

RedBush Technologies

209, Unitech Arcadia, South City 2 Sector 49, Gurgaon-122018

PH: 9971483222 / 9718212607

How many ways we can define String in Java?

There are basically two ways to define the string:

1. **String Literal:** Created by using double quotes. JVM first checks the "String Constant Pool". If the string already present in the same, a reference of the pooled instance is returned. If no such string is present in the pool, a new string instance is created and placed in the "String constant Pool". This makes Java more memory efficient.

```
String s="Ritu Bajar";
```

2. **By using a new Keyword:**

```
String s= new String ("Ritu Bajar");
```

Why Strings are immutable in java?

A string is an immutable object in java because it is stored in a string pool. Additionally strings are thread safe because they cannot be used at the same time with the same object. The Threads have to wait in line.

Difference between String buffer and String Builder?

String buffer: It is a mutable object which can change post definition. It is kept in a 'storage heap' and 'thread safe'. The thread has to wait in line before it can be processed.

String builder: This is a 'value changed' object also kept in a 'storage heap' and it is slow too.

The main difference between the StringBuffer and StringBuilder is that StringBuffer is not thread safe.

Difference between Function over loading and overriding?

Function overriding is an example of Runtime polymorphism in Java where we can have same method with same parameters or signature, but associated in a class & its subclass. Here, Java compiler does not understand which method is called at compilation time. Only JVM decides which method is called at run-time. For e.g.

netbanking()-parent

mobilebanking()-parent

offlinebanking()-parent

banking()-child

netbanking bn = new banking;

Now, the netbanking class methods can be used in the banking class. Here, we need to use the extends keyword while child class declaration.

What is Garbage collector?

The garbage collector is a program which runs on the Java Virtual Machine which gets rid of objects which are not being used by a Java application anymore. It is a form of automatic memory management.

When a typical Java application is running, it is creating new objects, such as Strings and Files, but after a certain time, those objects are not used anymore. For example, take a look at the following code:

```
for (File f : files) {  
    String s = f.getName();  
}
```

In the above code, the String s is being created on each iteration of the for loop. This means that in every iteration, a little bit of memory is being allocated to make a String object.

Going back to the code, we can see that once a single iteration is executed, in the next iteration, the String object that was created in the previous iteration is not being used anymore — that object is now considered “garbage”.

Eventually, we’ll start getting a lot of garbage, and memory will be used for objects which aren’t being used anymore. If this keeps going on, eventually the Java Virtual Machine will run out of space to make new objects.

That’s where the garbage collector steps in.

The garbage collector will look for objects which aren’t being used anymore, and gets rid of them, freeing up the memory so other new objects can use that piece of memory.

Difference between While and Do while?

1. The most important difference between While and Do-While is that in Do-While, the block of code is executed at least once. i.e., the Do-While loop runs at least once, even though the condition given is false.
2. While is Entry Controlled and Do-While is Exit Controlled.

3. //while

```
while(condition is true) {  
    something }
```

```
//do-while
```

```
do{  
    something  
}
```

```
while(condition is true)
```

Difference between HashMap and Hash Table?

Hashtable is synchronized, whereas HashMap is not. This makes HashMap better for non-threaded applications, as unsynchronized Objects typically perform better than synchronized ones.

Hashtable does not allow null keys or values. HashMap allows one null key and any number of null value.

Difference between Public and Private Access modifiers?

The Public access modifiers is specified using Public keyword. The classes, methods or data members declared as Public can be accessed from anywhere in the program. Where as Private Access modifier is accessible on in the same class where it has been declared.

Differences between List and Set interfaces in java?

1. List is an ordered collection it maintains the insertion order, which means upon displaying the list content it will display the elements in the same order in which they got inserted into the list.
Set is an unordered collection, it doesn't maintain any order. There are few implementations of Set which maintains the order such as LinkedHashSet (It maintains the elements in insertion order).
2. List allows duplicates while Set doesn't allow duplicate elements. All the elements of a Set should be unique if you try to insert the duplicate element in Set it would replace the existing value.
3. List implementations: ArrayList, LinkedList etc.
Set implementations: HashSet, LinkedHashSet, TreeSet etc.
4. List allows any number of null values. Set can have only a single null value at most.

Difference between final and finally keyword?

final is a keyword which cannot be used as an identifier as it is reserved keyword. It can be used with a class, method, variable.

final variable: A final variable cannot be changed once initialized.

final method: A final method cannot be overridden in subclass.

final class: A final class cannot be inherited.

Whereas finally is also a reserved keyword in java which is used in association with a try/catch block and guarantees that a section of code will be executed, even if an exception is thrown.

For example when the program is executed the flow starts from main program and when it enters try block

if it finds exception it will execute catch block and it will execute finally block, if no exception is found it will execute finally block directly.

that means in both cases irrespective of exception ,finally block will be executed.

Why Static is defined to the variables and classes? In what context it is required?

It is a variable which belongs to the class and not to object(instance) Static variables are initialized only once, at the start of the execution. These variables will be initialized first, before the initialization of any instance variables

A single copy to be shared by all instances of the class

A static variable can be accessed directly by the class name and doesn't need any object

Syntax:

<class-name>.<variable-name>

Static Method: It is a method which belongs to the class and not to the object(instance) A static method can access only static data. It can not access non-static data (instance variables)

A static method can call only other static methods and can not call a non-static method from it.

A static method can be accessed directly by the class name and doesn't need any object

A static method cannot refer to "this" or "super" keywords in anyway

Syntax :

<class-name>.<method-name>

What is Run time polymorphism?

In STATIC POLYMORPHISM, compiler itself determines which method should call. Method overloading is an eg of static polymorphism.

In RUNTIME POLYMORPHISM, compiler cannot determine the method at compile time. Method overriding is an eg of runtime polymorphism.

Because in runtime polymorphism, the signature of the method is same in both the class. So compiler cannot determine method at compile time which should be executed. Only after object creation(which is a runtime process), the runtime environment understands the exact method to call.

Difference between Interface and Abstract class?

Interface is like a contract with the class, which says I only accept stuff, which looks like the signature that I provide. And class that implements the interface says, sure I will make sure that I look like that. Interface is kind of a prototype; there are only signatures of methods. Methods do not have any implementation; they do not have behaviour of their own.

Abstract class on the other hand is like a class. They look like interface but they can have implementation of methods along with abstract methods. These methods may or may not have behaviour and can be overridden in the class that extends the abstract class. It like saying, abstract class tell the classes extending it, that all the classes should have implemented methods in common (might be overridden) and then these classes should implement the abstract methods also.

Is Multiple Inheritance is allowed in Java?

Multiple Inheritance is not allowed in Java directly, but through 'interfaces' it is allowed. The Reason why? It introduces more complexity and ambiguity.

Difference between HashMap and Hash Table?

A HashTable is some kind of lookup table using key hashes to lookup the corresponding value in a table like data structure. That's only one kind of a key-value Mapping. There are different implementations as you are probably aware. Different hashes, hash collusion solutions and table growing strategies and more under the hood. It's only interesting if you need to make your own hash table for whatever reason.

A HashMap is some kind of mapping of key-value pairs with a hashed key. Mapping is abstract as such and it may not be a table. Balanced trees or tries or other data structures/mappings are possible too.

You could simplify and say that a HashTable is the underlying data structure and the HashMap may be utilizing a HashTable.

Difference between List and Set?

List

- Is an Ordered grouping of elements.
- List is used to collection of elements with duplicates.
- New methods are defined inside List interface.

Set

- Is an Unordered grouping of elements.
- Set is used to collection of elements without duplicates.
- No new methods are defined inside Set interface, so we have to use Collection interface methods only with Set subclasses.

Difference between Super and This Keyword in Java?

'this' keyword refers to a reference of the current class.

'super' keyword refers to the parent of the current class (which called the super keyword).

By using 'this' keyword, it allows you to access methods/attributes of the current class (including its own private methods/attributes).

'super' keyword allows you to access public/protected method/attributes of parent(base) class. You cannot see the parent's private method/attributes.

What is the difference between an array and a list?

Array and linked list are both meant for storing information in parts. The difference mainly lies in their method of using to store and way of access

ARRAY

- the information is stored in form of continuous memory allocations. Each part follows other just after it in the m/y. there is no randomness in allocation.
- they provide random access like arr[0],arr[6] etc
- There is static allocation of memory. n this may lead to wastage of memory
- there can be only one type of data in each cell of an array
- insertion and deletion are bit more time consuming

LINKED LIST

- The information is stored RANDOMLY in parts. n each part is connected to other via a pointer to next cell (n to previous cell in case of doubly linked list)
 - they are to be accessed sequentially because of the dependency of each part
 - it is dynamically allocated that is m/y is allocated to each cell after processing request for it. hence there's no m/y wastage
 - a single cell can be divided into many parts each having info of different data type. but the last necessarily needs to be the pointer to the next cell
 - Insertion and deletion are lot easier and faster. Searching too is simpler.
- So overall linked list are better choice to use. But there are cases where arrays are used to avoid unnecessary pointers and lengthy code if not much manipulation of the information is required.

What is the difference between HashSet and Tree Set ?

HashSet is Implemented using a hash table. Elements are not ordered.

TreeSet is The elements in a set are sorted.

Other differences are:

1. First major difference between HashSet and TreeSet is performance. HashSet is faster than TreeSet and should be preferred choice if sorting of element is not required.
2. Second difference between HashSet and TreeSet is that HashSet allows null object but TreeSet doesn't allow null Object and throw NullPointerException
3. Now most important difference between HashSet and TreeSet is ordering. HashSet doesn't guaranteed any order while TreeSet maintains objects in Sorted order defined by either Comparable or Comparator method in Java.

Difference between Public and Private Access modifiers

Private: Limited access to class only

Default (no modifier): Limited access to class and package

Protected: Limited access to class, package and subclasses (both inside and outside package)

Public: Accessible to class, package (all), and subclasses... In short, everywhere.

Why Static is defined to the variables and classes? In what context it is required?

static variable is used as class level variable where its single copy is shared amongst all the instances of the class.

it can also directly accessed through the class and without creating a new object because it does not belong to any objects.

Some usage for static variable is with final keyword for creating constant variable and using one instance with other objects in the class.

Static classes are sealed and cannot be inherited therefore nested classes can only be static.

How to convert Array list to array and array to array list?

there are many different ways to to convert array string into array list, here's one of the simpler solutions.

```
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
public class ConvertArrayToArrayList {
public static void main (String [] args){
ArrayList<String> arrList= new ArrayList<String>();
String array[] = {"Raj","Kumar","Rajkumar"};

//array.length returns the current number of elements present in array
for(int i =0;i<array.length;i++) {

//We are adding each array's element to the ArrayList
arrList.add(array[i]);
}
System.out.println(arrList);
/*ArrayList content*/
for(String str: arrList)
{
System.out.println(str);
}
}
}
```

array list into array

```
List<String> list = new ArrayList<String>();
```

```
list.add("India");
list.add("Switzerland");
list.add("Italy");
list.add("France");

String [] countries = list.toArray(new String[list.size()]);
```

Mention the uses of Synchronized block?

In general, synchronized methods are used to protect access to resources that are accessed concurrently. When a resource that is being accessed concurrently belongs to each instance of your class, you use a synchronized instance method; when the resource belongs to all instances (i.e. when it is in a static variable) then you use a synchronized static method to access it.

Note that using synchronized without a specific lock object is generally not the safest choice when you are building a library to be used in code written by others. This is because malicious code could synchronize on your object or a class to block your own methods from executing. To protect your code against this, create a private "lock" object, instance or static, and synchronize on that object instead.

The synchronized keyword can be used to mark four different types of blocks:

1. Instance methods

Uses synchronized keyword in the method declaration. This tells Java that the method is synchronized.

A synchronized instance method in Java is synchronized on the instance (object) owning the method. Thus, each instance has its synchronized methods synchronized on a different object: the owning instance. Only one thread can execute inside a synchronized instance method

2. Static methods

Static methods are marked as synchronized just like instance methods using the synchronized keyword.

Synchronized static methods are synchronized on the class object of the class the synchronized static method belongs to. Since only one class object exists in the Java VM per class, only one thread can execute inside a static synchronized method in the same class.

If the static synchronized methods are located in different classes, then one thread can execute inside the static synchronized methods of each class. One thread per class regardless of which static synchronized method it calls.

3. Code blocks inside instance methods

You do not have to synchronize a whole method. Sometimes it is preferable to synchronize only part of a method.

Java synchronized block construct takes an object in parentheses. The object taken in the parentheses of a synchronized construct is called a monitor object. The code is said to be synchronized on the monitor object. A synchronized instance method uses the object it belongs to as monitor object. Only one thread can execute inside a Java code block synchronized on the same monitor object.

4. **Code blocks inside static methods**

For example, you could make a static factory method that keeps a “registry” of all objects that it has produced. A natural place for such registry would be a static collection. If your factory is used from multiple threads, you need to make the factory method synchronized (or have a synchronized block inside the method) to protect access to the shared static collection.

RedBush Technologies