

1. Classe Produto

Crie uma classe **Produto** com atributos:

- nome
- preço
- quantidade

E um método que calcule o **valor total em estoque**.

Depois, crie 3 produtos e mostre o valor total de cada um.

2. Classe ContaBancaria

Implemente:

- depositar(valor)
- sacar(valor)
- mostrarSaldo()

Crie um objeto e execute operações reais de conta.

3. Sistema de Funcionários

Crie uma classe **Funcionario** e duas subclasses:

- **Gerente** (recebe 20% de bônus)
- **Vendedor** (recebe 10% de comissão)

O método **calcularSalario()** deve ter comportamento diferente.

4. Carrinho de Compras

Crie classes:

- **Item**
- **Carrinho** (que agrega vários Item)

O carrinho deve:

- adicionar item
 - remover item
 - calcular total
-

5. Agenda Telefônica (Lista de Objetos)

Crie uma classe **Contato** e uma classe **Agenda** que armazena vários contatos.
Implemente:

- adicionar
 - buscar por nome
 - listar todos
-

6. Sistema de Login

Classe **Usuario** com:

- usuário
- senha (privado)

Implemente:

- autenticar(usuario, senha)
-

7. Classe Aluno com Média

Atributos:

- nome
- nota1
- nota2

Método:

- calcularMedia()

Monte um array/lista de 5 alunos.

8. Biblioteca

Classes:

- **Livro**
- **Biblioteca**

A biblioteca deve permitir:

- cadastrar livro
 - listar por autor
 - buscar por título
-

9. Sistema de Pedidos

Classes:

- Pedido
- ItemPedido
- Produto

O pedido calcula valor total somando itens.

10. Jogo simples

Classe **Personagem**:

- nome
- vida
- atacar(outroPersonagem)

Simule uma batalha.

11. Classe Carro

Métodos:

- ligar
- desligar
- acelerar
- frear

Simule o uso na main.

12. Sistema de Filmes

Classe **Filme** com título, ano, gênero.

Faça uma lista de filmes e implemente filtro por gênero.

13. Classe Aluno com Construtor Obrigatório

Crie obrigatoriamente: nome e matrícula.

14. Interface Pagamento

Crie uma interface:

- pagar()

Implemente classes:

- PagamentoPix
 - PagamentoCartao
-

15. Calculadora com Sobrecarga

Métodos:

- somar(int a, int b)
 - somar(double a, double b, double c)
-

16. Sistema de Transporte

Classe pai **Veiculo**

Filhas:

- Carro
- Moto

Ambas implementam **mover()** de forma diferente.

17. Turma de alunos

Classe **Turma** que contém uma lista de **Aluno**.

Método:

- calcular média geral da turma
-

18. Sistema de Mensagens

Classe **Mensagem**

- texto
- remetente
- destinatário

Lista de mensagens filtrável por destinatário.

19. Banco de dados fictício

Classe **BancoDados** simulando CRUD em lista:

- inserir()
 - atualizar()
 - deletar()
 - listar()
-

20. Empresa com funcionários

Crie composição:

- Empresa
- Lista de Funcionários

Método: totalFolhaPagamento()

1. Selecionar todos os produtos

Mostre todos os dados cadastrados na tabela produtos.

2. Selecionar somente nome e preço

Liste apenas os campos nome e preco.

3. Buscar produtos com preço maior que 50

Mostre somente produtos que custam mais de 50 reais.

4. Buscar produtos com estoque menor que 10

Liste produtos que estão ficando sem estoque.

5. Ordenar produtos por preço do menor para o maior

Ordene todos os produtos em ordem crescente de preço.

6. Ordenar produtos por estoque do maior para o menor

Organize a lista pelo maior estoque primeiro.

7. Buscar produtos cujo nome começa com a letra "A"

Use LIKE 'A%'.

8. Buscar produtos cujo nome termina com “o”

Use LIKE '%o'.

9. Buscar produtos que contém “tech” no nome

Use LIKE '%tech%'.

10. Exibir produtos com preço entre 20 e 100

Use BETWEEN.

11. Exibir produtos com estoque entre 5 e 50

Listar somente valores intermediários.

12. Mostrar produtos sem email cadastrado

Use IS NULL.

13. Mostrar produtos que possuem email cadastrado

Use IS NOT NULL.

14. Mostrar os 5 produtos mais caros

Use ORDER BY + LIMIT.

15. Mostrar os 10 produtos com menor estoque

Use ORDER BY estoque ASC LIMIT 10.

16. Contar quantos produtos existem

Use COUNT(*).

17. Calcular o preço médio dos produtos

Use AVG(preco).

18. Mostrar o estoque total somado

Use SUM(estoque).

19. Mostrar o produto mais caro

Use MAX(preco).

20. Mostrar o produto mais barato

Use MIN(preco).