

```
<?php
```

```
//ATRIBUTOS E MÉTODOS ESTÁTICOS:
```

```
//Só lembrando: atributos e métodos estáticos são atributos e métodos que só podem ser usados dentro da própria class...
```

```
class Forma {
```

```
    private static $contagem_de_bolos = 0; //Temos aqui um atributo estático e privado, só pode ser usado dentro da
                                           //classe...
```

```
    public function __construct(string $sabor, string $formato){
        $this->sabor = $sabor;
        $this->formato = $formato;
```

```
        Forma::acrescentaBolo(); //Veja como acessamos o método estático, temos que referencia a própria classe
    }                               //usando o nome dela, ou a palavra "self", ou a palavra "static", e usamos os "::"
                                   //para acessar os atributos ou métodos estáticos...
```

```
    public function getSabor():string {
        return $this->sabor;
    }
```

```
    public function getFormato():string {
        return $this->formato;
    }
```

```
    private static function acrescentaBolo(){ //Perceba que a função também é estática, só pode ser acessada pela
        return self::$contagem_de_bolos++;    //própria classe, e aqui usamos "self" para referenciar a classe...
    }
```

```
    public function verQuantidade(){
        return static::$contagem_de_bolos; //Aqui já usamos "static" para referenciar a classe...
    }
```

```
}
```

```
//Vamos criar 3 bolos e ver se ele conta...
```

```
$chocolate = new Forma('chocolate', 'quadrado');
```

```
$morango = new Forma('chocolate', 'quadrado');  
$leite_condensado = new Forma('leite condensado', 'quadrado');  
  
echo $leite_condensado->verQuantidade(); //Usando o método público que nos permite apenas ver a quantidade de bolos feita.
```

?>

[Running] php "c:\Users\Almoxarifado\Documents\php\arquivos\_das\_aulas\32-atributos\_e\_metodos\_estaticos.php"

3

[Done] exited with code=0 in 0.099 seconds