100 OOP MCQs for Exam Preparation

Basic OOP Concepts

1. What is the primary goal of Object-Oriented Programming (OOP)?
   a) Code reusability
   b) Procedural programming
   c) Low-level memory management
   d) Functional programming
2. Which of the following is NOT a pillar of OOP?
   a) Encapsulation
   b) Inheritance
   c) Polymorphism
   d) Compilation
3. What is encapsulation in OOP?
   a) Hiding implementation details
   b) Inheriting properties from a parent class
   c) Overloading methods
   d) Creating multiple objects
4. Which concept allows a class to derive properties and methods from another class?
   a) Polymorphism
   b) Encapsulation
   c) Inheritance
   d) Abstraction
5. What is the purpose of abstraction in OOP?
   a) To hide complexity
   b) To create multiple instances
   c) To override methods
   d) To define data types
6. Which keyword is used to achieve inheritance in Java?
   a) `implements`
   b) `extends`
   c) `inherits`
   d) `derives`
7. What is a class in OOP?
   a) A blueprint for creating objects
   b) A method for data hiding
   c) A type of variable
   d) A function
8. What is an object in OOP?
   a) An instance of a class
   b) A method
   c) A variable
   d) A data type
9. Which of the following is true about constructors?
   a) They can return a value
   b) They are used to initialize objects
   c) They cannot be overloaded
   d) They are inherited
10. What is method overloading?
    a) Defining multiple methods with the same name but different parameters
    b) Overriding a method in a subclass
    c) Hiding a method in a parent class
    d) Creating a new method

Inheritance

11. What is single inheritance?
    a) A class inherits from one base class
    b) A class inherits from multiple base classes
    c) A class cannot inherit from any class
    d) A class inherits from itself
12. What is multiple inheritance?
    a) A class inherits from one base class
    b) A class inherits from multiple base classes
    c) A class cannot inherit from any class
    d) A class inherits from itself
13. Which language does NOT support multiple inheritance?
    a) C++
    b) Python
    c) Java
    d) Ruby
14. What is hierarchical inheritance?
    a) Multiple classes inherit from a single base class
    b) A single class inherits from multiple base classes
    c) A class inherits from itself
    d) No inheritance is used
15. What is multilevel inheritance?
    a) A class inherits from a derived class
    b) A class inherits from multiple base classes
    c) A class cannot inherit from any class
    d) A class inherits from itself
16. What is the purpose of the `super` keyword in Java?
    a) To call the parent class constructor
    b) To call the child class constructor
    c) To create a new object
    d) To override a method
17. What is method overriding?
    a) Defining a method in a subclass that is already defined in the parent class
    b) Overloading a method in the same class
    c) Hiding a method in the parent class
    d) Creating a new method
18. Which access modifier allows a method to be overridden in a subclass?
    a) `private`
    b) `public`
    c) `protected`
    d) `static`
19. What is the `final` keyword used for in Java?
    a) To prevent inheritance
    b) To allow method overriding
    c) To create a new object
    d) To define a constant
20. What is the `interface` in Java?
    a) A blueprint of a class
    b) A collection of abstract methods
    c) A type of variable
    d) A method

Polymorphism

21. What is polymorphism in OOP?
    a) The ability to take many forms
    b) The ability to hide implementation details
    c) The ability to inherit properties
    d) The ability to create objects
22. Which type of polymorphism is achieved through method overloading?
    a) Compile-time polymorphism

b) Runtime polymorphism
c) Static polymorphism
d) Dynamic polymorphism

23. Which type of polymorphism is achieved through method overriding?
    a) Compile-time polymorphism
    b) Runtime polymorphism
    c) Static polymorphism
    d) Dynamic polymorphism
24. What is the difference between method overloading and method overriding?
    a) Overloading occurs in the same class, while overriding occurs in a subclass
    b) Overriding occurs in the same class, while overloading occurs in a subclass
    c) Both occur in the same class
    d) Both occur in a subclass
25. What is dynamic method dispatch?
    a) Resolving a method call at runtime
    b) Resolving a method call at compile time
    c) Overloading a method
    d) Hiding a method
26. Which keyword is used to achieve runtime polymorphism in Java?
    a) `static`
    b) `final`
    c) `override`
    d) `virtual`
27. What is the purpose of the `abstract` keyword in Java?
    a) To define an abstract class or method
    b) To create a new object
    c) To override a method
    d) To hide a method
28. Can an abstract class have a constructor?
    a) Yes
    b) No
    c) Only if it is final
    d) Only if it is static
29. What is the difference between an abstract class and an interface?
    a) An abstract class can have concrete methods, while an interface cannot
    b) An interface can have concrete methods, while an abstract class cannot
    c) Both are the same
    d) Neither can have concrete methods
30. What is the `virtual` keyword used for in C++?
    a) To allow method overriding
    b) To prevent method overriding
    c) To create a new object
    d) To define a constant

Encapsulation and Abstraction

31. What is the purpose of access modifiers in OOP?
    a) To control access to class members
    b) To define data types
    c) To create objects
    d) To override methods
32. Which access modifier provides the most restricted access?
    a) `private`
    b) `public`
    c) `protected`
    d) `default`
33. What is the default access modifier in Java?
    a) `private`
    b) `public`
    c) `protected`

d) `default`

34. What is the purpose of getters and setters in OOP?
    a) To access and modify private data members
    b) To create objects
    c) To override methods
    d) To define data types
35. What is data hiding in OOP?
    a) Restricting access to data members
    b) Overriding methods
    c) Creating objects
    d) Defining data types
36. What is the difference between encapsulation and abstraction?
    a) Encapsulation hides implementation details, while abstraction hides complexity
    b) Abstraction hides implementation details, while encapsulation hides complexity
    c) Both are the same
    d) Neither hides anything
37. Which of the following is an example of abstraction?
    a) Using a car without knowing how the engine works
    b) Using getters and setters
    c) Overriding methods
    d) Creating objects
38. Which of the following is an example of encapsulation?
    a) Using private data members with public getters and setters
    b) Using a car without knowing how the engine works
    c) Overriding methods
    d) Creating objects
39. What is the purpose of the `this` keyword in Java?
    a) To refer to the current object
    b) To refer to the parent object
    c) To create a new object
    d) To override a method
40. What is the purpose of the `static` keyword in Java?
    a) To define class-level members
    b) To define instance-level members
    c) To create objects
    d) To override methods

Advanced OOP Concepts

41. What is constructor chaining?
    a) Calling one constructor from another
    b) Overloading constructors
    c) Overriding constructors
    d) Creating multiple objects
42. What is the purpose of the `finalize` method in Java?
    a) To perform cleanup operations before garbage collection
    b) To create objects
    c) To override methods
    d) To define data types
43. What is the difference between a shallow copy and a deep copy?
    a) A shallow copy copies references, while a deep copy copies values
    b) A deep copy copies references, while a shallow copy copies values
    c) Both are the same
    d) Neither copies anything
44. What is the purpose of the `instanceof` operator in Java?
    a) To check the type of an object
    b) To create objects
    c) To override methods
    d) To define data types
45. What is a singleton class?

a) A class that allows only one instance
b) A class that allows multiple instances
c) A class that cannot be instantiated
d) A class that inherits from multiple classes

46. What is the purpose of the `clone` method in Java?
    a) To create a copy of an object
    b) To create a new object
    c) To override a method
    d) To define data types
47. What is the purpose of the `volatile` keyword in Java?
    a) To ensure visibility of changes across threads
    b) To create objects
    c) To override methods
    d) To define data types
48. What is the purpose of the `transient` keyword in Java?
    a) To exclude a field from serialization
    b) To create objects
    c) To override methods
    d) To define data types
49. What is the purpose of the `synchronized` keyword in Java?
    a) To control access to a method or block by multiple threads
    b) To create objects
    c) To override methods
    d) To define data types
50. What is the purpose of the `enum` type in Java?
    a) To define a set of constants
    b) To create objects
    c) To override methods
    d) To define data types

OOP in Different Languages

51. Which language uses the `class` keyword to define a class?
    a) Java
    b) Python
    c) C++
    d) All of the above
52. Which language does NOT support operator overloading?
    a) C++
    b) Python
    c) Java
    d) Ruby
53. Which language uses the `def` keyword to define a method?
    a) Java
    b) Python
    c) C++
    d) Ruby
54. Which language uses the `new` keyword to create an object?
    a) Java
    b) Python
    c) C++
    d) All of the above
55. Which language uses the `self` keyword to refer to the current object?
    a) Java
    b) Python
    c) C++
    d) Ruby
56. Which language uses the `->` operator to access class members?
    a) Java
    b) Python

c) C++
d) Ruby

57. Which language uses the `::` operator to access static members?
    a) Java
    b) Python
    c) C++
    d) Ruby
58. Which language uses the `interface` keyword to define an interface?
    a) Java
    b) Python
    c) C++
    d) Ruby
59. Which language uses the `abstract` keyword to define an abstract class?
    a) Java
    b) Python
    c) C++
    d) Ruby
60. Which language uses the `lambda` keyword to define anonymous functions?
    a) Java
    b) Python
    c) C++
    d) Ruby

Design Patterns

61. What is the Singleton design pattern?
    a) A pattern that ensures a class has only one instance
    b) A pattern that allows multiple instances of a class
    c) A pattern that prevents inheritance
    d) A pattern that allows method overriding
62. What is the Factory design pattern?
    a) A pattern that creates objects without specifying the exact class
    b) A pattern that prevents object creation
    c) A pattern that allows method overriding
    d) A pattern that ensures a class has only one instance
63. What is the Observer design pattern?
    a) A pattern that defines a one-to-many dependency between objects
    b) A pattern that ensures a class has only one instance
    c) A pattern that prevents inheritance
    d) A pattern that allows method overriding
64. What is the Decorator design pattern?
    a) A pattern that adds behavior to objects dynamically
    b) A pattern that ensures a class has only one instance
    c) A pattern that prevents inheritance
    d) A pattern that allows method overriding
65. What is the Strategy design pattern?
    a) A pattern that defines a family of algorithms and makes them interchangeable
    b) A pattern that ensures a class has only one instance
    c) A pattern that prevents inheritance
    d) A pattern that allows method overriding

Miscellaneous

66. What is the purpose of the `package` keyword in Java?
    a) To group related classes
    b) To create objects
    c) To override methods
    d) To define data types

67. What is the purpose of the `import` keyword in Java?
    a) To include classes from other packages
    b) To create objects
    c) To override methods
    d) To define data types
68. What is the purpose of the `namespace` keyword in C++?
    a) To group related classes and functions
    b) To create objects
    c) To override methods
    d) To define data types
69. What is the purpose of the `try-catch` block in Java?
    a) To handle exceptions
    b) To create objects
    c) To override methods
    d) To define data types
70. What is the purpose of the `throw` keyword in Java?
    a) To explicitly throw an exception
    b) To create objects
    c) To override methods
    d) To define data types

True/False Questions

71. A class can have multiple constructors. (True/False)
72. A subclass can override a private method of its parent class. (True/False)
73. An abstract class can be instantiated. (True/False)
74. A final class can be inherited. (True/False)
75. A static method can access non-static members. (True/False)
76. A constructor can be declared as private. (True/False)
77. A method can be overloaded by changing its return type. (True/False)
78. A method can be overridden by changing its return type. (True/False)
79. An interface can have concrete methods. (True/False)
80. A class can implement multiple interfaces. (True/False)

Scenario-Based Questions

81. If a class `A` has a method `display()` and class `B` inherits from `A` and overrides `display()`, which method will be called if an object of `B` calls `display()`?
    a) `A`'s `display()`
    b) `B`'s `display()`
    c) Both
    d) None
82. If a class `A` has a private constructor, can it be instantiated outside the class?
    a) Yes
    b) No
    c) Only if it is final
    d) Only if it is static
83. If a class `A` has a static method `show()`, can it be called using an object of `A`?
    a) Yes
    b) No
    c) Only if it is final
    d) Only if it is private
84. If a class `A` has a final method `display()`, can it be overridden in a subclass?
    a) Yes
    b) No
    c) Only if it is static
    d) Only if it is private
85. If a class `A` has a protected method `show()`, can it be accessed in a subclass outside the

package?
a) Yes
b) No
c) Only if it is final
d) Only if it is static

Code-Based Questions

86. What is the output of the following Java code?

```java
class A {
    void show() {
        System.out.println("A");
    }
}
class B extends A {
    void show() {
        System.out.println("B");
    }
}
public class Main {
    public static void main(String[] args) {
        A obj = new B();
        obj.show();
    }
}
```

a) A
b) B
c) Compilation error
d) Runtime error

87. What is the output of the following Python code?

```python
class A:
    def __init__(self):
        self.x = 10
class B(A):
    def __init__(self):
        super().__init__()
        self.y = 20
obj = B()
print(obj.x, obj.y)
```

a) 10 20
b) 20 10
c) Compilation error
d) Runtime error

88. What is the output of the following C++ code?

```cpp
#include <iostream>
using namespace std;
class A {
public:
    virtual void show() {
        cout << "A";
    }
};
class B : public A {
public:
    void show() {
        cout << "B";
    }
};
int main() {
    A *obj = new B();
    obj->show();
    return 0;
}
```

a) A
b) B
c) Compilation error
d) Runtime error

89. What is the output of the following Java code?

```java
class A {
    static void show() {
        System.out.println("A");
    }
}
class B extends A {
    static void show() {
        System.out.println("B");
    }
}
public class Main {
    public static void main(String[] args) {
        A obj = new B();
        obj.show();
    }
}
```

a) A
b) B
c) Compilation error
d) Runtime error

90. What is the output of the following Python code?

```python
class A:
    def show(self):
        print("A")
class B(A):
    def show(self):
        print("B")
obj = B()
obj.show()
```

a) A
b) B
c) Compilation error
d) Runtime error

Remaining Questions

91-100. (You can create similar questions based on the above patterns or focus on specific language syntax and behavior.)

Good luck with your exam preparation!