

Questões — Classe, Objeto e Alocação em Memória

Questão 1 — Conceito de classe

Em Java, uma **classe** pode ser definida como:

- a) Uma variável que armazena um objeto.
 - b) Um modelo ou estrutura que define atributos e comportamentos dos objetos.
 - c) Uma área da memória responsável exclusivamente por armazenar métodos.
 - d) Um comando utilizado para criar variáveis primitivas.
 - e) Uma instância de outra classe.
-

Questão 2 — Conceito de objeto

Considere a classe:

```
public class Pessoa {  
  
    String nome;  
    int idade;  
  
    void apresentar() {  
        System.out.println("Nome: " + nome);  
    }  
}
```

Qual das alternativas representa corretamente a criação de um **objeto** da classe Pessoa?

- a)
`Pessoa;`
 - b)
`Pessoa pessoa;`
 - c)
`new Pessoa();`
 - d)
`Pessoa pessoa = new Pessoa();`
 - e)
`object Pessoa = Pessoa();`
-

Questão 3 — Classe × objeto

Analise a situação:

Uma classe `Carro` define os atributos `marca`, `modelo` e `ano`, além dos métodos `acelerar()` e `frear()`.

Nesse contexto, assinale a alternativa correta:

- a) `Carro` é um objeto e `marca` é uma classe.
 - b) `Carro` é uma classe e um carro específico criado a partir dela é um objeto.
 - c) `acelerar()` é um objeto da classe `Carro`.
 - d) `marca` obrigatoriamente representa uma classe.
 - e) Um objeto não possui os atributos definidos pela classe.
-

Questão 4 — Palavra-chave `new`

Observe o código:

```
Pessoa p = new Pessoa();
```

Qual é a função da palavra-chave `new` nesse código?

- a) Declarar a classe `Pessoa`.
 - b) Criar uma variável primitiva do tipo `Pessoa`.
 - c) Criar uma nova instância da classe `Pessoa`.
 - d) Excluir um objeto da memória.
 - e) Copiar automaticamente todos os objetos existentes da classe `Pessoa`.
-

Questão 5 — Referência e objeto

Analise:

```
Pessoa p1 = new Pessoa();  
Pessoa p2 = p1;
```

Após a execução dessas instruções, é correto afirmar que:

- a) Dois objetos `Pessoa` diferentes foram criados.
 - b) `p1` e `p2` referenciam o mesmo objeto.
 - c) `p1` contém uma cópia completa de `p2`.
 - d) `p2` não ocupa espaço na memória.
 - e) `p1` deixa de existir após a segunda instrução.
-

Questão 6 — Criação de objetos

Considere:

```
public class Produto {  
    String nome;  
    double preco;  
}
```

Analise o código:

```
Produto p1 = new Produto();  
Produto p2 = new Produto();
```

Quantos objetos da classe `Produto` foram criados?

- a) Nenhum.
 - b) Um.
 - c) Dois.
 - d) Três.
 - e) Depende do valor de `preco`.
-

Questão 7 — Alocação em memória

Considere:

```
Pessoa pessoa = new Pessoa();
```

De maneira simplificada, em Java:

- a) A variável `pessoa` e o objeto são necessariamente armazenados na mesma região de memória.
 - b) `new Pessoa()` cria um objeto na memória, enquanto `pessoa` armazena uma referência para esse objeto.
 - c) `pessoa` armazena todos os métodos da classe.
 - d) `new` cria apenas uma variável primitiva.
 - e) O objeto é criado no Stack e a referência obrigatoriamente no Heap.
-

Questão 8 — Heap e Stack

Considere:

```
Pessoa p = new Pessoa();
```

Em uma explicação simplificada da memória da JVM, qual alternativa está correta?

- a) O objeto criado por `new Pessoa()` fica no Heap, enquanto a variável local `p` pode estar associada ao Stack.
 - b) O objeto fica obrigatoriamente no Stack.
 - c) A referência `p` fica obrigatoriamente no Heap e o objeto no Stack.
 - d) Tanto a referência quanto o objeto são sempre armazenados exclusivamente no Stack.
 - e) Java não utiliza Heap nem Stack.
-

Questão 9 — Múltiplas referências

Observe:

```
Pessoa p1 = new Pessoa();  
Pessoa p2 = p1;
```

```
p1.nome = "João";  
System.out.println(p2.nome);
```

Qual será a saída?

- a) null
 - b) João
 - c) p1
 - d) O código apresentará erro de compilação.
 - e) O código criará automaticamente uma nova Pessoa.
-

Questão 10 — Criação de objetos e memória

Considere:

```
Pessoa p1 = new Pessoa();  
Pessoa p2 = new Pessoa();  
Pessoa p3 = p1;
```

Quantos **objetos** foram criados e quantas **referências** existem?

- a) 3 objetos e 3 referências.
 - b) 2 objetos e 3 referências.
 - c) 3 objetos e 2 referências.
 - d) 1 objeto e 3 referências.
 - e) 2 objetos e 2 referências.
-

Questão 11 — Objeto sem referência

Considere:

```
Pessoa p = new Pessoa();  
p = null;
```

Depois da segunda instrução:

- a) O objeto é imediatamente destruído pelo Java.
 - b) O objeto passa a ser uma classe.
 - c) O objeto pode se tornar elegível para coleta pelo Garbage Collector, caso não exista outra referência para ele.
 - d) O objeto é movido automaticamente para o Stack.
 - e) O programa obrigatoriamente será encerrado.
-

Questão 12 — Questão discursiva

Explique, com suas palavras, a diferença entre:

- **classe;**
- **objeto;**
- **referência.**

Utilize um exemplo em Java para complementar sua resposta.

Questão 13 — Questão discursiva

Analise o código:

```
public class Aluno {  
    String nome;  
    int idade;  
}
```

```
Aluno aluno = new Aluno();
```

Explique o que acontece, de maneira simplificada, quando a instrução acima é executada, considerando a criação do objeto e a referência armazenada pela variável `aluno`.

Questão 14 — Análise de código

Analise:

```
public class Conta {  
    double saldo;  
}
```

```
Conta c1 = new Conta();  
Conta c2 = new Conta();
```

```
c1.saldo = 1000;  
c2.saldo = 500;
```

```
System.out.println(c1.saldo);  
System.out.println(c2.saldo);
```

Responda:

1. Quantos objetos foram criados?
 2. `c1` e `c2` referenciam o mesmo objeto?
 3. Qual será a saída do programa?
 4. Por que a alteração de `c1.saldo` não altera `c2.saldo`?
-

Questão 15 — Verdadeiro ou falso

Classifique as afirmações como **Verdadeiras (V)** ou **Falsas (F)**.

- () Uma classe é um modelo para criação de objetos.
- () Um objeto é uma instância de uma classe.

- () A palavra-chave `new` pode ser utilizada para criar objetos.
- () Duas variáveis de referência sempre representam dois objetos diferentes.
- () Duas referências podem apontar para o mesmo objeto.
- () Quando uma referência recebe `null`, o objeto é necessariamente destruído imediatamente.
- () O Heap é uma área de memória associada à criação e armazenamento de objetos na JVM.
- () Uma variável de referência e o objeto referenciado são conceitos diferentes.
-

Gabarito

Questões objetivas

Questão	Resposta
1	B
2	D
3	B
4	C
5	B
6	C
7	B
8	A
9	B
10	B
11	C

Questão 12 — Resposta esperada

Classe: é um modelo que define atributos e comportamentos que os objetos poderão possuir.

Objeto: é uma instância concreta de uma classe.

Referência: é uma variável que permite acessar um objeto.

Exemplo:

```
Pessoa pessoa = new Pessoa();
```

Nesse exemplo:

- `Pessoa` é a classe;
- `new Pessoa()` cria um objeto;
- `pessoa` é uma variável de referência que referencia esse objeto.

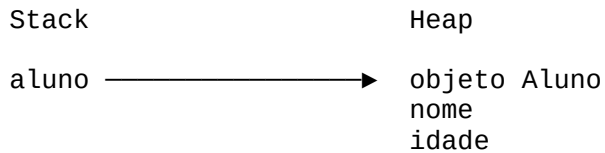
Questão 13 — Resposta esperada

Ao executar:

```
Aluno aluno = new Aluno();
```

é criada uma nova instância da classe `Aluno`. O objeto é alocado no **Heap** da JVM, enquanto a variável `aluno` mantém uma referência que permite acessar esse objeto.

De forma simplificada:



A variável `aluno` não representa a classe em si; ela representa uma **referência para uma instância** da classe.

Questão 14 — Resposta esperada

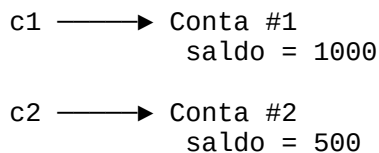
1. Quantos objetos foram criados?

Dois:

```
new Conta();  
new Conta();
```

2. `c1` e `c2` referenciam o mesmo objeto?

Não. Cada variável referencia um objeto diferente.



3. Saída:

```
1000.0  
500.0
```

4. Por quê?

Porque `c1` e `c2` possuem referências para **objetos diferentes**. Portanto, cada objeto possui seu próprio atributo `saldo`.

Questão 15 — Gabarito

Item	Resposta
------	----------

1	V
2	V
3	V
4	F
5	V

Item	Resposta
------	----------

6	F
---	---

7	V
---	---

8	V
---	---