Java Interview Question came across in recordings by Medha :

1.What is Java?

------------------------------------------------------------------------------------------

**Programs link:**

<http://javarevisited.blogspot.sg/2011/06/top-programming-interview-questions.html>

<http://www.tutorialspoint.com/javaexamples/>

<https://www.javatpoint.com/java-tutorial>

JAVA:

## What is Java

Java is a **programming language** and a **platform**.

Java is a high level, robust, secured and object-oriented programming language.

**Platform**: Any hardware or software environment in which a program runs, is known as a platform. Since Java has its own runtime environment (JRE) and API, it is called platform.

## Types of Java Applications

There are mainly 4 type of applications that can be created using java programming:

#### 1) Standalone Application

It is also known as desktop application or window-based application. An application that we need to install on every machine such as media player, antivirus etc. AWT and Swing are used in java for creating standalone applications.

#### 2) Web Application

An application that runs on the server side and creates dynamic page, is called web application. Currently, servlet, jsp, struts, jsf etc. technologies are used for creating web applications in java.

#### 3) Enterprise Application

An application that is distributed in nature, such as banking applications etc. It has the advantage of high level security, load balancing and clustering. In java, EJB is used for creating enterprise applications.

#### 4) Mobile Application

An application that is created for mobile devices. Currently Android and Java ME are used for creating mobile applications.

Java History :

 Originally developed by James Gosling at Sun Microsystems (which is now a subsidiary of Oracle Corporation) and released in 1995.

11) In 1995, Time magazine called **Java one of the Ten Best Products of 1995**.

12) JDK 1.0 released in(January 23, 1996).

There are many java versions that has been released. Current stable release of Java is Java SE 8.

# Features of Java

There is given many features of java. They are also known as java buzzwords. The Java Features given below are simple and easy to understand.

1. Simple: Java language is simple because, Java eliminates the pointers concepts which are used in C and C++ for memory allocations and declarations. Java Memory management taken care by JVM(java virtual Machine).
2. Object-Oriented: Object-oriented means we organize our software as a combination of different types of objects that incorporates both data and behaviour. That means everything is dealt in objects and classes.
3. Portable: We may carry the java bytecode to any platform.

Platform independent: A platform is the hardware or software environment in which a program runs.

There are two types of platforms software-based and hardware-based. Java provides software-based platform. It has two components:

1. Runtime Environment
2. API(Application Programming Interface)

Java code can be run on multiple platforms e.g. Windows, Linux, Sun Solaris, Mac/OS etc. Java code is compiled by the compiler and converted into bytecode. This bytecode is a platform-independent code because it can be run on multiple platforms i.e. Write Once and Run Anywhere(WORA).

java is platform independent

Secured: Java is secured because:

* **No explicit pointer**
* **Java Programs run inside virtual machine sandbox**

how java is secured

1. Robust: Robust simply means strong. Java uses strong memory management. There are lack of pointers that avoids security problem. There is automatic garbage collection in java. There is exception handling and type checking mechanism in java. All these points makes java robust.
2. Architecture neutral: There is no implementation dependent features e.g. size of primitive types is fixed. In java, DataTypes occupies 4 bytes of memory for both 32 and 64 bit architectures.
3. Dynamic: makes dynamic interactions possible on web applications.
4. Interpreted: Hello.java -compiles---.>hello.class(bytecode)--Interprets-🡪machine readable lang.code
5. High Performance: it handles complex logics execution using JIT compiler(JustInTime). Java is faster than traditional interpretation since byte code is "close" to native code still somewhat slower than a compiled language (e.g., C++)
6. Multithreaded: A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it doesn't occupy memory for each thread. It shares a common memory area. Threads are important for multi-media, Web applications etc.
7. Distributed: It works with various systems on various networks and can handle any network protocols(Tcp/ip,udp.ftp etc..)
8. System Independent: java byte code can be executed on any processor or OS.
9. Scalability: java has specific Api’s. it can connect to DB,web Services and any remote calls using its api packages.

Java Features

For executing any java program, you need to

* install the JDK if you don't have installed it, [download the JDK](http://www.oracle.com/technetwork/java/javase/downloads/index.html) and install it.
* set path of the jdk/bin directory. <http://www.javatpoint.com/how-to-set-path-in-java>
* create the java program
* compile and run the java program

## Understanding first java program

Let's see what is the meaning of class, public, static, void, main, String[], System.out.println().

* **class** keyword is used to declare a class in java.
* **public** keyword is an access modifier which represents visibility, it means it is visible to all.
* **static** is a keyword, if we declare any method as static, it is known as static method. The core advantage of static method is that there is no need to create object to invoke the static method. The main method is executed by the JVM, so it doesn't require to create object to invoke the main method. So it saves memory.
* **void** is the return type of the method, it means it doesn't return any value.
* **main** represents startup of the program.
* **String[] args** is used for command line argument.
* **System.out.println()** is used print statement.

## Valid java main method signature

1. **public** **static** **void** main(String[] args)
2. **public** **static** **void** main(String []args)
3. **public** **static** **void** main(String args[])
4. **public** **static** **void** main(String... args)
5. **static** **public** **void** main(String[] args)
6. **public** **static** **final** **void** main(String[] args)
7. **final** **public** **static** **void** main(String[] args)
8. **final** **strictfp** **public** **static** **void** main(String[] args)

## Invalid java main method signature

1. **public** **void** main(String[] args)
2. **static** **void** main(String[] args)
3. **public** **void** **static** main(String[] args)
4. **abstract** **public** **static** **void** main(String[] args)

## What happens at compile time?

At compile time, java file is compiled by Java Compiler (It does not interact with OS) and converts the java code into bytecode.

compilation of simple java program 

## What happens at runtime?

|  |
| --- |
| At runtime, following steps are performed: |
| what happens at runtime when simple java program runs |

|  |
| --- |
| **Classloader:**is the subsystem of JVM that is used to load class files. |
| **Bytecode Verifier:**checks the code fragments for illegal code that can violate access right to objects. |
| **Interpreter:**read bytecode stream then execute the instructions. |

### Q)Can you save a java source file by other name than the class name?

|  |
| --- |
| Yes, if the class is not public. It is explained in the figure given below: |
| how to save simple java program by another name |

|  |  |
| --- | --- |
| **To compile:** | javac Hard.java |
| **To execute:** | java Simple |

### Q)Can you have multiple classes in a java source file?

|  |
| --- |
| Yes, like the figure given below illustrates: |
| how to contain multiple class in simple java program |

## How to set Permanent Path of JDK in Windows

For setting the permanent path of JDK, you need to follow these steps:

* Go to MyComputer properties -> advanced tab -> environment variables -> new tab of user variable -> write path in variable name -> write path of bin folder in variable value -> ok -> ok -> ok

**Program: set of instructions to perform a particular task.**

**Compiler**: Machine can only understand binary language(0 and 1’s) ….compiler converts the high level language to machine understandable language all at once.

Example of Compiler language: C, C++, Scala

**Interpreter**: Converts the each line of high level language to machine level language line by line while it is executing.

Ex. Ruby, python Etc.

**Java**: is called as a platform and language..it follows WORA principle..Wrire Once run Anywhere….the compiled programs can be run on any other system also…like on windows , mac IBM…it’s a platform independent language.

Benefits of OOPS?

OOPS principles ?

JDK,JRE,JVM

**JIT Compiler** : Just In Time Compiler : when it is used ?

Garbage Collector

Access Modifiers : Public,Private,protected and Defualt

Global or Instance variables

**Constuctors**: A **constructor in Java** is a block of code similar to a method that's called when an instance of an object is created. Here are the key differences between a **constructor** and a method: A **constructor** doesn't have a return type. The name of the **constructor** must be the same as the name of the class.

Why we need constructors :

Memory management : Stack and Heap

What for stack

What is Encapsulation ?

What is this ? current reference

Inheritance :  is a mechanism in which one object acquires all the properties and behaviors of parent object.

Inheritance represents the **IS-A relationship**, also known as *parent-child* relationship.

**Inheritance in java** is a mechanism in which one object acquires all the properties and behaviors of parent object. The idea behind **inheritance in java** is that you can create new classes that are built upon existing classes.

**Note** − A subclass inherits all the members (fields, methods, and nested classes) from its superclass. Constructors are not members, so they are not inherited by subclasses, but the constructor of the superclass can be invoked from the subclass.

## The super keyword

The **super** keyword is similar to **this** keyword. Following are the scenarios where the super keyword is used.

* It is used to **differentiate the members** of superclass from the members of subclass, if they have same names.
* It is used to **invoke the superclass** constructor from subclass.

### Differentiating the Members

If a class is inheriting the properties of another class. And if the members of the superclass have the names same as the sub class, to differentiate these variables we use super keyword as shown below.

super.variable

super.method();

Why use inheritance in java

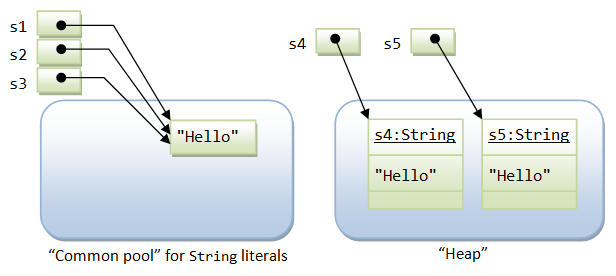
* For Method Overriding (so runtime polymorphism can be achieved).
* For Code Reusability.

Inheritance is a [**compile-time**](http://beginnersbook.com/2013/04/runtime-compile-time-polymorphism/) mechanism. A super-class can have any number of subclasses. But a subclass can have only one superclass. This is because Java does not support multiple inheritance.

What is type caseting : implicit and explicit

Why string class is Immutable ? what is immutable ? What is String Constant pool? How the objects are stored.

Where is this string constant pool exist ?



 As the name suggests, String Pool in java is a pool of Strings stored in Java Heap Memory.

 When we use double quotes to create a String, it first looks for String with same value in the String pool, if found it just returns the reference else it creates a new String in the pool and then returns the reference.

String pool helps in saving a lot of space for Java Runtime although it takes more time to create the String.

**String Buffer and String Builder** : mutable string classes : The String class is immutable, so that once it is created a String object cannot be changed. If there is a necessity to make a lot of modifications to Strings of characters, then you should use [String Buffer & String Builder](https://www.tutorialspoint.com/java/java_string_buffer.htm)Classes.

Only one method for buffer and builder…

Object is the root class in java hierarchy.

**Wrapper classes:**

All the wrapper classes (**Integer**, Long, Byte, Double, Float, Short) are subclasses of the abstract class Number. The object of the wrapper class contains or wraps its respective primitive data type. Converting primitive data types into object is called boxing, and this is taken care by the compiler.

Collections :

Collections in java is a Framework that provides an architecture to store and manipulate the group of objects.

Collection is single unit of objects.

Java Collections framework provides many Interfaces(LINK,SET,QUEUE,DEQUE, etc), Classes (ArrayList,Vector,LinkedList,HashSet,LinkedHastSet,TreeSet,PriorityQuesue) and Algorithms.

List Interface

1.Arraylist

2. Vector Class

3. LinkedList Class

|  |  |  |
| --- | --- | --- |
| LIST Interface | SET Interface | Queue Interface |
| ArrayList Class | HashSet Class | Prority Queue Class |
| Vector Class | Linked HashSet |  |
| LinkedList Class | Tree Set |  |
|  |  |  |

Creating the List Interface

Syntax :

List <dataType> list\_name = new I implementation ClassName<datatype>();

Ex. List<String> ls = new ArrayList<String>();

(Object we create for implementation class , so first we create for arraylist class). Need to import java.util package in java ctrl+Shift+O shortcut to importing.

When you are specifying the data type means you are making the list generic (similar kind of data is exist in the list)..new feature added in java

To input any data into collections we use add method.ex.ls.add(“name”);

We don’t create any size for collections….the size dynamically grows when we add the data.we can iterate the data using normal for loop or enhanced for loop

|  |  |  |
| --- | --- | --- |
| ArrayList | Vector | LinkedList |
| 1. Arraylist is faster in Iteration through the Random Access Interface. 2. Arraylist follows Order by Insertion 3. ArrayList grows dynamically. 4. Contains diff methods. 5. Duplicates values are allowed 6. NULL value is allowed 7. Because of indexing position Arraylist isprefered for searching and Iterations. 8. Implemets LIST Interface | 1. In Vectors ,we can give the initial capacity and incremental Capacity . 2. If Capacity is not specified it takes default capacity as 10. 3. Vector has Synchronized methods which makes them slower compared to ArrayList..preferable in multithread Environments.   now adays not used much in multithread env. Also implementing by adding extra logic to arrays only.   1. Vector also   follows Order by insertion only. | 1. Each node is linked to one another. Each elements share the relation.Every elements aware of the other elements previous node and ending node.double linking. 2. Inserting and deletion operations are easier in Linkedlist are faster. 3. Duplicates and null values allowed. 4. Follows insertion order. |

|  |  |  |
| --- | --- | --- |
| HashtSet | LinkedHashSet | TreeSet |
| 1. Allows Unique values, duplicate values are not allowed 2. Hashset is unordered and unsorted retrieval 3. Null is allowed only once 4. Data is retrieved by hashcode 5. We can remove by object…only one method don’t have overloaded methods 6. Hashcode wont follow index. 7. Hashset based on hashcode which is also called as hashing algorithm | 1. Will not allows duplicate values, unique values are allowed. 2. Linkedset is follows insertion order only | 1. Will not allow duplicate values, only unique values are allowed 2. Data is sorted in natural ascending/descending and ordered. 3. It sorts the data according to red black tree data structure. 4. Single NULL is also not allowed. |

Set will not allow duplicated. when we create user defined object type collection set ,then in memory for each object it allocates diff . location.

For string and wrapper classes jvm internally overrides the methods

Where as for customer defined object collectons we need to write override methods manually for hashcode and equals methods else it couldn’t be recognized .

**Vector** : vector methods are synchronized..hence it is slow compare dto ArrayList.

**Map** : Interface

In map is it stores data in key value pairs

Key and value can be of anything object ,string or any wrapper class

Key should be unique can’t be duplicated and Value can have the duplicate data. If the duplicated Key value is given , then the new value overrides the existing key value and give only one key.

We can iterate over themap data either by **using Entry Set or Key Set** methods.

Under hashmap specifically do not follow and order or sorting.

Allows one null key and multiple null values.

Data can be added using **put** method instead of add method.

For retrieving iteration is different….

|  |  |  |  |
| --- | --- | --- | --- |
| HashMap | HashTable | LinkedHashMap | TreeMap |
| 1. Data is unordered or unsorted. 2. One null key can be added and in value can have multiple null. 3. Hashmap internally uses hashCode. 4. In case if adding any user defined object as the Key then equals and hashCode methods should be overridden |  |  |  |

List

Arraylist

Vector :

Initial and incremental capacity can be

**List- is an interface:**

**They follow insertion order, they allow duplicates and null**

**ArrayList- arraylist is faster for iterations and search criteria**

**LinkedList- implements both List and Queue, faster for adding and removing elements**

**Vector- similar to arraylist but methods are synchronized and we can give the capacity and incrementor which saves memory.**

List :will allow duplicate values

List allows null values. list follows insertion order

Set : will not allow duplicate values

Set allows only one NULL value

Hashset : unordered and unsorted

Linkedhashset :

TestNG:

Goto eclispse market place..search for eclipse plugin—

Project – right click – java buildpath—add library

Cant for loop is used for retrieving the data in linked hash set how to get them

Features of Java:

Java is based on WORA ( Write Once Run Anywhere) principle.

Java is Platform independent language. Ex.Java compiled program of windows OS can run on MAC OS…that means .class file can be interpreted on anyother systems with jvm instaleed in it.

Java is simple language

Distributed Language :

Secured and system independent

Portable

Interpreted

Multithreded

Scalability

Java Arrays Concepts :

Arrays : Collection of elements of the same type wit fixed size.

Three types of Arrays :

1.Single dimensional array

2.Multi dimensional array

3.jogged array

Declaration :

1.int [] a = new int[5];

Int num,res ;

Res = 153