Q. Can we overload the main method in Java?

Yes, we can overload the main method in Java, but When we execute the class JVM starts execution with public static void main(String[] args) method.

public class Sample{

public static void main(){

System.out.println("This is the overloaded main method");

}

public static void main(String args[]){

Sample obj = new Sample();

obj.main();

}

}

For more:

<https://stackoverflow.com/questions/3759315/can-we-overload-the-main-method-in-java>

Q. Can we override main method?

**No**, we cannot override main method of java because a static method cannot be overridden.  
The static method in java is associated with class whereas the non-static method is associated with an object. Static belongs to the class area, static methods don’t need an object to be called. Static methods can be called directly by using the classname ( classname.static\_method\_name()).  
So, whenever we try to execute the derived class static method, it will automatically execute the base class static method.  
Therefore, it is not possible to override the main method in java.

Q. Can we overload or override static methods in java?

<https://www.geeksforgeeks.org/can-we-overload-or-override-static-methods-in-java/?ref=rp>

Q. Can We define the main method as different access modifier other than public?

No . main method can not have any access modifier other than public if you use private, protected or default access modifier then compilation will be fine but it will generate run time error, because program execution done by jvm and it only know the single definition of main method.

Q. what will happen if we declare main method as non static?

Program compilation will be success because compiler consider may the main method is overloaded

but program cannot run successfully because program run by jvm and it knows only one method public static void main(Sring[] args) , it will generate the run time error because.

Q. Why Main method is static?

Because Main method is called by JVM, and it can only search the main method if it is static so that it does not required to have the object it will be able to call directly with the class name.

Java **main**() **method** is always **static**, so that compiler can call it without the creation of an object or before the creation of an object of the class. ... **Static method** of a class can be called by using the class name only without creating an object of a class

Q. Can We Define Class Without Main?

No , we cannot define the class in java without main method starting from java7 .

In previous version we can declare class without main method with the help of static block but now from java7 it is not possible class StaticInitializationBlock{

   static{

      System.out.println("class without a main method");

      System.exit(0);

   }

}

Q. Can we Change Return type of main method in java?

No, it must be void , if we change the return type it will get compiled but we will get run time error.

Q. What is JVM.

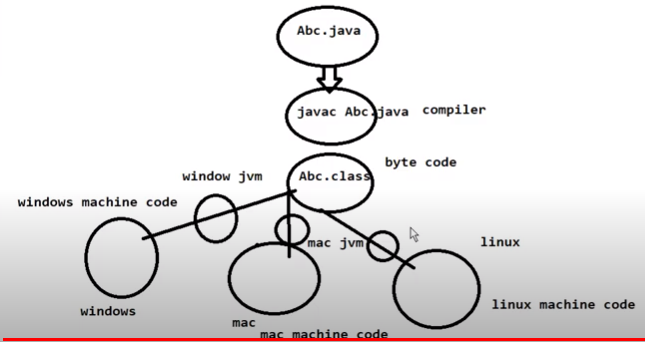
It is stands for java virtual machine, it is engine present in jre which is used for provide the environment for drive or execute the java compiled code in to the machine code and execute it.

First JDK compiler convert your source code to byte code and then JVM access the byte code and convert in machine code

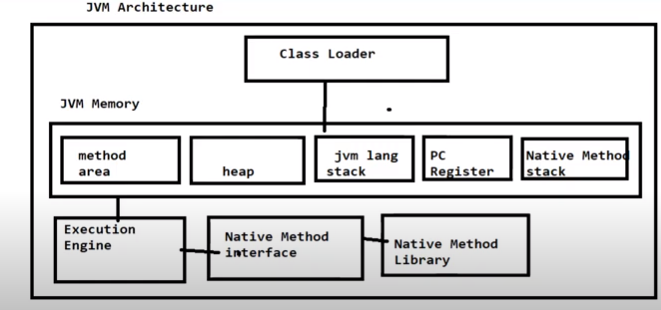
<https://www.geeksforgeeks.org/differences-jdk-jre-jvm/>

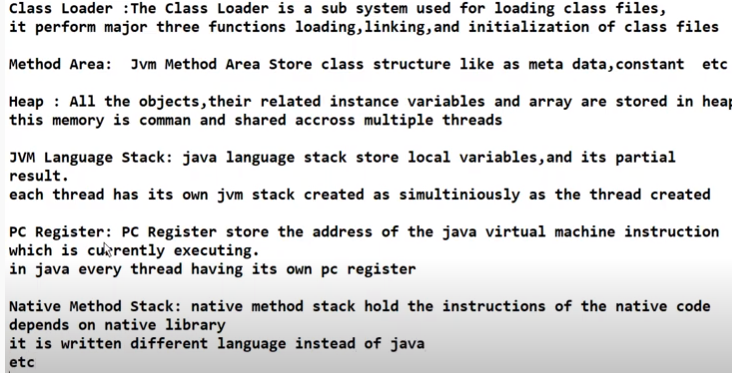
Q. Are JVM platform independent

No, each OS’s will have their own JVM because JVM needs to translate java byte code to machine code and machine code is strictly related with operating system. it changes by cpu to cpu and os to os.



Explain JVM architecture?





Q. can you pass the negative number as array size?

No, if we try to give the negative number as size of an array it will throw run time error NegativeArraySizeException

Q. Can you change the size of array once you define it?

No, we cannot modify the size of array once define

Q. Can we assign the different size of array to each other

Yes we can provided array type must be same

Eg.

Int a[] = new int[100]

Int b[] = new int[10]

a=b;//it is valid

Q. How do you check the equality of two array in java?

Arrays.equals() method to compare one dimensional and arrays.deepEquals() to multidimensional arrays

<https://www.geeksforgeeks.org/compare-two-arrays-java/>

Q. How do you Sort Array elements in java?

Using Arrays.sort() method , this method internally uses quick sort algorithm to sort array elements.

Q. what are the different way of copying array in to another array

By using loop

For(int i: a){a[i] = b[i]}

By Array.copyOf()

Int b[] = Array.copyOf(a,a.length);//we can also define the till what to what

Q. what is the anonymous array in java

If we create an array without reference in java called as anonymous array in java.

New[]{10,20,30,40,50};

public class AnonymousArray {

   public static void arrayToUpperCase(String [] array) {

      for(int i=0; i< array.length; i++) {

         char[] ch = array[i].toCharArray();

         for(int j=0; j<ch.length; j++){

            ch[j] = Character.toUpperCase(ch[j]);

         }

         System.out.println(new String(ch));

      }

   }

   public static void main(String args[]) {

      arrayToUpperCase(new String[] {"Krishna", "Vishnu", "Dhana", "Rupa", "Raja", "Kavya"});

   }

}

Q. what is the ArrayIndexOutOfBoundsException in java and when it occur?

An array is a data structure/container/object that stores a fixed-size sequential collection of elements of the same type. Whenever you used an –ve value or, the value greater than or equal to the size of the array, then the **ArrayIndexOutOfBoundsException** is thrown.

Q. What is the ArrayStoreException in JAVA and when you will get this

exceptionArrayStoreException in Java occurs whenever an attempt is made to store the wrong type of object into an array of objects. The ArrayStoreException is a class which extends RuntimeException, which means that it is an exception thrown at the runtime.

Ex.

public class ArrayStoreException {

    public static void main(String args[])

    {

        // Since Double class extends Number class

        // only Double type numbers

        // can be stored in this array

        Number[] a = new Double[2];

        // Trying to store an integer value

        // in this Double type array

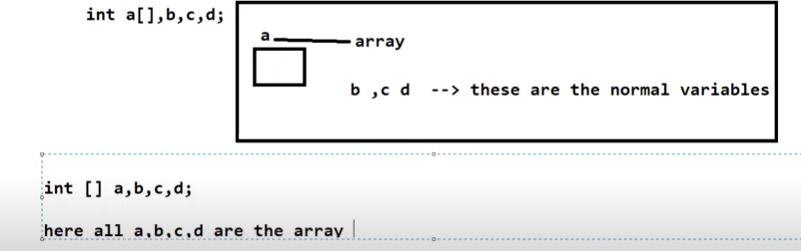
        a[0] = new Integer(4);

    }

}

Q. what is the diff between a[] and []a

Both are legal declaration technique of array in java. But the difference Is



Q. what is the diff between array and ArrayList?

**ArrayList** is part of collection framework in Java. Therefore **array** members are accessed using [], while **ArrayList** has a set of methods to access elements and modify them. **Array** is a fixed size data structure while **ArrayList** is not. One need not to mention the size of **Arraylist** while creating its object

<https://www.geeksforgeeks.org/array-vs-arraylist-in-java/#:~:text=ArrayList%20is%20part%20of%20collection,access%20elements%20and%20modify%20them.&text=Array%20is%20a%20fixed%20size,Arraylist%20while%20creating%20its%20object>.

Q. what value does array elements get if they are not initialized

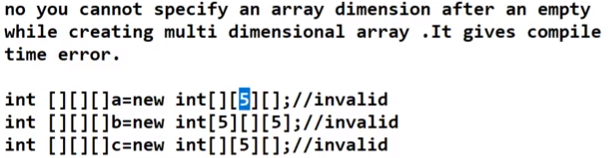
int a[] = new int[5]

by default it stores 0 at each index with the type of its declaration

Q. while creating the multi dimensional array can you specify an array dimension after empty dimension

No,

Invalid declarations



Q. what is the encapsulation and why use it

The idea behind the encapsulation is to hide the implementation details. It describes the idea of bundling data and methods that work on that data within one unit, e.g., a class in Java

To make the data secure (all the variable as private) and hence variable can not be accessed outside of the class using variable but we can access using object/function

Q. What is the static Block in java

A static block of code inside of java class that will be executed when class is first time loaded in to jvm , most static block will be used for initialize the variable

Static block is executed before main method as well as static block executed before constructor also

class Test {

    static int i;

    int j;

    // start of static block

    static {

        i = 10;

        System.out.println("static block called ");

    }

    // end of static block

}

class Main {

    public static void main(String args[]) {

        // Although we don't have an object of Test, static block is

        // called because i is being accessed in following statement.

        System.out.println(Test.i);

    }

}

Non static block :

Whenever an object is created, a **non-static block** will be executed **before the execution of the constructor. There is no protype defined**

**Ex.**

public class NonStaticBlockTest {

   {

      System.out.println("First Non-Static Block"); **// first non-static block**

   }

   {

      System.out.println("Second Non-Static Block"); **// second non-static block**

   }

   {

      System.out.println("Third Non-Static Block"); **// third non-static block**

   }

   NonStaticBlockTest() {

      System.out.println("Execution of a Constructor"); **// Constructor**

   }

   public static void main(String args[]) {

      NonStaticBlockTest nsbt1 = new NonStaticBlockTest();

      NonStaticBlockTest nsbt2 = new NonStaticBlockTest();

   }

}

## Output

First Non-Static Block

Second Non-Static Block

Third Non-Static Block

Execution of a Constructor

First Non-Static Block

Second Non-Static Block

Third Non-Static Block

Execution of a Constructor

Non static block also executes before constructor but on every new object.

Q. what is the static keyword

Static is non access specifier in java

Static can be applied to a variable, method and initialization block.

Static method.

It belongs to class rather than object it can be directly call by using a class name

Static class:

Note: in java only nested classes are allowed to be declared as a static, a top level class cannot be declared as static in java.

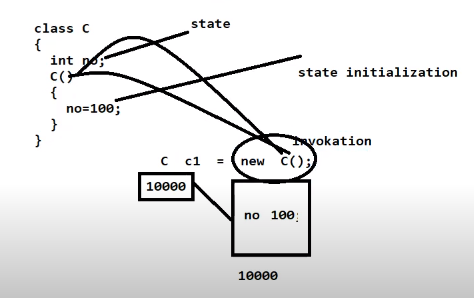
Q. Can we override static method

**We can** declare **static methods** with the same signature in the subclass, but it is not considered **overriding** as there won't **be** any run-time polymorphism. Hence the answer is 'No'.

Where as we can always overload the static method.

Q. What is a constructor?

It is a function same name as class name but without return type it is used for initialize the state of an object and will invoke during the time of object creation



Q. What are the rules of defining constructor?

1. Constructor name should be same as class name
2. No return type of constructor (if return type is given it will considered method)
3. If no constructor is defined it will create by default constructor without any parameter
4. It can have all access modifiers are allowed (private, public, protected , default)
5. It cannot have non access modifier (final, static, abstract, synchronized)
6. No return statement allowed
7. It can take any number of arguments
8. Constructor can throw exception; we can have throws class with constructor

Q. can constructor return any value

No , can’t return any value explicitly but it will be returning the instance of class internally

Q. can we have constructor in an interface

No, you cannot have a constructor within an interface in Java. Because it does not allow any method without return type

Q. can we have this and super constructor in same constructor

No, Because if you use this() and super() together in a constructor it will give compile time error. Because this() and super() must be the first executable statement. If you write this() first than super() will become the second statement and vice-versa. That's why we can't use this() and super() together.

Q. can abstract class in java have a constructor

Yes , an abstract class can have a constructor in java

Q. what is the constructor chaining in java

Constructor chaining is nothing but calling one constructor from another. This() constructor is used for performing constructor chaining in java

Q. What is the use of private constructor in java?

When we use the private constructor then object for the class can only be created internally in class no outside of class can created

Private constructor normally use when creating the singleton class in java

Q. abstract class

<https://medium.com/@codebyamir/when-to-use-an-abstract-class-in-java-c7697f664b3a>

Q. What is the use of final keyword in java

It is non access specifier in java , we can use with a variable, method and clss

Final variable means a variable cannot modify its value once we assign it means final variable is a constant in java

Final method means a method cannot override in child class :: method overriding means parent and child both class have same method name

Final class means a class cannot inherit in any another class.

Q. can a main method be declared as final?

Yes the main method can be declared as final and cannot be overridden

Q. can we declare constructor as final ?

No, since it is neither inherited nor overridden so there is no use to have final constructor, you will get illegal modifier only public, protected and private are permitted.

Q. can we make an abstract method final in java?

No, because both are opposite to each, abstract methods must be overridden where as final key word does not allow you to override a method.

Q. can we make local variable as final?

Yes, in fact it is the only access modifier is allowed to local variable. Other modifier will give the compilation error

Q. what is the difference between abstract class and final class?

Abstract class can be sub class final class cannot have sub class

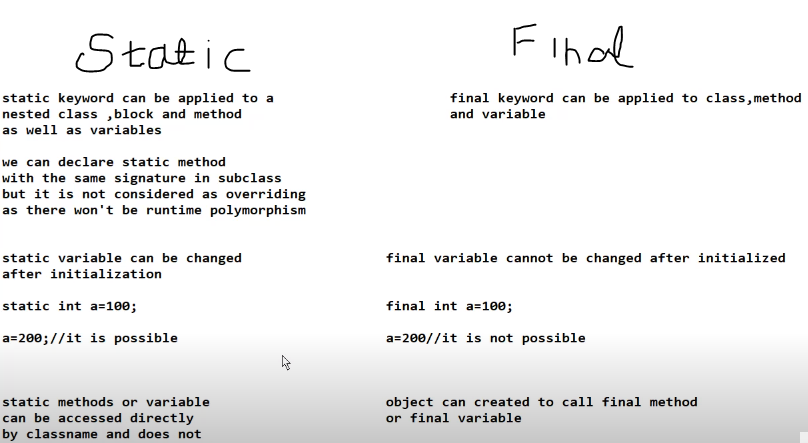
Abstract method should be overridden final method cannot be overridden

Can contain abstract method cannot contain abstract method

Abstract class can be instantiated final class can be instantiated

Immutable object cannot be created immutable object can be created

Q. Static vs Final



Q. What is static final variable in java

When we declare a variable as static final then the variable as static final then the variable become a strictly constant only one copy of variable exist which can not be change by any instance of class.

Q. what will happen if we add final to a List or ArrayList or any collection

This means that you cannot rebind the variable to point to a different *collection instance*:

final List<Integer> list = new ArrayList<Integer>();

list = new ArrayList<Integer>(); // Since `list' is final, this won't compile

I still can add to ArrayList new elements, remove elements and update it.

Q. What is the difference abstraction and encapsulation principles of object orientation?

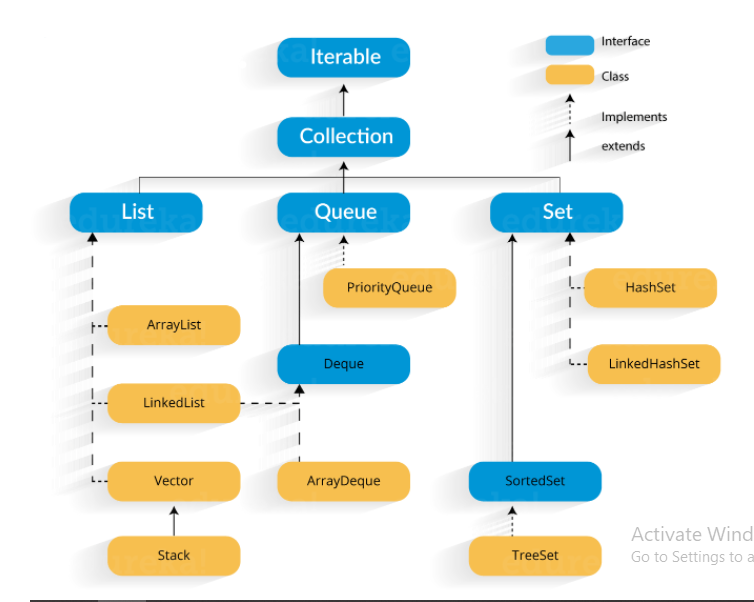
Abstraction solves problem at design level while Encapsulation solves problem at implementation level. Abstraction hides the irrelevant details found in the code whereas Encapsulation helps developers to organize the entire code easily.

Abstraction is the method of hiding the unwanted information. Whereas encapsulation is a method to hide the data in a single entity or unit along with a method to protect information from outside.

<https://www.geeksforgeeks.org/difference-between-abstraction-and-encapsulation-in-java-with-examples/>

Q. What is Collection?

Collection is ready implementation of dsa provided by java to us , it provides variables classes such as arrayList , vector , stack , linked list etc. and interfaces like as list , set , queue etc



Collection framework is a combination of classes and interfaces which is used to store and manipulate data in the form of objects as well as it is used for store the heterogenous data or different type of data.

Q. List vs set

Q. Difference between an iterator and list iterator

<https://www.geeksforgeeks.org/difference-between-an-iterator-and-listiterator-in-java/>

Q. what is the diff between HashSet and TreeSet

HashSet implements set interface where as TreeSet implements Sorted set

<https://www.tutorialspoint.com/difference-between-tree-set-and-hash-set-in-java#:~:text=Hash%20set%20and%20tree%20set,is%20backed%20by%20a%20hashmap.&text=The%20tree%20set%20does%20not%20allow%20the%20null%20object>

Q. What is the difference between set and Map

1) Set contains only values and map contains the key and value. Key is used to insert the value.Key is used for insert the value in map , retrive value from map, remove value from map etc.

2) Set collection store the unique values and map store the unique keys and values may be duplicated.

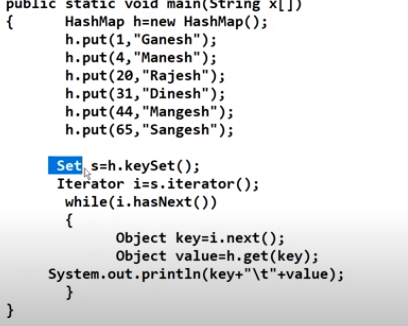
3) set can store single null value and map can store single null key and more than one null values

Q. HashSet vs HashMap

Hashset can generate the values randomly and HashMap can generate the keys randomly

Hashset implemetns a set interface nad hashmap implements map interface

Set can direct iterator through the iterator bu map need to convert it to set by using keyset() method for traveling purpose



Q. What is the diff between HashMap and TreeMap

1) HashMap not maintain the key sequence generate the keys randomly and TreeMap can arrange the all keys in ascending order.

2) hashmap implement the Map and treemap implement the tree structure

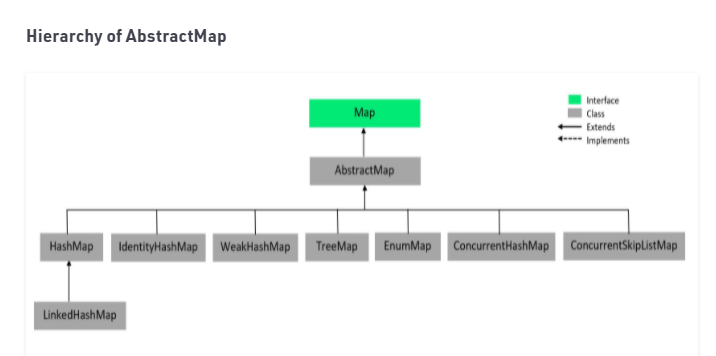
3) hashmap can store null keys where as no null keys allowed in tree map but null values are allowed

Q. What is the diff between hashMap and HashTable

1) hashMap is not a synchronized and hashtable is synchronized(lock)

2) HashMap can store single null key and multiple null values , hashtable not store null key as well as null value.

3) hashMap inherit the AbstractMap class and hashTable inherit the dictionary class



Q. What is the diff between collection and collections

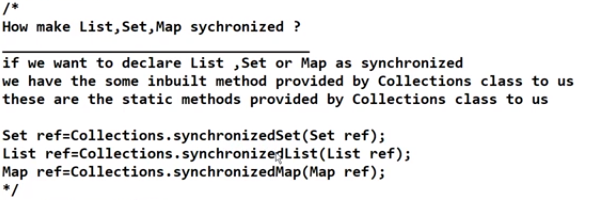
Collectiion is a interface and collections is class from java.util.pakcage

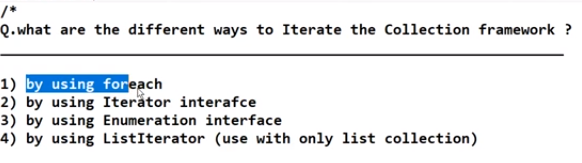
2) collection interface provides DS such as list,set ,queue etc…

But collections class contain the some static methods which help us to perform operation on collection like sort the collection m find minimum element from a collection, find disjoin element in collection etc purpose

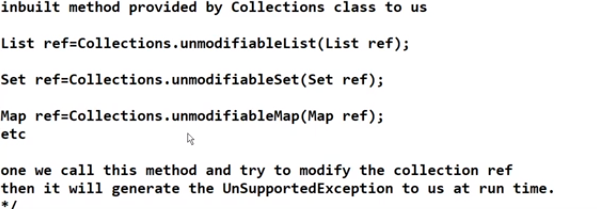
Collections.sort(collection): this method can sort the any collection in ascending order.

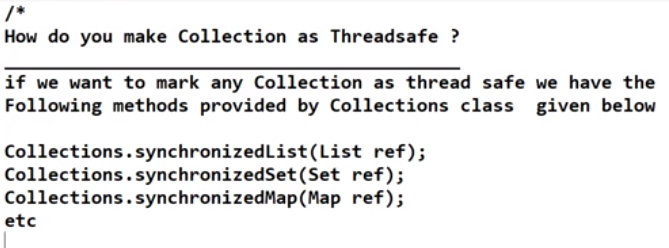
Collections.max(al): maximum element of the collection will be return.

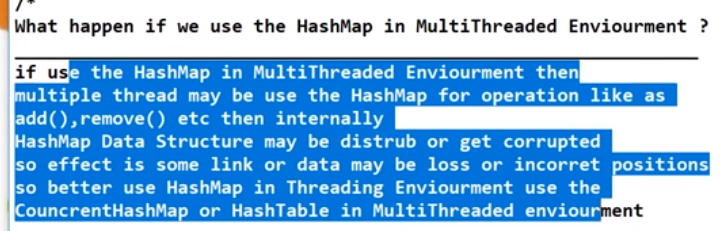


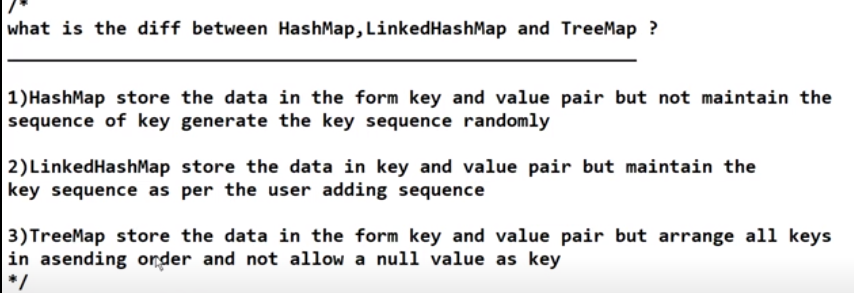


Q. How To Make Collection Read Only









Q. Generic Notations in Java?

Generic notations are the some standard to use with generic classes, interfaces, collection and type

Some of the nations are bellow

E - element

K - generic key type

V - value

T - Type, user defined types

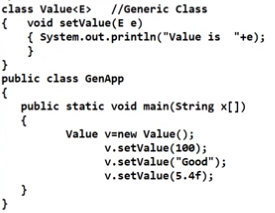
N - N stands number with Generics

Q. Types of Generic(generic class is something that can be used by anyone by providing their own data types

How to create the user define class as Generic class or what is the class with generics in java?

* User can create its own class as generic notations like as E,T

Example:



While creating an object we can also use value<Integer> v = new value<>()

See internal implementation of arrayList class

Q. wild cards

Bounded(upper bounded, lower bounded) and unbounded()

<https://www.geeksforgeeks.org/wildcards-in-java/#:~:text=The%20question%20mark%20(%3F)%20is,sometimes%20as%20a%20return%20type>.

Q. What is String pool constant?

Reference resides in stack memory where as value resides in heap memory

<https://www.geeksforgeeks.org/string-constant-pool-in-java/#:~:text=A%20string%20constant%20pool%20is,is%20created%20in%20the%20heap>.

Q.



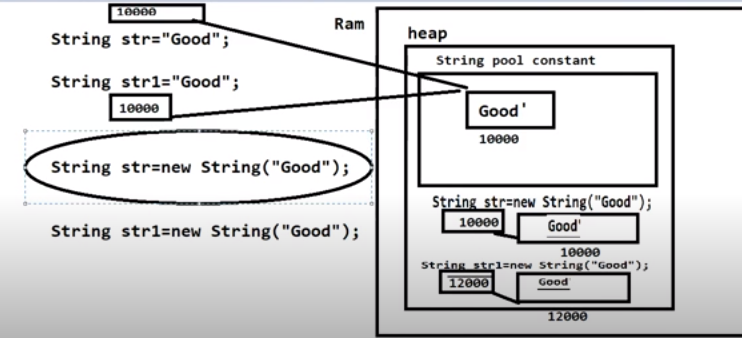
<https://www.geeksforgeeks.org/final-vs-immutability-java/#:~:text=In%20Java%2C%20we%20know%20that,to%20the%20existing%20String%20objects.&text=final%20means%20that%20you%20can,(using%20setter%20methods%20e.g)>.

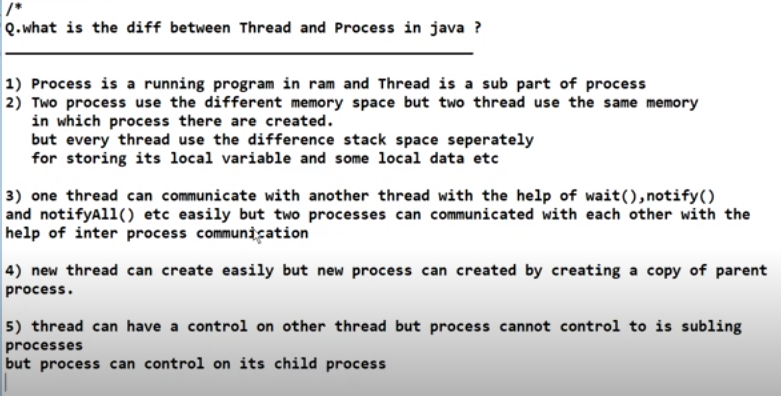
Q. what is intern string?

<https://www.geeksforgeeks.org/interning-of-string/>

Q. how many ways are there to create object in string

Using new key word and using string literal (stores in scp memory)

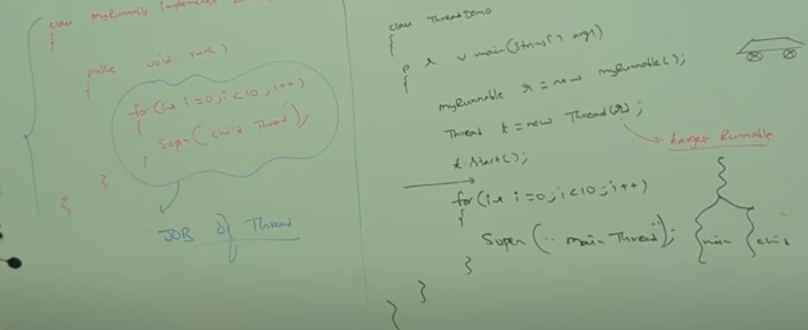




Q. how many ways we can define a Thread

2 ways.

1. Mythread extends Thread and thread implements Runnable internally
2. MyRunnable Implements Runnable interface



But which one is better to use ? : Implements Runnable

When you extends Thread class, after that you can’t extend any other class which you required. (As you know, Java does not allow inheriting more than one class).

When you implements Runnable, you can save space for your class to extend any other class in the future or now.

Thread schedule is responsible for scheduling the threads and it resides in JVM.

And the order of threads decided by scheduler. And behaviour of scheduler is dependent on JVM. Different jvm follows different algorithms like First Come First Serve (FCFS), 2) Shortest-Job-First (SJF) **Scheduling** 3) Shortest Remaining Time 4) Priority **Scheduling** 5) Round Robin **Scheduling** 6) Multilevel Queue **Scheduling**. ... In, Priority **Scheduling** the **scheduler** selects the tasks to work as per the priority.

What is MultiTasking?

Doing multiple task at a time is known as multitask.

Example: A student can listen, write, observe at a time.

1. Process Based Multitasking(each task is separate independent process)
   * It is OS level task, example: listening and downloading songs at a time.

1. Thread Based Multitasking
   * It is known as programmatically multitasking. Executing separate independent task(thread) known as thread based multitasking.

Q. Thread Life Cycle?

<https://www.journaldev.com/1044/thread-life-cycle-in-java-thread-states-in-java>