//criação da classe cliente

class cliente{

#nome\_cliente // foi utilizado o “#” para tornar o atributo privado e fazer o encapsulamento

#idade\_cliente

#saldo\_cliente

constructor (nome\_cliente, idade\_cliente, saldo\_cliente){

this.#nome\_cliente= nome\_cliente

this.#idade\_cliente= idade\_cliente

this.#saldo\_cliente= saldo\_cliente

// a utilização do “this” é para se referir ao objeto ou variável que foi criada

}

mostrarInformacoes(){

console.log(`o nome do cliente é: ${this.#nome\_cliente}`)

console.log(`a idade é: ${this.#idade\_cliente}`)

console.log(`o saldo é: ${this.#saldo\_cliente}`)

}

atualizarSaldo(valor){

if(valor > 0){

this.#saldo\_cliente += valor

console.log(`o saldo atualizado é: ${this.#saldo\_cliente}`)

}else{console.log(`valor invalido então o saldo é o mesmo: ${this.#saldo\_cliente}`)}

//aqui é para atualizar o saldo se o valor for menor que 0 o saldo não poderá ser atualizado e aparecera a mensagem “valor invalido então o saldo é o mesmo: saldo”

}

}

const cliente1= new cliente ("joão", 30, 1000)

cliente1.mostrarInformacoes()

cliente1.atualizarSaldo(55)

const cliente2= new cliente ("erison", 19, 500)

cliente2.mostrarInformacoes()

cliente2. atualizarSaldo(200)

// aqui são os testes que fiz