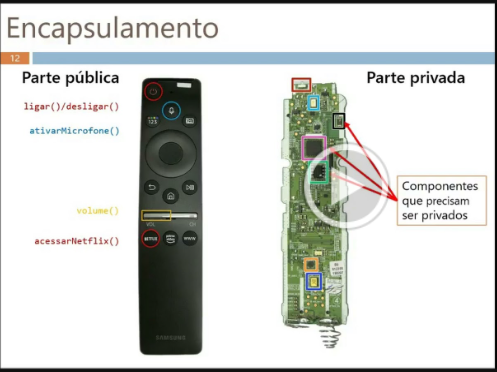
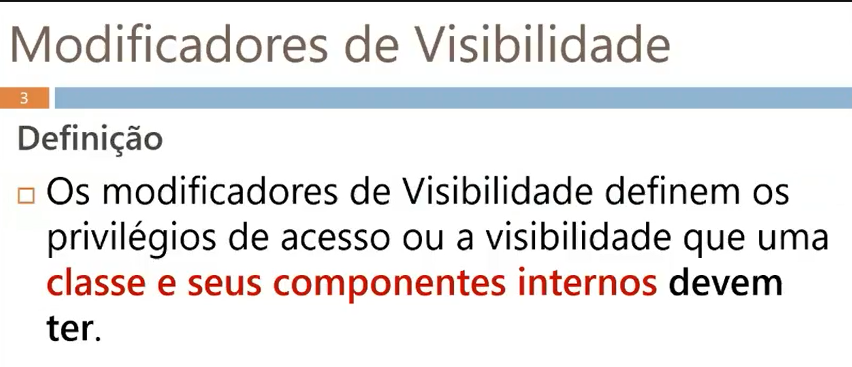
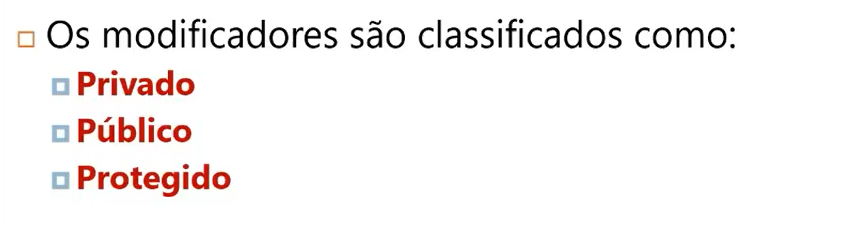
**04-POO - Modificadores de Acesso e Encapsulamento**

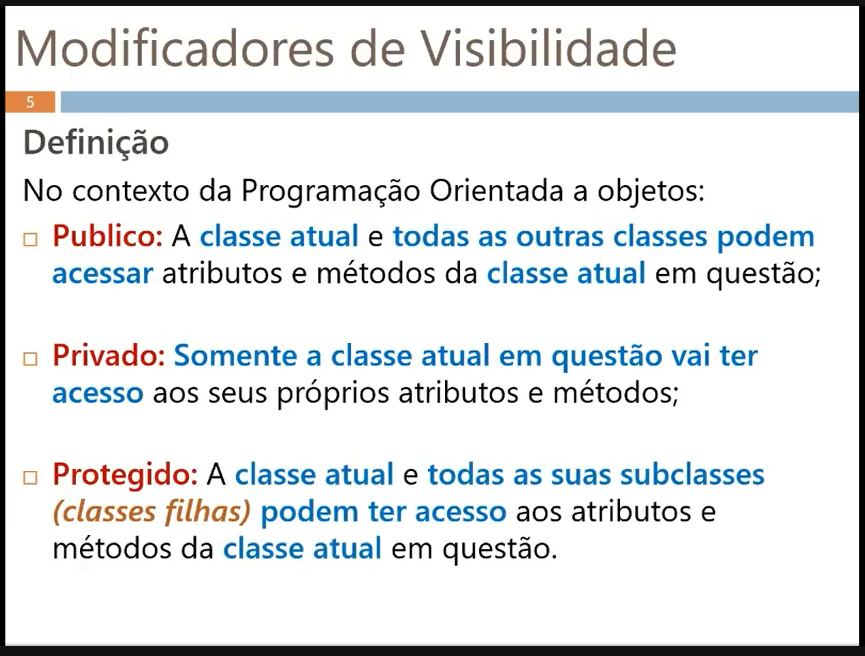
**s**

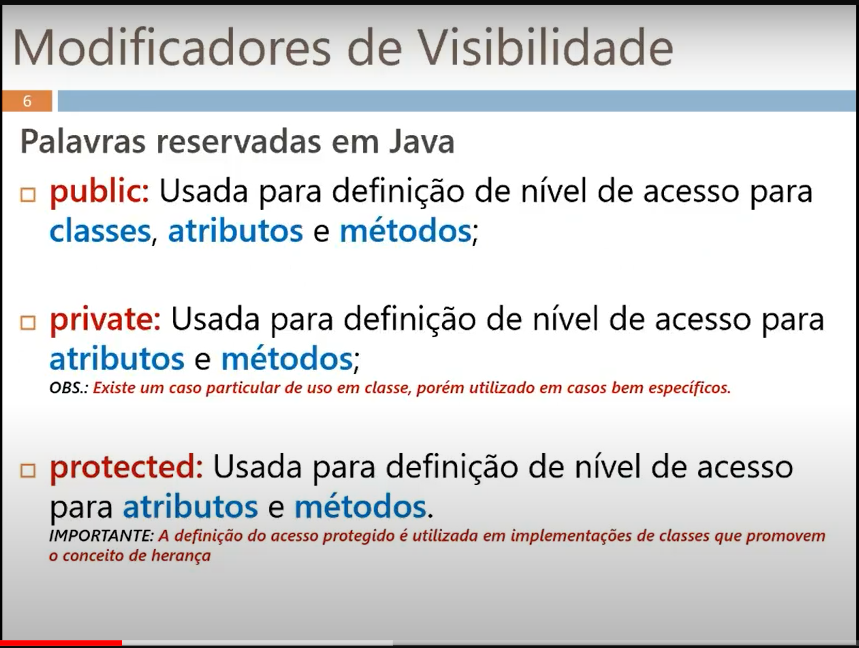
****

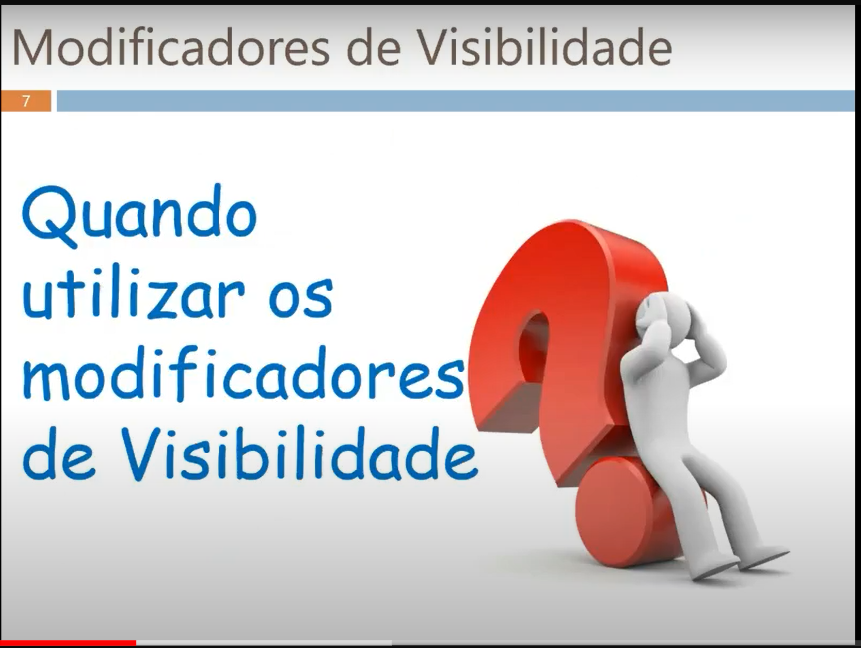
****

**** **** 

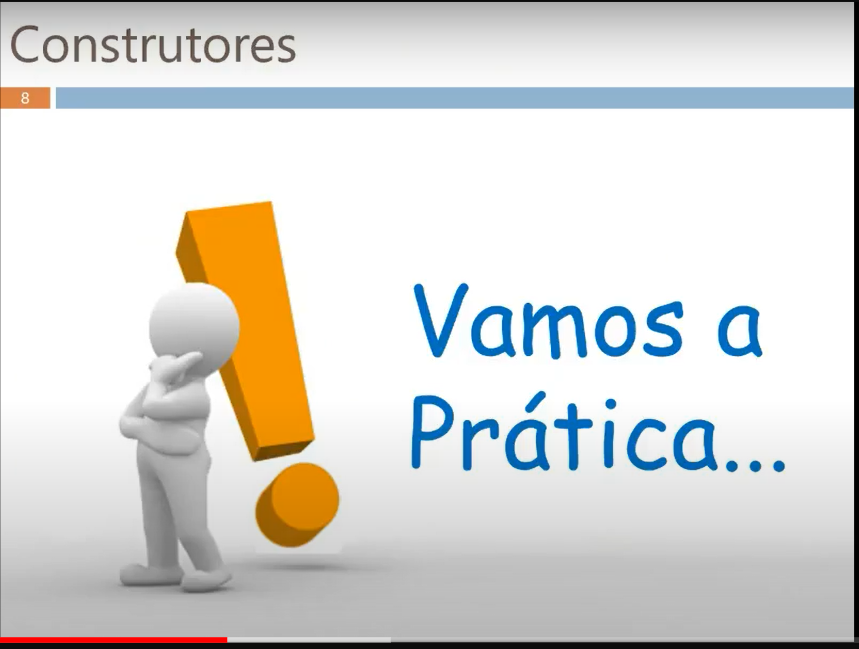
**PÚBLICO PRIVADO PROTEGIDO**

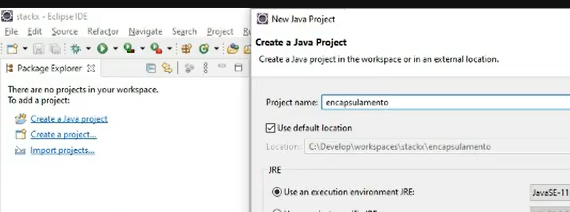
****

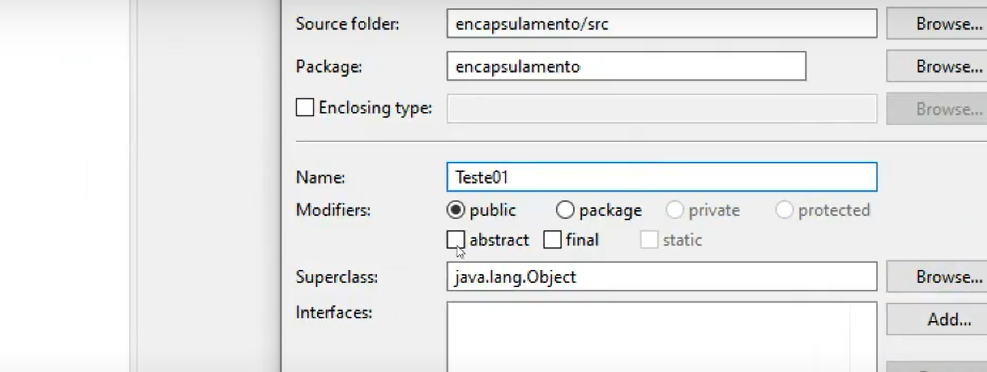
****

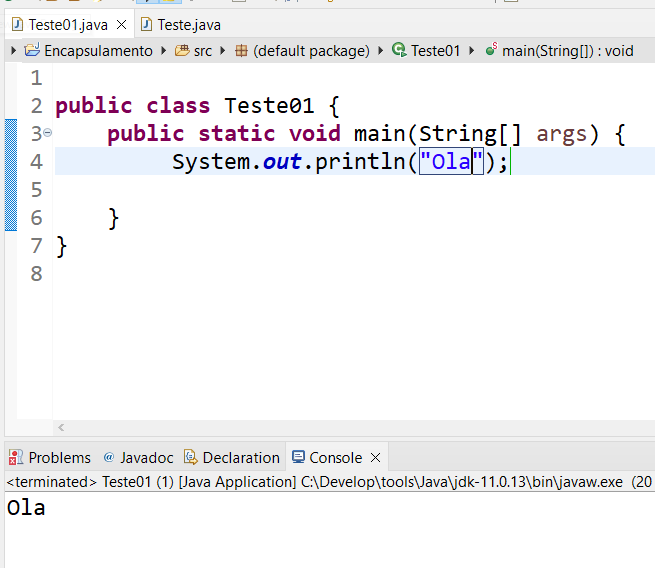
****

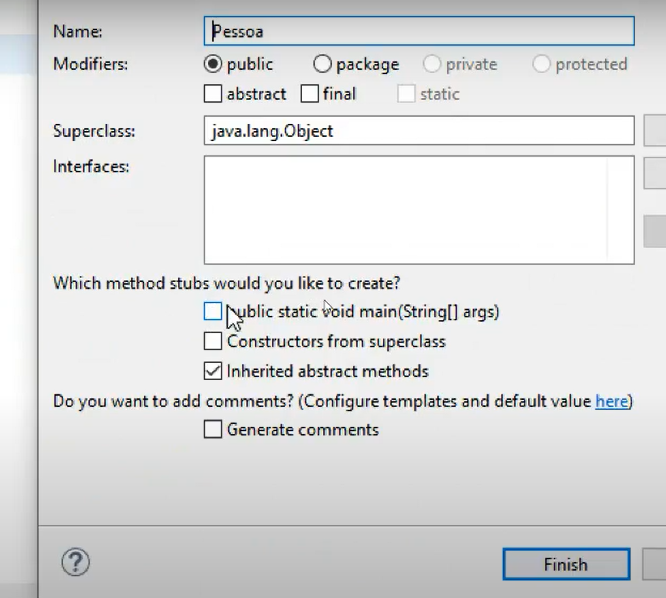
**AREA PRIVADA, CONTROLE PRIVADO**

****

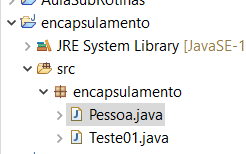
****

****

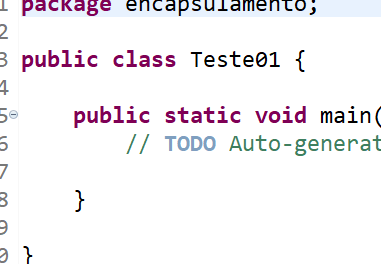
****

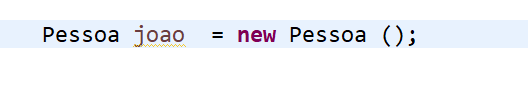
****

24:25 / 1:37:41

****

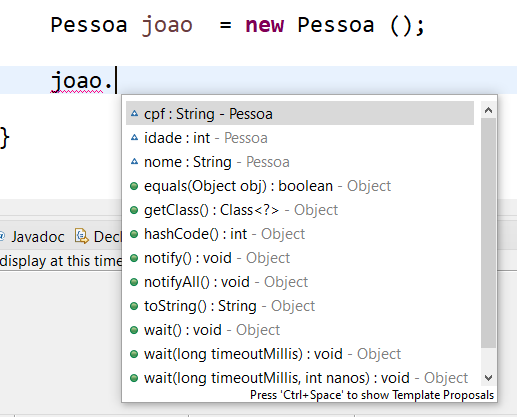
**Criar....**

**Agora irei instanciar o objeto Pessoa**

****

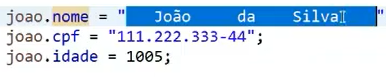
**Método construtor default ou vazio;**

**Quero acessar os atributos do objeto: joao.**

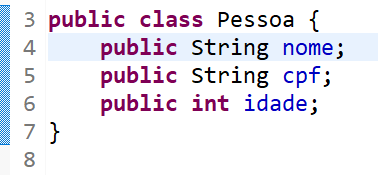
****

****

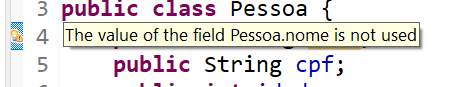
**Tenho que restringir os dados para que seja garantida a sua integridade;**

****

**Como funciona os MODIFICADORES DE ACESSO**

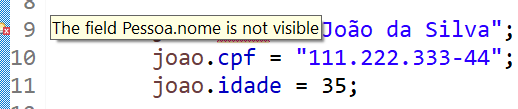
****

**quando não esta definido nem publico e nem privado... é considerado default**

****

****

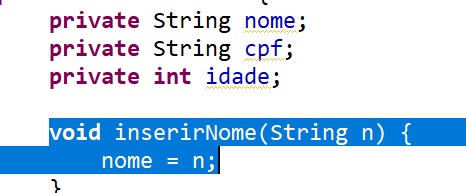
**TESTE01**

****

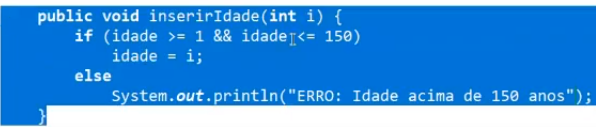
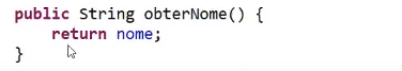
****

****

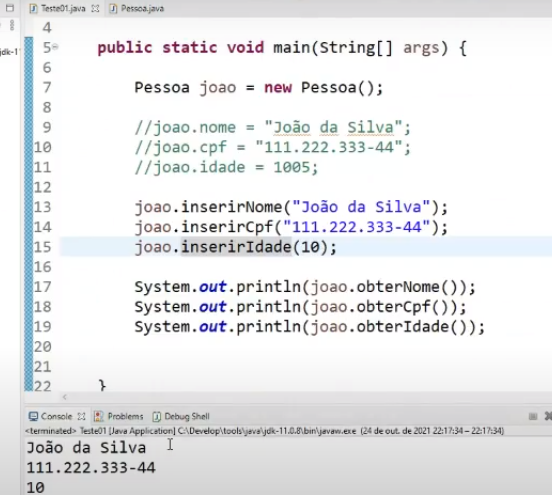
**Este é um METODO INSERIR NOME**

****

****

**** ****

49:22 / 1:37:41

****

**Teste01**

**package** encapsulamento;

**public** **class** Teste01 {

**public** **static** **void** main(String[] args) {

Pessoa joao = **new** Pessoa ();

//joao.nome = "João da Silva";

//joao.cpf = "111.222.333-44";

//joao.idade = 35;

joao.inserirNome("João da Silva");

joao.inserirCpf("111.222.333-44");

joao.inserirIdade(30);

System.***out***.println(joao.obterNome());

System.***out***.println(joao.obterCpf());

System.***out***.println(joao.obterIdade());

}

}

**package** encapsulamento;

**public** **class** Pessoa {

**private** String nome;

**private** String cpf;

**private** **int** idade;

**public** **void** inserirNome(String n) {

nome = n;

}

**public** **void** inserirCpf(String c) {

cpf = c;

}

**public** **void** inserirIdade(**int** i) {

**if** (i >= 1 && i <= 125) {

idade = i;

}**else** {

System.***out***.println("ERRO: Idade acima de 125 anos");

}

}

**public** String obterNome() {

**return** nome;

}

**public** String obterCpf() {

**return** cpf;

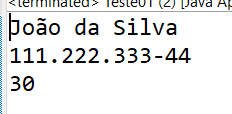
}

**public** **int** obterIdade() {

**return** idade;

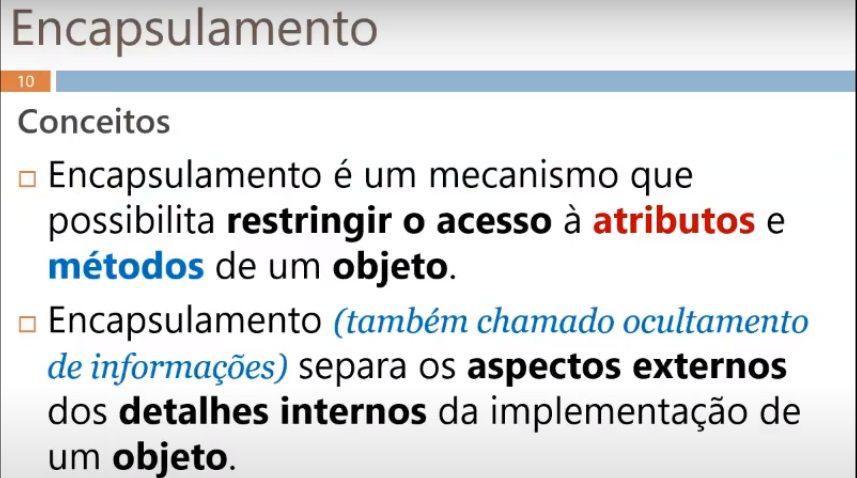
}

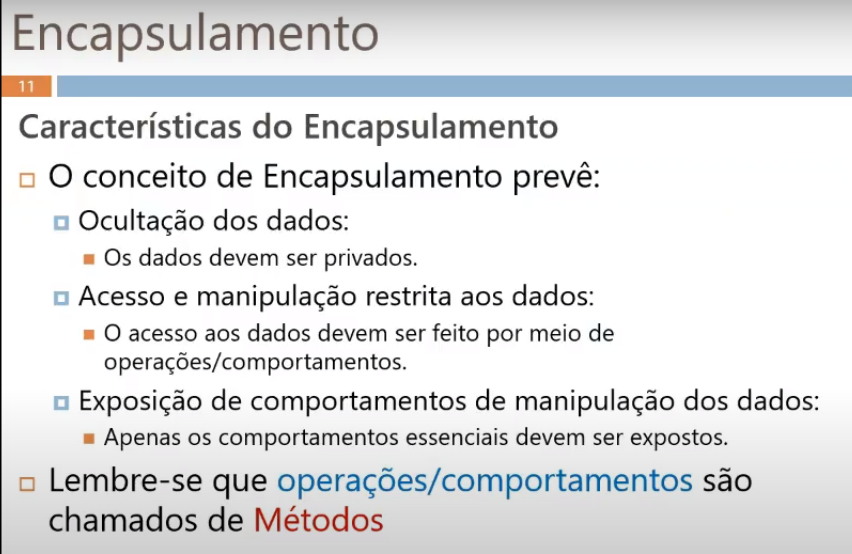
}

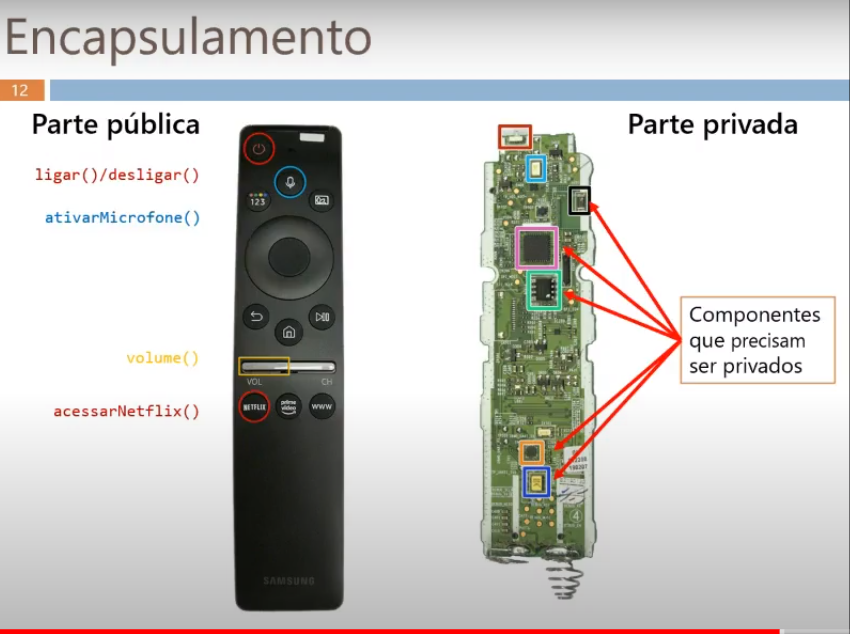
****

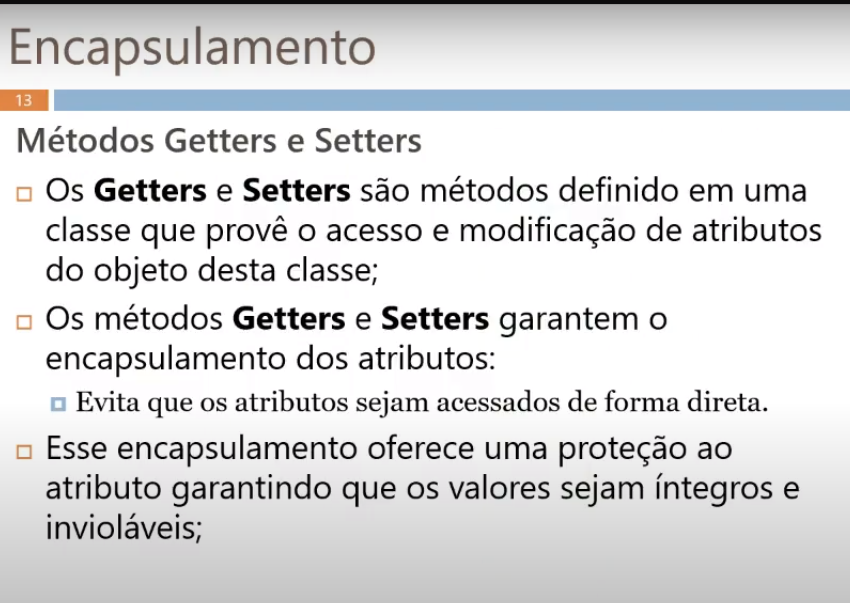
****

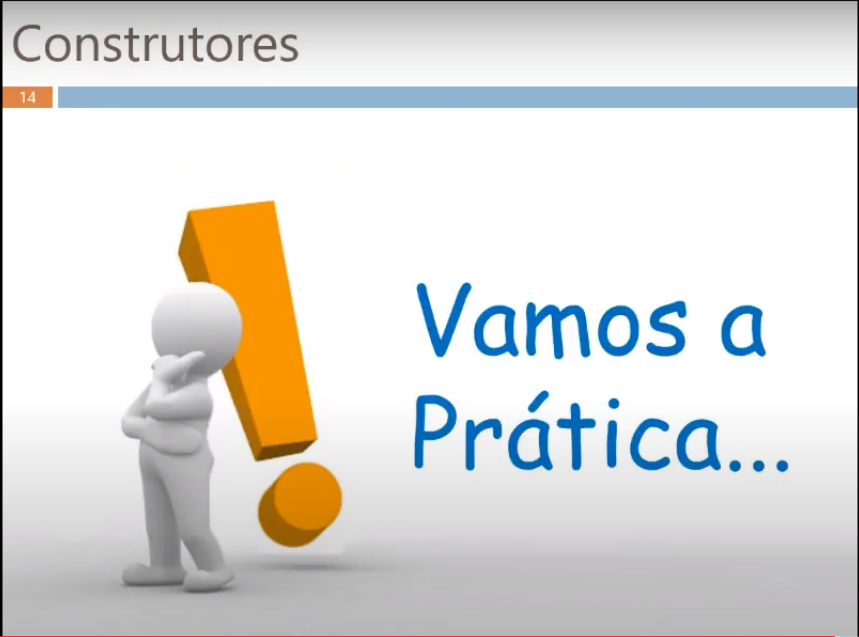
****

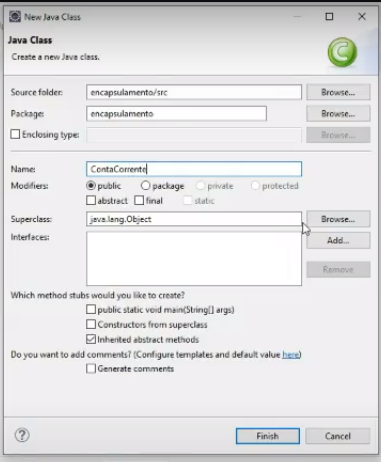
****

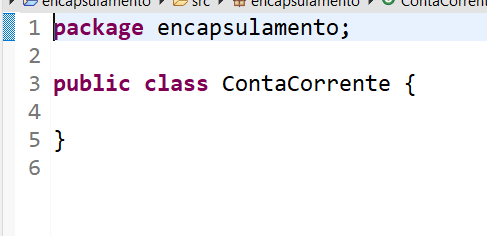
****

****

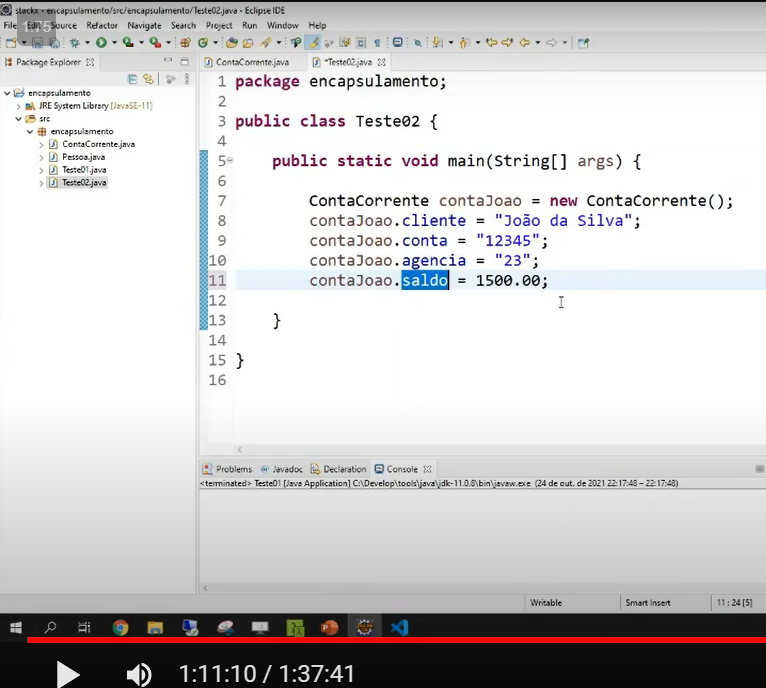
****

****

****

****

1:09:16 / 1:37:41

****

**METODO GETTER & SETTER:**

1. **MODELO ANTIGO:**

**package** encapsulamento;

**public** **class** ContaCorrente {

**private** String cliente;

**private** String conta;

**private** String agencia;

**private** **double** saldo;

**public** **void** inserirCliente(String c) {

cliente = c;

}

**public** **void** inserirConta(String ct) {

conta = ct;

}

**public** **void** inserirAgencia(String a) {

agencia = a;

}

**public** **void** inserirSaldo(**double** s) {

saldo = s;

}

**public** String obterCliente() {

**return** cliente;

}

**public** String obterConta() {

**return** conta;

}

**public** String obterAgencia() {

**return** agencia;

}

**public** **double** obterSaldo() {

**return** saldo;

}

}

**package** encapsulamento;

**public** **class** Teste02 {

**public** **static** **void** main(String[] args) {

ContaCorrente contaJoao = **new** ContaCorrente();

//contaJoao.cliente = "João da Silva";

//contaJoao.conta = "12345";

//contaJoao.agencia = "23";

//contaJoao.saldo = 1500.00;

contaJoao.inserirCliente("João da Silva");

contaJoao.inserirConta("12345");

contaJoao.inserirAgencia("23");

contaJoao.inserirSaldo(1500);

System.***out***.println(contaJoao.obterCliente());

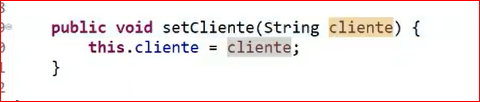
System.***out***.println(contaJoao.obterConta());

System.***out***.println(contaJoao.obterAgencia());

System.***out***.println(contaJoao.obterSaldo());

}

}



**O this representa o próprio objeto;**

 **o parâmetro cliente..**

**este é o atributo** 

A ideia do encapsulamento é : alem de restringir os dados

- ter o poder de controlar e validar o que vai entrar ou não para cada um dos atributos ,

Ou seja, é restringir mesmo de forma que os dados aqui dentro fiquem coerentes, coesos, integros e invioláveis



**2-MODELO NOVO:**

**ContaCorrente**

**package** encapsulamento;

**public** **class** ContaCorrente {

**private** String cliente;

**private** String conta;

**private** String agencia;

**private** **double** saldo;

**public** **void** setCliente(String cliente) {

**this**.cliente = cliente;

}

**public** **void** setConta(String conta) {

**this**.conta = conta;

}

**public** **void** setAgencia(String agencia) {

**this**.agencia = agencia;

}

/\* public void setSaldo(double saldo) {

this.saldo = saldo;

}\*/

**public** **void** sacar(**double** valor) {

**if** (**this**.saldo >= valor) {

**this**.saldo = **this**.saldo - valor;

}**else** {

System.***out***.println("ERRO: Saldo insuficiente!");

}

}

**public** **void** depositar(**double** valor) {

**this**.saldo = **this**.saldo + valor;

}

//--------------------------------------

**public** String getCliente() {

**return** **this**.cliente;

}

**public** String getConta() {

**return** **this**.conta;

}

**public** String getAgencia() {

**return** **this**.agencia;

}

**public** **double** getSaldo() {

**return** **this**.saldo;

}

**public** **void** imprimirExtrato() {

System.***out***.printf("CLIENTE : %s\n" , **this**.cliente);

System.***out***.printf("AGENCIA : %s\n" , **this**.agencia);

System.***out***.printf("CONTA : %s\n" , **this**.conta);

System.***out***.printf("SALDO : R$%.2f\n\n" , **this**.saldo);

}

}

**Teste02**

**package** encapsulamento;

**public** **class** Teste02 {

**public** **static** **void** main(String[] args) {

ContaCorrente contaJoao = **new** ContaCorrente();

//contaJoao.cliente = "João da Silva";

//contaJoao.conta = "12345";

//contaJoao.agencia = "23";

//contaJoao.saldo = 1500.00;

contaJoao.setCliente("João da Silva");

contaJoao.setConta("12345");

contaJoao.setAgencia("23");

//contaJoao.setSaldo(1500.00);

contaJoao.imprimirExtrato();

contaJoao.depositar(100);

contaJoao.imprimirExtrato();

contaJoao.depositar(100);

contaJoao.imprimirExtrato();

contaJoao.sacar(100);

contaJoao.imprimirExtrato();

contaJoao.sacar(1000);

contaJoao.imprimirExtrato();

}

}

CLIENTE : João da Silva

AGENCIA : 23

CONTA : 12345

SALDO : R$0,00

CLIENTE : João da Silva

AGENCIA : 23

CONTA : 12345

SALDO : R$100,00

CLIENTE : João da Silva

AGENCIA : 23

CONTA : 12345

SALDO : R$200,00

CLIENTE : João da Silva

AGENCIA : 23

CONTA : 12345

SALDO : R$100,00

ERRO: Saldo insuficiente!

CLIENTE : João da Silva

AGENCIA : 23

CONTA : 12345

SALDO : R$100,00

**GUSTAVO GUANABARA**