

Lecture 3-inheritance and polymorphism

1. What is the primary purpose of inheritance in object-oriented programming?

- a) To create multiple instances of a class
- b) To keep common behavior in one class and split different behavior into separate classes
- c) To override all methods in a superclass
- d) To create private variables

Answer: b

2. Which keyword is used to indicate that a class inherits from another class in Java?

- a) implements
- b) inherits
- c) extends
- d) derives

Answer: c

3. In UML diagrams, what does a solid line with a hollow triangle arrowhead represent?

- a) Association
- b) Aggregation
- c) Composition
- d) Inheritance

Answer: d

4. Which visibility modifier is the most restrictive?

- a) public
- b) protected
- c) default (package-private)
- d) private

Answer: d

5. What is the "is-a" relationship in inheritance?

- a) A subclass is a type of its superclass
- b) A superclass is a type of its subclass
- c) Two classes have the same methods
- d) Two classes have the same variables

Answer: a

6. In Java, if a class doesn't explicitly extend another class, what class does it implicitly extend?

- a) String
- b) Object
- c) Class
- d) None

Answer: b

7. What is the correct order of object construction in inheritance?

- a) Subclass, Superclass, Object
- b) Object, Superclass, Subclass
- c) Superclass, Object, Subclass
- d) Subclass, Object, Superclass

Answer: b

8. Which statement must be the first line in a constructor of a subclass?

- a) this();
- b) super();
- c) Either this() or super()
- d) new Object();

Answer: c

9. What is method overriding?

- a) Defining a method in a subclass with the same name and parameters as in the superclass
- b) Defining multiple methods with the same name but different parameters in the same class
- c) Defining a method with a different name but same parameters as in the superclass
- d) Defining a private method in a subclass

Answer: a

10. Which method from the Object class is commonly overridden to provide a string representation of an object?

- a) getString()
- b) print()
- c) toString()
- d) convertToString()

Answer: c

11. What is polymorphism in object-oriented programming?

- a) The ability to create multiple objects of the same class
- b) The ability to override methods in a subclass
- c) The ability for a superclass reference to call the appropriate method of a subclass object
- d) The ability to create multiple classes with the same name

Answer: c

12. What happens if you try to call a subclass-specific method on a superclass reference without casting?

- a) It works fine
- b) Runtime error
- c) Compile-time error
- d) The program crashes

Answer: c

13. Which keyword is used to check if an object is an instance of a particular class at runtime?

- a) instanceof

- b) instanceof
- c) typeOf
- d) checkType

Answer: b

14. What is the purpose of an abstract class?

- a) To create multiple instances of a class
- b) To define a class that cannot be instantiated and may contain abstract methods
- c) To override all methods in a superclass
- d) To create private variables

Answer: b

15. Which of the following is true about abstract methods?

- a) They have a method body
- b) They can be declared in non-abstract classes
- c) They must be implemented by non-abstract subclasses
- d) They can be declared as private

Answer: c

16. What is the main difference between an abstract class and an interface?

- a) Abstract classes can have constructors, interfaces cannot
- b) Interfaces can have implemented methods, abstract classes cannot
- c) Abstract classes support multiple inheritance, interfaces do not
- d) Interfaces can have instance variables, abstract classes cannot

Answer: a

17. Which of the following is a correct way to declare a class that implements an interface?

- a) public class MyClass extends MyInterface
- b) public class MyClass implements MyInterface
- c) public class MyClass inherits MyInterface
- d) public class MyClass using MyInterface

Answer: b

18. What is the purpose of the Comparable interface in Java?

- a) To compare two objects for equality
- b) To define a natural ordering for a class of objects
- c) To sort objects in a collection
- d) To implement the equals() method

Answer: b

19. Which method must be implemented when a class implements the Comparable interface?

- a) compare()
- b) compareTo()
- c) equals()
- d) hashCode()

Answer: b

20. What is the return type of the compareTo() method in the Comparable interface?

a) boolean

b) Object

c) int

d) void

Answer: c