Selvam College of Technology

An Autonomous Institution

Accredited by NAAC with "A" Grade, UGC Recognized 2(f) Status,
An ISO 9001:2015 Certified Institution, Approved by AICTE New Delhi, Affiliated to Anna University-Chennai
PONNUSAMY NAGAR, SALEM ROAD(NH-44), NAMAKKAL-637003. TAMILNADU.
Mobile: 9942099122, 9942099109, Web: www.selvamtech.edu.in

Name of the Bundle	Advanced Bundle V2	Subject	Java Programming V2
Topic	Inheritance	Last updated on	09 January 2025

1. What does inheritance allow a class to do?

    a. Create a new class from scratch.

    b. Copy all features from another class.

    c. Inherit fields and methods from another class.

    d. Delete features from another class.

   Ans: c. Inherit fields and methods from another class.

   Explanation: Inheritance lets a class use features (fields and methods) from another class.

2. What is the main benefit of inheritance in programming?

    a. To prevent creating new classes.

    b. To create new classes based on existing ones.

    c. To remove unnecessary features.

    d. To avoid using fields and methods.

   Ans: b. To create new classes based on existing ones.

   Explanation: Inheritance allows creating new classes that reuse and build upon the functionality of existing ones.

3. Which of the following does a subclass inherit from its parent class?

    a. Only the fields

    b. Only the methods

    c. Both fields and methods

    d. Only the constructor

   Ans: c. Both fields and methods

   Explanation: A subclass inherits both fields (variables) and methods from its parent class.

4. Which of these is inherited by a subclass?

   a. Only the constructor.

   b. Only private methods.

   c. Public and protected fields and methods.

   d. Only static methods.

   Ans: c. Public and protected fields and methods.

   Explanation: A subclass inherits public and protected fields and methods from its parent class, but not private ones or constructors.

5. Why is inheritance important in programming?

   a. It allows code reuse.

   b. It makes the program slower.

   c. It removes all the methods.

   d. It requires fewer classes.

   Ans: a. It allows code reuse.

   Explanation: Inheritance enables code reuse by allowing a subclass to inherit and use features (fields and methods) from a parent class, avoiding duplication.

6. What can a subclass do with the inherited methods from the parent class?

   a. Modify them

   b. Delete them

   c. Ignore them

   d. Add new fields

   Ans: a. Modify them

   Explanation: A subclass can change how inherited methods work.

IT Support and Development Training Programme
Creating Employable Engineers and Entrepreneurs

7. Which of the following is true about inheritance?

    a. Subclass can't use parent features.

    b. Subclass only gets methods, not fields.

    c. Subclasses can use parent features.

    d. Subclass only gets constructors.

Ans: c. Subclasses can use parent features.

Explanation: Inheritance allows subclasses to inherit fields and methods from the parent class, enabling code reuse and extension.

8. Inheritance allows a subclass to do what with its parent class?

    a. Create a completely independent class.

    b. Copy only the fields of the parent class.

    c. Access and use fields and methods from the parent class.

    d. Remove the parent class methods.

Ans: c. Access and use fields and methods from the parent class.

Explanation: Inheritance lets a subclass use the fields and methods of its parent class. This helps reuse code.

9. How does inheritance help with code maintenance?

    a. It removes all the errors.

    b. It allows code reuse and reduces repetition.

    c. It eliminates the need for methods.

    d. It makes the program run faster.

Ans: b. It allows code reuse and reduces repetition.

Explanation: Inheritance helps with code maintenance by allowing code reuse and reducing repetition, making the code easier to update and manage.

10. Which of the following can a subclass do with a method inherited from the parent class?

    a. Only use it.

    b. Modify it or use it.

    c. Ignore it.

    d. Delete it permanently.

Ans: b. Modify it or use it.

Explanation: A subclass can either use the inherited method as it is or modify it by overriding it to provide specialized behavior.

11. What can a class do when it inherits from another class?

    a. It can only reuse fields.

    b. It can only add new fields.

    c. It can reuse methods and fields, and add new ones.

    d. It cannot reuse any methods.

Ans: c. It can reuse methods and fields, and add new ones.

Explanation: When a class inherits from another class, it can reuse existing methods and fields while also adding new ones for extended functionality.

12. What can be added to a class that inherits from another class?

    a. Only new methods.

    b. Only new fields.

    c. Both new fields and methods.

    d. No new features can be added.

Ans: c. Both new fields and methods.

Explanation: A class that inherits from another class can add both new fields and methods, extending its functionality.

IT Support and Development Training Programme
Creating Employable Engineers and Entrepreneurs

13. Inheritance is an example of which relationship between classes?

    a. HAS-A relationship

    b. IS-A relationship

    c. WAS-A relationship

    d. CAN-DO relationship

Ans: b. IS-A relationship

Explanation: Inheritance represents an IS-A relationship because a subclass is a specialized version of its parent class.

14. What is an IS-A relationship in inheritance?

    a. A subclass is a type of the parent class.

    b. A parent class is a type of the subclass.

    c. A subclass cannot reuse the parent class features.

    d. A parent class does not have any features.

Ans: a. A subclass is a type of the parent class.

Explanation: An IS-A relationship in inheritance means a subclass is a specialized type of the parent class, inheriting its features.

15. Which keyword is used to inherit from a class in Java?

    a. super

    b. extends

    c. implements

    d. this

Ans: b. extends

Explanation: The extends keyword is used to inherit from a class in Java.

16. What does the extends keyword do in Java?

    a. It defines a new class without inheritance.

    b. It allows a class to inherit fields and methods from another class.

    c. It is used to implement interfaces.

    d. It makes the class static.

Ans: b. It allows a class to inherit fields and methods from another class.

Explanation: The extends keyword allows a class to inherit fields and methods from another class.

17. If Class A extends Class B, then which of the following is true?

    a. Class A is the superclass of Class B.

    b. Class B is the superclass of Class A.

    c. Class A and Class B are unrelated.

    d. Class A and Class B are the same class.

Ans: b. Class B is the superclass of Class A.

Explanation: If Class A extends Class B, then Class B is the superclass of Class A, meaning A inherits from B.

18. In an inheritance hierarchy, the class that is inherited from is known as the:

    a. Child class

    b. Superclass

    c. Sibling class

    d. Subclass

Ans: b. Superclass

Explanation: In an inheritance hierarchy, the class that is inherited from is known as the Superclass.

IT Support and Development Training Programme
Creating Employable Engineers and Entrepreneurs

19. Which of the following is true about the superclass in inheritance?

   a. The superclass inherits features.

   b. The superclass provides features to other classes.

   c. The superclass cannot be inherited.

   d. The superclass only contains methods, not fields.

Ans: b. The superclass provides features to other classes.

Explanation: The superclass provides features (fields and methods) to other classes that inherit from it.

20. When a class uses the extends keyword in Java, it is creating a relationship between which two classes?

   a. Parent class and sibling class

   b. Parent class and subclass

   c. Child class and sibling class

   d. Subclass and derived class

Ans: b. Parent class and subclass

Explanation: When a class uses the extends keyword in Java, it creates a relationship between the parent class and the subclass.

21. The class that inherits features from another class is called the:

   a. Superclass

   b. Derived class

   c. Parent class

   d. Existing class

Ans: b. Derived class

Explanation: The class that inherits features from another class is called the Derived class.

IT Support and Development Training Programme
Creating Employable Engineers and Entrepreneurs

22. Which keyword is used to inherit from an existing class in Java?

    a. derives

    b. extends

    c. super

    d. implements

Ans: b. extends

Explanation: The extends keyword is used to inherit from an existing class in Java.

23. What is the term for the class that is being inherited from in Java?

    a. Subclass

    b. Derived class

    c. Superclass

    d. Child class

Ans: c. Superclass

Explanation: The class that is being inherited from in Java is called the Superclass.

24. What is Single Inheritance in object-oriented programming?

    a. A class inherits from multiple classes.

    b. A class inherits from only one class.

    c. A class inherits from no classes.

    d. A class inherits only methods, not fields.

Ans: b. A class inherits from only one class.

Explanation: Single Inheritance in object-oriented programming means a class inherits from only one class.

25. What is Multilevel Inheritance?

    a. A class inherits from multiple classes.

    b. A class inherits from a parent class, which in turn inherits from another class.

    c. A class inherits from no classes.

    d. A class inherits only fields from its parent class.

Ans: b. A class inherits from a parent class, which in turn inherits from another class.

Explanation: Multilevel Inheritance occurs when a class inherits from another class, and that class also inherits from another class, forming a chain.

26. What is Hierarchical Inheritance?

    a. Multiple classes inherit from a single class.

    b. A class inherits from multiple classes.

    c. A class inherits from no other class.

    d. A class inherits from two unrelated classes.

Ans: a. Multiple classes inherit from a single class.

Explanation: Hierarchical Inheritance occurs when multiple classes inherit from a single parent class.

27. What is Multiple Inheritance?

    a. A class can inherit from two or more classes.

    b. Inherit from one class.

    c. Inherit from a chain of classes.

    d. Inherit from many interfaces, not classes.

Ans: a. A class can inherit from two or more classes.

Explanation: Multiple Inheritance means a class can inherit from two or more classes.

28. What is Hybrid Inheritance?

    a. Inherit from one parent class.

    b. Inherit from multiple classes.

    c. Inherit from no classes.

    d. Inherit only from interfaces.

    Ans: b. Inherit from multiple classes.

    Explanation: Hybrid Inheritance means inheriting from multiple classes.

29. Which inheritance type involves a class inheriting from multiple classes?

    a. Single Inheritance

    b. Multilevel Inheritance

    c. Multiple Inheritance

    d. Hierarchical Inheritance

    Ans: c. Multiple Inheritance

    Explanation: Multiple Inheritance involves a class inheriting from multiple classes.

30. Which inheritance type can be a combination of Single, Multilevel, and Multiple Inheritance?

    a. Hierarchical Inheritance

    b. Hybrid Inheritance

    c. Single Inheritance

    d. Multilevel Inheritance

    Ans: b. Hybrid Inheritance

    Explanation: Hybrid Inheritance is a combination of Single, Multilevel, and Multiple Inheritance.

IT Support and Development Training Programme
Creating Employable Engineers and Entrepreneurs

31. Which of the following is a disadvantage of Multiple Inheritance?

   a. It increases code reusability.

   b. It can cause ambiguity if two parent classes have the same method.

   c. It does not support polymorphism.

   d. It simplifies class design.

Ans: b. It can cause ambiguity if two parent classes have the same method.

Explanation: Multiple Inheritance can cause ambiguity if two parent classes have the same method, leading to potential conflicts.

32. A derived class can extend how many base classes?

   a. One base class

   b. Two base classes

   c. Multiple base classes

   d. No base classes

Ans: a. One base class

Explanation: In single inheritance, a derived class can extend one base class.

33. A derived class can implement how many interfaces?

   a. One interface

   b. Two interfaces

   c. Any number of interfaces

   d. No interfaces

Ans: c. Any number of interfaces

Explanation: A derived class can implement any number of interfaces in Java.

IT Support and Development Training Programme
Creating Employable Engineers and Entrepreneurs

34. In an inheritance hierarchy, where should you create an object?

    a. Only for the top-most base class

    b. Only for the bottom-most derived class

    c. Only for the derived classes, not base classes

    d. You can create an object for any class in the hierarchy

Ans: b. Only for the bottom-most derived class

Explanation: In an inheritance hierarchy, you typically create an object for the bottom-most derived class, as it contains the complete functionality.

35. Can a derived class extend multiple base classes?

    a. Yes, it can extend multiple base classes.

    b. No, it can extend only one base class.

    c. It can extend multiple classes but only if they are interfaces.

    d. A derived class cannot extend any base classes.

Ans: b. No, it can extend only one base class.

Explanation: In single inheritance, a derived class can extend only one base class. However, it can implement multiple interfaces.

36. Can a derived class implement multiple interfaces?

    a. Only one interface.

    b. Any number of interfaces.

    c. Cannot implement interfaces.

    d. Only two interfaces.

Ans: b. Any number of interfaces.

Explanation: A derived class can implement any number of interfaces.

IT Support and Development Training Programme
Creating Employable Engineers and Entrepreneurs

37. When we create an object of the most derived class, where is memory allocated first?

    a. Data members of the derived class

    b. Data members of the topmost base class

    c. Methods of the derived class

    d. Constructors of the base classes

Ans: b. Data members of the topmost base class

Explanation: Memory is first allocated for the data members of the topmost base class when creating an object.

38. After memory is allocated for the topmost base class, where is memory allocated next?

    a. For the methods of the topmost base class

    b. For the data members of the derived class

    c. For the constructors of the base class

    d. For the base class of the derived class

Ans: b. For the data members of the derived class

Explanation: After memory is allocated for the topmost base class, it is allocated next for the data members of the derived class.

39. What does the derived class logically contain when an object is created?

    a. Data members of the derived class

    b. Data members of the topmost base class

    c. All data members from base classes

    d. No data members

Ans: c. All the data members from base classes

Explanation: When an object is created, the derived class logically contains all data members from base classes.

IT Support and Development Training Programme
Creating Employable Engineers and Entrepreneurs

40. When an object is created in a class hierarchy, which data members are included in the object?

   a. Only the derived class data

   b. Only the base class data

   c. All data from both derived and base classes

   d. No base class data

   Ans: c. All data from both derived and base classes

   Explanation: When an object is created in a class hierarchy, it includes all data from both derived and base classes.

41. Can a base class constructor be inherited?

   a. Constructors are inherited.

   b. Constructors are not inherited.

   c. Only in multilevel inheritance.

   d. Only if public.

   Ans: b. Constructors are not inherited.

   Explanation: Constructors are not inherited by a derived class.

42. Why must the base class be defined first in inheritance?

   a. To prevent it from being reused or inherited.

   b. To allow the derived class to override its methods.

   c. To ensure that the derived class can access private members.

   d. To enable polymorphism.

   Ans: a. To prevent it from being reused or inherited.

   Explanation: The base class must be defined first so the derived class can use it.

43. Which class is the implicit superclass for all Java classes?

    a.   java.util.List

    b. java.lang.Object

    c. java.lang.Class

    d. java.lang.System

Ans: b. java.lang.Object

Explanation: The implicit superclass for all Java classes is java.lang.Object.

44. What is the role of the java.lang.Object class in Java?

    a. For interfaces.

    b. Superclass for all Java classes.

    c. For constructors only.

    d. For garbage collection.

Ans: b. Superclass for all Java classes.

Explanation: The java.lang.Object class is the superclass for all Java classes.

45. What feature does java.lang.Object provide to improve application performance?

    a. Object serialization

    b. Exception handling

    c. Garbage collection to free unused memory

    d. Multithreading

Ans: c. Garbage collection to free unused memory

Explanation: java.lang.Object indirectly supports garbage collection, which helps improve application performance by freeing unused memory.

IT Support and Development Training Programme
Creating Employable Engineers and Entrepreneurs

46. What is the scope of a base class object?

    a. It can access both the base class and derived class features.

    b. It only holds details about the features of the derived class.

    c. It only holds details about its own features.

    d. It cannot access any features of the class.

Ans: c. It only holds details about its own features.

Explanation: A base class object only holds details about its own features and cannot access features of the derived class.

47. Why is multiple inheritance not supported in Java?

    a. To reduce the complexity and simplify the language.

    b. To support dynamic polymorphism.

    c. To make it easier to write complex code.

    d. To allow inheritance from multiple interfaces.

Ans: a. To reduce the complexity and simplify the language.

Explanation: Multiple inheritance is not supported in Java to reduce complexity and simplify the language, preventing issues like ambiguity.

48. What is Single Inheritance?

    a. A class inherits from multiple classes

    b. A class inherits from only one class

    c. A class inherits from no classes

    d. A class inherits only methods, not fields

Ans: b. A class inherits from only one class

Explanation: Single inheritance means a class derives properties and behavior from one parent class, enabling code reuse and a clear class hierarchy.

49. How many base classes can a derived class extend?

    a. Only one base class

    b. Two base classes

    c. Multiple base classes

    d. No base classes

Answer: a. Only one base class

Explanation: In Java, a derived class can extend only one base class because Java does not support multiple inheritance with classes to avoid complexity and ambiguity.

50. Which of these is not inherited by a subclass?

    a. Constructor

    b. Private and static methods

    c. Private fields and methods

    d. All the above are not inherited

Ans: d. All the above are not inherited

Explanation: Constructors, private fields, and private methods are not inherited by a subclass.

IT Support and Development Training Programme

Creating Employable Engineers and Entrepreneurs