|  | **AARE**: **Paradigmas de Linguagens de Programação**  **Professor**: Dr. Ausberto S. Castro V.  **Orientação a Objetos** |
| --- | --- |



**Prova JAVA**

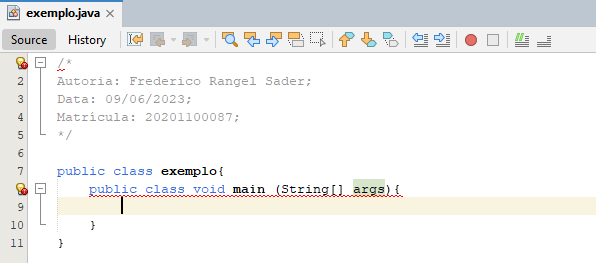
Nome Completo: Frederico Rangel Sader

Data: 31 de maio de 2023

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

* 1. Explique como se define uma classe em Java



‘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.

* 1. 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.**

* 1. 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.**

* 1. 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.**

1. **(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
   1. Docente

| superclasse; |
| --- |
| Docente; |
| * nome; * departamento; |
| * darAula(); * fazPesquisa(); |

↓ ↓

| subclasse; | subclasse; |
| --- | --- |
| Professor; | Assistente; |
| * Disciplina; * Grau de formação; | * Projeto; * Experiência; |
| * orientaAluno(); * corrigeProva(); | * organizarAulas(); * darMonitoria(); |

* 1. Verdura

| superclasse; |
| --- |
| Verdura; |
| * nome; * cor; |
| * cozinhar(); * armazenar(); |

↓ ↓

| subclasse; | subclasse; |
| --- | --- |
| Folhosa; | raiz; |
| * tipoDeFolha; * textura; | * parteComestível; * origem; |
| * prepararSalada(); * lavarFolha(); | * prepararSopa(); * descascar(); |

* 1. Ave

| superclasse; |
| --- |
| Ave; |
| * nome; * habitat; |
| * voar(); * fazerNinho(); |

↓

| subclasse; |
| --- |
| Pássaro; |
| * corPenas; * envergadura; |
| * cantar(); * migrar(); |

↓

| subclasse; |
| --- |
| Predador; |
| * tipoPresas; * velocidadeVoo; |
| * caçar(); * descansar(); |

* 1. Computador

| superclasse; |
| --- |
| Computador; |
| * marca; * modelo; |
| * ligar(); * desligar(); |

↓

| subclasse; |
| --- |
| Desktop; |
| * peso; * tamanho; |
| * conectarInternet(); * reproduzirMidia(); |

↓

| subclasse; |
| --- |
| DesktopGamer; |
| * Placa de vídeo; * Overclock; |
| * jogarGame(); * receberDadosGrandes(); |

* 1. Rede

| superclasse; |
| --- |
| Rede; |
| * nome; * velocidade; |
| * conectarDispositivo(); * monitorarTrafego(); |

↓

| subclasse; |
| --- |
| RedeLocal; |
| * quantidadeDispositivos; * endereçoIP; |
| * CompartilharArquivos(); * monitorarDispositivos(); |

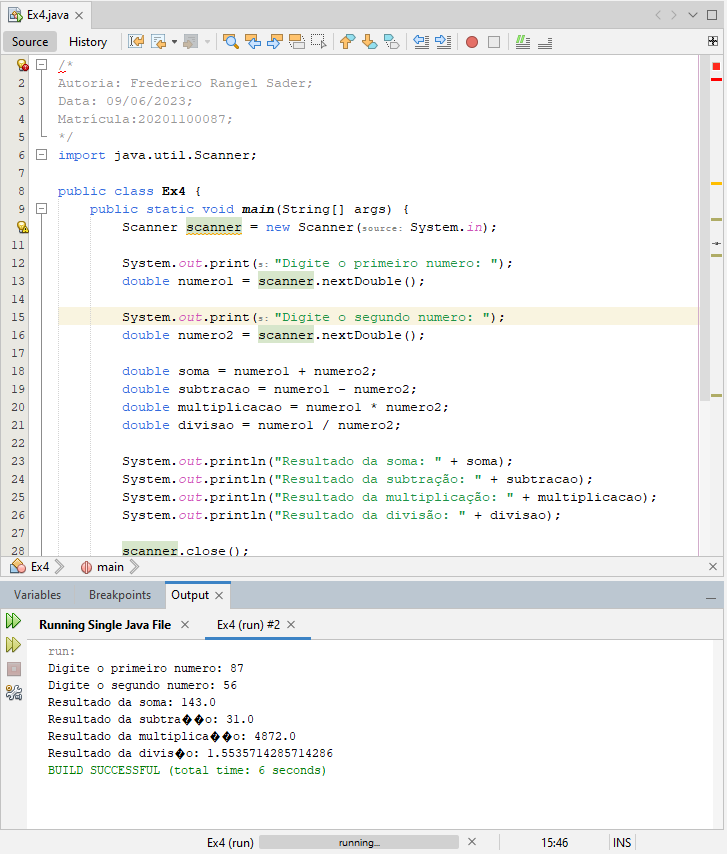
↓

| subclasse; |
| --- |
| RedeSemFio; |
| * alcance; * canalFrequencia; |
| * conectarDispositivosMovéis(); * gerenciarSenha(); |

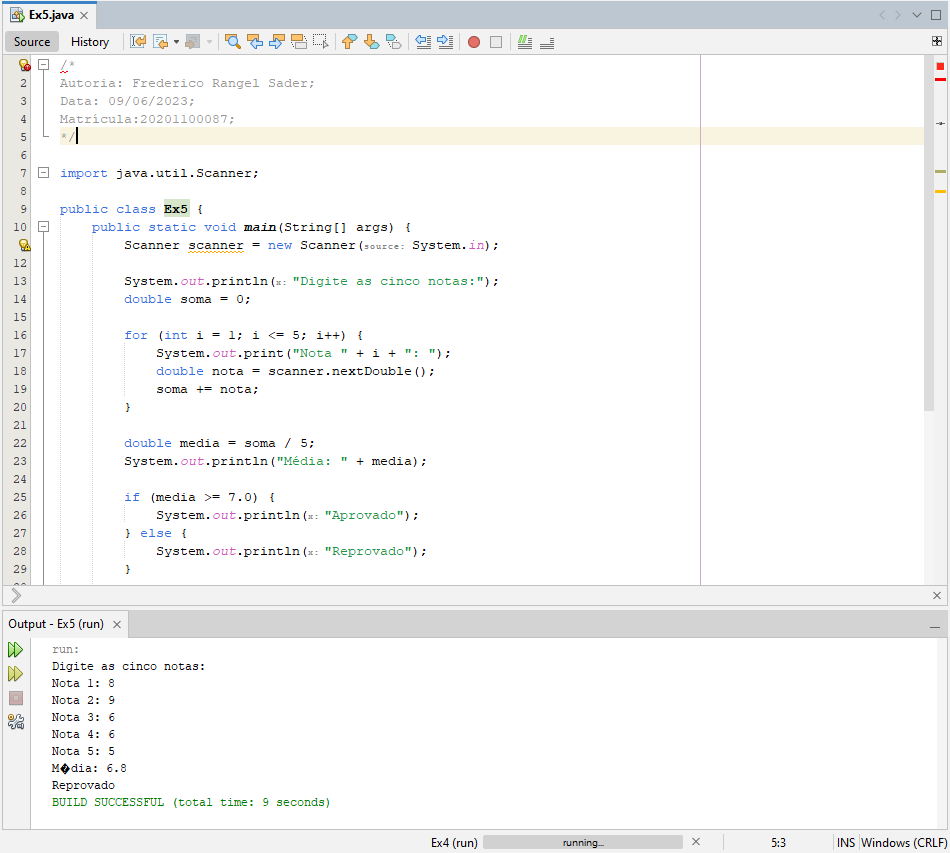
1. **(01 ponto)** Dar cinco exemplos de cada operação abstrata em orientação a objetos. Ilustre com imagens ou diagramas

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

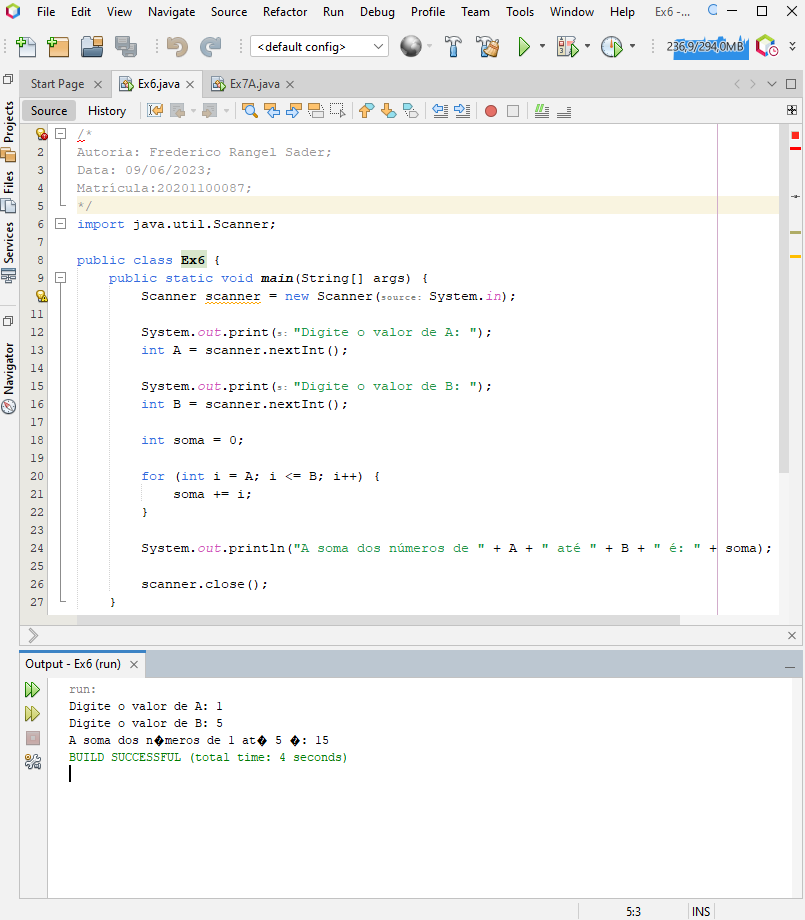
1. **(01 ponto)** Escreva um programa em Java que execute as quatro operações aritméticas de números reais



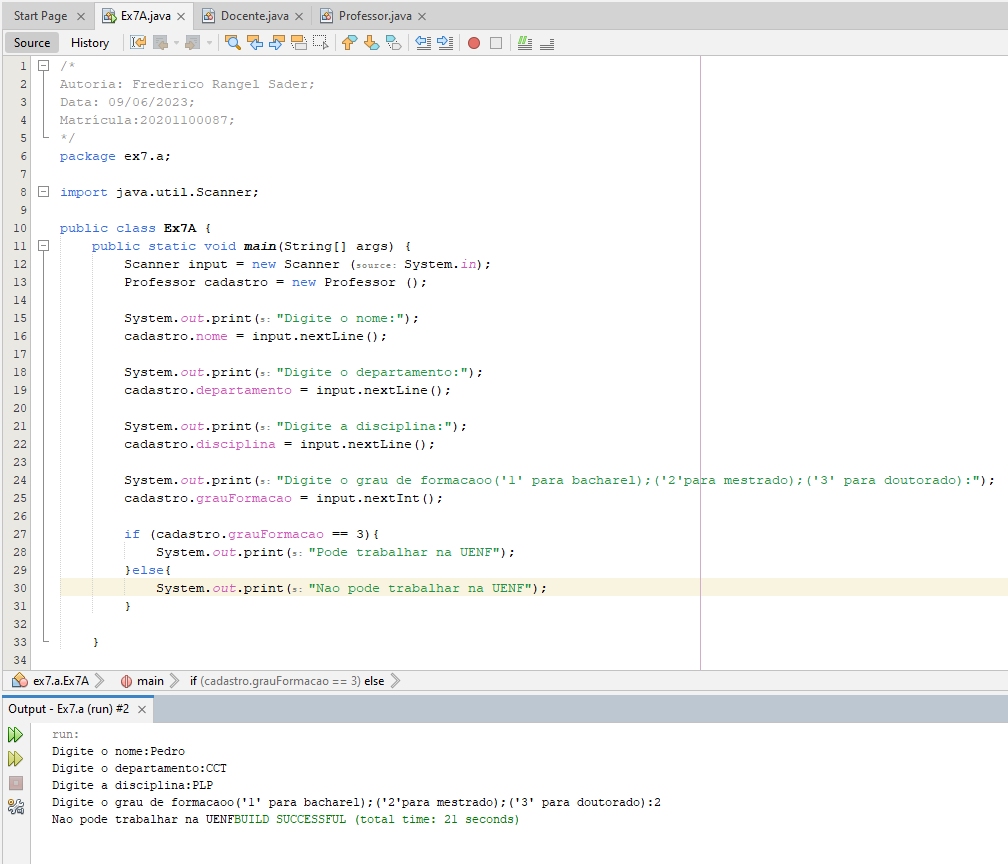
1. **(01 ponto)** Escreva um programa JAVA que calcule a média de cinco notas. O programa deve mostrar o valor da média e indicar se foi aprovado ou reprovado

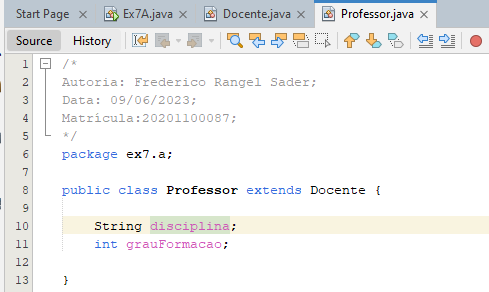


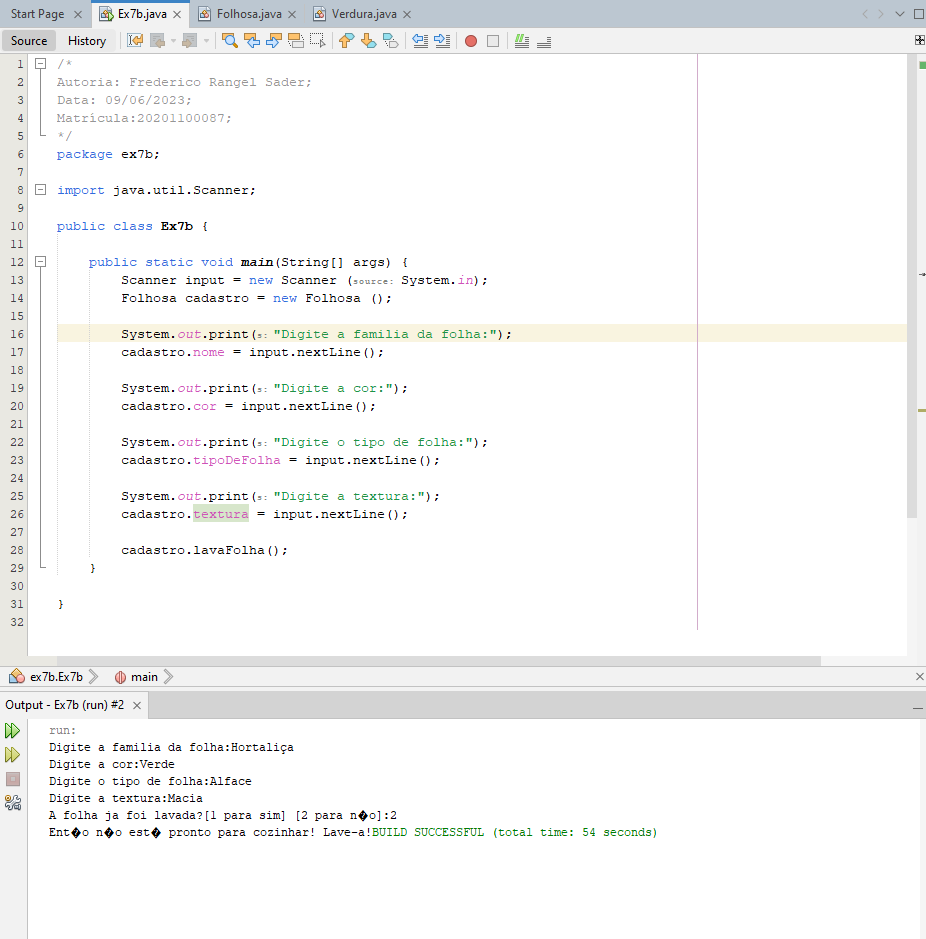
1. **(01 ponto)** Escreva um programa JAVA que calcule a soma dos inteiros maiores o iguales a um número inteiro A e menores ou iguais a um inteiro B.

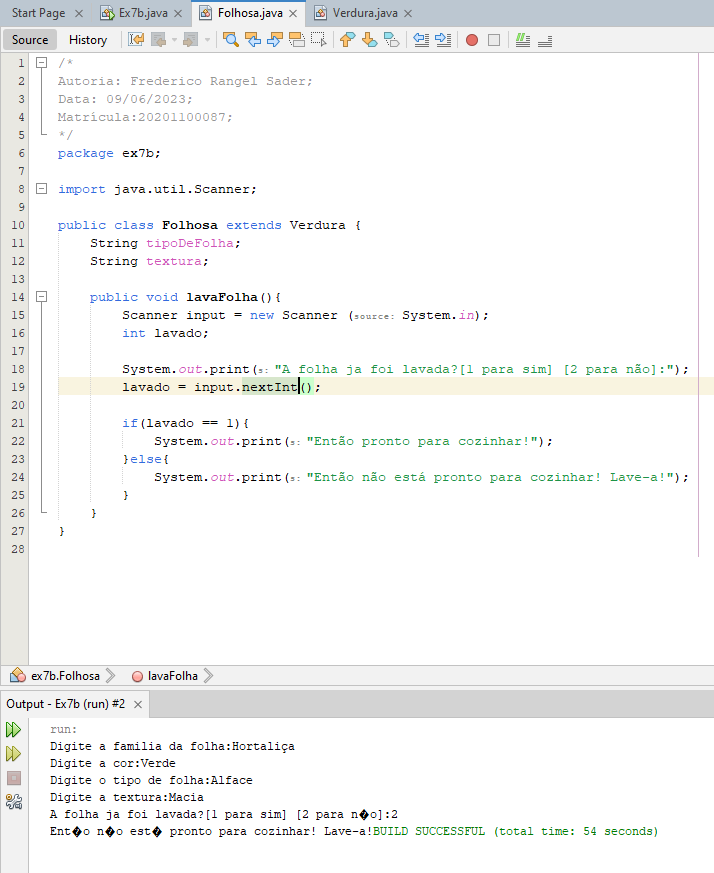


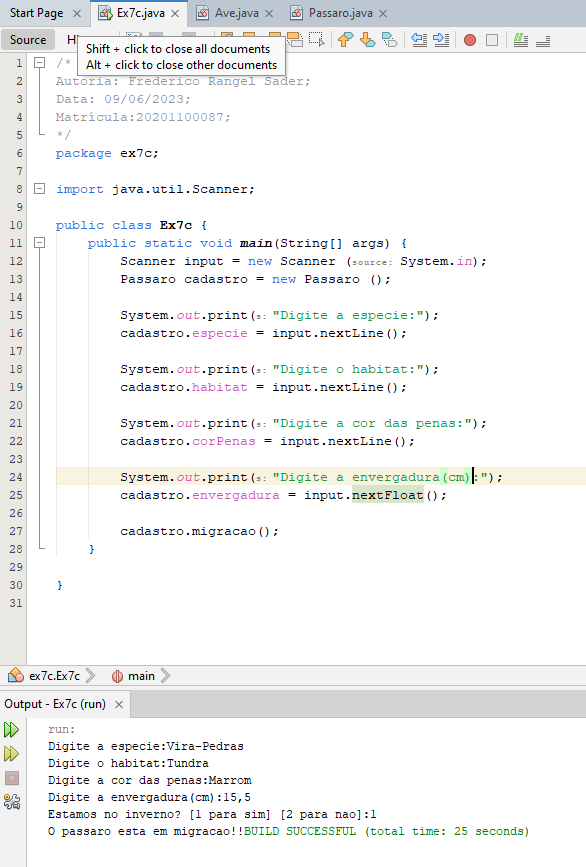
1. **(2 pontos)** Para cada caso da questão 2, escrever um programa JAVA, mostrando as classes, as operações (métodos), a criação de objetos e outras características da Programação Orientada a Objetos.

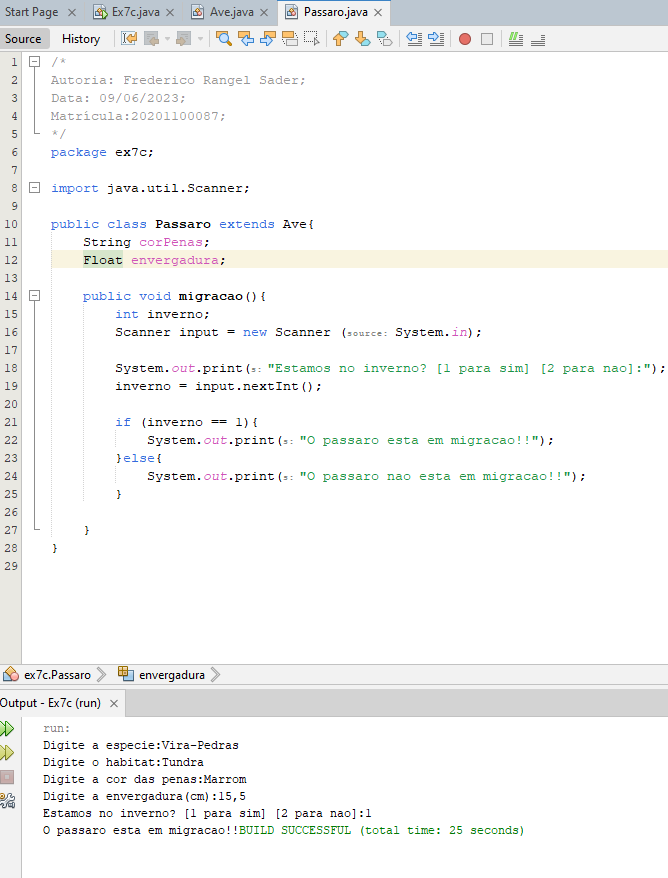


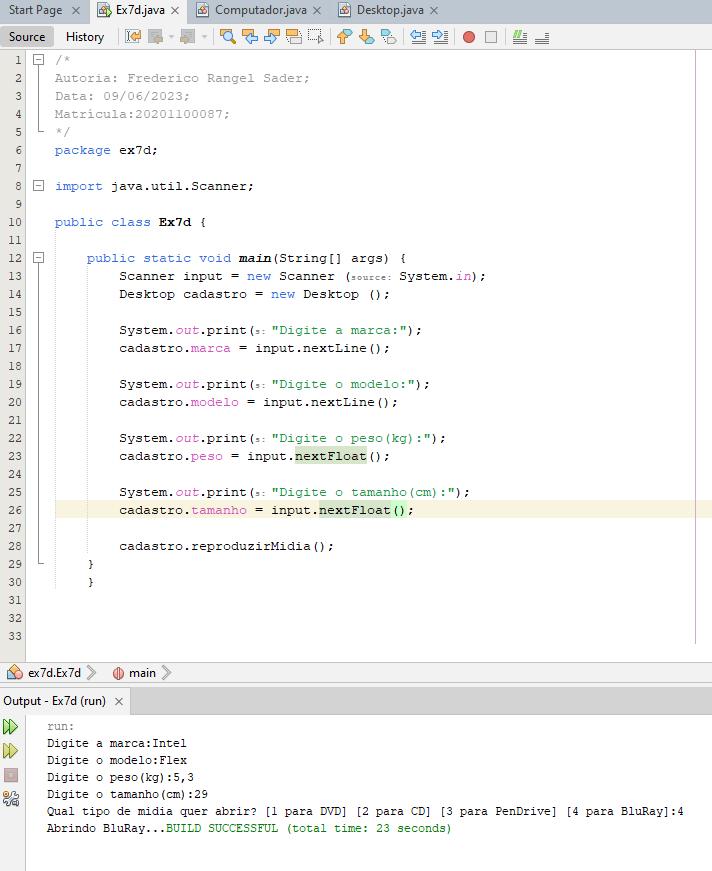


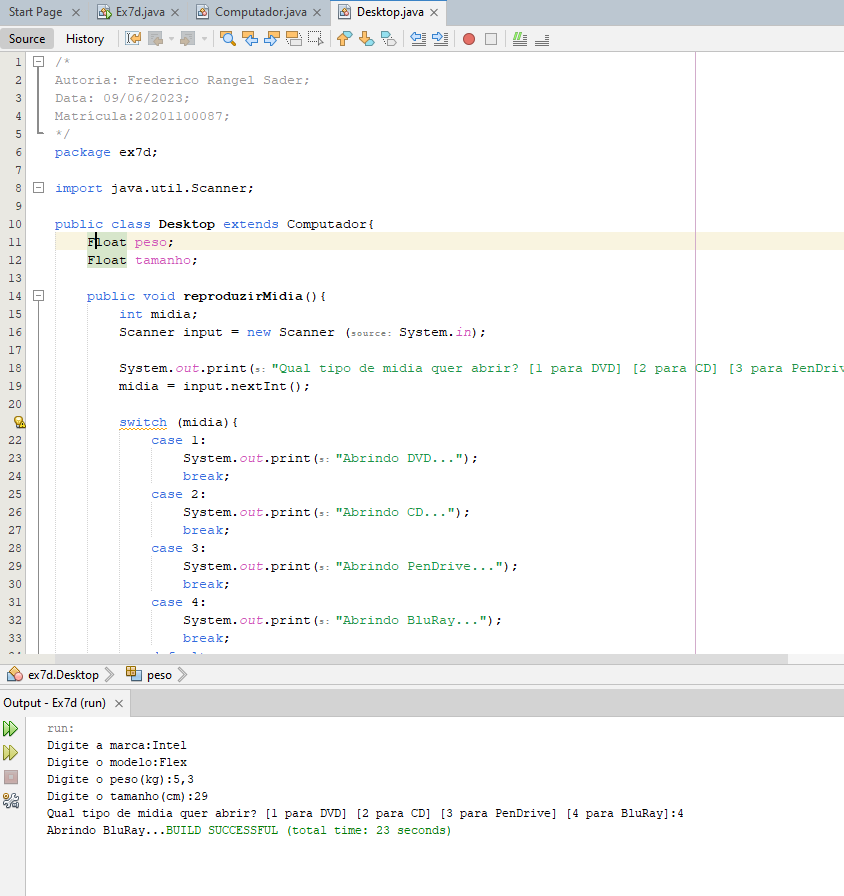


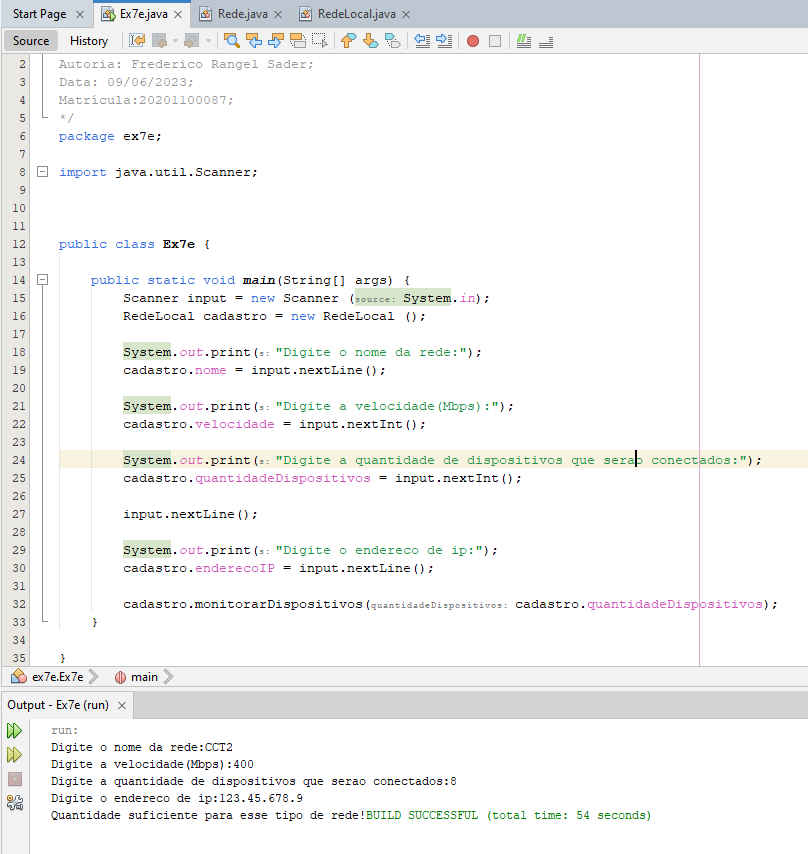


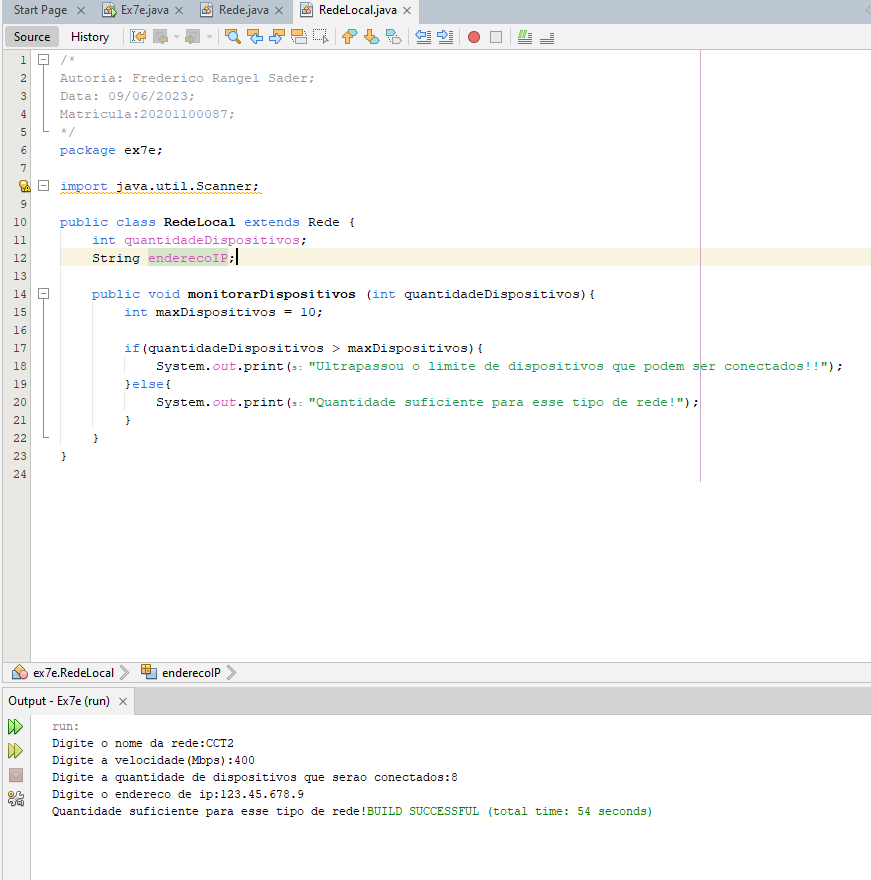












1. **(02 pontos)** Resolver a Questão **9.10** e **9.11** (pág.363) do livro “Introduction to JAVA Programming”, Y. Daniel Liang, 2014. Não precisa desenhar os diagramas UML. “Implemente a classe” = “Faça um programa em Java”

**O professor**

