

## Orientação a Objetos

# Prova JAVA

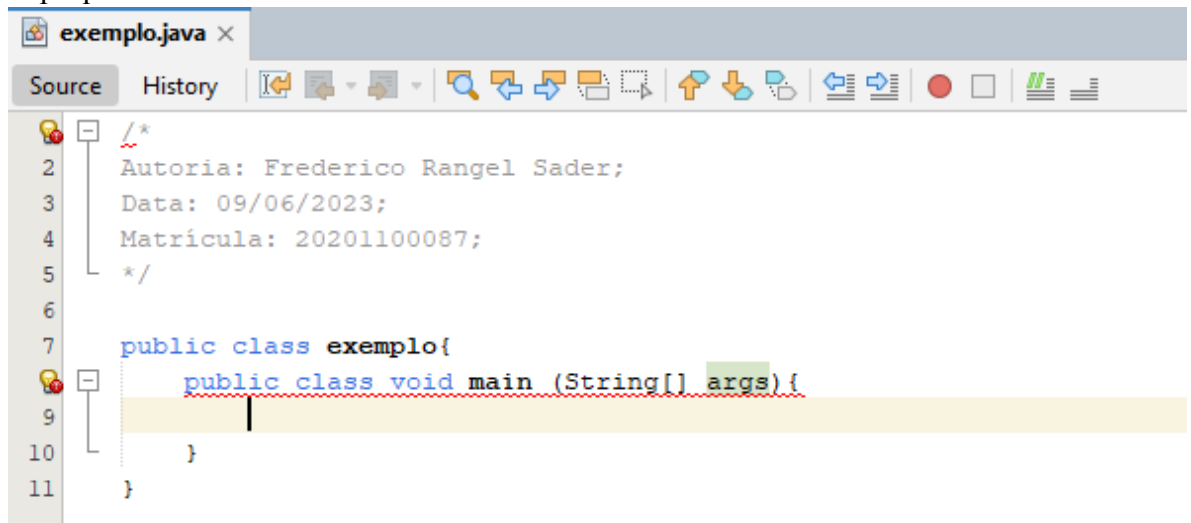
Nome Completo: Frederico Rangel Sader  
Data: 31 de maio de 2023

1. **(1 pontos)** Responda detalhadamente as seguintes questões (**exemplifique**):

a. Qual é a diferença entre uma classe e um objeto?

**Uma classe é uma estrutura que define as características e o comportamento de um objeto, enquanto um objeto é uma instância concreta de uma classe, que possui valores reais para os atributos e pode executar os métodos definidos na classe. A classe é como um molde, enquanto o objeto é o objeto real criado a partir desse molde ou planta.**

b. Explique como se define uma classe em Java



```
exemplo.java x
Source History
1  /*
2  Autoria: Frederico Rangel Sader;
3  Data: 09/06/2023;
4  Matrícula: 20201100087;
5  */
6
7  public class exemplo{
8      public class void main (String[] args){
9
10     }
11 }
```

‘public’ sendo o modificador de acesso, sendo o que controla a visibilidade da classe, podendo ter outras propriedades. ‘class’ cria a classe. ‘exemplo’ sendo o nome da classe.

c. Por que uma classe é uma abstração lógica?

**Por ser uma generalização de detalhes de certos objetos, e por buscar simplificar entes até uma forma básica, pode ser considerada uma abstração lógica.**

d. Como os objetos são criados?

**Os objetos são criados através da instanciação. A instanciação é o ato de criar uma instância única de uma classe, ou seja, criar um objeto específico com base em uma definição de classe.**

- e. O que é coleta de lixo e como ela funciona?

**A coleta de lixo é um mecanismo automático que gerencia a memória em um programa. Ele identifica e libera a memória ocupada por objetos que não são mais acessíveis ou utilizados pelo programa. Isso tira a necessidade de liberar manualmente a memória alocada pelos objetos inutilizados, tornando o processo mais conveniente e evitando vazamentos de memória.**

2. **(1 pontos)** Desenhar todas as classes, subclasses e superclasses associadas aos seguintes objetos, indicando em cada caso, pelo menos 2 atributos e 2 métodos, e a relação de herança entre classes

- a. Docente

superclasse;
Docente;
<ul style="list-style-type: none"> <li>- nome;</li> <li>- departamento;</li> </ul>
<ul style="list-style-type: none"> <li>- darAula();</li> <li>- fazPesquisa();</li> </ul>



subclasse;	subclasse;
Professor;	Assistente;
<ul style="list-style-type: none"> <li>- Disciplina;</li> <li>- Grau de formação;</li> </ul>	<ul style="list-style-type: none"> <li>- Projeto;</li> <li>- Experiência;</li> </ul>
<ul style="list-style-type: none"> <li>- orientaAluno();</li> <li>- corrigeProva();</li> </ul>	<ul style="list-style-type: none"> <li>- organizarAulas();</li> <li>- darMonitoria();</li> </ul>

- b. Verdura

superclasse;
Verdura;
<ul style="list-style-type: none"> <li>- nome;</li> <li>- cor;</li> </ul>

<ul style="list-style-type: none"> <li>- cozinhar();</li> <li>- armazenar();</li> </ul>
---



subclasse;	subclasse;
Folhosa;	raiz;
<ul style="list-style-type: none"> <li>- tipoDeFolha;</li> <li>- textura;</li> </ul>	<ul style="list-style-type: none"> <li>- parteComestível;</li> <li>- origem;</li> </ul>
<ul style="list-style-type: none"> <li>- prepararSalada();</li> <li>- lavarFolha();</li> </ul>	<ul style="list-style-type: none"> <li>- prepararSopa();</li> <li>- descascar();</li> </ul>

c. Ave

superclasse;
Ave;
<ul style="list-style-type: none"> <li>- nome;</li> <li>- habitat;</li> </ul>
<ul style="list-style-type: none"> <li>- voar();</li> <li>- fazerNinho();</li> </ul>



subclasse;
Pássaro;
<ul style="list-style-type: none"> <li>- corPenas;</li> <li>- envergadura;</li> </ul>
<ul style="list-style-type: none"> <li>- cantar();</li> <li>- migrar();</li> </ul>



subclasse;
------------

Predador;
<ul style="list-style-type: none"> <li>- tipoPresas;</li> <li>- velocidadeVoo;</li> </ul>
<ul style="list-style-type: none"> <li>- caçar();</li> <li>- descansar();</li> </ul>

d. Computador

superclasse;
Computador;
<ul style="list-style-type: none"> <li>- marca;</li> <li>- modelo;</li> </ul>
<ul style="list-style-type: none"> <li>- ligar();</li> <li>- desligar();</li> </ul>



subclasse;
Desktop;
<ul style="list-style-type: none"> <li>- peso;</li> <li>- tamanho;</li> </ul>
<ul style="list-style-type: none"> <li>- conectarInternet();</li> <li>- reproduzirMidia();</li> </ul>



subclasse;
DesktopGamer;
<ul style="list-style-type: none"> <li>- Placa de vídeo;</li> <li>- Overclock;</li> </ul>
<ul style="list-style-type: none"> <li>- jogarGame();</li> <li>- receberDadosGrandes();</li> </ul>

e. Rede

superclasse;
Rede;
<ul style="list-style-type: none"><li>- nome;</li><li>- velocidade;</li></ul>
<ul style="list-style-type: none"><li>- conectarDispositivo();</li><li>- monitorarTrafego();</li></ul>



subclasse;
RedeLocal;
<ul style="list-style-type: none"><li>- quantidadeDispositivos;</li><li>- enderecoIP;</li></ul>
<ul style="list-style-type: none"><li>- CompartilharArquivos();</li><li>- monitorarDispositivos();</li></ul>



subclasse;
RedeSemFio;
<ul style="list-style-type: none"><li>- alcance;</li><li>- canalFrequencia;</li></ul>
<ul style="list-style-type: none"><li>- conectarDispositivosMovéis();</li><li>- gerenciarSenha();</li></ul>

3. **(01 ponto)** Dar cinco exemplos de cada operação abstrata em orientação a objetos. Ilustre com imagens ou diagramas

	<b>Ex. 1</b>	<b>Ex. 2</b>	<b>Ex. 3</b>
<b>Instanciação</b> (classe para objeto)	Pessoa→João	Cidade→Campos	Estado→RJ
<b>Classificação</b> (objeto para classe)	Coca Cola→Refrigerantes	Fusca→Carros	Indicador→Dedos
<b>Generalização</b> (Remover detalhes de classes)	Salas→Cômodos da casa	Cardiologistas→Médicos	Humanos→Mamíferos
<b>Especialização</b> (especificar uma classe)	Programador→Programador em C	Animais→Animais Domésticos	Humanos→Loiros
<b>Agregação</b> (Objetos simples para complexos)	Tela→Televisão	Tampa→Garrafa	Cordas→Violão
<b>Decomposição</b> (objetos complexos para simples)	Prédio→Colunas	Beatles→Lennon	Matriz→Linha

4. **(01 ponto)** Escreva um programa em Java que execute as quatro operações aritméticas de números reais

The screenshot shows an IDE window titled 'Ex4.java' with a source code editor and an output console. The source code is a Java program named 'Ex4' that uses a 'Scanner' to read two numbers from the user and then performs arithmetic operations: addition, subtraction, multiplication, and division. The output console shows the program's execution with the user inputting 87 and 56, followed by the calculated results for each operation. The program runs successfully in 6 seconds.

```
Ex4.java x
Source History
/*
2  Autoria: Frederico Rangel Sader;
3  Data: 09/06/2023;
4  Matricula:20201100087;
5  */
6  import java.util.Scanner;
7
8  public class Ex4 {
9      public static void main(String[] args) {
10         Scanner scanner = new Scanner(System.in);
11
12         System.out.print("Digite o primeiro numero: ");
13         double numero1 = scanner.nextDouble();
14
15         System.out.print("Digite o segundo numero: ");
16         double numero2 = scanner.nextDouble();
17
18         double soma = numero1 + numero2;
19         double subtracao = numero1 - numero2;
20         double multiplicacao = numero1 * numero2;
21         double divisao = numero1 / numero2;
22
23         System.out.println("Resultado da soma: " + soma);
24         System.out.println("Resultado da subtração: " + subtracao);
25         System.out.println("Resultado da multiplicação: " + multiplicacao);
26         System.out.println("Resultado da divisão: " + divisao);
27
28         scanner.close();
29     }
30 }
Ex4 main
Variables Breakpoints Output x
Running Single Java File x Ex4 (run) #2 x
run:
Digite o primeiro numero: 87
Digite o segundo numero: 56
Resultado da soma: 143.0
Resultado da subtração: 31.0
Resultado da multiplicação: 4872.0
Resultado da divisão: 1.5535714285714286
BUILD SUCCESSFUL (total time: 6 seconds)
Ex4 (run) running_ 15:46 INS
```

5. (01 ponto) Escreva um programa JAVA que calcule a média de cinco notas. O programa deve mostrar o valor da média e indicar se foi aprovado ou reprovado

The screenshot shows an IDE window titled 'Ex5.java'. The code is as follows:

```
1  /*
2  Autoria: Frederico Rangel Sader;
3  Data: 09/06/2023;
4  Matricula:20201100087;
5  */
6
7  import java.util.Scanner;
8
9  public class Ex5 {
10     public static void main(String[] args) {
11         Scanner scanner = new Scanner(System.in);
12
13         System.out.println("Digite as cinco notas:");
14         double soma = 0;
15
16         for (int i = 1; i <= 5; i++) {
17             System.out.print("Nota " + i + ": ");
18             double nota = scanner.nextDouble();
19             soma += nota;
20         }
21
22         double media = soma / 5;
23         System.out.println("Média: " + media);
24
25         if (media >= 7.0) {
26             System.out.println("Aprovado");
27         } else {
28             System.out.println("Reprovado");
29         }
30     }
31 }
```

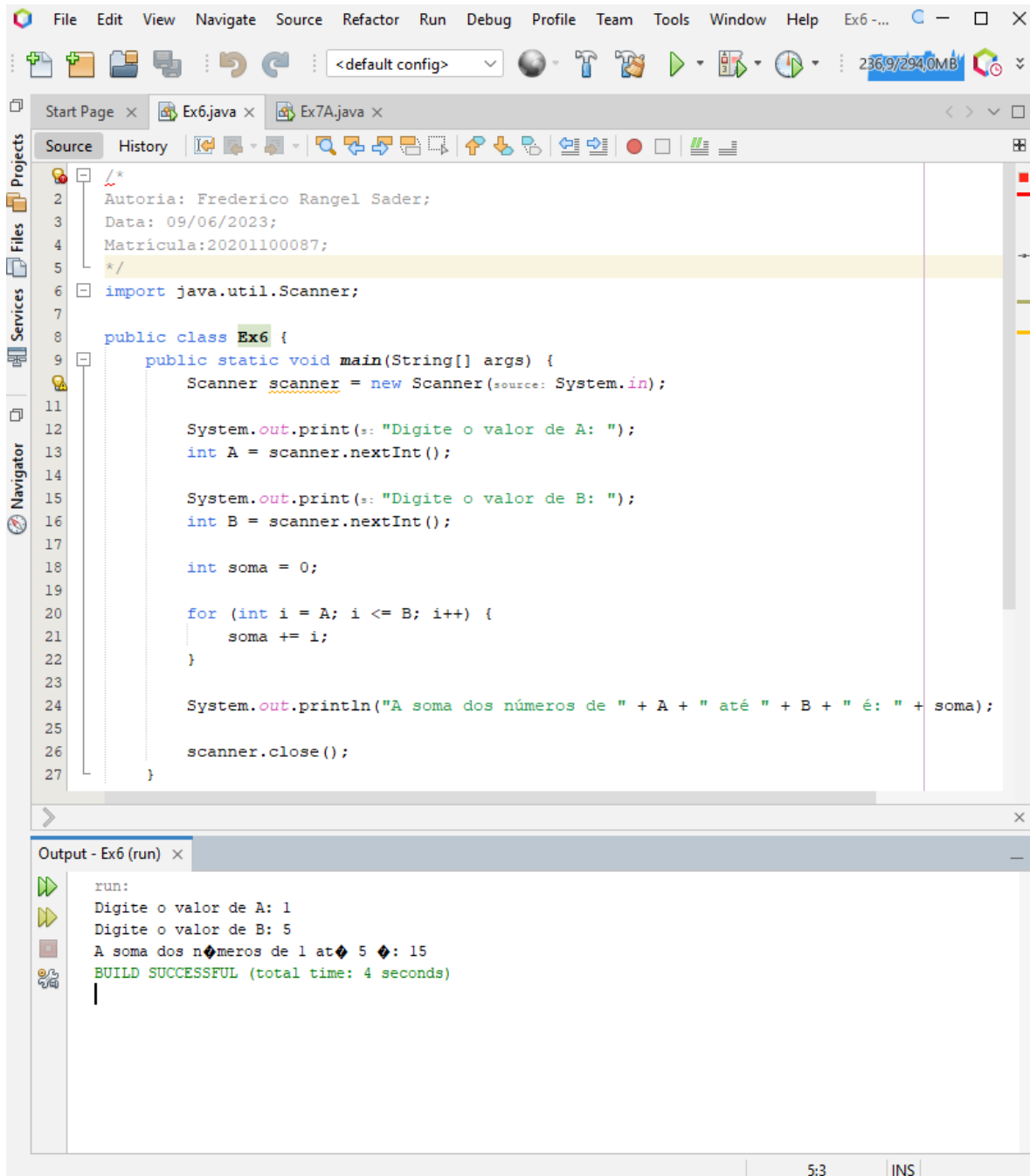
Below the code editor is the 'Output - Ex5 (run)' window, which displays the following text:

```
run:
Digite as cinco notas:
Nota 1: 8
Nota 2: 9
Nota 3: 6
Nota 4: 6
Nota 5: 5
Média: 6.8
Reprovado
BUILD SUCCESSFUL (total time: 9 seconds)
```

The status bar at the bottom indicates 'Ex4 (run) running...' and '5:3 INS Windows (CRLF)'.

6. **(01 ponto)** Escreva um programa JAVA que calcule a soma dos inteiros maiores o iguais a um número inteiro A e menores ou iguais a um inteiro B.





7. **(2 pontos)** Para cada caso da questão 2, escrever um programa JAVA, mostrando as classes, as operações (métodos), a criação de objetos e outras características da Programação Orientada a Objetos.

Start Page x Ex7A.java x Docente.java x Professor.java x

Source History

```
1  /*
2  |  Autoria: Frederico Rangel Sader;
3  |  Data: 09/06/2023;
4  |  Matricula:20201100087;
5  |  */
6  |  package ex7.a;
7  |
8  |  import java.util.Scanner;
9  |
10 |  public class Ex7A {
11 |      public static void main(String[] args) {
12 |          Scanner input = new Scanner (source: System.in);
13 |          Professor cadastro = new Professor ();
14 |
15 |          System.out.print(=: "Digite o nome:");
16 |          cadastro.nome = input.nextLine();
17 |
18 |          System.out.print(=: "Digite o departamento:");
19 |          cadastro.departamento = input.nextLine();
20 |
21 |          System.out.print(=: "Digite a disciplina:");
22 |          cadastro.disciplina = input.nextLine();
23 |
24 |          System.out.print(=: "Digite o grau de formacao('1' para bacharel);('2'para mestrado);('3' para doutorado):");
25 |          cadastro.grauFormacao = input.nextInt();
26 |
27 |          if (cadastro.grauFormacao == 3){
28 |              System.out.print(=: "Pode trabalhar na UENF");
29 |          }else{
30 |              System.out.print(=: "Nao pode trabalhar na UENF");
31 |          }
32 |      }
33 |  }
34 |  }
```

ex7.a.Ex7A > main > if (cadastro.grauFormacao == 3) else >

Output - Ex7.a (run) #2 x

run:

```
run:
>> Digite o nome:Pedro
>> Digite o departamento:CCT
>> Digite a disciplina:PLP
>> Digite o grau de formacao('1' para bacharel);('2'para mestrado);('3' para doutorado):2
%> Nao pode trabalhar na UENFBUILD SUCCESSFUL (total time: 21 seconds)
```

Start Page x Ex7A.java x Docente.java x Professor.java x

Source History

```
1  /*
2  |  Autoria: Frederico Rangel Sader;
3  |  Data: 09/06/2023;
4  |  Matrícula:20201100087;
5  |  */
6  |  package ex7.a;
7  |
8  |  public class Professor extends Docente {
9  |
10 |      String disciplina;
11 |      int grauFormacao;
12 |
13 |  }
```

Start Page x Ex7b.java x Folhosa.java x Verdura.java x

Source History

```
1  /*
2  Aatoria: Frederico Rangel Sader;
3  Data: 09/06/2023;
4  Matrícula:20201100087;
5  */
6  package ex7b;
7
8  import java.util.Scanner;
9
10 public class Ex7b {
11
12     public static void main(String[] args) {
13         Scanner input = new Scanner (source: System.in);
14         Folhosa cadastro = new Folhosa ();
15
16         System.out.print(=: "Digite a familia da folha:");
17         cadastro.nome = input.nextLine();
18
19         System.out.print(=: "Digite a cor:");
20         cadastro.cor = input.nextLine();
21
22         System.out.print(=: "Digite o tipo de folha:");
23         cadastro.tipoDeFolha = input.nextLine();
24
25         System.out.print(=: "Digite a textura:");
26         cadastro.textura = input.nextLine();
27
28         cadastro.lavaFolha();
29     }
30
31 }
32
```

ex7b.Ex7b main

Output - Ex7b (run) #2 x

```
run:
Digite a familia da folha:Hortaliça
Digite a cor:Verde
Digite o tipo de folha:Alface
Digite a textura:Macia
A folha ja foi lavada?[1 para sim] [2 para nãoo]:2
Entãoo nãoo estãoo pronto para cozinhar! Lave-a!BUILD SUCCESSFUL (total time: 54 seconds)
```






Start Page × Ex7b.java × Folhosa.java × Verdura.java ×

Source History

```
1  /*
2   * Autoria: Frederico Rangel Sader;
3   * Data: 09/06/2023;
4   * Matricula:20201100087;
5   */
6   package ex7b;
7
8   import java.util.Scanner;
9
10  public class Folhosa extends Verdura {
11      String tipoDeFolha;
12      String textura;
13
14      public void lavaFolha() {
15          Scanner input = new Scanner (source: System.in);
16          int lavado;
17
18          System.out.print(s: "A folha ja foi lavada?[1 para sim] [2 para não]:");
19          lavado = input.nextInt();
20
21          if(lavado == 1){
22              System.out.print(s: "Então pronto para cozinhar!");
23          }else{
24              System.out.print(s: "Então não está pronto para cozinhar! Lave-a!");
25          }
26      }
27  }
28
```

ex7b.Folhosa > lavaFolha >

Output - Ex7b (run) #2 ×

 run:  
 Digite a familia da folha:Hortaliça  
 Digite a cor:Verde  
 Digite o tipo de folha:Alface  
 Digite a textura:Macia  
A folha ja foi lavada?[1 para sim] [2 para não]:2  
Então não está pronto para cozinhar! Lave-a!BUILD SUCCESSFUL (total time: 54 seconds)

Start Page x Ex7c.java x Ave.java x Passaro.java x

Source H

Shift + click to close all documents  
Alt + click to close other documents

```
1  /*
2  Autoria: Frederico Rangel Sader;
3  Data: 09/06/2023;
4  Matrícula:20201100087;
5  */
6  package ex7c;
7
8  import java.util.Scanner;
9
10 public class Ex7c {
11     public static void main(String[] args) {
12         Scanner input = new Scanner (source: System.in);
13         Passaro cadastro = new Passaro ();
14
15         System.out.print(s: "Digite a especie:");
16         cadastro.especie = input.nextLine();
17
18         System.out.print(s: "Digite o habitat:");
19         cadastro.habitat = input.nextLine();
20
21         System.out.print(s: "Digite a cor das penas:");
22         cadastro.corPenas = input.nextLine();
23
24         System.out.print(s: "Digite a envergadura(cm):");
25         cadastro.envergadura = input.nextFloat();
26
27         cadastro.migracao();
28     }
29 }
30
31
```

ex7c.Ex7c > main >

Output - Ex7c (run) x

```
run:
Digite a especie:Vira-Pedras
Digite o habitat:Tundra
Digite a cor das penas:Marrom
Digite a envergadura(cm):15,5
Estamos no inverno? [1 para sim] [2 para nao]:1
O passaro esta em migracao!!BUILD SUCCESSFUL (total time: 25 seconds)
```

Start Page x Ex7c.java x Ave.java x Passaro.java x

Source History

```
1  /*
2  |  Autoria: Frederico Rangel Sader;
3  |  Data: 09/06/2023;
4  |  Matricula:20201100087;
5  |  */
6  package ex7c;
7
8  import java.util.Scanner;
9
10 public class Passaro extends Ave{
11     String corPenas;
12     Float envergadura;
13
14     public void migracao() {
15         int inverno;
16         Scanner input = new Scanner (source: System.in);
17
18         System.out.print(s: "Estamos no inverno? [1 para sim] [2 para nao]:");
19         inverno = input.nextInt();
20
21         if (inverno == 1){
22             System.out.print(s: "O passaro esta em migracao!!");
23         }else{
24             System.out.print(s: "O passaro nao esta em migracao!!");
25         }
26     }
27 }
28
29
```

ex7c.Passaro envergadura

Output - Ex7c (run) x

```
run:
Digite a especie:Vira-Pedras
Digite o habitat:Tundra
Digite a cor das penas:Marrom
Digite a envergadura(cm):15,5
Estamos no inverno? [1 para sim] [2 para nao]:1
O passaro esta em migracao!!BUILD SUCCESSFUL (total time: 25 seconds)
```

Start Page × Ex7d.java × Computador.java × Desktop.java ×

Source History

```
1  /*
2  Autoria: Frederico Rangel Sader;
3  Data: 09/06/2023;
4  Matrícula:20201100087;
5  */
6  package ex7d;
7
8  import java.util.Scanner;
9
10 public class Ex7d {
11
12     public static void main(String[] args) {
13         Scanner input = new Scanner (source: System.in);
14         Desktop cadastro = new Desktop ();
15
16         System.out.print(=: "Digite a marca:");
17         cadastro.marca = input.nextLine();
18
19         System.out.print(=: "Digite o modelo:");
20         cadastro.modelo = input.nextLine();
21
22         System.out.print(=: "Digite o peso(kg):");
23         cadastro.peso = input.nextFloat();
24
25         System.out.print(=: "Digite o tamanho(cm):");
26         cadastro.tamanho = input.nextFloat();
27
28         cadastro.reproduzirMidia();
29     }
30 }
31
32
33
```

ex7d.Ex7d > main >

Output - Ex7d (run) ×

```
run:
Digite a marca:Intel
Digite o modelo:Flex
Digite o peso(kg):5,3
Digite o tamanho(cm):29
Qual tipo de midia quer abrir? [1 para DVD] [2 para CD] [3 para PenDrive] [4 para BluRay]:4
Abrindo BluRay...BUILD SUCCESSFUL (total time: 23 seconds)
```

Start Page × Ex7d.java × Computador.java × Desktop.java ×

Source History

```
1  /*
2  |  Autoria: Frederico Rangel Sader;
3  |  Data: 09/06/2023;
4  |  Matrícula:20201100087;
5  |  */
6  package ex7d;
7
8  import java.util.Scanner;
9
10 public class Desktop extends Computador{
11     float peso;
12     Float tamanho;
13
14     public void reproduzirMidia(){
15         int midia;
16         Scanner input = new Scanner (source: System.in);
17
18         System.out.print(s: "Qual tipo de midia quer abrir? [1 para DVD] [2 para CD] [3 para PenDrive] [4 para BluRay]:");
19         midia = input.nextInt();
20
21         switch (midia){
22             case 1:
23                 System.out.print(s: "Abrindo DVD...");
24                 break;
25             case 2:
26                 System.out.print(s: "Abrindo CD...");
27                 break;
28             case 3:
29                 System.out.print(s: "Abrindo PenDrive...");
30                 break;
31             case 4:
32                 System.out.print(s: "Abrindo BluRay...");
33                 break;
34         }
35     }
36 }
```

ex7d.Desktop > peso >

Output - Ex7d (run) ×

```
run:
Digite a marca:Intel
Digite o modelo:Flex
Digite o peso(kg):5,3
Digite o tamanho(cm):29
Qual tipo de midia quer abrir? [1 para DVD] [2 para CD] [3 para PenDrive] [4 para BluRay]:4
Abrindo BluRay...BUILD SUCCESSFUL (total time: 23 seconds)
```



Start Page x Ex7e.java x Rede.java x RedeLocal.java x

Source History

```
2  Autoria: Frederico Rangel Sader;
3  Data: 09/06/2023;
4  Matricula:20201100087;
5  */
6  package ex7e;
7
8  import java.util.Scanner;
9
10
11
12  public class Ex7e {
13
14      public static void main(String[] args) {
15          Scanner input = new Scanner (source: System.in);
16          RedeLocal cadastro = new RedeLocal ();
17
18          System.out.print(=: "Digite o nome da rede:");
19          cadastro.nome = input.nextLine();
20
21          System.out.print(=: "Digite a velocidade(Mbps):");
22          cadastro.velocidade = input.nextInt();
23
24          System.out.print(=: "Digite a quantidade de dispositivos que serao conectados:");
25          cadastro.quantidadeDispositivos = input.nextInt();
26
27          input.nextLine();
28
29          System.out.print(=: "Digite o endereco de ip:");
30          cadastro.enderecoIP = input.nextLine();
31
32          cadastro.monitorarDispositivos(quantidadeDispositivos: cadastro.quantidadeDispositivos);
33      }
34
35  }
```

ex7e.Ex7e main

Output - Ex7e (run) x

```
run:
Digite o nome da rede:CCT2
Digite a velocidade(Mbps):400
Digite a quantidade de dispositivos que serao conectados:8
Digite o endereco de ip:123.45.678.9
Quantidade suficiente para esse tipo de rede!BUILD SUCCESSFUL (total time: 54 seconds)
```

The screenshot shows an IDE with three tabs: Start Page, Ex7e.java, Rede.java, and RedeLocal.java. The RedeLocal.java tab is active, displaying the following code:

```
1  /*
2  Autoria: Frederico Rangel Sader;
3  Data: 09/06/2023;
4  Matricula:20201100087;
5  */
6  package ex7e;
7
8  import java.util.Scanner;
9
10 public class RedeLocal extends Rede {
11     int quantidadeDispositivos;
12     String enderecoIP;
13
14     public void monitorarDispositivos (int quantidadeDispositivos){
15         int maxDispositivos = 10;
16
17         if(quantidadeDispositivos > maxDispositivos){
18             System.out.print(=: "Ultrapassou o limite de dispositivos que podem ser conectados!!");
19         }else{
20             System.out.print(=: "Quantidade suficiente para esse tipo de rede!");
21         }
22     }
23 }
24
```

Below the code editor, the IDE shows the package and class names: ex7e.RedLocal and enderecoIP. The Output window shows the following text:

```
run:
Digite o nome da rede:CCT2
Digite a velocidade(Mbps):400
Digite a quantidade de dispositivos que serao conectados:8
Digite o endereco de ip:123.45.678.9
Quantidade suficiente para esse tipo de rede!BUILD SUCCESSFUL (total time: 54 seconds)
```

8. **(02 pontos)** Resolver a Questão 9.10 e 9.11 (pág.363) do livro “Introduction to JAVA Programming”, Y. Daniel Liang, 2014. Não precisa desenhar os diagramas UML. “Implemente a classe” = “Faça um programa em Java”

**O professor**

**\*9.10** (Algebra: quadratic equations) Design a class named `QuadraticEquation` for a quadratic equation  $ax^2 + bx + c = 0$ . The class contains:

- Private data fields `a`, `b`, and `c` that represent three coefficients.
- A constructor for the arguments for `a`, `b`, and `c`.
- Three getter methods for `a`, `b`, and `c`.
- A method named `getDiscriminant()` that returns the discriminant, which is  $b^2 - 4ac$ .
- The methods named `getRoot1()` and `getRoot2()` for returning two roots of the equation

$$r_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{and} \quad r_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

These methods are useful only if the discriminant is nonnegative. Let these methods return 0 if the discriminant is negative.

Draw the UML diagram for the class and then implement the class. Write a test program that prompts the user to enter values for `a`, `b`, and `c` and displays the result based on the discriminant. If the discriminant is positive, display the two roots. If the discriminant is 0, display the one root. Otherwise, display “The equation has no roots.” See Programming Exercise 3.1 for sample runs.

**\*9.11** (Algebra:  $2 \times 2$  linear equations) Design a class named `LinearEquation` for a  $2 \times 2$  system of linear equations:

$$\begin{aligned} ax + by &= e \\ cx + dy &= f \end{aligned} \quad x = \frac{ed - bf}{ad - bc} \quad y = \frac{af - ec}{ad - bc}$$

The class contains:

- Private data fields `a`, `b`, `c`, `d`, `e`, and `f`.
- A constructor with the arguments for `a`, `b`, `c`, `d`, `e`, and `f`.
- Six getter methods for `a`, `b`, `c`, `d`, `e`, and `f`.
- A method named `isSolvable()` that returns true if  $ad - bc$  is not 0.
- Methods `getX()` and `getY()` that return the solution for the equation.

Draw the UML diagram for the class and then implement the class. Write a test program that prompts the user to enter `a`, `b`, `c`, `d`, `e`, and `f` and displays the result. If  $ad - bc$  is 0, report that “The equation has no solution.” See Programming Exercise 3.3 for sample runs.

**\*\*9.12** (Geometry: intersecting point) Suppose two line segments intersect. The two end-

Start Page x Ex9a.java x QuadraticEquation.java x

Source History

```
1  /*
2  Autoria: Frederico Rangel Sader;
3  Data: 09/06/2023;
4  Matrícula:20201100087;
5  */
6  package ex9a;
7
8  import java.util.Scanner;
9
10 public class Ex9a {
11     public static void main(String[] args) {
12         Scanner scanner = new Scanner(System.in);
13
14         System.out.print("Digite o valor de a: ");
15         double a = scanner.nextDouble();
16
17         System.out.print("Digite o valor de b: ");
18         double b = scanner.nextDouble();
19
20         System.out.print("Digite o valor de c: ");
21         double c = scanner.nextDouble();
22
23         QuadraticEquation equation = new QuadraticEquation(a, b, c);
24
25         double discriminant = equation.getDiscriminant();
26         if (discriminant >= 0) {
27             double root1 = equation.getRoot1();
28             double root2 = equation.getRoot2();
29             System.out.println("Raiz 1: " + root1);
30             System.out.println("Raiz 2: " + root2);
31         } else {
32             System.out.println("Não existem raízes reais.");
33         }
34     }
35 }
```

ex9a.Ex9a main

Output - Ex9a (run) x

run:  
Digite o valor de a: 2  
Digite o valor de b: 6  
Digite o valor de c: 4  
Raiz 1: -1.0  
Raiz 2: -2.0  
BUILD SUCCESSFUL (total time: 6 seconds)




Start Page x Ex9a.java x QuadraticEquation.java x

Source History

```
1  /*
2   | Autoria: Frederico Rangel Sader;
3   | Data: 09/06/2023;
4   | Matrícula:20201100087;
5   | */
6   package ex9a;
7
8   public class QuadraticEquation {
9       private final double a;
10      private final double b;
11      private final double c;
12
13      public QuadraticEquation(double a, double b, double c) {
14          | this.a = a;
15          | this.b = b;
16          | this.c = c;
17      }
18
19      public double getA() {
20          | return a;
21      }
22
23      public double getB() {
24          | return b;
25      }
26
27      public double getC() {
28          | return c;
29      }
30
31      public double getDiscriminant() {
32          | return b * b - 4 * a * c;
33      }
34  }
```

ex9a.QuadraticEquation > getRoot2 >

Output - Ex9a (run) x



run:  
Digite o valor de a: 2  
Digite o valor de b: 6  
Digite o valor de c: 4  
Raiz 1: -1.0  
Raiz 2: -2.0  
BUILD SUCCESSFUL (total time: 6 seconds)

```
public double getRoot1() {  
    double discriminant = getDiscriminant();  
    if (discriminant >= 0) {  
        return (-b + Math.sqrt(a: discriminant)) / (2 * a);  
    } else {  
        return 0;  
    }  
}
```

```
public double getRoot2() {  
    double discriminant = getDiscriminant();  
    if (discriminant >= 0) {  
        return (-b - Math.sqrt(a: discriminant)) / (2 * a);  
    } else {  
        return 0;  
    }  
}
```

Start Page × Ex9b.java × LinearEquation.java ×

Source History

```
1  /*
2   Aatoria: Frederico Rangel Sader;
3   Data: 09/06/2023;
4   Matrícula:20201100087;
5   */
6   package ex9b;
7
8   import java.util.Scanner;
9
10  public class Ex9b {
11      public static void main(String[] args) {
12          Scanner scanner = new Scanner(System.in);
13
14          System.out.print("Digite o valor de a: ");
15          double a = scanner.nextDouble();
16
17          System.out.print("Digite o valor de b: ");
18          double b = scanner.nextDouble();
19
20          System.out.print("Digite o valor de c: ");
21          double c = scanner.nextDouble();
22
23          System.out.print("Digite o valor de d: ");
24          double d = scanner.nextDouble();
25
26          System.out.print("Digite o valor de e: ");
27          double e = scanner.nextDouble();
28
29          System.out.print("Digite o valor de f: ");
30          double f = scanner.nextDouble();
31
32          LinearEquation equation = new LinearEquation(a, b, c, d, e, f);
33
34          if (equation.isSolvable()) {
```

ex9b.Ex9b > main > d >

Output - Ex9b (run) ×

```
run:
Digite o valor de a: 5
Digite o valor de b: 2
Digite o valor de c: 5
Digite o valor de d: 6
Digite o valor de e: 1
Digite o valor de f: 4
Solucao do sistema linear:
x = -0.1
y = 0.75
BUILD SUCCESSFUL (total time: 7 seconds)
```

```
if (equation.isSolvable()) {  
    double x = equation.getX();  
    double y = equation.getY();  
    System.out.println(x: "Solucao do sistema linear:");  
    System.out.println("x = " + x);  
    System.out.println("y = " + y);  
} else {  
    System.out.println(x: "O sistema linear não possui solução.");  
}  
  
scanner.close();
```



```
/*
Autoria: Frederico Rangel Sader;
Data: 09/06/2023;
Matricula:20201100087;
*/
package ex9b;

public class LinearEquation {
    private final double a;
    private final double b;
    private final double c;
    private final double d;
    private final double e;
    private final double f;

    public LinearEquation(double a, double b, double c, double d, double e, double f) {
        this.a = a;
        this.b = b;
        this.c = c;
        this.d = d;
        this.e = e;
        this.f = f;
    }

    public double getA() {
        return a;
    }

    public double getB() {
        return b;
    }

    public double getC() {
        return c;
    }
}
```

ex9b.LinearEquation > getX >

out - Ex9b (run) x

```
run:
Digite o valor de a: 5
Digite o valor de b: 2
Digite o valor de c: 5
Digite o valor de d: 6
Digite o valor de e: 1
Digite o valor de f: 4
Solucao do sistema linear:
x = -0.1
y = 0.75
BUILD SUCCESSFUL (total time: 7 seconds)
```

Start Page × Ex9b.java × LinearEquation.java ×

Source History

```
31     }
32
33     public double getC() {
34         return c;
35     }
36
37     public double getD() {
38         return d;
39     }
40
41     public double getE() {
42         return e;
43     }
44
45     public double getF() {
46         return f;
47     }
48
49     public boolean isSolvable() {
50         return a * d - b * c != 0;
51     }
52
53     public double getX() {
54         return (e * d - b * f) / (a * d - b * c);
55     }
56
57     public double getY() {
58         return (a * f - e * c) / (a * d - b * c);
59     }
60 }
61
62
```

ex9b.LinearEquation > getX >

Output - Ex9b (run) ×

run:

Digite o valor de a: 5

Digite o valor de b: 2

Digite o valor de c: 5

Digite o valor de d: 6

Digite o valor de e: 1

Digite o valor de f: 4

Solucao do sistema linear:

x = -0.1

y = 0.75

BUILD SUCCESSFUL (total time: 7 seconds)