Unit

Here are some short answer type questions for inheritance in Java:

1. **What is inheritance in Java?**
2. **Name the keyword used for inheritance in Java.**
3. **How do you implement inheritance in Java?**
4. **What is the extends keyword used for in Java?**
5. **What is single inheritance? Give an example.**
6. **What is the difference between single and multilevel inheritance?**
7. **What is the difference between extends and implements in Java?**
8. **What is method overriding? Provide an example.**
9. **Can a class inherit from multiple classes in Java? Why or why not?**
10. **What is the super keyword used for?**
11. **What is the difference between super and this keywords?**
12. **Can constructors be inherited in Java? Explain.**
13. **What is the role of the Object class in Java inheritance?**
14. **How do you prevent a class from being inherited?**
15. **What is hierarchical inheritance? Provide an example.**
16. **Why is multiple inheritance not supported in Java using classes?**
17. **How is polymorphism related to inheritance?**
18. **What is the purpose of final keyword in inheritance?**
19. **What is an abstract class in Java?**
20. **How do you declare an abstract class in Java?**
21. **What is the purpose of an abstract class?**
22. **Can an abstract class have concrete (non-abstract) methods?**
23. **What is the difference between an abstract class and an interface in Java?**
24. **Can you create an instance of an abstract class? Why or why not?**
25. **What is an abstract method in Java?**
26. **Can an abstract class have constructors? If yes, explain why.**
27. **Can an abstract class implement an interface?**
28. **Can an abstract class be final in Java? Why or why not?**
29. **Can a class be abstract without any abstract methods?**
30. **How does inheritance work with abstract classes?**
31. **What happens if a subclass does not implement all abstract methods of an abstract class?**
32. **Can you declare an abstract method as static in Java? Why or why not?**
33. **What are the advantages of using abstract classes in a program?**

**Arrays:**

1. **What is an array in Java?**  
   An array is a collection of elements of the same data type stored in contiguous memory locations, accessed using an index.
2. **How do you declare and initialize an array in Java?**  
   You can declare an array like: int[] arr;  
   You can initialize it as: arr = new int[5]; or int[] arr = {1, 2, 3, 4, 5};
3. **How is an array stored in computer memory?**  
   Arrays are stored in contiguous memory locations, where each element occupies space based on its data type (e.g., 4 bytes for int).
4. **How can you access elements of an array?**  
   You access elements by specifying the index: arr[2] accesses the third element.
5. **What are the different operations you can perform on array elements?**  
   Operations include accessing, updating, traversing, sorting, searching, and arithmetic operations on elements.
6. **How can you assign one array to another in Java?**  
   You can assign arrays directly: arr2 = arr1;  
   However, both arr2 and arr1 will reference the same array object.
7. **How can you dynamically change the size of an array in Java?**  
   Arrays have a fixed size. To change the size dynamically, you need to create a new array and copy the elements, or use dynamic data structures like ArrayList.
8. **What are the different methods for sorting arrays in Java?**  
   You can use Arrays.sort(arr) to sort an array in ascending order.
9. **How do you search for a specific value in an array?**  
   You can search using a loop or by using Arrays.binarySearch(arr, value) if the array is sorted.
10. **What is the Arrays class in Java, and what utility methods does it provide?**  
    The Arrays class provides utility methods for manipulating arrays, such as sort(), binarySearch(), copyOf(), equals(), and fill().
11. **How do you declare and use a two-dimensional array in Java?**  
    You declare a 2D array as int[][] matrix = new int[3][4]; and access elements like matrix[1][2].
12. **What are arrays of varying lengths (jagged arrays) in Java?**  
    Jagged arrays are arrays where each row may have a different length. For example: int[][] jagged = new int[3][]; jagged[0] = new int[2]; jagged[1] = new int[4];
13. **How do you declare and work with three-dimensional arrays in Java?**  
    You declare a 3D array like int[][][] arr = new int[3][4][5]; and access elements as arr[i][j][k].
14. **How can arrays be used to represent vectors in Java?**  
    Arrays can represent vectors by using a 1D array to store vector components, e.g., int[] vector = {x, y, z};.

**Inheritance:**

1. **What is inheritance in Java, and why is it used?**  
   Inheritance is a mechanism in Java where one class (subclass) inherits the properties and behavior (fields and methods) of another class (superclass). It promotes code reuse and hierarchical classification.
2. **What is the difference between "Has-a" and "Is-a" relationships in Java?**
   * "Is-a" represents inheritance, where a subclass extends a superclass (e.g., Dog is a Animal).
   * "Has-a" represents composition, where a class contains another class as a field (e.g., Car has a Engine).
3. **Explain the process of inheritance in Java.**  
   A subclass inherits the fields and methods of a superclass using the extends keyword. The subclass can add its own fields and methods or override inherited methods.
4. **What are the different types of inheritance in Java?**  
   Java supports single, multilevel, and hierarchical inheritance. It does not support multiple inheritance through classes but supports it via interfaces.
5. **What is the Object class in Java, and why is it considered the universal superclass?**  
   The Object class is the root class of all Java classes. Every class in Java implicitly extends Object, which provides common methods like toString(), hashCode(), and equals().
6. **How can inheritance be inhibited using the final keyword?**  
   The final keyword can be used to prevent inheritance by marking a class as final, which means it cannot be extended. Example: public final class Car { }
7. **What is access control in Java, and how does it affect inheritance?**  
   Access control defines the visibility of class members. Members marked private are not inherited, while protected members are accessible within subclasses.
8. **Explain multilevel inheritance with an example.**  
   In multilevel inheritance, a class inherits from a class that is itself a subclass. Example:  
   class A {}  
   class B extends A {}  
   class C extends B {}.
9. **What is the use of the super keyword in Java?**  
   The super keyword refers to the superclass. It is used to call the superclass constructor or to access superclass methods and fields.
10. **How do constructors work in inheritance?**  
    Constructors of the superclass are called automatically when a subclass object is created, starting from the topmost superclass down to the subclass. You can explicitly call the superclass constructor using super().
11. **What is method overriding, and how is it related to inheritance?**