What is jdk ?

A Java Development Kit (JDK) is a program development environment for writing Java applets and applications.

|  |
| --- |
| What jdk contains or composed of or components comes with and define each components? |
| Answer: |
| Java Compiler- Compiles Java File |
| Java Interpreter- Interpret the compiled Java File |
| Java Disassembler- Disassembles Java Class files |
| Java Header File Generator-to generate C language header file and source files to implement native methods. |
| Java Documentation-Used for easy maintenance of code. |
| Java Debugger-Used to debug java files |
| Java Applet Viewer-Used to view java applet |
| What is IDE? |
| A Java IDE (Integrated Development Environment) is a software application which enables users to more easily write and debug java program.  What are the IDEs available in the market to support java? Answer: There are many IDEs in the market, but most popular ones: Eclipse, NetBeans, IntelliJ IDEA, JetBrains. Jcreator and etc. |
| Explain the entire java life cycle. |
| Answer: |
| 1) Write source code using editor or IDE such as Eclipse. |
| 2) Compile source code to bytecode using Java Compiler (javac.exe), and get Classname.class file. |
| 3) JVM load (by java.exe) and verify this bytecode (class) file. |
| 4) The Execution Engine in JVM presents the result on the operating system. |
| what is class? |
| Answer: Class is a code template that creates for objects. |
| Exactly, a class is an extensible program-code-template for creating objects, providing initial values for state or member variables, and implementations of member functions/methods. |
| What is object? |
|  |
| Object is an instantiated instance of the class. |
|  |
| What is the entry point of an application? |
|  |
| An entry point is where control enters a program or piece of code. In many programming languages, the main function is where |
| a program starts its execution. |

Why main is static?

The public **static** void keywords mean the **Java** virtual machine (JVM) interpreter can call the program's **main** method to start the program (public) without creating an instance of the class (**static**), and the program does not return data to the **Java** VM interpreter (void) when it ends.

public static void main(String[] args){

...

}

|  |
| --- |
| What is the difference between an Interface and Abstract class? |
| Answer:  Main difference is methods of a Java interface are implicitly abstract and cannot have implementations. A Java abstract class can have instance methods that implements a default behavior. |
| Interface: only can declare method. |
| Abstract Class: can declare method, can implement method. |
| Can you make an Object from an Interface and Abstract class ? if not how do you use it ? |
| Answer: We can't create object out of an interface or Abstract class because main intention of creating an object is to utilize the wrapped methods and data. As interface don't have any concrete implementation hence we cannot. |
|  |
|  |
|  |
|  |
|  |
|  |
| what is Access Specifier? |
| What is OOP ? Define each feature of java OOP. |
| OOP-object oriented programming. |
|  |
| Polymorphism: Polymorphism is the ability of an object to take on many forms. |
|  |
| Encapsulation: Encapsulation can be described as a protective barrier that prevents the code and data being randomly accessed by other code defined outside the class. Access to the data |
| and code is tightly controlled by an interface. |
|  |
| Inheritance: Inheritance is a mechanism wherein a new class is derived from an existing class. |
|  |
| Abstraction - Abstraction is the process of abstraction in Java is used to hide certain details and only show the essential features of the object. |
|  |
|  |
| What is API? |
| Application Programming Interface. |
| Does java support multiple inheritance? |
| Answer: Yes, it supports multiple inheritance of type, which is the ability of a class to implement more than one interface. |
| What is method overloading and when it happens? |
|  |
| Method overloading deals with the notion of having two or more methods(functions) in the same class with the same name but different arguments. Overloading happens at compile time. |
|  |
|  |
| What is method overriding and when it happens? |
|  |
| Method overriding means having two methods with the same arguments, but different implementation. One of them would exist in the Parent class (Base Class) while another will be in the derived class(Child Class).Overriding happens at the run time. |
| Explain exceptions in java and how to handle it. |
| Answer: An exception is an event that occurs during the execution of a program that disrupts the normal flow of instructions. We can use the three exception handler components - the try, catch, and finally blocks - to write an exception handler. |
|  |
|  |
| What is static keyword in java? How it has been used in variables and methods? |
| Answer: The static keyword in Java means that the variable or function is shared between all instances of that class as it belongs to the type, not the actual objects themselves. |
| Static variable belongs to class, not belong to object. |
| What is final and how it has been used variables and methods? |
|  |
| Finalizing the value of the variable so that no change can be made. |
| A final class cannot be subclassed.All methods in a final class are implicity final. |
| A final method cannot be overridden or hidden by subclass. |
|  |
| Static Variable - Declaring any variable as static, it is known static variable. The static variable can be used to refer the |
| common property of all objects (that is not unique for each object) e.g. company name of employees,college name of students etc. |
| The static variable gets memory only once in class area at the time of class loading. It makes your program memory efficient. |
|  |
| Static Method - A static method belongs to the class rather than object of a class. A static method can be invoked without the |
| need for creating an instance of a class.static method can access static data member and can change the value of it. |
| What is final, finally and finalize? |
|  |
| Final: final is a keyword. The variable declared as final should be initialized only once and cannot be changed. Java classes declared as final cannot be extended. Methods declared as final cannot be overridden.  Finally: finally is a block. The finally block always executes when the try block exits This ensures that the finally block is executed even if an unexpected exception occurs But finally is useful for more than just exception handling - it allows the programmer to avoid having cleanup code accidentally bypassed by a return, continue, or break. Putting cleanup code in a finally block is always a good practice, even when no exceptions are anticipated.  Finalize: finalize is a method. Before an object is garbage collected, the runtime system calls its finalize() method. We can write system resources release code in finalize() method before getting garbage collected. |
| What is a constructor ? |
| A constructor is a bit of code that allows you to create objects from a class |
|  |
| What is "this" keyword in java ? |
| Within an instance method or a constructor, this is a reference to the current object — the object whose method or constructor |
| is being called. You can refer to any member of the current object from within an instance method or a constructor by using this. |
|  |
| What is JVM stand for ? |
| Java virtual machine. |
|  |
| What version of java are you using? Java SE-1.8 |
|  |
| What is JAR stand for ? |
| JAR stands for "Java Archive". A JAR file is basically a zip-file which contains a couple of class files and a text file containing some meta information. |
|  |
| JAR files can be used to bundle a Java library containing many classes into a single file (which does not need to be unzipped |
| in order to be used by a java programmer) or to bundle a Java application into a so-called executable JAR file (an executable |
| JAR file is just a JAR file whose meta-information contains the information, which class is the main class, i.e. the class |
| whose main method should be run to execute the application). |
|  |
| What is compile time and run time? |
|  |
| A compiled program can be opened and run by a user. When an application is running, it is called runtime.  What is heap? |
| Heap is an area of pre-reserved computer main storage ( memory ) that a program process can use to store data in some variable amount that won't be known until the program is running.   |  | | --- | | How java manage it's memory? | |  | | The JVM divided the memory into following sections. | | 1. Heap: The Heap section contains Objects (may also contain reference variables). | | 2. Stack: The Stack section of memory contains methods, local variables and reference variables | | 3. Code: The code section contains your byte code. | | 4. Static: The Static section contains Static data/methods. | |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |