



Formação Java

Para você que quer se tornar um EXPERT no assunto



<< 5/5 >>

Área do Aluno

[Home do aluno](#)[Meus dados](#)[Meus pedidos](#)[Meus cursos](#)[Atendimento](#)[Sair](#)

Fundamentos de Java 10

Avaliação

1) A sobrecarga de métodos é o processo de definir, na mesma classe, mais de um método com o mesmo nome. Porém, métodos sobrecarregados devem, obrigatoriamente, ter os parâmetros diferentes. Esta afirmação é verdadeira ou falsa?

- ☒ Verdadeira (alternativa correta)
- ☐ Falso (alternativa incorreta)

Correto**Resposta correta!**

A afirmação é verdadeira, uma vez que os parâmetros podem ser alterados. Esta alteração pode ser tanto em quantidade quanto no tipo de dado dos parâmetros passados.

2) Quais das alternativas abaixo são corretas a respeito do garbage collector?

* Marque todas as alternativas que respondem o enunciado da questão

- ☐ A chamada `System.gc()` força a execução do garbage collector. (alternativa incorreta)
- ☐ Devido ao garbage collector, uma aplicação Java nunca fica sem memória para executar. (alternativa incorreta)
- ☒ Não é possível saber quando o garbage collector será executado. (alternativa correta)
- ☒ O mecanismo do garbage collector é dependente de implementação (isto é, JVMs diferentes podem implementá-lo de formas diferentes). (alternativa correta)
- ☐ O garbage collector, em situações onde existe pouca memória disponível, pode remover da memória áreas que ainda podem ser acessadas pelo programa, mudando os valores de algumas variáveis. (alternativa incorreta)

Correto**Resposta correta!**

O garbage collector é um mecanismo que pode ser implementado de diversas formas (depende da empresa que implementa a JVM). O desenvolvedor não consegue executá-lo manualmente, ficando esta tarefa a cargo da JVM (que toma a decisão de executá-lo no momento em que achar apropriado). O garbage collector limpa apenas áreas de memória localizadas no heap que não são mais acessíveis pela aplicação. Logo, se a aplicação utiliza toda a memória disponível e continua referenciando toda esta memória que foi alocada, ela pode sim ficar sem memória para executar.

3) O que acontece quando o código abaixo é compilado e executado?

```
01 public class Exemplo {
02     int y = 10;
03
04     void mudar(Exemplo e) {
05         e.y = y + 5;
06     }
07     public static void main(String[] args) {
08         Exemplo e1 = new Exemplo();
09         Exemplo e2 = new Exemplo();
10         e1.mudar(e2);
11         System.out.println(e1.y);
12         System.out.println(e2.y);
13     }
14 }
```

- ☐ Imprime 10 e 10 (alternativa incorreta)
- ☐ Imprime 15 e 10 (alternativa incorreta)
- ☐ Imprime 15 e 15 (alternativa incorreta)
- ☒ Imprime 10 e 15 (alternativa correta)
- ☐ O código não compila devido a problemas na linha 11 (alternativa incorreta)

Correto**Resposta correta!**

Quando o objeto e2 é passado como parâmetro para o método `mudar()`, o valor do atributo `y` do objeto assume o valor inicial de `y` do objeto `e1` (10) acrescido de 5. O objeto `e1` não sofre qualquer alteração, mas o objeto `e2`, sim. É importante lembrar que variáveis que representam objetos possuem, na verdade,

referências a objetos criados no heap. Logo, quando um objeto é alterado, todas as referências que apontam para aquele objeto sofrem o efeito da alteração.

4) Qual afirmação abaixo é falsa com relação às classes e objetos?

- ☒ Classes e objetos são conceitos bastante parecidos, significando a mesma coisa em alguns casos. **(alternativa correta)**
- ☐ A classe representa uma estrutura de dados, enquanto um objeto da classe é algo criado com base nesta estrutura. **(alternativa incorreta)**
- ☐ Diversos objetos podem ser criados com base na mesma classe. **(alternativa incorreta)**
- ☐ Objetos de uma classe podem se comunicar (isto é, chamar métodos) com os objetos de outras classes. **(alternativa incorreta)**
- ☐ Um objeto sempre é criado com base em uma classe. Esta criação de objetos é também chamada de instanciação. **(alternativa incorreta)**

Correto

Resposta correta!

Todas as afirmações são corretas, exceto a que diz que os conceitos de classe e objeto podem ser iguais em alguns casos. Classes e objetos são dois conceitos completamente diferentes. Enquanto a classe representa um tipo de dados (ou uma estrutura de dados), o objeto é a materialização da classe. Imagine a classe como se fosse a planta de uma casa, que define como será a sua estrutura; e o objeto como se fosse a casa construída com base nessa planta.

5) O que acontece quando o código abaixo é compilado e executado?

```
01 public class Exemplo {
02     int i;
03     boolean b;
04     public static void main(String[] args) {
05         Exemplo e = new Exemplo();
06         e.i = e.i + 10;
07         System.out.println(e.i);
08         System.out.println(e.b);
09     }
10 }
```

- ☒ Imprime 10 e false **(alternativa correta)**
- ☐ O código não compila porque os atributos i e b não foram inicializados **(alternativa incorreta)**
- ☐ Imprime -1 e false **(alternativa incorreta)**
- ☐ Imprime 10 e true **(alternativa incorreta)**
- ☐ Não imprime qualquer valor **(alternativa incorreta)**

Correto

Resposta correta!

Quando os atributos não forem explicitamente inicializados, o Java os inicializa com valores padrão. No caso de atributos do tipo `int`, o valor 0 é assumido. No caso de atributos do tipo `boolean`, `false` é assumido.

[.: Retornar para a página do curso](#)