

1. Em Java, qual palavra-chave é usada para indicar que uma classe herda de outra?

- a) inherit
- b) super
- c) extends (X)
- d) implements

2. Qual palavra-chave permite que uma classe acesse a implementação da sua superclasse?

- a) parent
- b) super (X)
- c) base
- d) this

3. Em Java, uma classe pode herdar de quantas superclasses diretamente?

- a) Apenas 1 (X)
- b) 2
- c) 3
- d) Quantas forem necessárias

4. O que significa encapsulamento em Java?

- a) Reutilização de código por meio da herança
- b) Agrupamento de classes em pacotes
- c) Proteção dos atributos através de modificadores de acesso e métodos (X)
- d) Organização de código em interfaces

5. Qual modificador de acesso permite que um atributo seja acessado apenas dentro da própria classe?

- a) public
- b) protected
- c) private (X)
- d) default

6. O modificador de acesso protected permite acesso:

- a) Apenas na mesma classe
- b) Apenas no mesmo pacote
- c) Na mesma classe, no mesmo pacote e por subclasses (X)
- d) Em qualquer lugar do programa

7. Em herança, se uma subclasse precisa chamar o construtor da superclasse, qual palavra-chave deve ser usada?

- a) base()
- b) super() (X)
- c) parent()
- d) extends()

8. O que acontece se uma classe filha define um método com a mesma assinatura de um método da classe pai?

- a) Sobrecarga (overloading)

- b) Sobrescrita (overriding) (X)
- c) Encapsulamento
- d) Polimorfismo estático

9. Em relação à herança, qual é a diferença entre extends e implements?

- a) extends é usado para herdar classes, implements para interfaces (X)
- b) extends é usado para interfaces, implements para classes
- c) Ambos fazem a mesma coisa
- d) extends só funciona com classes abstratas

10. Se um atributo é declarado como private, como ele pode ser acessado por outras classes?

- a) Diretamente pelo nome
- b) Através de métodos públicos getters e setters (X)
- c) Apenas com herança
- d) Não pode ser acessado em nenhuma hipótese

11. Uma classe declarada como final:

- a) Não pode ser instanciada
- b) Não pode ser herdada (X)
- c) Não pode ter atributos
- d) Não pode ter métodos

12. Em Java, qual anotação é usada para indicar que um método sobrescreve um da superclasse?

- a) @Override (X)
- b) @Overload
- c) @Overwritten
- d) @Inherited

13. É possível herdar múltiplas classes diretamente em Java?

- a) Sim, sempre
- b) Sim, mas apenas classes abstratas
- c) Não, apenas uma classe pode ser herdada diretamente (X)
- d) Sim, mas somente no mesmo pacote

14. O encapsulamento promove qual dos seguintes benefícios?

- a) Acesso irrestrito a todos os atributos
- b) Segurança e controle sobre os dados da classe (X)
- c) Permite múltipla herança
- d) Reduz a necessidade de construtores

15. Qual dos seguintes é um exemplo correto de herança em Java?

- a) class Car extends Vehicle { } (X)
- b) class Car implements Vehicle { }
- c) class Car inherit Vehicle { }
- d) class Car super Vehicle { }

---

Quer que eu monte isso em formato de gabarito de prova (com tabela) ou prefere nesse estilo de lista com o X?