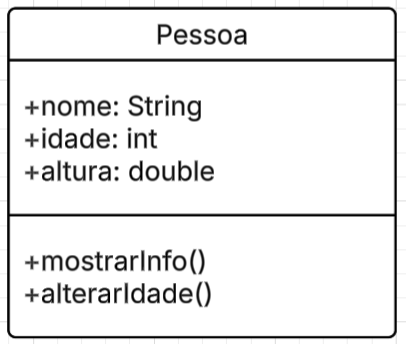
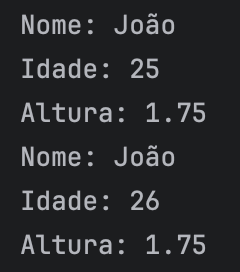
**Tarefa 2**

**Aula 2: Estrutura Básica da POO - classes**, **objetos, atributos**, **métodos, passagem de dados (parâmetros e argumento)**

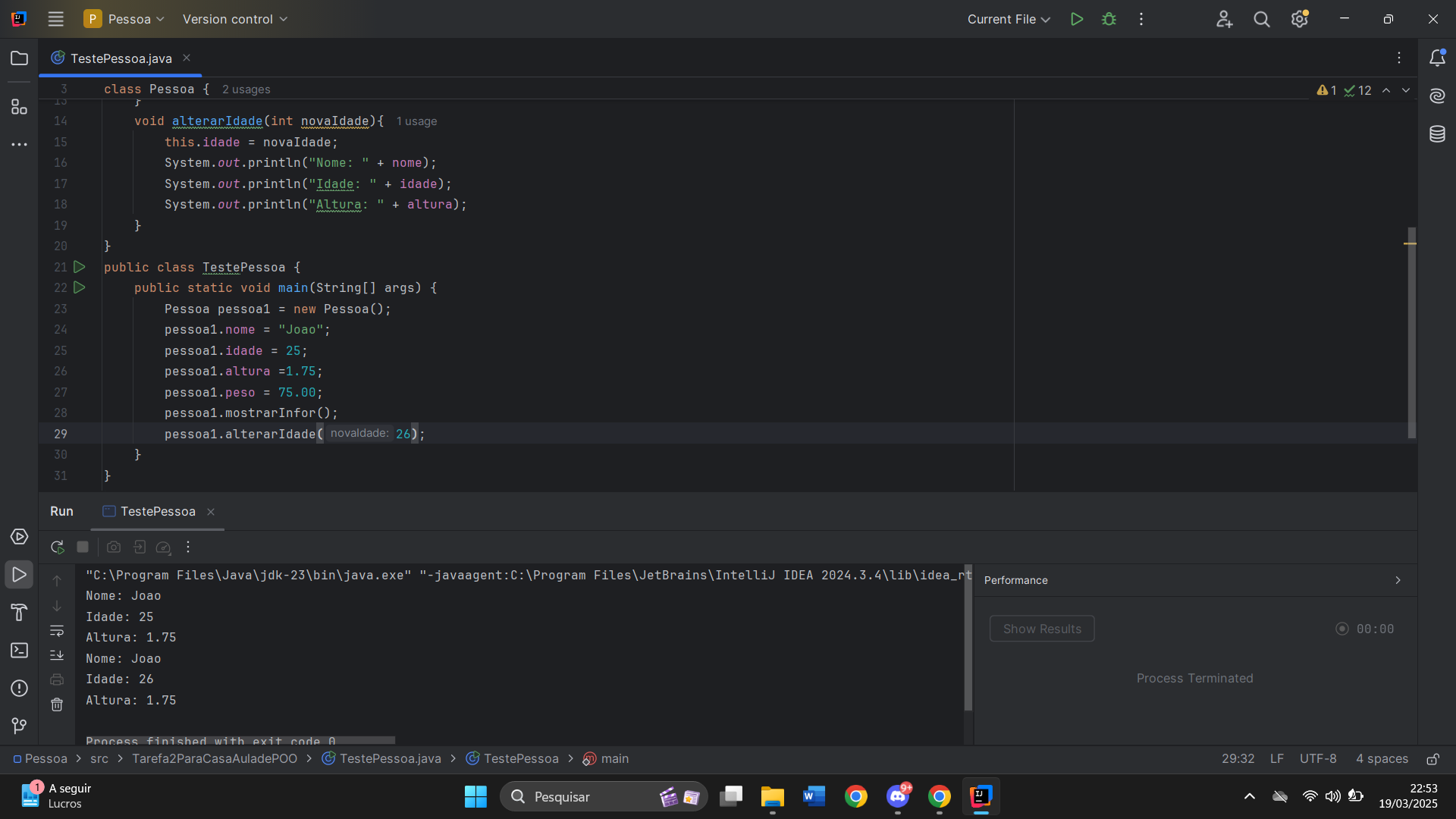
Questão 1) Implemente a classe Pessoa, é necessário criá-la com os atributos nome, do tipo String, para armazenar o nome da pessoa; idade, do tipo int, para representar a idade; e altura, do tipo double, para registrar a altura. Além disso, deve-se desenvolver o método mostrarInfo(), responsável por exibir as informações da pessoa no formato adequado, incluindo nome, idade e altura. Também é essencial criar o método alterarIdade(int novaIdade), que permite modificar o valor do atributo idade com base no parâmetro recebido. Após a implementação, a classe deve ser testada por meio da execução do código da classe TestePessoa, garantindo que a saída esteja conforme o esperado.



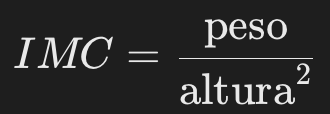
Saída Esperada:



Minha Saída:

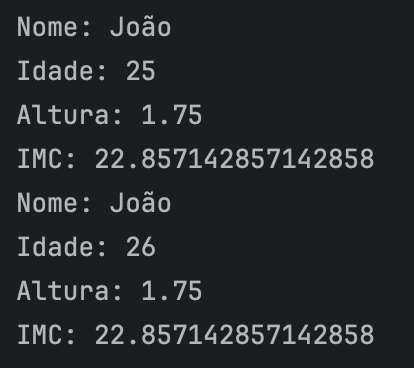


1. Modifique a classe Pessoa para incluir um novo método chamado calcularImc(), que calcula o Índice de Massa Corporal (IMC) da pessoa. O cálculo deve ser feito utilizando a fórmula:



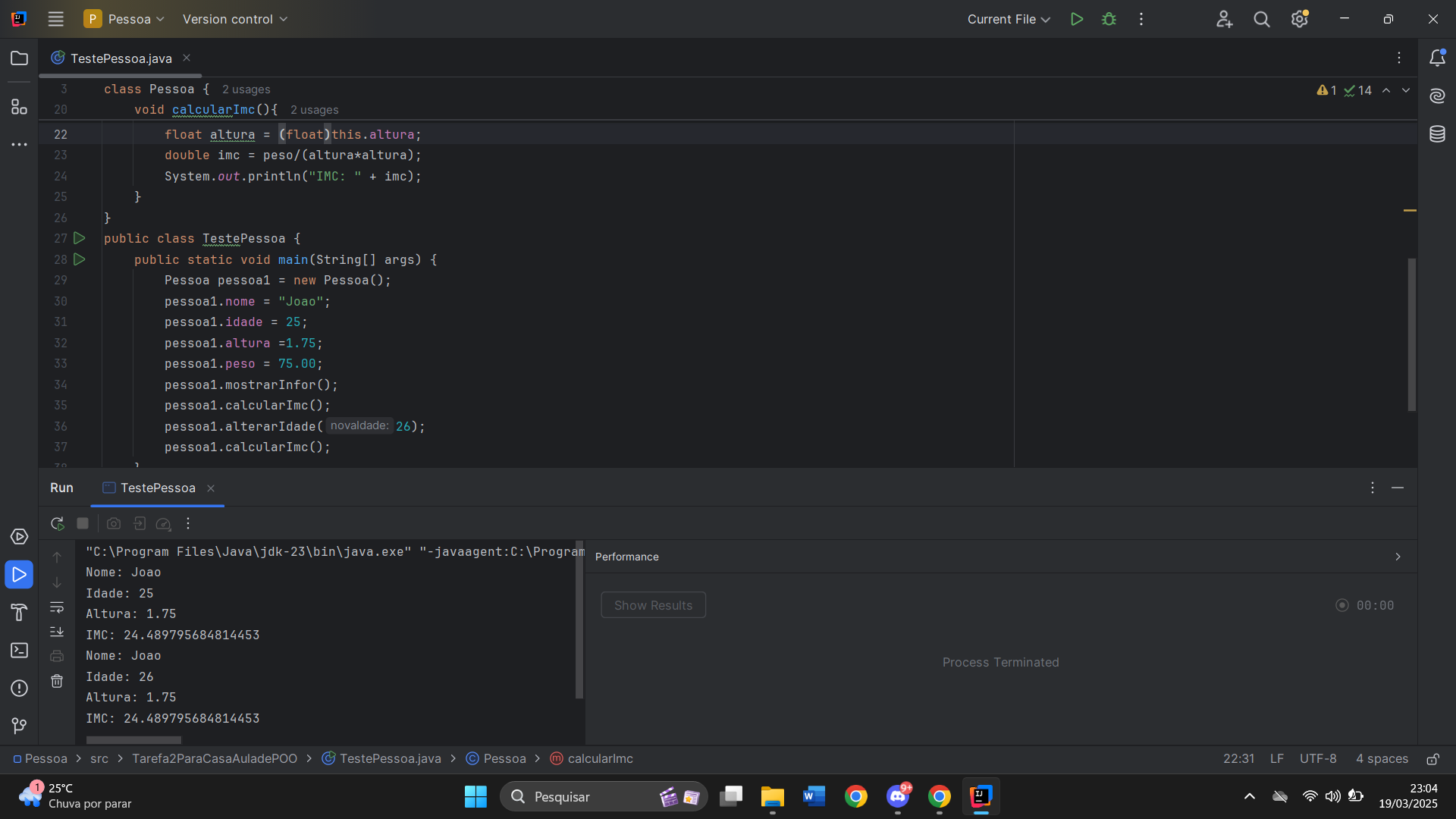
O método deve receber o peso como parâmetro, realizar o cálculo e retornar o valor do IMC. Essa modificação permite que a classe Pessoa forneça uma informação relevante sobre a saúde do indivíduo.

Saída Esperada:

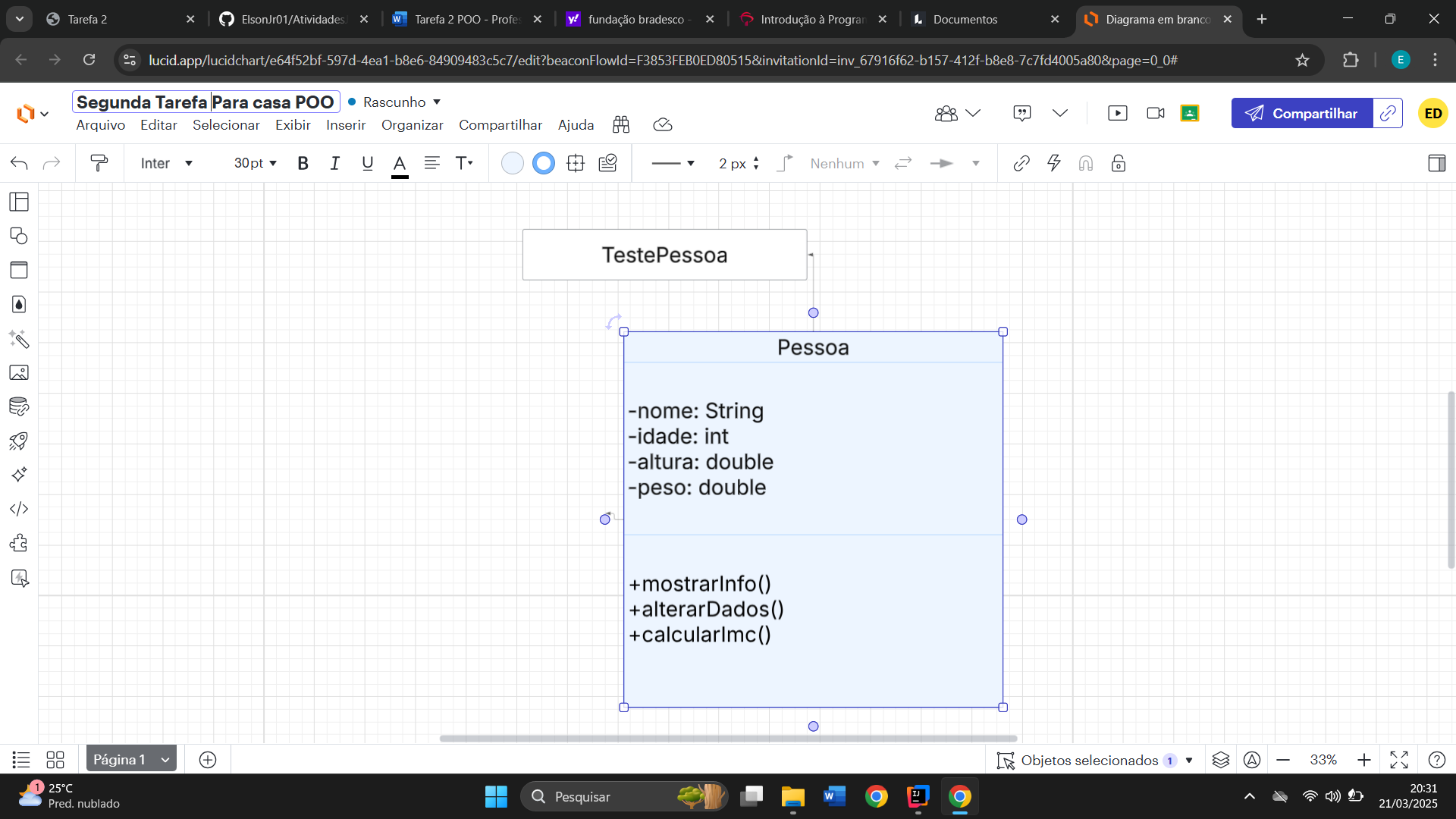


Minha Saída:

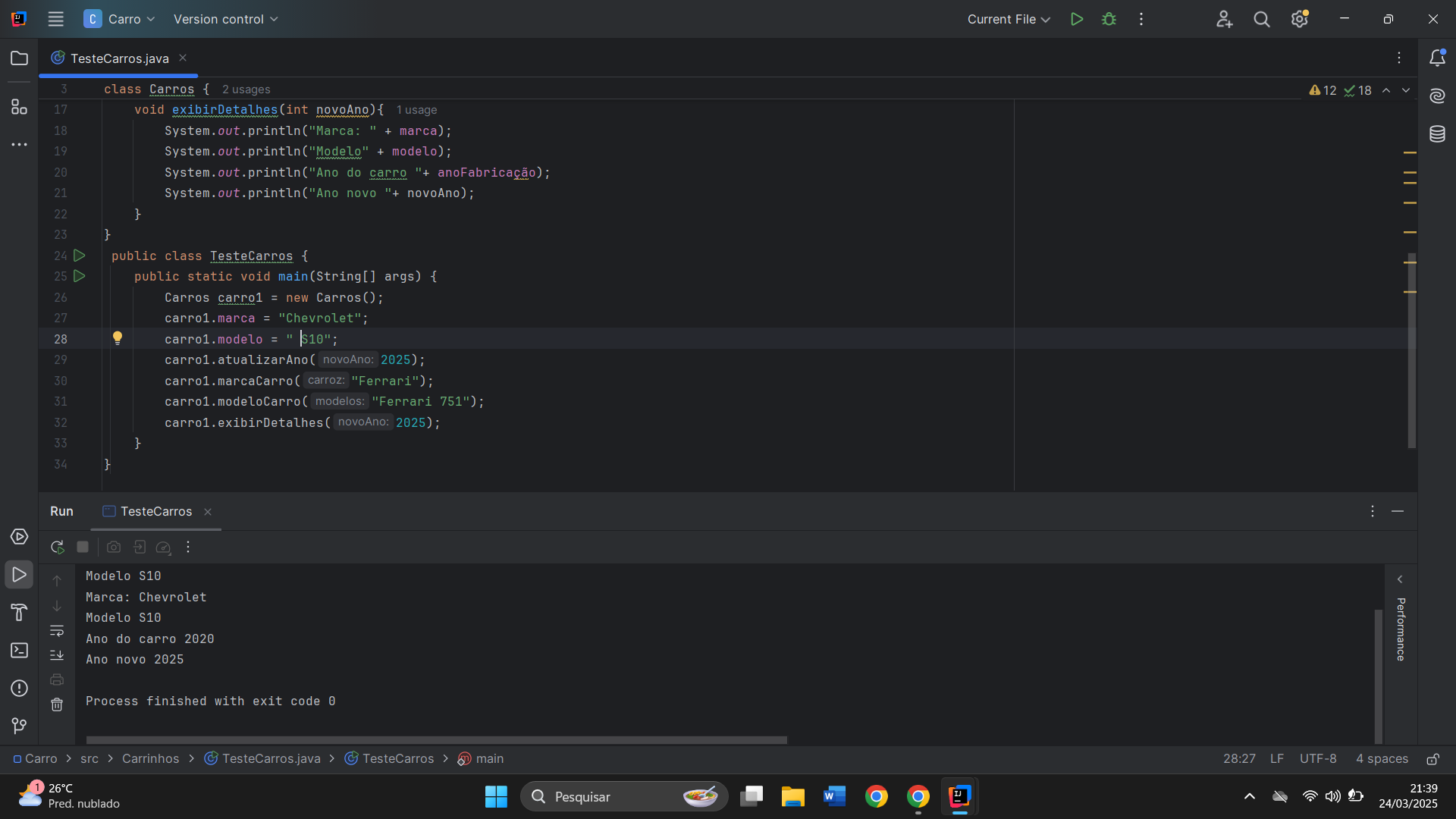
R: Usei o que foi aprendido em sala de aula e adicionei uma casting como teste, mas apenas para aprimorar meus conhecimentos.



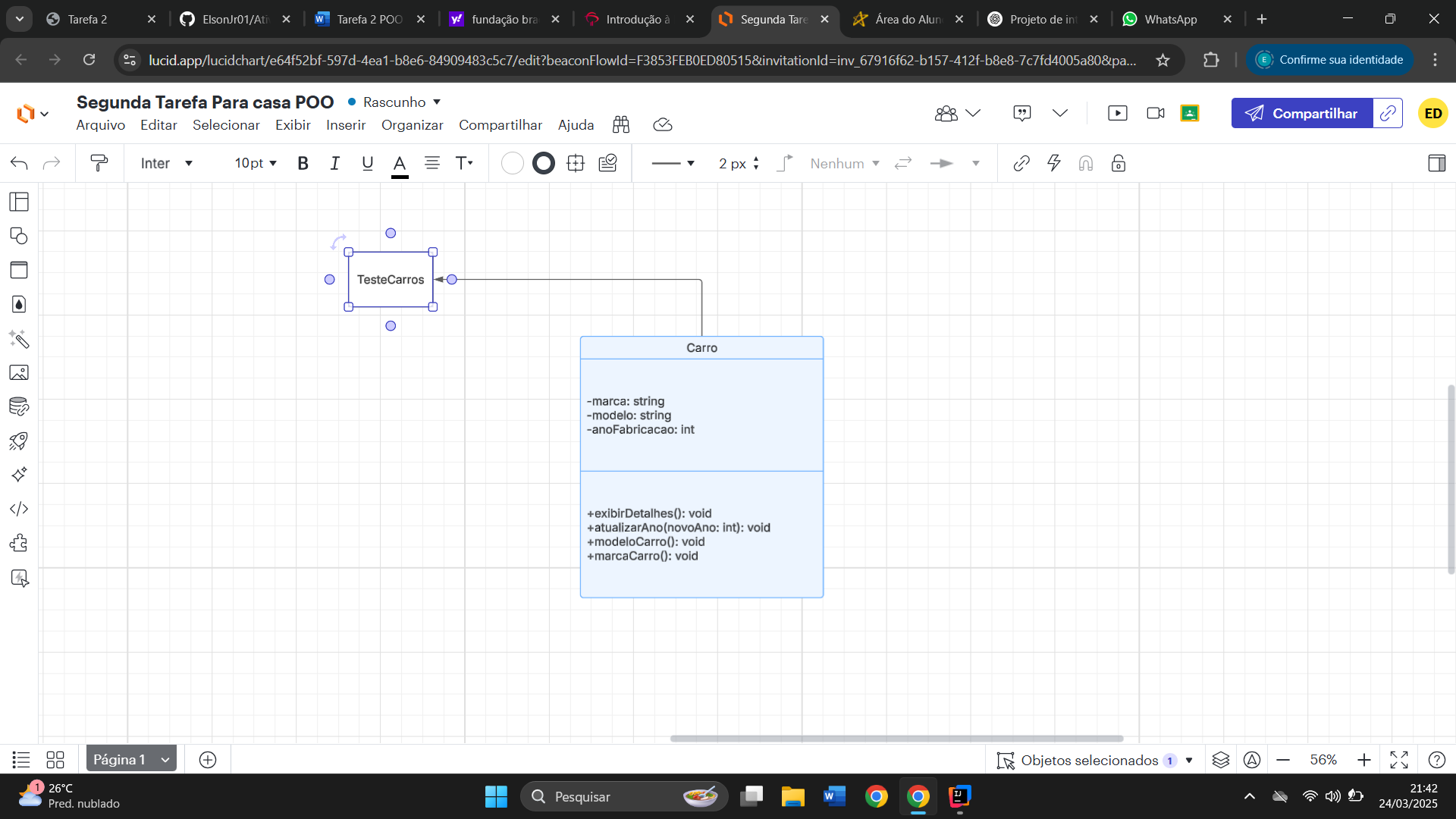
1. Atualize o Diagrama de Classe.



Questão 3) Crie uma classe Carro com os atributos marca, modelo e anoFabricacao. Em seguida, implemente o método exibirDetalhes(), que imprime as informações do carro. Por fim, crie o método atualizarAno(int novoAno), que permite modificar o ano de fabricação do veículo. No método main, instancie dois objetos da classe Carro, exiba suas informações iniciais, altere o ano de fabricação de um deles e, por fim, exiba novamente os dados atualizados.

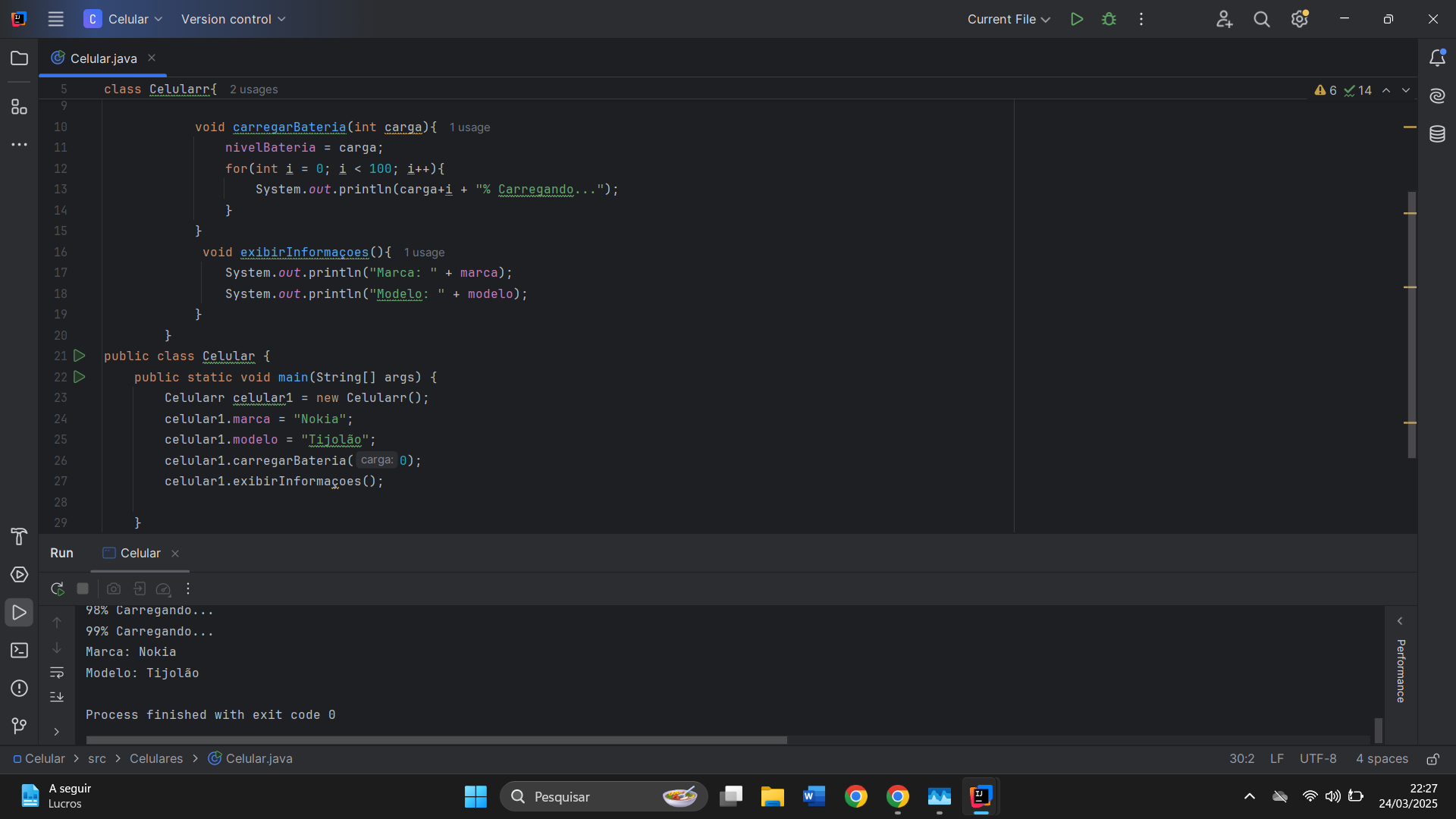
R: 

1. Crie o diagrama de Classe nessa ferramenta: **lucid.app**

R: 

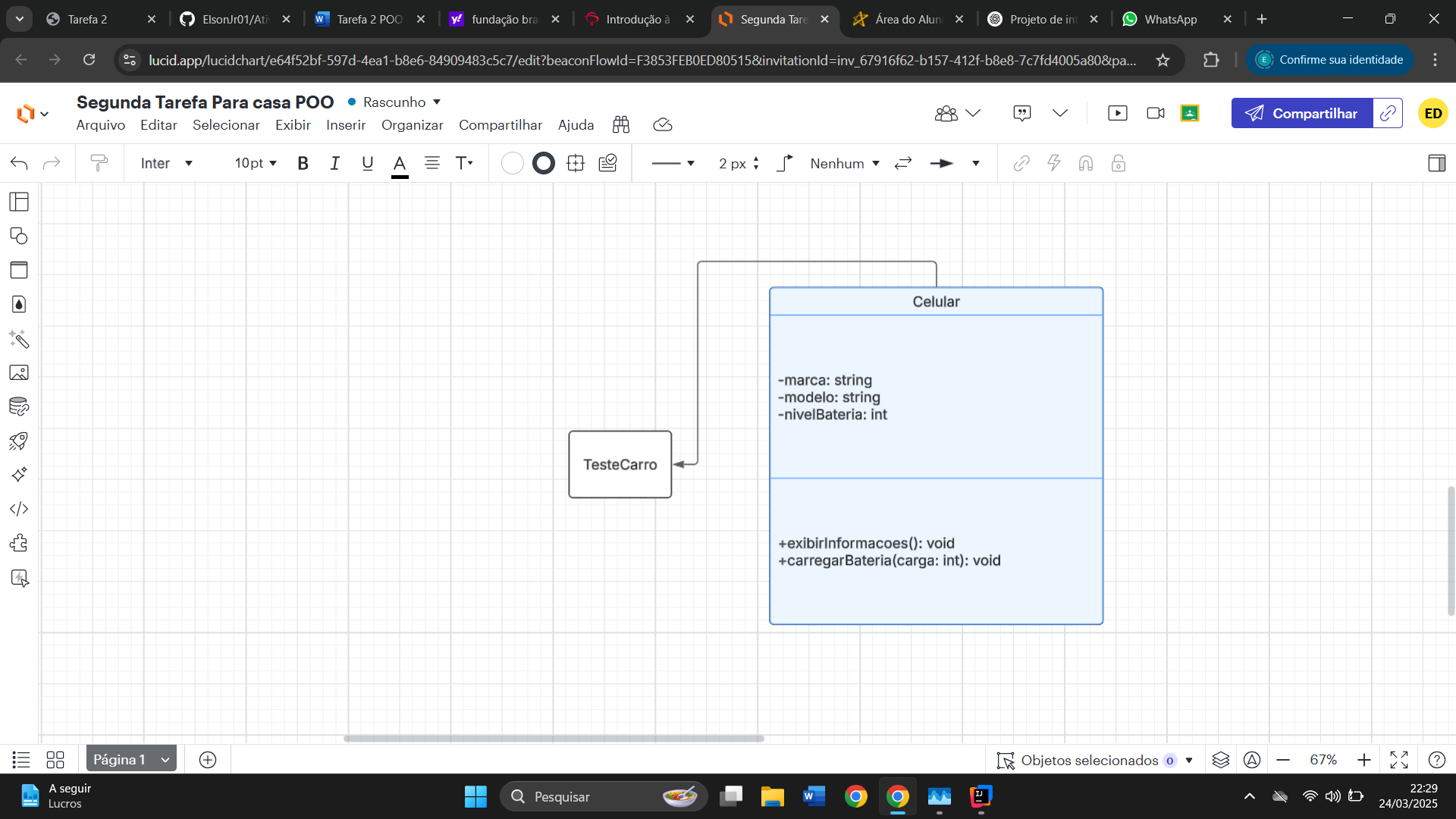
Questão 4) crie uma classe Celular com os atributos marca, modelo e nivelBateria. Em seguida, implemente o método exibirInformacoes(), que imprime os detalhes do celular. Depois, crie o método carregarBateria(int carga), que aumenta o nível da bateria em uma determinada porcentagem, sem ultrapassar 100%. No método main, instancie dois objetos da classe Celular, exiba suas informações iniciais, carregue a bateria de um deles e, por fim, exiba novamente os dados atualizados.

R:

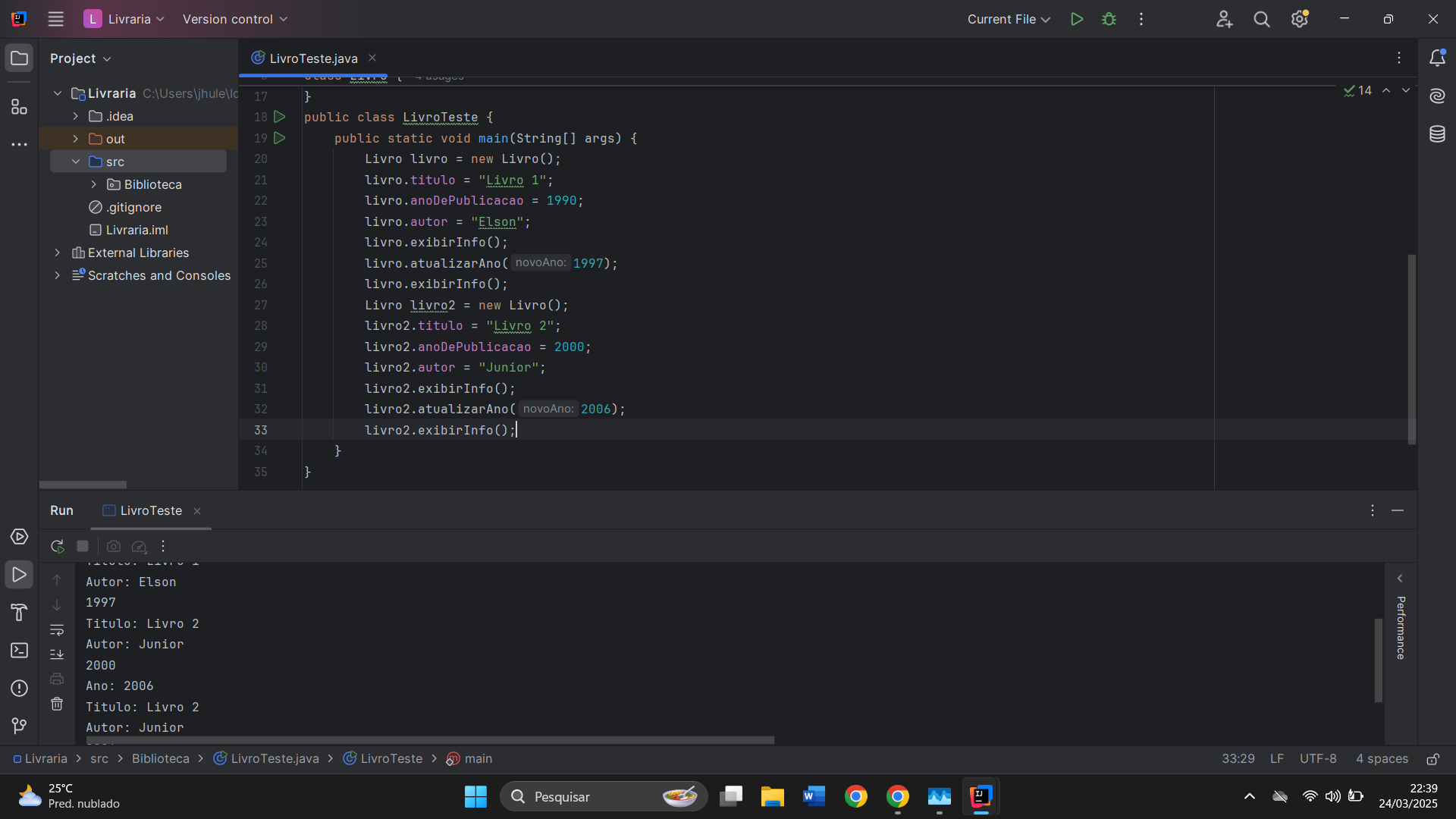


1. Crie o diagrama de Classe nessa ferramenta: **lucid.app**

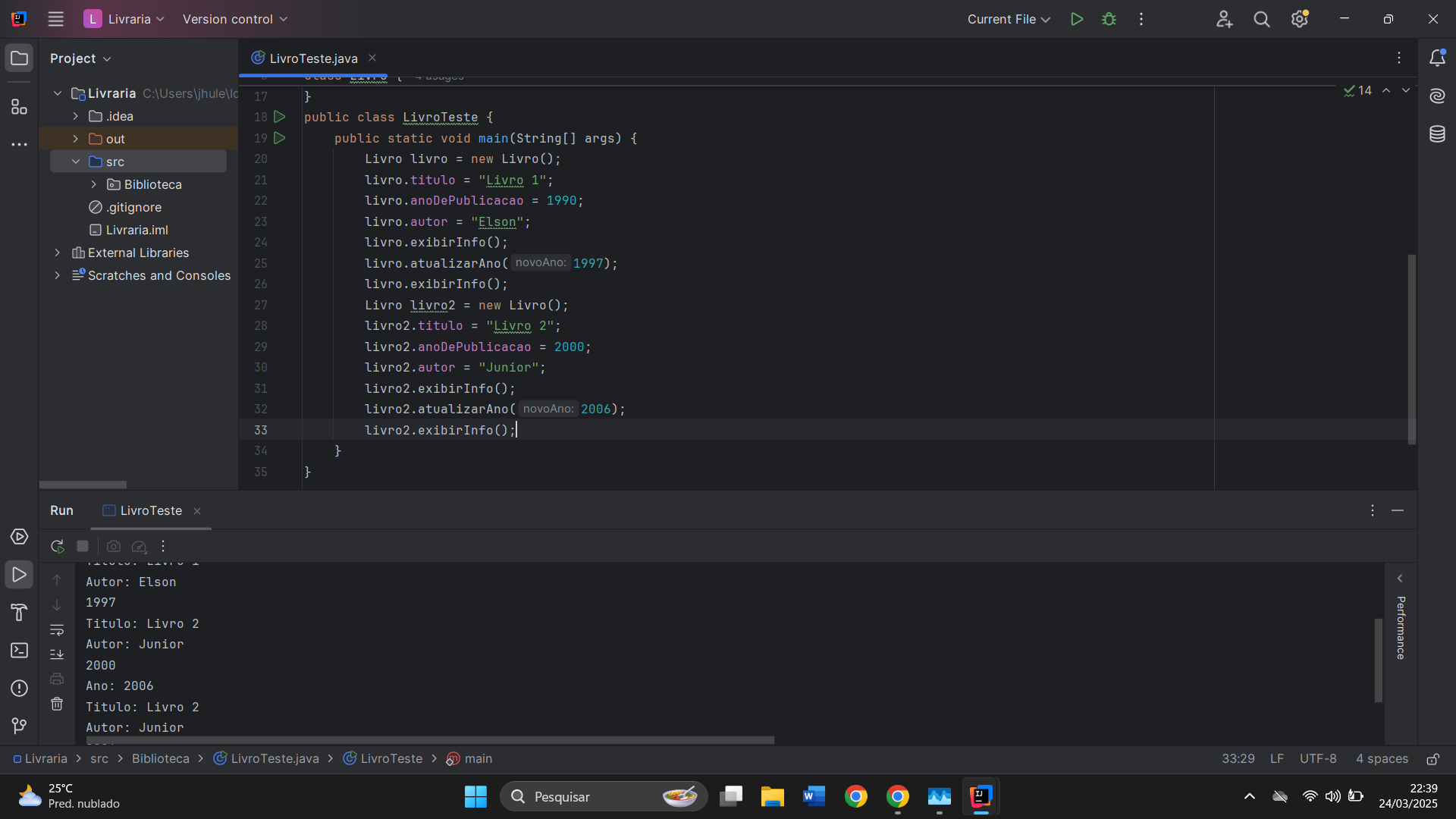
R:



Questão 5) Implemente a classe Livro em Java, incluindo os atributos titulo, autor e anoDePublicacao. Crie os métodos exibirInfo(), que exibe as informações do livro, e atualizarAno(int novoAno), que altera o ano de publicação. Em seguida, no método main, instancie dois objetos da classe Livro, exiba suas informações iniciais, altere o ano de publicação de um deles e, por fim, exiba novamente os dados atualizados.

R: 

1. Crie o diagrama de Classe nessa ferramenta: **lucid.app**

R: 

Questão 2) Para cada questão e de acordo com o código-fonte, Identifique:

a) Quais são as classes? Explique

No meu código, eu criei três classes principais:

**Pessoa**: Representa uma pessoa com atributos como nome, idade, altura e peso.

**Livro**: Representa um livro com título, autor e ano de publicação.

**Celular**: Representa um celular, armazenando a marca, modelo e nível de bateria. Além disso, tenho as classes de teste (TestePessoa, LivroTeste e TesteCelular), que possuem o método main() e servem para testar o funcionamento dos objetos.

* b) Quais são os atributos? Explique

1.Na classe Pessoa:

nome (String): Nome da pessoa

idade (int): Idade da pessoa

altura (double): Altura da pessoa

peso (double): Peso da pessoa

2. Na classe Livro:

titulo (String): Nome do livro

autor (String): Nome do autor

anoDePublicacao (int): Ano de publicação do livro

3.Na classe Celular:

marca (String): Marca do celular

modelo (String): Modelo do celular

nivelBateria (int): Indica o percentual da carga da bateria

Cada atributo é usado para armazenar informações sobre os objetos que eu criei.

c) Quais são os métodos? Explique

Os métodos são as ações que cada classe pode realizar. No meu código, tenho os seguintes métodos:

Na classe Pessoa:

mostrarInfor(): Exibe as informações da pessoa.

alterarIdade(int novaIdade): Atualiza a idade da pessoa.

calcularImc(): Calcula o Índice de Massa Corporal (IMC) da pessoa.

Na classe Livro:

exibirInfo(): Mostra as informações do livro.

atualizarAno(int novoAno): Atualiza o ano de publicação do livro.

Na classe Celular:

carregarBateria(int carga): Simula o carregamento da bateria do celular.

exibirInformacoes(): Exibe os detalhes do celular, como marca, modelo e nível da bateria.

Os métodos permitem que eu manipule os atributos dos objetos e execute ações neles.

d) Quais são os objetos? Explique

Na classe Pessoa:

pessoa1 = new Pessoa(); Cria um objeto chamado pessoa1 baseado na classe Pessoa.

Na classe Livro:

livro = new Livro(); Cria um objeto chamado livro.

livro2 = new Livro(); Outro objeto da mesma classe.

Na classe Celular:

celular1 = new Celular(); Cria um objeto baseado na classe Celular.

Cada um desses objetos representa um elemento único e pode armazenar valores diferentes dentro dos seus atributos.

e) Quais os parâmetros e argumentos? Explique

Os parâmetros são as variáveis que os métodos recebem para processar as informações, e os argumentos são os valores reais que eu passo quando chamo esses métodos.

Exemplo na classe Pessoa:

O método alterarIdade(int novaIdade) tem um parâmetro chamado novaIdade.

Quando eu chamo pessoa1.alterarIdade(26);, o valor 26 é o argumento passado para o método.

Exemplo na classe Livro:

O método atualizarAno(int novoAno) tem o parâmetro novoAno.

Quando eu chamo livro.atualizarAno(2006);, o argumento passado é 2006.

Exemplo na classe Celular:

O método carregarBateria(int carga) tem o parâmetro carga.

Quando eu chamo celular1.carregarBateria(80);, o argumento passado é 80.

Os parâmetros servem para tornar os métodos mais flexíveis, permitindo que eu envie diferentes valores como entrada e obtenha diferentes resultados.