

1

Crie uma classe chamada Pessoa com o atributo nome.  
Crie dois objetos (p1 e p2) e atribua nomes diferentes.  
Imprima o nome de cada pessoa.

---

2.

Crie uma classe Animal com o atributo tipo (ex: cachorro, gato).  
Crie dois objetos e mostre o tipo de cada um.

---

3.

Crie uma classe Carro com o atributo estado definido como 'novo'.  
Crie um carro fusca e outro ferrari, mude o estado do ferrari para 'usado' e imprima os dois estados.

---

4.

Adicione à classe Carro um atributo cor.  
Crie um fusca azul e uma ferrari vermelha.  
Imprima o nome e a cor de cada um.

---

5.

Crie uma classe Aluno com os atributos nome e nota.  
Peça para o usuário digitar os valores e mostre o resultado.

---

6.

Crie uma classe ContaBancaria com o atributo saldo.  
Crie duas contas com saldos diferentes e mostre ambos.

---

7.

Na classe ContaBancaria, adicione um método mostrar\_saldo().  
Chame esse método em dois objetos.

---

**8.**

Crie uma classe Produto com atributos nome e preco.  
Crie três produtos e exiba seus nomes e preços.

---

**9.**

Crie uma classe Computador com o atributo ligado iniciado como False.  
Crie um método ligar() que muda o estado para True.  
Teste o método com um objeto.

---

**10.**

Crie uma classe Lampada com o atributo estado (acesa ou apagada).  
Implemente um método ligar() que altera o estado.  
Mostre o resultado.

---

**11.**

Crie uma classe Funcionario com os atributos nome e salario.  
Imprima o nome e o salário de dois funcionários.

---

**12.**

Crie uma classe Cachorro com o método latir() que imprime "Au Au!".  
Crie um objeto e chame o método.

---

**13.**

Crie uma classe Celular com o atributo bateria.  
Adicione um método carregar() que aumenta o nível da bateria.  
Simule a recarga.

---

**14.**

Crie uma classe Livro com os atributos titulo e autor.  
Crie dois livros diferentes e imprima as informações.

---

**15.**

Crie uma classe Aluno que receba nome e curso pelo método `__init__`.  
Crie dois alunos e exiba seus dados.

---

**16.**

Crie uma classe Conta com saldo = 0.  
Crie um método depositar(valor) que soma ao saldo.  
Teste com dois depósitos diferentes.

---

**17.**

Crie uma classe Conta com método sacar(valor) que diminui o saldo.  
Teste o saque com valores válidos e inválidos (maiores que o saldo).

---

**18.**

Crie uma classe Pessoa com método falar().  
Ao chamar, o método deve imprimir "Olá, tudo bem?".

---

**19.**

Crie uma classe Carro que tenha marca, modelo e ano.  
Peça ao usuário para preencher e depois exiba as informações.

---

**20.**

Crie uma classe Retangulo que receba largura e altura.  
Crie um método area() que retorne o valor da área.  
Teste com um objeto.

---

**21.**

Crie uma classe Aluno que tenha método media(n1, n2) e retorne a média.  
Mostre o resultado.

---

**22.**

Crie uma classe Produto com método desconto(percentual) que reduz o preço.  
Teste com um valor de 10%.

---

**23.**

Crie uma classe Pessoa e adicione um atributo idade.  
Crie um método maior\_de\_idade() que retorna True se idade  $\geq$  18.

---

**24.**

Crie uma classe Banco com atributo clientes (lista).  
Adicione três nomes à lista e imprima todos.

---

**25.**

Crie uma classe Motor com atributo ligado=False.  
Adicione método ligar\_motor() e mude o estado para True.

---

**26.**

Crie uma classe Casa com atributos cor e tamanho.  
Crie duas casas com valores diferentes e exiba os dados.

---

**27.**

Crie uma classe Pessoa com método \_\_init\_\_ que mostra uma mensagem “Pessoa criada”.  
Crie dois objetos e veja o resultado.

---

**28.**

Crie uma classe Carro com método mostrar\_estado() que imprime o valor atual do atributo estado.  
Teste com dois carros.

---

**29.**

Crie uma classe Computador com método info() que mostra “Computador ligado” se o atributo ligado=True.  
Teste com dois objetos.

---

**30.**

Crie uma classe Aluno e um método situacao(nota) que imprime “Aprovado” se  $\text{nota} \geq 7$  e “Reprovado” caso contrário.  
Teste com duas notas diferentes.