

# Lista de Exercícios 2 - Programação Orientada a Objetos

## Orientações:

- Entrega deve ser realizada via Tarefa do Microsoft Teams
- Entregar as respostas em arquivo formato PDF com identificação (Nome, Curso e Matrícula.)

## Objetivos da Lista:

- Diferença entre classe e objeto
- Diferença entre variáveis primitivas e variáveis de referência
- Construtores, modificadores de acesso e *static*

Considere o código apresentado na Figura 1 para responder as Questões 1 - 6

```
1 public class Nave{
2     private String nome;
3     private int vida;
4     public Nave(String nome, int vida){
5         this.nome = "Nave";
6         this.vida = 70;
7     }
8     //Getter e Setter omitido
9 }
10 public class Principal{
11     public static void main(String args[]) {
12         Nave n1 = new Nave("Nave1", 100);
13         Nave n2 = new Nave("Nave2", 200);
14         if(n1 != n2)
15             n1.setVida(50);
16         else
17             n2.setVida(70);
18         n2 = n1;
19
20         System.out.println(n2.getVida() + " " + n1.getVida());
21     }
22 }
```

Figura 1:

**Questão 1** O que será impresso no terminal ao executar esse código?

**Questão 2** Explique o que as variáveis n1 e n2 significam. O que elas possuem salvo na memória?

**Questão 3** Qual o resultado do teste condicional da linha 14? Justifique sua resposta

**Questão 4** O que está ocorrendo na linha 18? Qual o resultado dessa atribuição? Justifique sua resposta!

---

**Questão 5** Qual o valor dos membros da classe Nave após a inicialização?

**Questão 6** O que significa o operador “this”? O que `this.nome` significa? Nesse exemplo, dentro do construtor da Nave, faz alguma diferença usar ou não o `this`? Justifique!

Considere o código apresentado na Figura 2 para responder as Questões 7 - 10. Considere também que não existe erro de compilação.

```
1  public class Nave {
2      public static int cont = 0;
3      public int vida;
4      private int numNaves;
5      private String nome;
6      public Nave(){
7          this.vida = 100;
8          cont++;
9      }
10     public Nave(String nome){
11         this.nome = nome;
12         this.numNaves++;
13     }
14     public void mostraInfo(){
15         System.out.println(vida);
16     }
17     public void trocaVida(Nave nave) {
18         this.vida = nave.getVida();
19         nave.setVida(this.vida);
20     }
21     //Getter e Setter omitido
22 }
23 public class Main{
24
25     public static void main(String args[]) {
26         Nave n1 = new Nave();
27         Nave n2 = new Nave();
28         Nave n3 = new Nave();
29         n3.setVida(250);
30
31         n2.trocaVida(n3);
32         System.out.println(n2.cont);
33         n1.mostraInfo();
34         n2.mostraInfo();
35     }
36 }
```

Figura 2:

**Questão 7** O que será impresso na tela ao executar o código na linha 32? Justifique esse valor impresso.

**Questão 8** A variável membro “vida” (linha 3) está com o modificador `public`. O que isso significa?

---

Seria possível fazer na classe `Main` uma chamada `System.out.println(n1.vida)`? Caso não seja possível, quais alternativas permitem ter acesso ao conteúdo da variável “vida”? Justifique sua resposta

**Questão 9** A variável membro `cont` está sendo incrementada? Explique o funcionamento dessa variável. Qual o valor máximo que ela assume nesse código?

**Questão 10** Seria possível no método `main` executar o código `Nave n4 = new Nave("Nave1")`? Justifique.

**Questão 11** O que significa comportamento da classe? E estado da classe? Dê exemplos.

**Questão 12** O que é o construtor da classe? Quando ele é invocado? Ele pode ser parametrizado? Ele precisa ficar explícito? Justifique.