh says

SEPTEMBER 29, 2016 AT 12:33 PM

My program is:::

```
package interviews;
public class InternMethod {
public static void main(String[] args) {
String str1="java";
String str2=str1.intern();
String str3=new String(str1.intern());
System.out.println("hash1="+str1.hashCode());
System.out.println("hash2="+str2.hashCode());
System.out.println("hash3="+str3.hashCode());
System.out.println("str1==str2==>>"+(str1==str2));
System.out.println("str1==str3==>>"+(str1==str3));
}
}
```

==========output===>

```
hash1=3254818
hash2=3254818
hash3=3254818
str1==str2==>>true
str1==str3==>>false
```

==========

Can anyone explain how == returns false even though s1 and s3 having same hashcode?

Reply

Pankaj says

SEPTEMBER 30, 2016 AT 12:09 AM

str1 and str2 points to two different String objects, hence == is false.

Reply

Abhi says

AUGUST 11, 2016 AT 4:37 AM

Could you please add some details about hashcode() and equals operator in String class

Reply

Srinivas says
AUGUST 10, 2016 AT 9:00 AM
Stringbuffer sb=new Stringbuffer("ab");
Stringbuffer sb1=new Stringbuffer("ab");
syso(sb==sb1);
O/P ?
Reply

Ujjwal says
AUGUST 14, 2016 AT 2:11 PM
False
Reply

siva ranjan says
NOVEMBER 7, 2016 AT 7:50 PM
false
Reply

anshu says
JANUARY 1, 2017 AT 11:53 AM
false because == operator works on reference comparison . It means if one object is referred by two reference variable then it will return true . but here there are two objects are created in heap memory so it returns false.
Reply

GAURAV PANT says
JULY 5, 2016 AT 10:21 AM
Hi Pankaj,
In different ways to create string you have said:
"When we use new operator, JVM creates the String object but DON'T store it into the String Pool. "
And in string programming question:
String s1 = new String("Hello");
String s2 = new String("Hello");
Athough string s1 is creating using new operator, you are saying Hello will be saved in String pool.
Isn't both statements are contradictory.

Kindly let me know if I am missing something here.
Thanks,
Gaurav
Reply

Pankaj says
JULY 5, 2016 AT 12:02 PM
"Hello" in the String constructor argument will be created first in the Pool, since it's a string literal. Then s1 will be created in the heap.
Reply

GAURAV PANT says
JULY 5, 2016 AT 11:21 PM
So whenever we are creating string using "new" operator, JVM is storing the string into the String Pool also?
Reply

Pankaj says
JULY 6, 2016 AT 9:15 AM
It is because you are also providing string literal in double quotes as constructor argument. So Yes, if there is not a string with same value in the pool. No, if a string with same value exists in the pool.
Reply

arun says
FEBRUARY 18, 2017 AT 9:44 AM
total 3 objects created
new keyword creates objects in heap area———————2 objects
in string constant pool same content so it create ———- 1 object
————————————————————————
total 3 objects created
Reply

Jigar says
JULY 1, 2016 AT 3:13 AM
How many object will create?
String str1 = new String("abc");
String str2 = new String("xyz");
Reply

Ankita says
JULY 5, 2016 AT 5:10 AM
4 objects
First – line 1, "abc" object in the string pool.
Second – line 1, new String with value "abc" in the heap memory.
Third – line 2, new String with value "xyz" in the heap memory.
Fourth – line 2, "xyz" object in the string pool
Here since the value of string is different for both str1 and str2 hence "abc","xyz" string from string pool is not reused.
Reply

arun says
FEBRUARY 18, 2017 AT 9:47 AM
yes it is correct
Reply

Ajay Dwivedi says
MARCH 10, 2016 AT 11:37 AM
Difference between String, StringBuffer and StringBuilder?
Ans : So when multiple threads are working on same String, we should use StringBuffer but in single threaded environment we should use StringBuilder.
Hi please correct this answer it makes lots of confusion. it must be like below
Ans : So when multiple threads are working on same String, we should use StringBuilder but in single threaded environment we should use StringBuffer.
Reply

Pankaj says
MARCH 10, 2016 AT 8:25 PM
The given answer is correct. StringBuffer is thread safe, not StringBuilder.

Reply

chandini says
MARCH 20, 2017 AT 3:52 AM
Strings are immutable(we cant modify once created). These are thread safe and uses more memory.
StringBuffer : Theses are mutable(modifyable) . the objects created using StringBuffer are stored in heap.Methods under StringBuffer class are Synchronized so these are thread safe.Due to this it does not allow two threads to simultaneously access the same method . Each method can be accessed by one thread at a time .
StringBuilder : StringBuilder are similar to StringBuffer but the only difference is they are not Thread safe
Reply

Vijay says
MARCH 5, 2016 AT 12:53 AM
How many String objects got created in below code snippet?
1. String s1 = new String("Hello");
2. String s2 = new String("Hello");
Answer is 3.
First – line 1, "Hello" object in the string pool.
Second – line 1, new String with value "Hello" in the heap memory.
Third – line 2, new String with value "Hello" in the heap memory. Here "Hello" string from string pool is reused.
Here,
how the "Hello" will be created in String Pool since we have not used either double quote String s1 = "Hello" or s1 = s1.intern() method, then how it will be created in Pool as you explained.
If I am not wrong, you might have missed any one of the above way of creation. Please explain me..
Reply

Puneet Srivastava says
JUNE 8, 2016 AT 9:35 PM
whenever you write any string in double quotes it automatically creates an object of string in string pool.
Reply

Ashutosh says
FEBRUARY 17, 2016 AT 1:00 PM
Hi Pankaj, I am very use to go through with your tutorials they are excellent and simple to understand, but while reading above string interview questions I saw that "when you create string using new keyword it would create 2 object one in pool and another one in heap(if the same string was not exist in pool)". So can you please explain how exactly it works internally , I checked across on internet but unluckily didn't find any satisfactory answer or logic?
I hope you will reply. Thanks- Your's Reader .
Reply

Pankaj says
FEBRUARY 19, 2016 AT 11:49 AM
When new keyword is used, it will just create in the heap and not in String pool. You will have to use intern() method to move it to Pool.
Reply

Pankaj Verma says
JUNE 2, 2016 AT 6:28 AM
1. String s1 = new String("Hello");
2. String s2 = new String("Hello");
Answer is 3.
Then how three object created here.
I am very confused here.
According to you when you create String through new Keyword it will create two object one is on constant pool and another one is on heap.please correct me if am wrong.
Reply

Pankaj says
JUNE 2, 2016 AT 7:16 AM
"Hello" object in pool, then s1 and s2 in heap memory. Total 3 objects, i hope it clarifies your doubt.
Reply

Pankaj Verma says
JUNE 19, 2016 AT 1:39 AM
Thanks Pankaj for the reply it really help.

Navneet says
JULY 1, 2016 AT 3:54 AM
How will “Hello” go in pool? We have not used intern() method here….Am I missing anypoint here?

Mandeep says
JULY 24, 2016 AT 1:35 AM
without calling intern() method how can Hello can be stored in pool.

chandini says
MARCH 20, 2017 AT 4:01 AM
How will “Hello” go in pool? We have not used intern() method here….Am I missing anypoint here?
For the above question(as of i know)–>
in java Anything we write between double quotes is a compile time constant and it will be moved into the String pool.
String s1 = new String(“Hello”);
“Hello” will be compiled(because it is inside double quotes) and thus will be added to the String constants pool for the current JVM at compile time.
and the value of “s1′ will be resolved at run-time and will be added to the heap during run-time.

raju says
JUNE 14, 2016 AT 11:12 PM
when we create string object using new keyword two objects will be created one is in heap and onother one is scp(string constant pool) . if u creating another object with same content like above that time only aboject will be created in heap but not scp. bxz in scp duplicate not allowed
Reply

Hareesh says
JANUARY 4, 2017 AT 7:35 PM
Thanks Raju.

Reply

Sundara Baskaran says
FEBRUARY 6, 2016 AT 5:06 PM
String str2 = new String("abc");
String str1 = "abc";
System.out.println("value = " + str1.equals(str2));
The above program returns value true, can you explain why?
Reply

dilli says
FEBRUARY 8, 2016 AT 1:12 AM
Equals() method always compares contents of the strings.
Reply

Amit says
MARCH 17, 2016 AT 4:05 AM
because equals method in String class is overridden.
Reply

mani_v says
DECEMBER 19, 2015 AT 6:43 PM
Hii pankaj,
Your articles are super,first i tq u to ur thinking like to make it all useful info into one place.
Reply

Ravi says
DECEMBER 12, 2015 AT 11:19 AM
It is tremendously written thanks for such great explanation .. Keep writing we will hope for more and more concept

Reply

Pankaj says
DECEMBER 12, 2015 AT 11:15 PM
Thanks Ravi, we are determined to write best articles. You should also subscribe to our newsletter where we send exclusive tips and free eBooks.
Reply

Praveen Kumar says
DECEMBER 17, 2015 AT 7:42 AM
Hi ,
I am a great fan of your articles. It would be awesome if you could integrate disqus in your blog for comments. I have a little doubt regarding strings in java. Can you tell me how many objects are created in these two lines of code ?
StringBuilder sb = new StringBuilder("abc");
sb.append("def");
Kindly mail me if possible.
Thanks a lot.
Reply

Hareesh says
JANUARY 4, 2017 AT 7:44 PM
Hi Praveen,
StringBuffer is immutable.
at this line StringBuilder sb = new StringBuilder("abc"); —>one object created
sb.append("def"); —->you are trying to change content. so existing object goes for GC and a new object created with "abcdef"
Reply

Vishnu Prasad says
MARCH 25, 2017 AT 7:58 AM
Hi Hareesh..
You are wrong.
StringBuilder is Mutable .
If you create StringBuilder sb=new StringBuilder("abc"); then only one object will be created..and if you are try to modify StringBuilder Object then new object are not create.Only modify on that object..
Because StringBuilder capacity is ByDefault 16.and It's Mutable
StringBuilder sb = new StringBuilder("abc");
System.out.println(sb.capacity());// 16
For Example-

StringBuilder sb = new StringBuilder("abc");
System.out.println(sb.capacity());//19 => 3 abc+16 default
System.out.println(sb.length());//3
sb.append("def");
System.out.println(sb.capacity());//19
System.out.println(sb.length());//6
If Default capacity is full then JVM will create a new Object ..
sb.append("I self Vishnu");
System.out.println(sb.capacity());//19
System.out.println(sb.length());//19
Hear Default capacity is full again if you try to modify then JVM will create new Object and copy all data of previous Object after then destroy the previous Object
like :formula of New Object Capacity:
New Capacity=(Initial Capacity*2)+2
=(19*2)+2
=38+2=40
Ex:
sb.append("Prasad");
System.out.println(sb.capacity());//40
System.out.println(sb.length());//25
If you Define our own capacity then Create the StringBuilder Object to Following way..
StringBilder(int Initial Capacity);
Ex:=>
StringBuilder sb1=new StringBuilder(100);
System.out.println(sb1.capacity()); //100
System.out.println(sb1.length()); //0

Walter Frobin Ekka says
NOVEMBER 10, 2015 AT 8:17 AM
Pankaj, you said like char array is more preferable than String to store password but if you go with the security concern String immutability is big advantage to store such sensitive information like Username,password.
Looks conflicting to me.
Reply

Pankaj says

NOVEMBER 11, 2015 AT 10:28 PM

Strings are stored in String Pool as plain text and have longer life, anybody having access to memory dump can find it.

In favor of this, getPassword() method of JPasswordField returns a char[] and deprecated getText() method which returns password in clear text stating security reason.

Reply

ARVIND AGGARWAL says

NOVEMBER 6, 2015 AT 9:05 PM

Hey Pankaj…..

Where is comparison operator in Question 1 for which U R saying that don't confuse with that?

Reply

Senthil Narayanan says

AUGUST 11, 2015 AT 10:51 AM

Very helpful…. thanks pankaj:)

Reply

shank's says

OCTOBER 4, 2015 AT 11:45 AM

deep approach in string…..its really helpful

Reply

irfan says

JULY 29, 2015 AT 4:07 PM

Amazing work Pankaj… keep up the good work…I have been a regular visitor of your site and posts.

Reply

maitrey says

JULY 12, 2015 AT 1:55 AM

Dear sir,

I have one basic question.

Why java supports both features of string string creation.
String str="abc";
String strObj=new String("abc");
str="abc" is better than new String().
So why Java supports new String();
Reply

Chetan says
SEPTEMBER 23, 2015 AT 12:59 PM
I am also having the same question. Recently interviewer asked this to me that what is the significance of string object creation with new keyword. If you have any thoughts then please add it
Reply

ashish says
NOVEMBER 20, 2015 AT 10:23 PM
In case of instance of string in constant string pool
while in second case two instance will be created one will be stored constant string pool while other in heap (as commonly object reference are stored in case of using new keyword).
so String str="abc" is memeory efficient compared to String strObj=new String("abc");
Reply

Ashutosh says
FEBRUARY 17, 2016 AT 1:09 PM
Only designer of Java can give you more specific and correct answer we can only make few assumption on that (1) if someone want to create new string each and ever time so they can use new keyword.etc
Reply

Ashutosh says
FEBRUARY 17, 2016 AT 1:20 PM
one more point is on why to create string using new keyword, you can use different constructors to create a string.
For example:
creating string by char array
creating string for specific encoding etc.
Reply

srikanth says
MAY 31, 2015 AT 12:03 AM
I have two Strings str1="ABC",str2="BC" ,then output will come op1=A, op2=null, and next string are change str1="B" str2= "BANGALORE", then op1="ANGALORE" op2=null, how can write write the program please could you explian ?
Reply

Pankaj says
MAY 31, 2015 AT 12:30 AM
What is the logic for this, what are you trying to achieve here? If you have logic, create an algorithm for that and then writing code is the easy part.
Reply

Lukman says
MARCH 23, 2015 AT 10:49 PM
Hi Pankaj,
Very useful work……
keep it up…
Reply

ram says
MARCH 18, 2015 AT 7:43 AM
I need program:
write a java program to give the spaces b/w words
Ex:
i/o : ramisagoodboy
o/p : ram is a good boy
Reply

siddharth says
SEPTEMBER 9, 2015 AT 5:02 AM
use trim() method when need spaces
Reply

siddharth says
SEPTEMBER 9, 2015 AT 5:03 AM
sorry use append(" "); method
Reply

Hareesh says
JANUARY 4, 2017 AT 8:00 PM
search in GOOGLE with below words:
how to trim a string in java
Reply

amar says
JANUARY 16, 2015 AT 5:59 AM
How to Print source code as output in java??
Reply

amar says
JANUARY 16, 2015 AT 5:55 AM
hi..
How to print program source code as output in java language..?
Reply

arun says
DECEMBER 25, 2014 AT 10:43 PM
hi I have one small doubt which data type is used to check whether the given string student is present or absent
Reply

Sarita says
OCTOBER 2, 2014 AT 12:31 AM

Hi pankaj,
I have a question in string permutation?
for example: Input: sarita
key :ri
the out put should be :sarita, sarIta, saRita, saRIta
Reply

Sanjana says
SEPTEMBER 30, 2014 AT 10:35 PM
Hi,
I need a solution for this
Consider String str1 = “hari”;
String str2 = “malar”;
now find the same characters in both the string and delete the repeated characters
So, the output must be
str1 = hi
str2 = mla
Reply

sudhakar says
SEPTEMBER 22, 2014 AT 2:03 AM
can you expalin about thread in real time?
Reply

Ishan Aggarwal says
SEPTEMBER 20, 2014 AT 6:09 AM
Hi Pankaj,
I have one doubt. I want to know if we execute the below statements then how many objects and references will be created on each line 1, 2 and 3.
1.) String s1 = “abc”;
2.) String s2 = “abc”;
3.) String s3= new String(“abc”);
Thanks,
Ishan
Reply

Pankaj says

SEPTEMBER 20, 2014 AT 10:36 AM

1. “abc” in String Pool and s1 reference in the stack memory.
2. s2 reference created in stack memory, referring to the same string object “abc” in the pool.
3. s3 reference created and new String object with value “abc” in the heap memory.

So total 2 string objects and 3 references.

Reply

vallabhi says

SEPTEMBER 25, 2014 AT 2:20 AM

Hi Pankaj

Thanks for these wonderful questions

I have a doubt that if we can create objects through String literals than why do we use creating string objects through new keyword as it is not memory efficient way. Is it possible to completely remove the feature of creating String objects through new keyword..Please Explain

Reply

Pankaj says

SEPTEMBER 25, 2014 AT 2:55 AM

Since String is an Object and we can use **new** operator to instantiate it. It’s not possible to remove this option.

Reply

vallabhi says

SEPTEMBER 25, 2014 AT 5:42 AM

Thanks for the reply.. But the question is that is there any situation where we can only use String Object with new keyword instead of String literal. Actually an interviewer asked me the question that when String literal is memory efficient than why do we create object in string using new keyword..

Ruchi Gupta says

SEPTEMBER 29, 2014 AT 5:12 AM

Strings created in pool are not garbage collected.

Please correct me Pankaj if i am wrong.

Reply

chandini says

MARCH 20, 2017 AT 4:24 AM

I hope those strings will be there as long as their scope

Anoop says
DECEMBER 17, 2014 AT 3:26 AM
If the references were defined as instance variables, they are created in the heap memory.
Reply

aditya says
OCTOBER 16, 2014 AT 10:26 AM
Only two objects will be created … one objects for String s1 = “abc” and another for new String(“abc”)
Reply

Dhruba Jyoti Talukdar says
DECEMBER 21, 2014 AT 11:01 AM
Hello Ishan,
String x = “abc”;
// It creates 1 String object and 1 reference variable.
//”abc” will go to pool memory and x will refer to it.
String y = new String(“xyz”);
//It creates 2 objects and one reference variable.
//In this case, because we used the new keyword, java will create a new String object in the normal (non-pool) memory, one object in the pool memory(for storing “xyz”), and y will refer to it.
1.) String s1 = “abc”; -> 1 object and 1 refercene
2.) String s2 = “abc”; -> 0 object and 1 reference
3.) String s3= new String(“abc”); -> 2 objects and 1 reference.
So the correct answer to your question would be 3 objects and 3 reference.
Regards
Dhruba
Reply

Ruchi Gupta says
SEPTEMBER 11, 2014 AT 6:28 AM
public void testFinally(){
System.out.println(setOne().toString());
}
protected String setOne(){
String s1=new String();
try{
s1.concat("Cool");
return s1=s1.concat("Return");
}finally{
s1=null; /* ☺ */
}
}
When s1 string concatenates with cool it has content as "cool". In next statement it points it to same reference s1.so the output should be "returncool".Why its "return".
Reply

Pankaj says
SEPTEMBER 11, 2014 AT 11:09 PM
The point to remember is that String is immutable, so in `s1.concat("Cool");` line, a new String is created but since you didn't assigned it, it's lost. s1 value is still "".
`s1=s1.concat("Return");`: here you are assigning after concat, so s1 is "return" now, hence the output.
Reply

rose says
SEPTEMBER 19, 2014 AT 2:12 AM
String is Immutable ,whenever we are adding anything using concat method it points new ref after concatination if string not modified then it point same ref ,1st time u r not assigned any variable so it will create other obj and second time your storing in again s1 so it returns return only,but this process not applicable for Adding string using "+ " operator.
Reply

Harikrishna says
MAY 28, 2014 AT 3:24 AM
String is not final by default since you have mentioned "String immutable and final in java"?

Reply

Pankaj says
MAY 28, 2014 AT 9:09 AM
String is a final class and immutable too.
Reply

Rajesh says
SEPTEMBER 24, 2014 AT 11:40 AM
not satisfied…..see the code
class StringFinal{
public static void main(String args[]){
//String s1 = new String("hi");
String s1 = "hello";
s1 = "hi";
System.out.println(s1);
}
}
it will print hi,,,,not hello
Reply

Pankaj says
SEPTEMBER 25, 2014 AT 1:19 AM
You need to understand that s1 is reference only to object. First it's referring to "hello" and then "hi", hence the output. I would suggest you to read about immutability.
Reply

Anshul says
NOVEMBER 14, 2014 AT 9:27 AM
lol….pankaj is saying String class is declared final in JDK source code dumbo…
final class String {} in JDK
Reply

Hareesh says
JANUARY 4, 2017 AT 9:42 PM
whatever Pankaj said that is True.

String s1 = “hello”; –> new object created
System.out.println(s1.toUpperCase()); –>In String class we have methods like toUpper() and toLower() you are trying to make Uppercase to s1. this will result “HELLO” but this will not store in s1.
System.out.println(s1) –>results hello only
s1 = s1.toUpperCase
System.out.println(s1) —->results HELLO
s1 = “hi”; —> here you are not changing object. you are changing object reference.
Reply

saurabh moghe says
MAY 21, 2014 AT 10:44 PM
Hi ,
in answer of ‘What are different ways to create String Object?’ you have said when we use new operator with string .it will create one object..but in scjp 6 book author is saying there will be 2 objects , one in string pool and second is in heap. see this link also ‘http://www.coderanch.com/t/245707/java-programmer-SCJP/certification/String-object-created’ (http://www.coderanch.com/t/245707/java-programmer-SCJP/certification/String-object-created%27) .. please elaborate this string concept
Reply

Gary says
APRIL 18, 2014 AT 1:05 PM
Very helpful, just wanted to say that I appreciate the clarity with which the answers are presented.
Reply

ADITYA VALLURU says
APRIL 8, 2014 AT 5:30 PM
Hi Pankaj,
I liked you post because the coding question you put in it. We can get the theory knowledge from any web site, but questions like coding based we can’t get from other site. Keep up good job and keep concentrate on coding related question.
I really like one ans. i.e Write a method that will remove given character from the String?
I know this method, but you used in a different manner. Expecting more like this from you. I will be in touch with you after this. I met this site first time.

Reply

Pankaj says
APRIL 8, 2014 AT 7:46 PM
Thanks Aditya, please look into other posts too. I am sure you find a lot of good stuffs. □
Reply

kiki says
MARCH 29, 2014 AT 4:13 PM
excellent post !!! could you discuss programming questions related to strings such as reverse a stringand other string operations.. which algorithm will be best and its time complexity.. thanks
Reply

Ashi says
FEBRUARY 18, 2014 AT 2:57 PM
Please also put some light on the implementation of substring? How substring method can cause memory leaks? How can we avoid these memory leaks in java?
Reply

senthil hari says
DECEMBER 9, 2013 AT 2:15 PM
Hi pankaj,
Thanks a lot for providing examples and good explanations.if you have stuff about GWT framework(Google Web Toolkit) please post that frame work concepts too.it will be really help to for those who new to this framework
Reply

devs says
NOVEMBER 5, 2013 AT 11:26 AM
All your tutorials are great. Helping me in my job jump. Thanks a lot.
Reply

nilesh shinde says
JULY 6, 2013 AT 4:35 PM
after readind all your questions i realfeel like am getting real basic and clear from all my doubts. pls increase your collections. i read String and COllection and it helped m e
Reply

Pankaj says
JULY 7, 2013 AT 6:47 AM
Thanks Nilesh, it really feels good that my articles are helping and cleared your doubts.
Reply

dinesh says
FEBRUARY 3, 2014 AT 6:03 AM
its really good and useful…try to post some typical and tricky programs on strings which will be helpful for interviews and thanks……
Reply

Trilochan says
JUNE 27, 2013 AT 10:49 AM
really this material is gud. definitely it will help a lot 2 give a good concept in string…
Reply

Pankaj says
JULY 7, 2013 AT 6:48 AM
Thanks for liking it.
Reply

tamil says
AUGUST 15, 2013 AT 2:09 PM
sir pls clearly explain the java in string concept
Reply

Pankaj says

AUGUST 15, 2013 AT 8:43 PM

I have already wrote a lot about String class in java, String is like other classes in java with some additional features because of it's heavy usage.

Read these posts for better understanding:

https://www.journaldev.com/802/why-string-is-immutable-or-final-in-java

https://www.journaldev.com/797/what-is-java-string-pool

Reply

nandan says

JUNE 24, 2013 AT 7:58 PM

Hi pankaj ,

your stuff regarding strings in java is really great . it is really helpful in prospective of interviews and gaining some knowledge over java strings .. Loved it ..Keep up good work buddy!!!

Reply

Pankaj says

JULY 7, 2013 AT 6:48 AM

Thanks Nandan for kind words. It's these appreciations that helps me keep going.

Reply

aicotutorial.com says

JUNE 1, 2013 AT 3:37 PM

Thanks for finally talking about > Java String Interview Questions and Answers | JournalDev < Loved it!

Reply

Prosenjit says

MAY 3, 2013 AT 1:24 AM

Nice Material….

Keep it up to make us knowledgeable..

Reply

Ravi says
APRIL 16, 2013 AT 1:10 PM
Hi Pankaj,
There is always something new to learn in your posts.
But I would like to correct one thing in above post.
When we create a String using **new** keyword, a whole new reference is being assigned, not from a literal. Up to this its right. But this instance is not added into the String Pool until we call intern(). We can check it out by below example:

```
String s1 = new String("abc"); // new object - not from pool.
String s2 = "abc"; // tries to search it from pool.
System.out.println(s1 == s2); // returns false that means s1 has not been added to pool.
```

Reply

Pankaj says
JULY 7, 2013 AT 7:02 AM
Thanks Ravi, yes you are right. Corrected the post.
Reply

Amit Malik says
SEPTEMBER 10, 2013 AT 4:46 AM
Pankaj, I think you were right before.
when we create string with String s1 = new String("abc") then two objects are created.
One is created on heap and second on constant pool(created only if it is not there in the pool).
Since we are using new operator to create string so s1 always refers to the object created on the heap.
When we create string with String s2 = "abc" then s2 will refer to object created in the constant pool. No object will be created on the heap.
Since s1 and s2 are referring to different objects s1 == s2 will return false.
If we want s1 to refer object created on the pool then use s1.intern().
Reply

Pankaj says
SEPTEMBER 10, 2013 AT 5:34 PM
Thats what is mentioned here…
Reply

deepak says
AUGUST 31, 2016 AT 11:20 PM

please explain why intern() is needed if the object will be created in both constant pool and heap when we use new operator.

deepak says
AUGUST 31, 2016 AT 11:18 PM
Hi
I've a query!!
please correct me if I'm wrong.
String s = new String("Hello");
When above code is executed two objects will be created. One in heap another in SCP(String Constant Pool)
So what is the need to use intern() explicitly.? Or when do we use intern() exactly??
by above discussion i got that object will not be added to SCP but its created.!! then where it is created.?
Reply

Ayan says
FEBRUARY 1, 2013 AT 7:17 AM
Can you please explain the statement "Strings are immutable, so we can't change it's value in program. " by a sample program
Reply

Pankaj says
FEBRUARY 2, 2013 AT 10:55 PM

```
String str = "abc";
str = "xyz"; // here the value of Object str is not changed, a new String literal "xyz" is created and str is now holding reference to this new String.
```

I would suggest you to go through this post to know the properties of a immutable class and how you can write your own immutable class.
Immutable Classes in Java
Reply

siddu says
OCTOBER 13, 2013 AT 7:49 PM
String s1=new String("Hello")
above code How many objects will create?
Reply

Deepak Chauhan says
NOVEMBER 14, 2013 AT 6:28 PM
There will be two objects created one is in heap and one is in constant pool.
Ques.. how and Why ..
How
Everything that is inserted within the " " (double quote) is a string and JVM forces to allocate the memory in the Constantpool.. ok fine. And we also know the new Keyword that is used to allocate the memory in the Heap.
And as the SCJP standard in case of string making the object with new keyword is certainly memory lose.
example:
String s=new String("Deepak");//line 1
String s1="Deepak";
the reference Id of "Deepak" object will be assigned to s1.because It is already in the pool.//see line1
Question Arise:
What kind of Memory is used by the ConstantPool….Heap or other..
Reply

Paramjeet Singh says
SEPTEMBER 13, 2014 AT 11:55 PM
String s=new String("Deepak");
when u have create a String object by using new keword than every time create new object in heap;
but by using literal String than check in the memory
but here there are two object is created
one is heap , and second in pool;
public class java {
public static void main(String arr[])
{
String s=new String("Deepak");
String s1="Deepak";
String s2=new String("Deepak");
System.out.println(s==s2);
System.out.println(s==s1);

```
}
}
output is false
false
that is solution
```

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Save my name, email, and website in this browser for the next time I comment.

POST COMMENT

Search for tutorials...

DOWNLOAD ANDROID APP

CORE JAVA TUTORIAL

Java 10 Tutorials Java 9 Tutorials Java 8 Tutorials Java 7 Tutorials Core Java Basics OOPS Concepts Data Types and Operators String Manipulation Java Arrays Annotation and Enum Java Collections Java IO Operations Java Exception Handling MultiThreading and Concurrency Regular Expressions Advanced Java Concepts

IMPORTANT INTERVIEW QUESTIONS

Java Interview Questions

- Core Java Interview Questions
- String Interview Questions
- Multithreading Interview Questions
- Collections Interview Questions
- Exception Interview Questions
- Java Programming Interview Questions
- Java 8 Interview Questions Part 1
- Java 8 Interview Questions Part 2
- Servlet Interview Questions
- JSP Interview Questions
- Struts 2 Interview Questions
- JDBC Interview Questions
- Spring Interview Questions
- Hibernate Interview Questions
- JSF Interview Questions
- Web Services Interview Questions
- Scala Basic Interview Questions
- Scala Intermediate Interview Questions
- Scala Advanced Interview Questions
- Scala Interview Questions Summary
- Common Job Interview Questions

Miscellaneous

- Java ClassLoader
- String StringBuffer StringBuilder
- Java is Pass By Value
- Java Heap vs Stack Memory

RECOMMENDED TUTORIALS

Java Tutorials

- Java IO
- Java Regular Expressions
- Multithreading in Java
- Java Logging
- Java Annotations
- Java XML
- Collections in Java
- Java Generics
- Exception Handling in Java
- Java Reflection
- Java Design Patterns
- JDBC Tutorial

Java EE Tutorials

- Servlet JSP Tutorial
- Struts2 Tutorial
- Spring Tutorial
- Hibernate Tutorial
- Primefaces Tutorial
- Apache Axis 2
- JAX-RS
- Memcached Tutorial