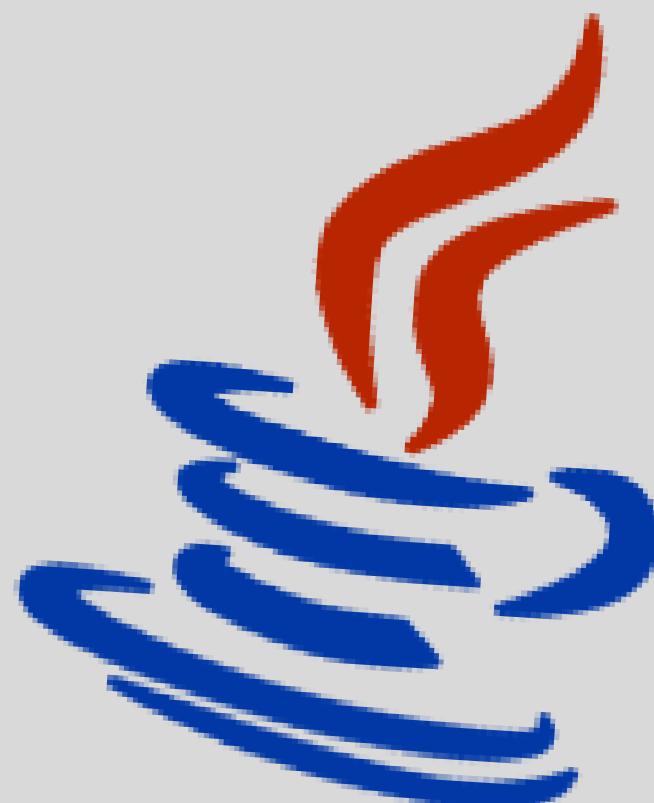


code_buddy_
Ganesh Kokate

100 Java Interview Question 1 to 10 WITH ANSWERS



Q1. Explain JDK, JRE and JVM?

JDK	JRE	JVM
It stands for Java Development Kit.	It stands for Java Runtime Environment.	It stands for Java Virtual Machine.
It is the tool necessary to compile, document and package Java programs.	JRE refers to a runtime environment in which Java bytecode can be executed.	It is an abstract machine. It is a specification that provides a runtime environment in which Java bytecode can be executed.
It contains JRE + development tools.	It's an implementation of the JVM which physically exists.	JVM follows three notations: Specification, Implementation, and Runtime Instance.

Q2. Explain **public static void main(String args[])** in Java.

→ **main()** in Java is the entry point for any Java program. It is always written as **public static void main(String[] args)**.

- **public:** Public is an access modifier, which is used to specify who can access this method. Public means that this Method will be accessible by any Class.
- **static:** It is a keyword in java which identifies it is class-based. main() is made static in Java so that it can be accessed without creating the instance of a Class. In case, main is not made static then the compiler will throw an error as main() is called by the JVM before any objects are made and only static methods can be directly invoked via the class.
- **void:** It is the return type of the method. Void defines the method which will not return any value.
- **main:** It is the name of the method which is searched by JVM as a starting point for an application with a particular signature only. It is the method where the main execution occurs.
- **String args[]:** It is the parameter passed to the main method.



Q9. How is Java different from C++?

→ Java and C++ are both object-oriented programming languages, but they have some key differences.

1. Platform independence: Java is a platform-independent language, while C++ is a platform-dependent language. This means that Java code can run on any platform that has a Java virtual machine (JVM), while C++ code can only run on the platform that it was compiled for.
2. Memory management: Java uses automatic memory management, while C++ requires manual memory management. This means that Java programmers do not need to worry about allocating and freeing memory, while C++ programmers need to be careful to manage memory correctly to avoid memory leaks.
3. Safety: Java is a safer language than C++. This is because Java has built-in security features that help to protect applications from malicious code. For example, Java does not allow direct access to the operating system, which makes it more difficult for attackers to exploit security vulnerabilities.
4. Performance: Java is typically not as fast as C++. This is because Java uses a virtual machine, which adds an extra layer of abstraction between the code and the hardware. However, Java applications are typically more portable and secure than C++ applications.

Q10.What is singleton class in Java and how can we make a class singleton?

→ A singleton class in Java restricts the instantiation of a class to a single instance. To create a singleton class, you make its constructor private, create a private static instance of the class, and provide a public static method to return that instance. This ensures that only one instance of the class can exist in the application.



Q3. Why Java is platform independent?

→ Java is considered platform-independent due to its 'Write Once, Run Anywhere' philosophy, which is achieved through several key mechanisms. Firstly, Java source code is compiled into bytecode, an intermediate and platform-independent format. Secondly, Java applications are executed using a Java Virtual Machine (JVM), which is available for different platforms, making it possible to run Java applications without modification. Additionally, Java's standard library provides an abstraction layer for common tasks, and the language specification enforces rules for consistent behavior across platforms. Java also restricts direct access to hardware, abstracts memory management through garbage collection, and allows dynamic classloading. All of these features contribute to Java's high portability and its ability to run on diverse operating systems. This platform independence has made Java a popular choice for developing cross-platform software.

Q4. Why Java is not 100% Object-oriented?

→ Java is not considered 100% object-oriented for several reasons. Firstly, it includes primitive data types that are not objects for efficiency reasons. Secondly, it allows static methods and variables that are associated with classes, not instances. Additionally, Java supports procedural programming constructs and lacks multiple inheritance for classes. Finally, it includes non-object-oriented constructs like operators and control flow statements. However, Java strikes a balance between object-oriented principles and practicality, making it a versatile and widely used language.

Q5. What are wrapper classes in Java?

→ Wrapper classes convert the Java primitives into the reference types (objects). Every primitive data type has a class dedicated to it. These are known as wrapper classes because they "wrap" the primitive data type into an object of that class.



Q6. What is the difference between ArrayList and vector in Java?

ArrayList	Vector
ArrayList is not synchronized.	Vector is synchronized.
ArrayList is fast as it's non-synchronized.	Vector is slow as it is thread safe.
If an element is inserted into the ArrayList, it increases its Array size by 50%.	Vector defaults to doubling size of its array.
ArrayList does not define the increment size.	Vector defines the increment size.
ArrayList can only use Iterator for traversing an ArrayList.	Vector can use both Enumeration and Iterator for traversing.

Q7.What is the difference between equals() and == in Java?

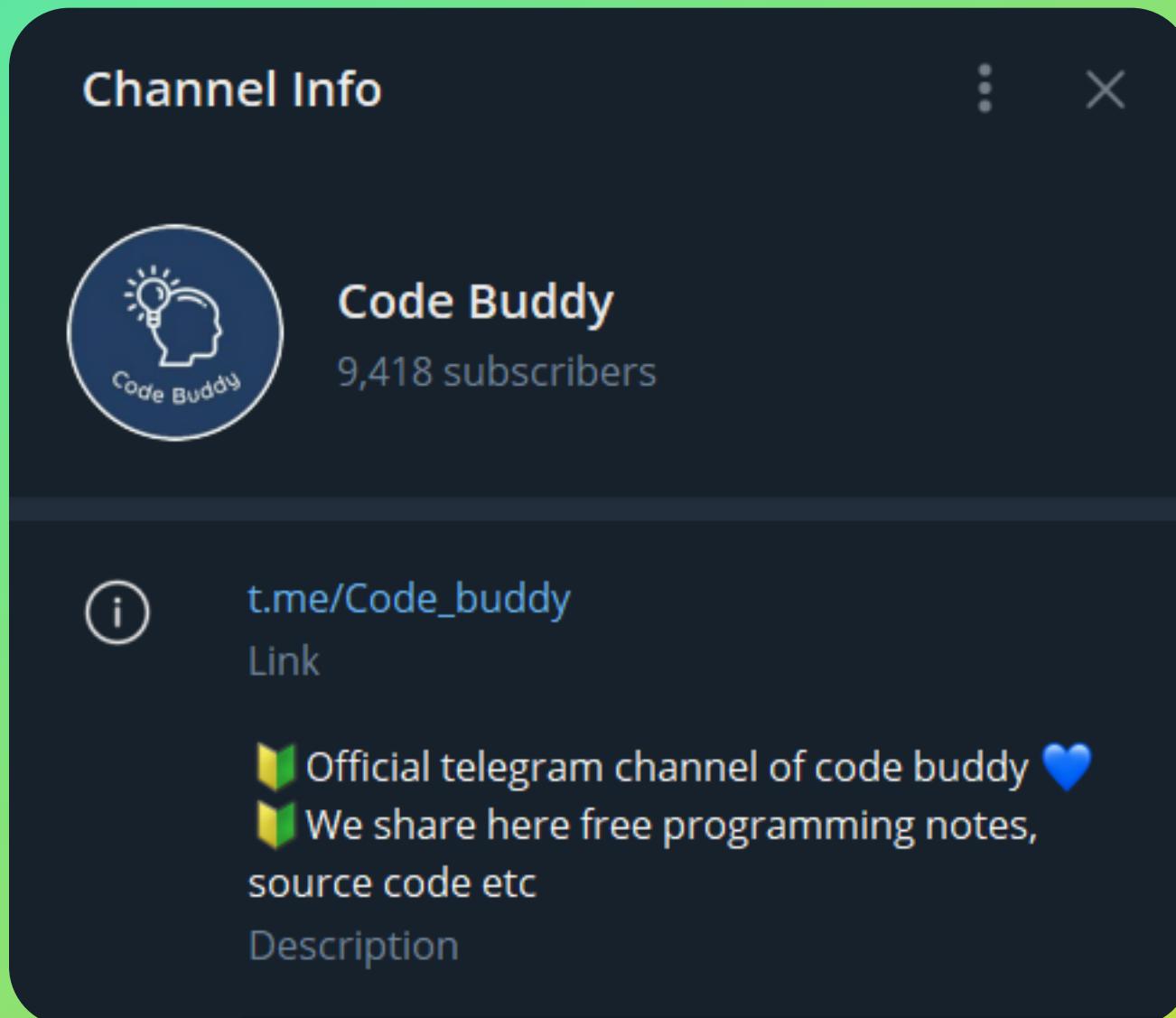
→ Equals() method is defined in Object class in Java and used for checking equality of two objects defined by business logic.
“==” or equality operator in Java is a binary operator provided by Java programming language and used to compare primitives and objects. public boolean equals(Object o) is the method provided by the Object class. The default implementation uses == operator to compare two objects. For example: method can be overridden like String class. equals() method is used to compare the values of two objects.

Q8.What is Java?

→ Java is a class-based, object-oriented programming language with minimal implementation dependencies. A general-purpose programming language, Java allows programmers to write once and run anywhere (WORA), meaning compiled Java code can execute on all Java-supported platforms without recompilation. Java applications are usually compiled to bytecode that works on any JVM, regardless of computer architecture.



FREE PDF UPLOADED ON TELEGRAM



LETS MAKE IT 10,000



Link in Bio