**Classe**

O que é uma classe?

Classe é a arquitetura de um objeto ou um molde de um objeto

Assim como uma casa é construída através de plantas.

Um **OBJETO** é construído ou instanciado através de sua **classe**.

Então, objeto é igual a instancia de uma classe e suas propriedades

**Objeto**

O objeto consiste em duas coisas

Estado e Comportamento

O **Estado do objeto** é composto pelos atributos/dados e ele não é exposto para fora do objeto, visível somente para nos programadores dentro da nossa interface.

O **Comportamento** é todo o código que atua neste estado por meio de uma interface visível, onde a lógica é exposta através de métodos, onde ai sim alguém de fora (outros objetos) tem contado com o estado do objeto

**O que é fundamental para um objeto?**

**Comportamentos =**

visualizarMetadados()

curtirPost()

**Estado =**

quantidadeDeCurtidas

(**comportamento** é o que vaza do objeto, **Estado** é o que fica dentro do objeto)

**Em um post no instagran o comportamento é**:

A imagem exibida

Os comentários

As ações de curtir, comentar, compartilhar.

Os metadados

A caixa de comentário

**Em um post no instagran o Estado é:**

Quantidade de likes

Quem não pode ver a postagem

Filtro usado na foto do post

Identificados do post

Tags utilizadas

Tipo de post(tipo imagem ou tipo vídeo)

A maneira pelas quais os Objetos vão agir são limitadas, ou seja, o objeto só pode ser interagido através de uma interface.

**Encapsulamento**

O Objeto oculta seus detalhes internos por trás de uma interface.

**Ocultação de informação e Acesso restrito**

**Ocultação de informação:**

O código pode tocar o objeto, mas ele é limitado dentro do próprio objeto.

**Acesso Restrito:**

Uma interação com o objeto é restrita através do que foi exposto através de uma interface

O comportamento do objeto é determinado pela sua interface.

**Hierarquia de um objeto**

Existem dois tipos de hierarquia

**Interna (taxonomia)**

**Externa (coleções)**

Interna (taxonomia):

Evitar um pouco, gerar associações, exemplo, um post no insta, esse post pode ser, curtido, se pode ser curtido, outras pessoas podem ver, se outras pessoas podem ver, podem curtir.

(o cachorro é um mamífero, um livro é uma publicação)

Externa (coleções):

É uma hierarquia por contenção, tipo pastas que dentro contêm arquivos.

Ex: uma empresa tem departamentos, e departamentos tem funcionários.

**Herança**

Quando um objeto estende a características de outro objeto.

Post do tipo foto ou vídeo

Post é um objeto pai, que pega os objetos filhos, vídeo e foto

Ícone

Descrição gerada automaticamente com confiança baixa

A foto e o veio herdam características do post, essas características são comportamentos, representados por uma interface.

O Objetivo é diminuir o número de linhas duplicadas no projeto.

Se o post tem o comportamento de curtir algo, o objeto foto pode herdar esse comportamento.

Detalhe que sempre tentar fazer **composição** no lugar de **herança.**

**Polimorfismo**

Um objeto que apresenta a mesma interface, pode se comportar de maneira diferente.

EX: enquanto em um vídeo o objeto mostra como metadados os viwes, no objeto foto mostra como metadado as curtidas

Diagrama

Descrição gerada automaticamente

**Princípios orientados a função**

**(USANDO ESSES CONCEITOS EM KOTLIN)**

Orientação orientada a função, vem ganhando bastante forca nos últimos anos.

Composta de três princípios

**1º Imutabilidade**

As funções não alteram o estado do sistema

**2º funções de primeira classe**

As funções são definidas e usadas no mesmo nível que as classes.

**3º Funções puras**

As funções apenas recebem parâmetros, executam o cálculo e retornam os resultados.

Um objeto imutável é aquele objeto no qual o estado dele não pode ser alterado, nem mesmo dentro dele.