



Fichas de trabalho

2025.01 - MDCOO - Revisão - 2ºB

Total questions: 30

Tempo da planilha: 43 minutos

Nome do instrutor: Fernando Alves

Nome

Turma

Data

1. Dentro da temática de Orientação a Objetos, pode-se definir o seguinte exemplo: dada a classe CarroDePasseio, ela representa todos os atributos e métodos dos carros de passeio. Assim, essa classe consiste em um modelo para reservar um espaço de memória para manter e transformar os dados. A partir desse exemplo, assinale a alternativa que apresenta corretamente os conceitos de Atributo, Método e Instanciação.
 - a) Atributos são o comportamento que os elementos podem adotar, Método é o conjunto de características de um dado Elemento e Instanciação é a capacidade de criar classes a partir de um objeto.
 - b) Atributos são a capacidade que um Elemento tem de criar informações diferentes da Classe pai, Método é a característica que a classe tem de se comunicar com elementos externos e Instanciação é a capacidade da classe corrigir seus próprios erros.
 - c) Atributos são o conjunto de características de um dado Elemento, Método é o comportamento que os elementos podem adotar e Instanciação é a capacidade de criar objetos a partir de uma classe.
 - d) Atributos são o conjunto de exceções para criação de um Elemento, Método é o passo a passo para construir a classe e Instanciação é a capacidade de conexão entre outras classes.
2. Na orientação a objetos, há o conceito de encapsulamento que estabelece que
 - a) aspectos internos de implementação de uma classe não são visíveis por outras classes.
 - b) classes com métodos encapsulados não podem possuir atributos.
 - c) subclasses assumem características das superclasses correspondentes.
 - d) classes com métodos encapsulados não admitem ter subclasses.

3. João, Maria e José são clientes de um banco. O sistema de informatização do banco é capaz de cadastrar tais pessoas empregando características particulares como nome, endereço e outras informações julgadas relevantes. Além disso, o sistema pode realizar o cadastro ou a exclusão do cliente.

Dentro do conceito de programação orientada a objetos

I. João, Maria e José são exemplos de classes de um objeto que pode ser denominado CLIENTE;

II. as características que definem João, Maria e José são denominados atributos;

III. as operações de cadastro e exclusão de clientes são métodos implementados na classe.

- | | |
|--|--|
| a) se somente a afirmativa I estiver correta | b) se somente as afirmativas I e II estiverem corretas |
| c) se somente as afirmativas II e III estiverem corretas | d) se somente a afirmativa II estiver correta |
4. No domínio da orientação a objetos, a ideia de existência de dados e funcionalidades acessados somente pelos objetos de uma classe, de forma interna, e de que deva existir alguma forma protegida de acesso a esses dados e funcionalidades, de modo que objetos de classes externas os enxerguem, está associada aos conceitos, respectivamente, de:
- | | |
|--------------------------------|---------------------------------|
| a) associação e herança. | b) polimorfismo e virtualidade. |
| c) encapsulamento e interface. | d) atributo e método. |
5. Considere o cenário onde uma Classe B lega suas estruturas e comportamentos de uma Classe A. Essa relação entre a Classe A e a Classe B é caracterizada por:
- | | |
|----------------------------|----------------------------|
| a) um polimorfismo | b) uma instância da classe |
| c) uma instância do objeto | d) uma herança |
6. Um dos conceitos do paradigma orientado a objetos consiste na alteração do funcionamento interno de um método herdado de um objeto pai. Assinale a alternativa correta que apresenta este conceito.
- | | |
|-------------------|-----------------|
| a) Encapsulamento | b) Abstração |
| c) Herança | d) Polimorfismo |

- <https://wayground.com/print/quiz/686da8ae37de4a907eee70a7>

11. Na programação orientada a objetos, a instanciação de objetos tem o objetivo de:
- a) criar um objeto
 - b) abstrair os atributos de um objeto
 - c) inicializar uma classe
 - d) definir atributos e métodos de uma classe
12. Associe os Princípios de Orientação a Objetos com os seus significados:
- | | | |
|---|---|------------------|
| Permite ao usuário definir tipos de forma incremental, a partir de tipos existentes. | ○ | ○ Abstração |
| Define que uma determinada operação pode se comportar de diferentes formas em diferentes classes. | ○ | ○ Classe |
| Elucida apenas as propriedades comuns de um conjunto de objetos, omitindo os detalhes. | ○ | ○ Polimorfismo |
| Conjunto de objetos que possuem o mesmo tipo. | ○ | ○ Herança |
| Ocultamento de partes da classe que não podem ser acessada diretamente por outras. | ○ | ○ Encapsulamento |
13. A forma de reutilização de *software* em que novas classes adquirem os membros de outras já existentes e aprimoram essas classes com novas capacidades é denominada
- a) encapsulamento.
 - b) herança.
 - c) polimorfismo.
 - d) tratamento de exceções.

14. Procedures, funções e subrotinas são conceitos das técnicas de programação tradicionais que correspondem, nas técnicas orientadas ao objeto:
- a) à hereditariedade.
 - b) à classe.
 - c) aos métodos.
 - d) às variáveis de instância.
15. Na programação orientada a objetos, os métodos representam.
- a) os tipos de linguagens de programação utilizados
 - b) o tipo de herança existente entre as classes.
 - c) as associações estabelecidas entre as classes
 - d) a implementação das ações das classes definidas.
16. Na Programação orientada a objetos, é o princípio que oferece a capacidade de um método pode ser implementado de diferentes formas, ou mesmo de realizar coisas diferentes, ou seja, um único serviço pode oferecer variações, conforme se aplique a diferentes subclasses de uma superclasse.
- a) Abstração.
 - b) Herança.
 - c) Encapsulamento.
 - d) Polimorfismo.
17. No âmbito dos princípios de concepção e programação orientada a objeto, é correto afirmar que "um objeto da subclasse é um objeto da superclasse, ou seja, os objetos da subclasse podem ser tratados como objetos da superclasse". Esta afirmação é possível quando se refere ao contexto de
- a) Encapsulamento.
 - b) Abstração.
 - c) Herança.
 - d) Polimorfismo.
18. Um subconjunto de entidades, dentro de um conjunto de entidades, que tem atributos distintos das demais entidades do conjunto (refinamento em subgrupos topdown) denomina-se
- a) especialização.
 - b) herança.
 - c) sistematização.
 - d) generalização.

19. Uma técnica que consiste em separar aspectos externos dos internos da implementação de um objeto, isto é, determinados detalhes ficam ocultos aos demais objetos e dizem respeito apenas ao próprio objeto. Trata-se de:
- a) encapsulamento.
 - b) generalização.
 - c) polimorfismo.
 - d) herança.
20. Na programação orientada a objetos, há os métodos chamados de construtores, sobre os quais é correto afirmar que
- a) há, obrigatoriamente, pelo menos dois construtores em cada classe
 - b) sempre retornam um dos dois seguintes valores: falso ou verdadeiro.
 - c) são executados sempre que um atributo de um dado objeto for consultado.
 - d) são métodos executados na criação do objeto
21. São considerados princípios do Paradigma da Orientação a Objetos:
- a) Dados, Funções, Encapsulamento e Hereditariedade.
 - b) Abstrações (Classes, Dados, Atributos dos dados e Funções), Encapsulamento e Diagramas de Classe.
 - c) Abstração (Classes, Objetos, Atributos e Métodos), Encapsulamento, Herança e Polimorfismo.
 - d) Unicamente Classes, Encapsulamento, Hereditariedade e polimorfismo.
22. Considerando o paradigma da orientação a objeto, para o enunciado abaixo, assinale a alternativa correta.
Paulo pode andar, correr, pular. Ele tem 30 anos, é casado e trabalha com Tecnologia da Informação e Comunicação.
- a) Paulo é uma classe pois há outros Paulos que tem idade diferente de 30 anos e que exercem outras profissões.
 - b) Andar, correr e pular, são objetos que podem ter como variável Paulo.
 - c) Paulo é um objeto. Tem como métodos andar, correr e pular e, 30 anos, ser casado e trabalhar com TICs são atributos.
 - d) Ter 30 anos, ser casado, andar e correr são estados que podem ser associados ao objeto Paulo.

23. Marque a alternativa que NÃO representa uma característica da orientação a objetos.

- a) Polimorfismo.
- b) Encapsulamento.
- c) Recursão.
- d) Reutilização de Código.

24. Na programação orientada a objeto,

- a) métodos são variáveis que se encontram dentro de cada um dos objetos de uma classe.
- b) atributos são classes que se encontram dentro de cada um dos objetos restritos a determinados tipos.
- c) encapsulamento consiste na aglutinação de aspectos internos e externos de um objeto.
- d) um construtor serve para inicializar os atributos e é executado automaticamente sempre que ocorre a criação de um novo objeto.

25. Em conformidade com a metodologia orientada a objetos, com a finalidade de evitar que partes de um programa se tornem tão independentes que uma pequena alteração tenha grandes efeitos em cascata, é aplicado um recurso que separa os aspectos externos e acessíveis de um objeto dos detalhes internos de implementação.

Esse recurso utiliza um princípio da *Orientação a Objetos* que propõe ocultar determinados elementos de uma classe das demais classes. O objetivo ao colocar uma proteção ao redor é prevenir contra os efeitos colaterais indesejados ao ter essas propriedades modificadas de forma inesperada.

Este recurso é conhecido por:

- a) encapsulamento.
- b) coesão.
- c) acoplamento.
- d) polimorfismo.

26.

```
JS passaro.js > ...
1  class Passaro{
2      vida=3;
3      tamanho= 15;
4      #velocidade=0;
5      constructor(v,h){
6          this.#velocidade = v;
7          this.alturaSolo=h;
8      }
9      voar(){
10         this.alturaSolo++;
11     }
12     morrer(){
13         this.vida--;
14     }
15 }
16 let passaro = new Passaro(10,100);
17 console.log(passaro.alturaSolo)
```

Organize essas opções nas categorias certas

Categorize o seguinte

Passaro

Passaro

Vida

Tamanho

Velocidade

AlturaSolo

Voar()

Morrer()

Objeto

Classe

Atributos

Métodos

27.

```
JS passaro.js > ...
1  class Passaro{
2    vida=3;
3    tamanho= 15;
4    #velocidade=0;
5    constructor(v,h){
6      this.#velocidade = v;
7      this.alturaSolo=h;
8    }
9    voar(){
10     this.alturaSolo++;
11   }
12   morrer(){
13     this.vida--;
14   }
15 }
16 let passaro = new Passaro(10,100);
17 console.log(passaro.alturaSolo)
```

Qual valor será impresso no console do navegador ao executar o trecho do código ao lado?

- a) 3
- b) 0
- c) 10
- d) 100

28.

```
JS passaro.js > ...
1  class Passaro{
2    vida=3;
3    tamanho= 15;
4    #velocidade=0;
5    constructor(v,h){
6      this.#velocidade = v;
7      this.alturaSolo=h;
8    }
9    voar(){
10     this.alturaSolo++;
11   }
12   morrer(){
13     this.vida--;
14   }
15 }
16 let passaro = new Passaro(10,100);
17 console.log(passaro.alturaSolo)
```

Qual atributo é privado?

- a) vida
- b) alturaSolo
- c) velocidade
- d) tamanho

29.

```
JS passaro.js > ...
1  class Passaro{
2      vida=3;
3      tamanho= 15;
4      #velocidade=0;
5      constructor(v,h){
6          this.#velocidade = v;
7          this.alturaSolo=h;
8      }
9      voar(){
10         this.alturaSolo++;
11     }
12     morrer(){
13         this.vida--;
14     }
15 }
16 let passaro = new Passaro(10,100);
17 console.log(passaro.alturaSolo)
```

Como eu executo o método morrer ?

- a) passaro.morrer
- b) call method morrer;
- c) exec morrer
- d) passaro.morrer()

30.

```
JS passaro.js > ...
1  class Passaro{
2      vida=3;
3      tamanho= 15;
4      #velocidade=0;
5      constructor(v,h){
6          this.#velocidade = v;
7          this.alturaSolo=h;
8      }
9      voar(){
10         this.alturaSolo++;
11     }
12     morrer(){
13         this.vida--;
14     }
15 }
16 let passaro = new Passaro(10,100);
17 console.log(passaro.alturaSolo)
```

Considerando apenas o trecho do código ao lado, é possível deduzir que:

- a) O objeto passaro está sempre parado.
- b) O objeto passaro voa a uma altura constante.
- c) O objeto passaro voa para cima.
- d) O objeto passaro voa para baixo.

Chaves de resposta

1. c) Atributos são o conjunto de características de um dado Elemento, Método é o comportamento que os elementos podem adotar e Instanciação é a capacidade de criar objetos a partir de uma classe.
2. a) aspectos internos de implementação de uma classe não são visíveis por outras classes.
3. c) se somente as afirmativas II e III estiverem corretas
4. c) encapsulamento e interface.
5. d) uma herança
6. d) Polimorfismo
7. d) Apenas I e II.
8. b) agregação.
9. d) Polimorfismo
10. a) esconder ou ocultar detalhes da implementação de uma dada classe de outras classes.
11. a) criar um objeto
12. Elucida apenas as propriedades comuns de um conjunto de objetos, omitindo os detalhes.
 - Abstração, Conjunto de objetos que possuem o mesmo tipo.
 - Classe, Define que uma determinada operação pode se comportar de diferentes formas em diferentes classes.
 - Polimorfismo,

Permite ao usuário definir tipos de forma incremental, a partir de tipos existentes.

- Herança,

Ocultamento de partes da classe que não podem ser acessada diretamente por outras.

- Encapsulamento

- | | | |
|---|--|---|
| 13. b) herança. | 14. c) aos métodos. | 15. d) a implementação das ações das classes definidas. |
| 16. d) Polimorfismo. | 17. c) Herança. | 18. a) especialização. |
| 19. a) encapsulamento. | 20. d) são métodos executados na criação do objeto | 21. c) Abstração (Classes, Objetos, Atributos e Métodos), Encapsulamento, Herança e Polimorfismo. |
| 22. c) Paulo é um objeto. Tem como métodos andar, correr e pular e, 30 anos, ser casado e trabalhar com TICs são atributos. | 23. c) Recursão. | 24. d) um construtor serve para inicializar os atributos e é executado automaticamente sempre que ocorre a criação de um novo objeto. |
| 25. a) encapsulamento. | 26. Objeto | 27. d) 100 |

Passaro

Classe

Passaro

Atributos

Vida

Tamanho

Velocidade

AlturaSolo

Métodos

Voar()

Morrer()

28. c) velocidade

29. d) passaro.morrer()

30. c) O objeto passaro
voa para cima.