

Java completo

Operadores Java - Tabela de Precedência

Quando maior o número, maior é a precedência

Precedência	Operador	Tipo	Associatividade
15	() [] .	Parentesis Seleção em array Seleção de membros	Esquerda p/ Direita
14	++ --	Incremento pós-fixado Decremento pós-fixado	Direita p/ Esquerda
13	++ -- + - ! ~ (type)	Incremento pré-fixado Decremento pré-fixado Mais Menos Negação lógica Complemento binário Conversão explícita (cast)	Direita p/ Esquerda
12	* / %	Multiplicação Divisão Módulo	Esquerda p/ Direita
11	+ -	Adição Subtração	Esquerda p/ Direita
10	<< >> >>>	Deslocamento aritmético p/ esquerda Deslocamento aritmético p/ direita Deslocamento lógico p/ direita	Esquerda p/ Direita
9	< <= > >= instanceof	Menor que Menor que ou igual Maior que Maior que ou igual Comparação de tipos (objects only)	Esquerda p/ Direita
8	== !=	Igualdade Desigualdade	Esquerda p/ Direita
7	&	E lógico (bit a bit)	Esquerda p/ Direita
6	^	Ou exclusivo (bit a bit)	Esquerda p/ Direita

5		Ou lógico (bit a bit)	Esquerda p/ Direita
4	&&	E lógico	Esquerda p/ Direita
3		OU lógico	Esquerda p/ Direita
2	? :	Condição ternária	Direita p/ Esquerda
1	= += -= *= /= %=	Atribuição Atribuição aditiva Atribuição subtrativa Atribuição de Multiplicação Atribuição de Divisão Atribuição de Módulo	Direita p/ Esquerda

JRE vs JDK

JRE - Java Runtime Environment (quem usa é o usuário) → é um ambiente de execução/ sinônimo da JVM

JDK - Java Development Kit (quem usa é o desenvolvedor)

JVM - Java Virtual Machine (lê ByteCode e interpreta/traduz para os códigos de máquina do SO)

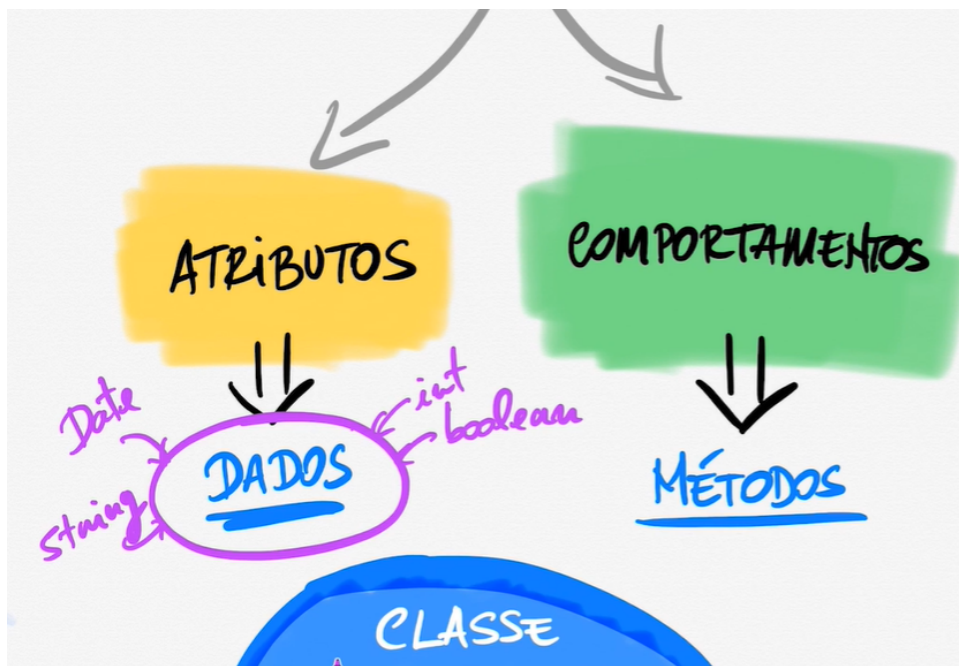
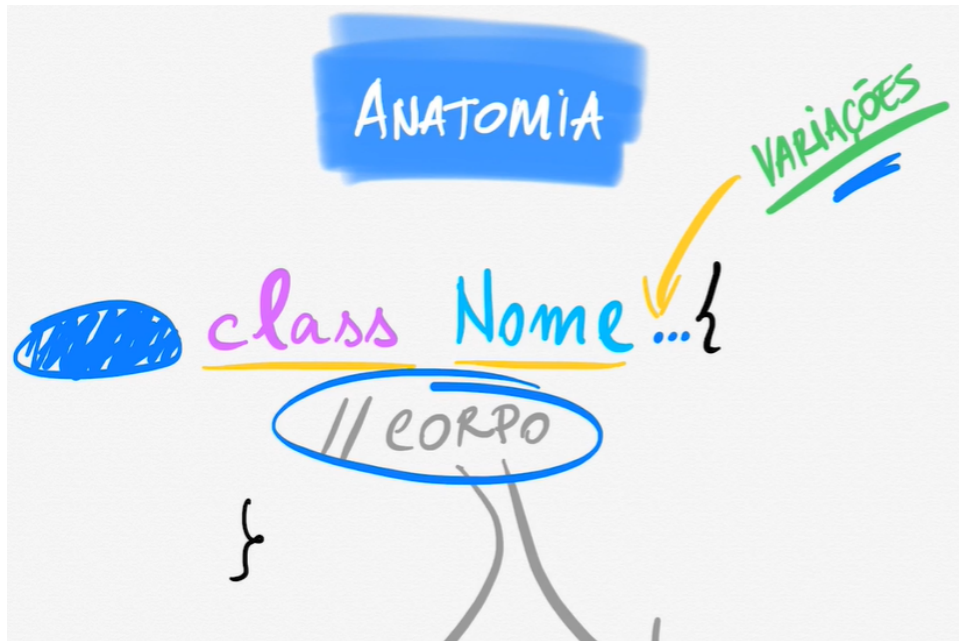
- (.java → .class (ByteCode))

Javac - java compiler - é o que transforma um arquivo .java em um arquivo .class

Programa em Java - chama um arquivo .class pra maquina virtual interpretar esse arquivo e executar-lo

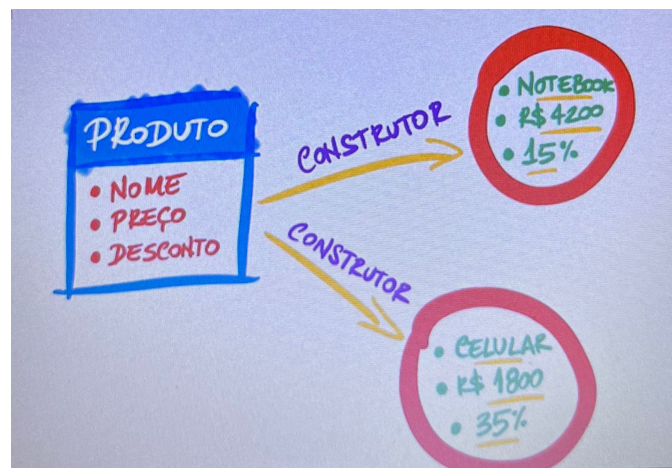
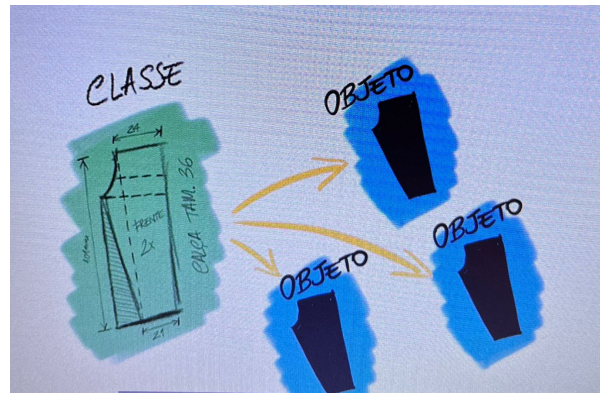
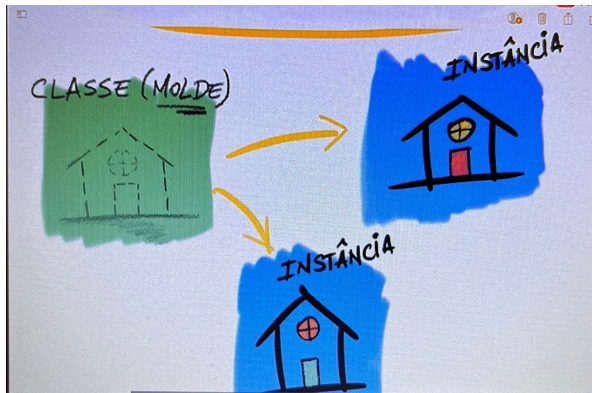
Classes

- Uma classe define um bloco de código
- Uma classe pública deve ter o mesmo nome de seu arquivo



Membros de uma classe = são os atributos e comportamentos

- Uma classe define um TIPO (e um tipo é uma estrutura de dados)
- Uma classe representa uma abstração (simplificação do mundo real)
- A partir de uma classe, criamos objetos, que são dados.
- Objeto == Instância



Uma classe é composta por:

- Variáveis/Constantes
- Construtores
- Métodos
- Classes

Há variáveis que pertencem à classe e há variáveis que pertencem ao objeto!

objeto.membro:

Para acessar os membros que pertencem à um objeto ou à uma classe, usamos a notação ponto. Ou seja, a partir de um objeto, podemos acessar os membros dele por meio da notação ponto (exemplo → `System.out.println`)