

Consolidando o seu conhecimento

Chegou a hora de você pôr em prática o que foi visto na aula. Para isso, execute os passos listados abaixo.

- 1) Crie na raiz do projeto o arquivo **banco.php**, com o seguinte conteúdo:

```
<?php

require_once 'src/Conta.php';

$primeiraConta = new Conta();
$primeiraConta->depositar(500);
$primeiraConta->sacar(300);
$primeiraConta->saldo -= 300;
```

- 2) Na classe `Conta`, torne todos os atributos privados, da seguinte forma:

```
<?php

class Conta
{
    private string $cpfTitular;
    private string $nomeTitular;
    private float $saldo = 0;

    // ...
}
```

- 3) Execute este arquivo (no terminal, digite `php banco.php`) e veja que o PHP exibe um erro. Não é mais possível acessar o `saldo` diretamente.

- 4) Crie um método para recuperar o saldo de uma conta:

```
// ...
public function recuperarSaldo(): float
{
    return $this->saldo;
}
// ...
```

- 5) Agora, através de `$primeiraConta->recuperarSaldo()`, é possível acessar o saldo da conta.

- 6) Crie o restante dos métodos de acesso, inclusive para alterar os valores do CPF e nome do titular. Sua classe deve se parecer com a classe abaixo:

```
<?php

class Conta
{
    private string $cpfTitular;
    private string $nomeTitular;
    private float $saldo = 0;

    public function recuperarSaldo(): float
    {
        return $this->saldo;
    }

    public function depositar(float $valor)
    {
        $this->saldo += $valor;
    }

    public function sacar(float $valor)
    {
        $this->saldo -= $valor;
    }

    public function setNomeTitular(string $nome)
    {
        $this->nomeTitular = $nome;
    }

    public function setCpfTitular(string $cpf)
    {
        $this->cpfTitular = $cpf;
    }
}
```

```
private string $cpfTitular;
private string $nomeTitular;
private float $saldo = 0;

public function saca(float $valorASacar): void
{
    if ($valorASacar > $this->saldo) {
        echo "Saldo indisponível";

        return;
    }

    $this->saldo -= $valorASacar;
}

public function deposita(float $valorADepositar): void
{
    if ($valorADepositar < 0) {
        echo "Valor precisa ser positivo";
        return;
    }

    $this->saldo += $valorADepositar;
}

public function transfere(float $valorATransferir, Conta $contaDestino): void
{
    if ($valorATransferir > $this->saldo) {
        echo "Saldo indisponível";
        return;
    }

    $this->sacar($valorATransferir);
    $contaDestino->depositar($valorATransferir);
}

public function recuperaSaldo(): float
{
    return $this->saldo;
}

public function defineCpfTitular(string $cpf): void
{
    $this->cpfTitular = $cpf;
}

public function recuperaCpfTitular(): string
{
    return $this->cpfTitular;
}

public function defineNomeTitular(string $nome): void
{
    $this->nomeTitular = $nome;
}

public function recuperaNomeTitular(): string
{
```

```
    return $this->nomeTitular;  
}  
}
```

7) Note que os métodos foram renomeados para manter um padrão na conjugação dos verbos no imperativo, ao invés do infinitivo, como estava antes.