

1º -

Linha 01

true

false

false

true

false

Linha 02

false

false

false

false

true

Linha 03

true

true

false

true

True

3º -

Classe:

Uma classe é uma forma de definir um tipo de dado, onde definimos suas características, atributos, comportamentos, dados é a base da programação.

Objeto:

Objeto é uma instancia que contém todos os dados de uma determinada classe, com esse objeto podemos acessar todos os dados dessas mesmas classes e até mesmo manipular esses dados.

Herança:

Herança é basicamente uma determinada classe que tem uma “Classe pai”, quando se cria uma classe herdando de outra, teremos acesso a todos dados do pai, incluindo atributos e métodos, por exemplo, se eu tenho uma classe Estagiário e crio uma classe filho chamada EstagioDev e extendo a classe Estagio nela, eu vou ter todos os atributos da classe Pai (Estagiário) no Filho (EstagioDev), porem o pai não tem os dados do filho.

Polimorfismo:

Polimorfismo envolve de certa forma a herança, é quando você herda de uma classe abstrata, e seus métodos abstratos são herdados, porém são herdados somente as assinaturas do método, a implementação desse método na classe aonde você está herdando fica com você.

Interfaces

Interface é usada para a padronização do projeto, ela funciona bem parecido com o polimorfismo, a diferença da interface para uma classe comum é que não podemos instanciar ela, é na interface que criamos nossos métodos (somente as assinaturas dos métodos) e implementamos em nossa classe, podemos implementar quantas interface quisermos (na herança da classe concreta só podemos herdar de 1 classe).

5º

Não usaria esse código,

O que eu mudaria:

A assinatura do método está incorreta, o método de comparação está incorreto também, na assinatura está "cargo" e no corpo do método está "carg", está com vírgula ao invés de '.', o que iria dar erro de compilação e execução, essa estrutura condicional poderia ser uma estrutura de swift case, reduzindo o tempo de execução, pois o if analisa if por if, já o case não, e iria transformar o tipo de funcionário em um ENUM.

Ficaria um pouco parecido com isso minha solução:

```
public double calcularSalario(String tipoFuncionario) {

    switch (tipoFuncionario) {

        case "fullstack1":

            return 2500.00;

        case "fullstack2":

            return 4500.00;

        case "fullstack3":

            return 6500.00;

        case "gerente":

            return 12000.00;

    }

    throw new RuntimeException("Não existe esse cargo");

}
```

9º -

```
SELECT lei.*, ende.*, empre.emprestimo_data
FROM endereco ende JOIN leitor lei ON lei.leitor_endereco_fk = ende.endereco_idem
JOIN emprestimo empre ON lei.leitor_idem = empre.emprestimo_leitor_fk
JOIN livro ON livro.livro_idem = empre.emprestimo_livro_fk
WHERE livro.livro_idem = (SELECT livro.livro_idem FROM livro WHERE livro.livro_titulo =
'Criptografia e Segurança de Redes')
```

10º -

```
SELECT COUNT(*) FROM emprestimo WHERE date_part('month',emprestimo.emprestimo_data) =
date_part('month', CURRENT_DATE)
```

11º -

A tecnologia WebSocket serve para manter uma conexão contínua com o servidor, evitando que faça uma nova requisição a todo momento, um exemplo que poderia dar é um chat de conversa online, evitar uma atualização a todo momento que o usuário mande uma mensagem, a metodologia do WebSocket mantém um tipo de conexão entre os chats.

[Watlas/ProvaSoluti \(github.com\)](https://github.com/Watlas/ProvaSoluti) - Repositório com os programas que foram pedidos na prova