

01) O que é POO?

- a) Uma linguagem de programação
- b) Um paradigma de programação
- c) Um modelo de banco de dados
- d) Um sistema operacional

02) Qual a principal vantagem da POO?

- a) Código mais fácil de entender e manter
- b) Melhor desempenho
- c) Menor consumo de memória
- d) Maior compatibilidade com outras linguagens

03) O que é uma classe em POO?

- a) Uma função que retorna um valor
- b) Uma estrutura de dados que contém propriedades e métodos
- c) Um objeto que contém outros objetos
- d) Um conjunto de variáveis globais

04) O que é um objeto em POO?

- a) Um conjunto de variáveis globais
- b) Uma estrutura de dados que contém propriedades e métodos
- c) Uma função que retorna um valor
- d) Uma classe abstrata

05) O que é herança em POO?

- a) Uma forma de reutilizar código de uma classe existente em uma nova classe
- b) Uma forma de proteger o código de uma classe
- c) Uma forma de criar uma cópia de uma classe existente
- d) Uma forma de alterar o comportamento de uma classe existente

06) O que é polimorfismo em POO?

- a) A capacidade de uma classe de ser estendida para outras classes
- b) A capacidade de uma classe de assumir várias formas
- c) A capacidade de uma classe de ser dividida em partes menores
- d) A capacidade de uma classe de ser usada em diferentes linguagens de programação

07) O que é encapsulamento em POO?

- a) A capacidade de esconder a implementação interna de uma classe
- b) A capacidade de usar uma classe em diferentes linguagens de programação
- c) A capacidade de alterar o comportamento de uma classe existente
- d) A capacidade de reutilizar código de uma classe existente em uma nova classe

08) O que é uma interface em POO?

a) Um conjunto de métodos e propriedades que uma classe deve implementar

b) Um tipo especial de classe abstrata

c) Um tipo de classe que pode ser instanciado diretamente

d) Uma forma de criar objetos sem usar classes

09) O que é composição em POO?

a) Uma forma de reutilizar código de uma classe existente em uma nova classe

b) Uma forma de criar uma cópia de uma classe existente

c) Uma forma de proteger o código de uma classe

d) Uma forma de criar um objeto composto por outros objetos

10) O que é agregação em POO?

a) Uma forma de criar um objeto composto por outros objetos

b) Uma forma de reutilizar código de uma classe existente em uma nova classe

c) Uma forma de proteger o código de uma classe

d) Uma forma de criar uma cópia de uma classe existente

11) Qual é o principal benefício da POO?

a) Maior modularidade e reutilização de código.

b) Menor modularidade e reutilização de código.

c) Maior complexidade e dificuldade de manutenção.

d) Menor complexidade e facilidade de manutenção.

12) O que é uma classe em Dart?

a) Uma instância de um objeto.

b) Um tipo de dado.

c) Um construtor de objetos.

d) Um modelo para criar objetos.

13) Como se define uma classe em Dart?

a) `class NomeDaClasse { }`

b) `class { }`

c) `classe NomeDaClasse { }`

d) `Classe NomeDaClasse { }`

14) O que é um objeto em Dart?

a) Um tipo de dado.

b) Uma instância de uma classe.

c) Um construtor de objetos.

d) Uma variável global.

15) O que é o construtor de uma classe em Dart?

a) Um método especial que é executado quando um objeto é criado.

b) Uma variável global.

c) Um tipo de dado.

d) Um método que não pode ser executado.

16) O que é a herança em POO?

a) Uma técnica que permite criar novas classes a partir de classes existentes.

b) Uma técnica que permite criar novos objetos a partir de objetos existentes.

c) Uma técnica que permite criar novas classes a partir de objetos existentes.

d) Uma técnica que permite criar novos objetos a partir de classes existentes.

17) O que é o polimorfismo em POO?

a) Uma técnica que permite que objetos de diferentes classes sejam tratados de maneira semelhante.

b) Uma técnica que permite que objetos de diferentes classes sejam tratados de maneira diferente.

c) Uma técnica que permite que objetos de uma mesma classe sejam tratados de maneira semelhante.

d) Uma técnica que permite que objetos de uma mesma classe sejam tratados de maneira diferente.

18) Qual é a diferença entre uma classe abstrata e uma classe concreta em Dart?

a) Uma classe abstrata não pode ser instanciada diretamente, enquanto uma classe concreta pode ser.

b) Uma classe abstrata pode ser instanciada diretamente, enquanto uma classe concreta não pode ser.

c) Uma classe abstrata não pode ser estendida, enquanto uma classe concreta pode ser.

d) Uma classe abstrata pode ser estendida, enquanto uma classe concreta não pode ser.

19) O que é uma interface em Dart?

a) Um tipo especial de classe que define um conjunto de métodos que uma classe deve implementar.

b) Um tipo de dado.

c) Uma classe que não pode ser instanciada diretamente.

d) Uma classe que não pode ser estendida.

20) O que é um relacionamento de associação em um diagrama de classes em POO?

a) É quando uma classe é uma extensão de outra classe, herdando seus atributos e métodos.

b) É quando uma classe usa a funcionalidade de outra classe sem herdar seus atributos e métodos.

c) É quando uma classe contém outra classe como propriedade.

d) É quando uma classe é composta por outras classes.