**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

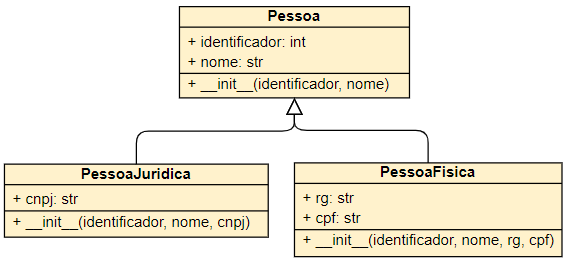
**Exercício 01**

O diagrama abaixo fornece uma hierarquia de classes onde a classe Pessoa é a superclasse (classe mãe), e as classes PessoaFisica e PessoaJuridica são as subclasses (classes filhas).

Crie a classe Pessoa com os atributos identificador e nome.

Crie a classe PessoaJuridica que herda da classe Pessoa e acrescenta o atributo cnpj.

Crie a classe PessoaFisica que herda da classe Pessoa e acrescenta os atributos rg e cpf.



Utilize o trecho de programa abaixo para testar as classes

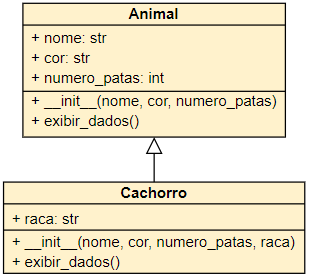
| pessoa1 = Pessoa(1, "Nome da Pessoa")  p\_juridica = PessoaJuridica(2, "Nome da Pessoa Juridica", "1111111111")  p\_fisica = PessoaFisica(3, "Nome da Pessoa Fisica", "222222222", "333333333")  print(pessoa1.identificador) # 1  print(pessoa1.nome) # Nome da Pessoa  print(p\_juridica.identificador) # 2  print(p\_juridica.nome) # Nome da Pessoa Juridica  print(p\_juridica.cnpj) # 1111111111  print(p\_fisica.identificador) # 3  print(p\_fisica.nome) # Nome da Pessoa Fisica  print(p\_fisica.rg) # 222222222  print(p\_fisica.cpf) # 333333333 |
| --- |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Exercício 02**

Crie a classe Animal com os atributos nome, cor e numero\_patas. Crie também o método exibir\_dados, que imprime na tela os dados do animal (nome, cor e numero\_patas).

Crie a classe Cachorro que herda da classe Animal e que possui como atributo adicional a raça do cachorro. Crie também o método exibir\_dados, que imprime na tela os dados do cachorro (nome, cor, numero\_patas e raca)



Utilize o trecho de programa abaixo para testar as classes

| animal = Animal("Passarinho", "Azul", 2)  animal.exibir\_dados() # exibe os atributos do animal  dog = Cachorro("Rex", "Marrom", 4, "Vira lata")  dog.exibir\_dados() # exibe os atributos do cachorro |
| --- |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

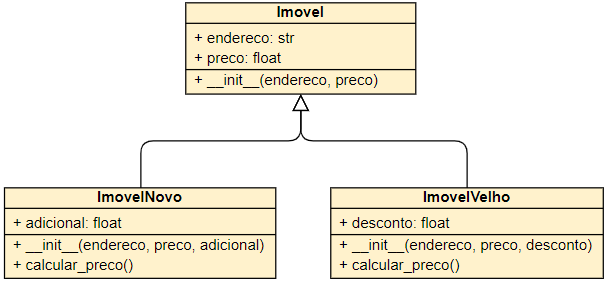
**Exercício 03**

Crie a classe Imovel, que possui um endereço e um preço.

Crie a classe ImovelNovo, que herda de Imovel e possui um adicional no preço.

Crie a classe ImovelVelho, que herda de Imovel e possui um desconto no preço.

O método calcular\_preco das classes deve retornar o preço atualizado de acordo com o adicional ou desconto.



Utilize o programa abaixo para testar as classes

| imovel = Imovel("Rua Silva, 123", 300000.0)  imovel\_novo = ImovelNovo("Rua Joaquim, 999", 250000.0, 20000.0)  imovel\_velho = ImovelVelho("Av. Brasil, 777", 500000.0, 35000.0)  print(imovel.endereco) # Rua Silva, 123  print('Preço:', imovel.preco) # 300000.0  print(imovel\_novo.endereco) # Rua Joaquim, 999  print('Preço:', imovel\_novo.preco) # 250000.0  print('Preço Atualizado:', imovel\_novo.calcular\_preco()) # 270000.0  print(imovel\_velho.endereco) # Av. Brasil, 777  print('Preço:', imovel\_velho.preco) # 500000.0  print('Preço Atualizado:', imovel\_velho.calcular\_preco()) # 465000.0 |
| --- |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Exercício 04**

Escreva um programa para armazenar dados de veículos.

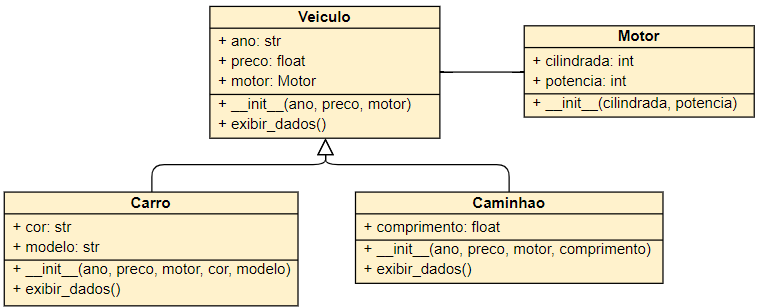
Crie a classe Motor que contém cilindrada e potencia.

Crie a classe Veiculo contendo ano de fabricação, preco e motor. Crie também o metodo exibir\_dados para mostrar os dados do Veículo.

Crie a classe Carro, que herda da classe Veiculo e adiciona os atributos cor e modelo. Crie também o metodo exibir\_dados para mostrar os dados do Carro.

Crie a classe Caminhão, que herda da classe Veiculo e adiciona o atributos comprimento (em metros). Crie também o metodo exibir\_dados para mostrar os dados do Caminhão.

Obs.: *A classe Motor não possui relação de herança com a classe Veiculo, possui apenas uma relação de associação (o veiculo possui um motor)*



Utilize o programa abaixo para testar as classes

| motor1 = Motor(1000, 500)  motor2 = Motor(8000, 900)  carro = Carro(2010, 20000, motor1, "branca", "gol")  caminhao = Caminhao(2015, 80000, motor2, 10)  carro.exibir\_dados() # imprime os valores de todos os atributos do carro  caminhao.exibir\_dados() # imprime os valores de todos os atributos do caminhão |
| --- |