

Programação Orientada a Objetos

Instruções: Escolha a melhor resposta para cada questão.

Tópicos:

- Pilares da POO
- Classes
- Objetos
- Propriedades

Questões:

1. Qual dos seguintes NÃO é um pilar da Programação Orientada a Objetos (POO)?

- a) Abstração
- b) Encapsulamento
- c) Modularidade**
- d) Herança
- e) Polimorfismo

2. Qual conceito da POO permite que uma classe herde atributos e métodos de outra classe?

- a) Abstração
- b) Encapsulamento
- c) Herança**
- d) Polimorfismo
- e) Modularidade

3. Qual conceito da POO permite que um método tenha diferentes implementações em classes diferentes, mas com a mesma assinatura?

- a) Abstração
- b) Encapsulamento
- c) Herança
- d) Polimorfismo**
- e) Modularidade

4. Qual o principal objetivo da abstração na POO?

- a) Ocultar detalhes de implementação
- b) Criar novas classes a partir de classes existentes
- c) Definir métodos com diferentes implementações
- d) Criar objetos a partir de classes
- e) Agrupar métodos e atributos em uma estrutura lógica

5. O que é uma classe na POO?

- a) Uma instância de um objeto
- b) Um conjunto de instruções para criar objetos
- c) Um conjunto de métodos
- d) Um conjunto de propriedades
- e) Um objeto específico

6. O que é um objeto na POO?

- a) Uma instância de uma classe
- b) Um conjunto de instruções para criar objetos
- c) Um conjunto de métodos
- d) Um conjunto de propriedades
- e) Uma classe específica

7. Qual é a relação entre classes e objetos na POO?

- a) Classes são instâncias de objetos
- b) Objetos são instâncias de classes
- c) Classes e objetos são sinônimos
- d) Não existe relação entre classes e objetos
- e) Objetos são a estrutura base para a criação de classes

8. Qual o principal objetivo do encapsulamento na POO?

- a) Ocultar detalhes de implementação de uma classe

- b) Criar novas classes a partir de classes existentes
- c) Definir métodos com diferentes implementações
- d) Criar objetos a partir de classes
- e) Agrupar métodos e atributos em uma estrutura lógica

9. O que são propriedades em uma classe na POO?

- a) Métodos que definem o comportamento de um objeto
- b) Variáveis que armazenam dados de um objeto
- c) Classes que definem a estrutura de um objeto
- d) Objetos que representam a instância de uma classe
- e) Métodos que interagem com outras classes

10. Qual é a diferença entre métodos e propriedades em uma classe?

- a) Métodos definem o comportamento, enquanto propriedades armazenam dados
- b) Propriedades definem o comportamento, enquanto métodos armazenam dados
- c) Métodos e propriedades são sinônimos
- d) Não existe diferença entre métodos e propriedades
- e) Métodos são usados para criar objetos, enquanto propriedades são usadas para interagir com objetos

11. Qual o conceito da POO que permite a reutilização de código?

- a) Abstração
- b) Encapsulamento
- c) Herança
- d) Polimorfismo
- e) Modularidade

12. Qual dos seguintes é um exemplo de polimorfismo na POO?

- a) Criar uma classe com o mesmo nome de outra classe
- b) Criar um método com o mesmo nome em classes diferentes

- c) Criar uma classe que herda atributos e métodos de outra classe
- d) Criar uma classe com métodos abstratos
- e) Criar uma classe com um atributo privado

13. Qual o principal benefício da POO?

- a) Facilita a escrita de programas complexos
- b) Permite a criação de software mais eficiente
- c) Promove o reuso de código
- d) Aumenta a segurança do software
- e) Todas as alternativas acima

14. Qual o papel das interfaces na POO?

- a) Definir o comportamento de uma classe
- b) Criar objetos a partir de classes
- c) Implementar métodos abstratos
- d) Definir contratos que as classes devem seguir
- e) Implementar o encapsulamento

15. Qual dos seguintes é um exemplo de abstração na POO?

- a) Criar uma classe com o mesmo nome de outra classe
- b) Criar um método com o mesmo nome em classes diferentes
- c) Criar uma classe que herda atributos e métodos de outra classe
- d) Criar uma classe com métodos abstratos
- e) Criar uma classe com um atributo privado

16. Qual a diferença entre uma classe abstrata e uma interface?

- a) Classes abstratas podem ter implementações, enquanto interfaces não
- b) Interfaces podem ter implementações, enquanto classes abstratas não
- c) Classes abstratas e interfaces são sinônimos
- d) Não existe diferença entre classes abstratas e interfaces

e) Classes abstratas são usadas para criar objetos, enquanto interfaces são usadas para definir contratos

17. Qual o papel dos construtores na POO?

- a) Inicializar os atributos de um objeto
- b) Criar um objeto a partir de uma classe
- c) Definir o comportamento de um objeto
- d) Implementar métodos abstratos
- e) Implementar o encapsulamento

18. Qual a diferença entre atributos e métodos em uma classe?

- a) Atributos definem o comportamento de um objeto, enquanto métodos armazenam dados
- b) Métodos definem o comportamento de um objeto, enquanto atributos armazenam dados
- c) Atributos e métodos são sinônimos
- d) Não existe diferença entre atributos e métodos
- e) Atributos são usados para criar objetos, enquanto métodos são usados para interagir com objetos

19. Qual o papel do encapsulamento na segurança do software?

- a) O encapsulamento impede o acesso direto aos dados de um objeto
- b) O encapsulamento permite o acesso direto aos dados de um objeto
- c) O encapsulamento não tem relação com a segurança do software
- d) O encapsulamento aumenta o risco de erros no software
- e) O encapsulamento diminui a performance do software

20. Qual é o principal objetivo da modularidade na POO?

- a) Dividir o código em partes menores e mais fáceis de gerenciar
- b) Criar novas classes a partir de classes existentes
- c) Definir métodos com diferentes implementações
- d) Criar objetos a partir de classes

e) Agrupar métodos e atributos em uma estrutura lógica