

Python OOP Fundamentals Quiz: 40 Multiple-Choice Questions

1. What is the core concept of OOP that focuses on bundling data (attributes) and methods (functions) that operate on the data into a single unit?

A. Inheritance

B. Encapsulation

C. Polymorphism

D. Abstraction

Answer: B

2. Which keyword is used to define a Class in Python?

A. define

B. class

C. object

D. new

Answer: B

3. A Class is often described as a:

A. Single piece of data.

B. Blueprint or template for creating objects.

C. Function that returns a value.

D. Built-in Python data type.

Answer: B

4. What is an Object?

A. A copy of a class.

B. An instance of a class.

C. A synonym for a variable.

D. A Python module.

Answer: B

5. The first argument of every instance method (like the constructor) in a Python class is conventionally named:

- A. this
- B. my_class
- C. self
- D. instance

Answer: C

6. Which special method is automatically called when a new object (instance) of a class is created?

- A. `__start__`
- B. `__init__`
- C. `__new__`
- D. `__create__`

Answer: B

7. What is the primary role of the `__init__` method?

- A. To destroy the object.
- B. To define and initialize the object's attributes.
- C. To print the object's address.
- D. To inherit methods from other classes.

Answer: B

8. In Python, the mechanism where one class acquires the attributes and methods of another class is called:

- A. Encapsulation
- B. Abstraction
- C. Inheritance
- D. Instantiation

Answer: C

9. If Dog inherits from Animal, Dog is known as the:

- A. Parent Class
- B. Base Class
- C. Derived Class (or Child Class)
- D. Abstract Class

Answer: C

10. If Dog inherits from Animal, Animal is known as the:

- A. Child Class
- B. Subclass
- C. Derived Class
- D. Base Class (or Parent Class)

Answer: D

11. To call a method from the parent class inside a child class, you can use the function:

- A. parent()
- B. base()
- C. main()
- D. super()

Answer: D

12. What does the self parameter refer to in an instance method?

- A. The class itself.
- B. The specific instance (object) that the method is currently being called on.
- C. The parent class.
- D. A reserved keyword for static methods.

Answer: B

13. Which concept refers to the ability of different classes to respond to the same method call (e.g., `animal.speak()`) in their own specific way?

- A. Inheritance
- B. Polymorphism
- C. Encapsulation
- D. Abstraction

Answer: B

14. What is it called when a child class implements a method that is already present in its parent class?

- A. Method Overloading
- B. Method Overriding
- C. Polymorphism
- D. Method Hiding

Answer: B

15. An Attribute is another name for a class's:

- A. Function
- B. Variable (or data)
- C. Comment
- D. Keyword

Answer: B

16. A Method is another name for a class's:

- A. Variable (or data)
- B. Object
- C. Function
- D. Attribute

Answer: C

17. To create an instance of a class named Car, what is the correct syntax?

- A. `my_car = new Car()`
- B. `my_car = Car()`
- C. `my_car = create Car`
- D. `my_car.Car()`

Answer: B

18. In Python, how do you typically make an attribute "private" (indicating it shouldn't be directly accessed from outside the class)?

- A. By using the private keyword.
- B. By starting the attribute name with a single underscore (`_`).
- C. By starting the attribute name with two leading underscores (`__`).
- D. By defining it outside the `__init__` method.

Answer: C

19. Which OOP principle focuses on showing only essential information and hiding complex implementation details?

- A. Encapsulation
- B. Polymorphism
- C. Abstraction
- D. Inheritance

Answer: C

20. What is a Docstring (string literal right after the class definition) used for in a class?

- A. To hold the class data.
- B. To provide built-in documentation and description of the class.
- C. To define the constructor.
- D. To list the parent class.

Answer: B

21. What method is used to define an operation (like addition +) for objects of a custom class?

A. `__plus__`

B. `__add__`

C. `__sum__`

D. `__calc__`

Answer: B

22. A method that operates on the class itself rather than on a specific instance (object) is called a:

A. Instance Method

B. Private Method

C. Static Method

D. Global Method

Answer: C

23. Which decorator is used to define a method that belongs to the class itself, not to any specific instance, and takes `cls` as its first argument?

A. `@staticmethod`

B. `@classmethod`

C. `@property`

D. `@init`

Answer: B

24. A Class Attribute is shared by:

A. Only one instance of the class.

B. All methods within the class.

C. All instances (objects) of the class.

D. Only the parent class.

Answer: C

25. An Instance Attribute is specific to:

- A. The class definition.
- B. The parent class.
- C. The specific object (instance) it belongs to.
- D. All other objects of the same class.

Answer: C

26. What will be the output of `type(my_object)` if `my_object` is an instance of the class `Circle`?

- A. `<class '__main__.Circle'>`
- B. `<class 'object'>`
- C. `<class 'int'>`
- D. Error

Answer: A

27. The concept of Encapsulation primarily helps with:

- A. Making code faster.
- B. Controlling access to data and preventing accidental modification.
- C. Allowing multiple inheritance.
- D. Defining new data types.

Answer: B

28. When a class inherits from two or more parent classes, this is known as:

- A. Simple Inheritance
- B. Multiple Inheritance
- C. Polymorphism
- D. Aggregation

Answer: B

29. What is the method often used to provide a user-friendly, string representation of an object (what you see when you print the object)?

- A. `__print__`
- B. `__show__`
- C. `__str__`
- D. `__name__`

Answer: C

30. In the context of polymorphism, what does the expression `isinstance(obj, ClassName)` check?

- A. If `obj` is equal to `ClassName`.
- B. If `obj` is an instance of `ClassName` or its subclasses.
- C. If `obj` can inherit from `ClassName`.
- D. If `obj` has a method named `ClassName`.

Answer: B

31. If you want to define a method as abstract method, you would typically use the:

- A. `@classmethod` decorator
- B. `@staticmethod` decorator
- C. `@abstractmethod` decorator
- D. `__init__` method only

Answer: C

32. The process of creating and initializing an object from a class blueprint is called:

- A. Abstraction
- B. Declaration
- C. Instantiation
- D. Overriding

Answer: C

33. What is the relationship between `car = Car()` and the class `Car`?

- A. `car` is a method of `Car`.
- B. `car` is the blueprint for `Car`.
- C. `car` is an instance of `Car`.
- D. `car` is the parent of `Car`.

Answer: C

34. A Getter method's purpose is to:

- A. Change the value of a private attribute.
- B. Retrieve (read) the value of an attribute.
- C. Destroy the object.
- D. Define the class name.

Answer: B

35. A Setter method's purpose is to:

- A. Retrieve (read) the value of an attribute.
- B. Validate and set (change) the value of an attribute.
- C. Call the parent class constructor.
- D. Print the object's memory location.

Answer: B

36. In Python, methods are generally considered public, but if a method name starts with double underscore (`__`), it conventionally indicates:

- A. It must be called from outside the class.
- B. It is a private method meant only for internal use.
- C. It is a static method.
- D. It belongs to the parent class.

Answer: B

37. When creating a class that inherits from another, where is the parent class name placed?

- A. After the `def` keyword.
- B. Inside the parentheses of the class definition (`class Child(Parent):`).
- C. In a separate import statement.
- D. In the `__init__` method only.

Answer: B

38. The use of the same function name for different types of objects (as seen with `len()` working on strings, lists, and tuples) is an example of:

- A. Encapsulation
- B. Method Overriding
- C. Duck Typing
- D. Abstraction

Answer: C

39. What is the fundamental requirement for a method to be classified as a Static Method?

- A. It must be defined in the parent class.
- B. It must take self as the first argument.
- C. It does not operate on the instance (self) or the class (cls).
- D. It must be private.

Answer: C

40. If an object is no longer needed, Python's automatic process of reclaiming the memory it occupies is called:

- A. Memory Leaking
- B. Garbage Collection
- C. Destructor Calling
- D. Polymorphism

Answer: B