

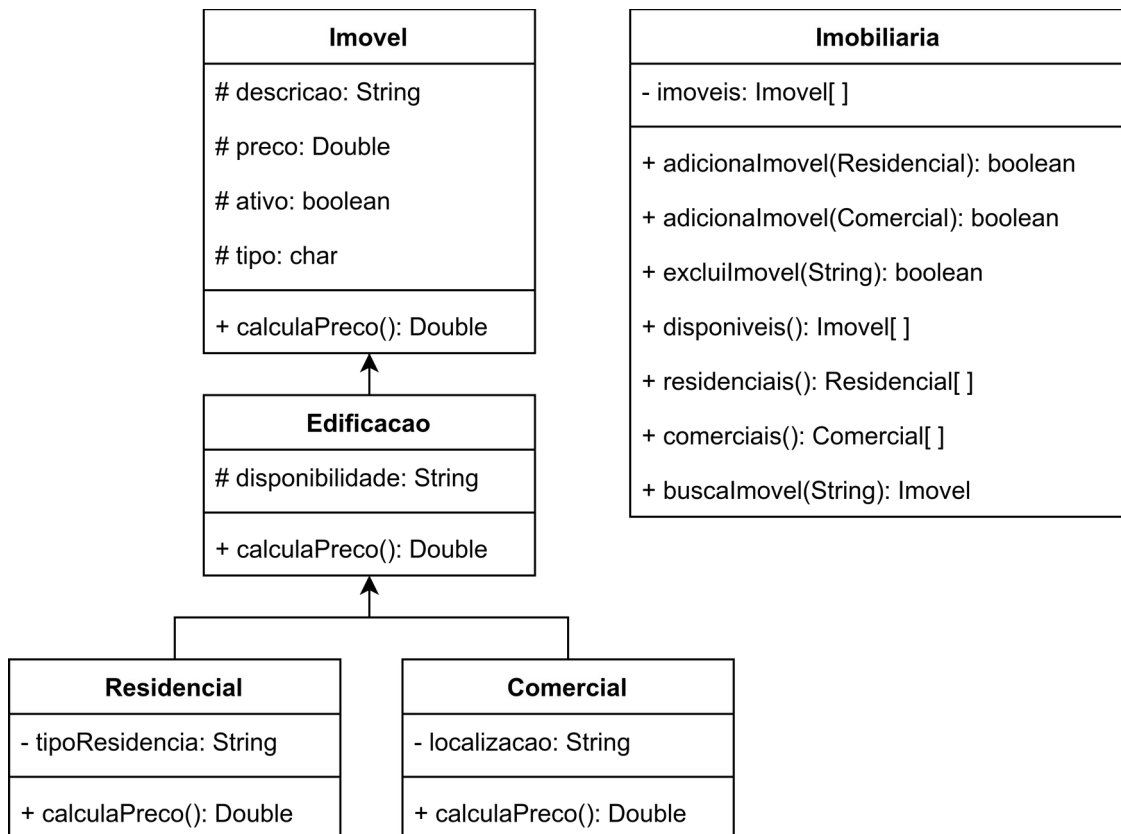
Exercício POO - Casting

Os construtores estão descritos abaixo.

Não esqueça do encapsulamento.

APÓS CRIAR AS CLASSES ABAIXO NO MESMO DIRETÓRIO DO EXERCÍCIO EXECUTE O COMANDO “java Principal”

Crie um sistema para uma Imobiliária que possui imóveis comerciais e residenciais para venda e locação. O sistema deve possuir as classes abaixo:



Classe Imovel:

Construtor `Imovel(Double preco, String descricao, boolean ativo)`

calculaPreco(): Retorna o preço do imóvel se estiver ativo.

Classe Edificacao:

Construtor `Edificacao(String disponibilidade, Double preco, String descricao, boolean ativo)`

disponibilidade informa se o imóvel está disponível para “COMPRA” ou “LOCAÇÃO”.

Classe Residencial:

Construtor Residencial(String tipoResidencia, String disponibilidade, Double preco, String descricao, boolean ativo)

O construtor registra 'R' no atributo tipo.

tipoResidencia informa se o imóvel é "CASA" ou "APARTAMENTO".

calculaPreco(): Retorna o preço do imóvel com as seguintes opções de tipo e disponibilidade:

CASA/LOCAÇÃO: preço aumenta 10% para pagamento do IPTU.

CASA/COMPRA: preço aumenta em 5%.

APARTAMENTO/LOCAÇÃO: preço aumenta 15% para pagamento de IPTU e condomínio.

APARTAMENTO/COMPRA: preço aumenta em 3% por taxas de liberação.

Classe Comercial:

Construtor Comercial(String localizacao, String disponibilidade, Double preco, String descricao, boolean ativo)

O construtor registra 'C' no atributo tipo.

localização informa se o imóvel se localiza no "CENTRO" ou em "BAIRRO".

calculaPreco(): Retorna o preço do imóvel com as seguintes opções de tipo e disponibilidade:

BAIRRO/LOCAÇÃO: preço aumenta 5%

BAIRRO/COMPRA: preço aumenta em 7%.

CENTRO/LOCAÇÃO: preço aumenta 15%.

CENTRO/COMPRA: preço aumenta em 20%.

Classe Imobiliária

Construtor padrão (vazio).

imoveis: vetor para armazenar até 10 imóveis.

adicionarImovel(Residencial) e **adicionarImovel**(Comercial) recebem um imóvel residencial ou comercial por parâmetro e insere na primeira posição vazia do vetor e retorna um boolean indicando se o imóvel foi adicionado ao vetor.

excluiImovel(String) recebe a descrição por parâmetro e exclui o imóvel do vetor, caso ele exista, e retorna um boolean indicando se o imóvel foi excluído do vetor.

diponiveis() retorna um vetor com os imóveis cadastrados.

residenciais() retorna um vetor de Residenciais com os imóveis residenciais cadastrados.

comerciais() retorna um vetor de Comerciais com os imóveis comerciais cadastrados.

buscaImovel(String) recebe uma descrição por parâmetro e retorna o imóvel com a descricao ou retorna nulo caso não encontre.