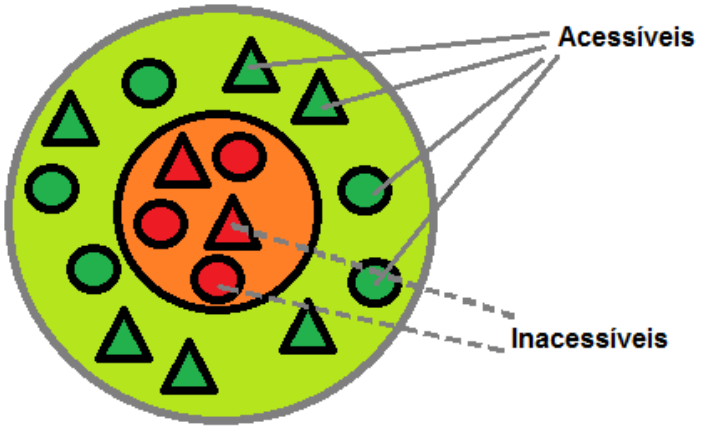
INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

Orientação a Objetos

Prof. Daniel Di Domenico

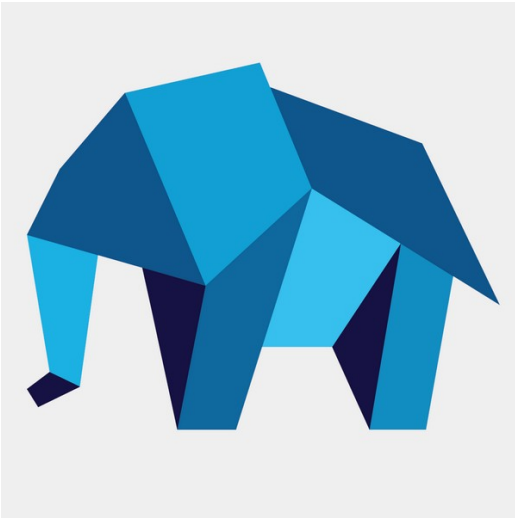
Encapsulamento e controle de acesso

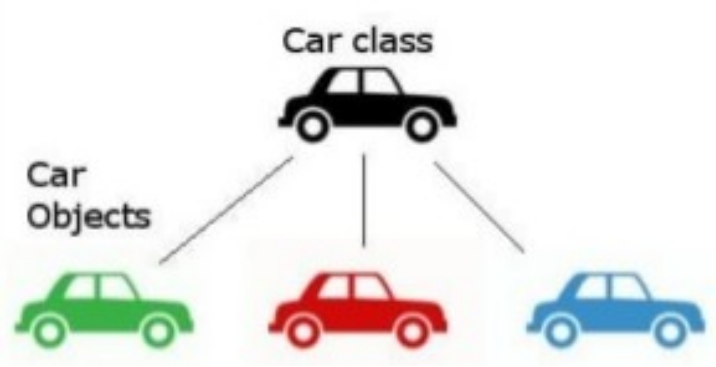


O que já sabemos?

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Linguagem PHP
  - Variáveis, tipos, operadores
  - Comandos de fluxo:
    - IF/ELSE, SWITCH, WHILE, DO-WHILE, FOR
  - Leitura de dados e vetores
- Orientação a objetos
  - Abstração
  - Classes e Objetos
  - Atributos e Métodos





Objetivos da aula

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- **Conhecer o conceito de encapsulamento:**
  - Objetivos e vantagens
- **Conhecer os atributos de controle de acesso**
  - Público
  - Privado
  - Protegido

Encapsulamento

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- É um **conceito/princípio** para implementar um código de um programa
  - Pilar da programação orientada a objetos
- Para que utilizamos o **encapsulamento**?
  - **Organizar os dados** relacionados em um domínio único
    - Domínio único = objeto
  - **Restringir o acesso externo** a estes dados
    - Não há necessidade de um usuário externo acessar dados que são pertinentes somente ao domínio/objeto
    - Aumenta-se a segurança dos dados do objeto
    - A restrição de acesso é feita utilizando atributos de controle de acesso

Encapsulamento

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- **Exemplo:** uso de um liquidificador
  - Uma pessoa vai utilizar um liquidificador
  - Ela não precisa saber detalhes do funcionamento interno do aparelho
    - A pessoa precisa configurar o motor do aparelho?
    - Se o motor foi trocado, faz diferença?
  - Basta conhecer a sua **interface** (botões do liquidificador)
- Com o **encapsulamento**, ocultam-se no objeto os dados pertinentes ao domínio do mesmo:
  - Assim, o **acesso externo** a estes dados é concedido **de uma forma controlada**

Encapsulamento

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Como implementar o encapsulamento?
  - Regras gerais:
    - Os atributos da classe devem ser acessados apenas no código/escopo da própria classe
    - O acesso externo aos atributos deve ser concedido através de métodos
      - Esses métodos são chamados de GETs e SETs
  - Para controlar o acesso, deve-se utilizar os **atributos de controle**
    - Privado, público e protegido

Atributos de controle de acesso

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Também chamados de modificadores de acesso
  - **private** (privado):
    - símbolo: **-**
    - permite acesso a um atributo/método apenas dentro do escopo da classe
  - **public** (público):
    - símbolo: **+**
    - permite acesso a um atributo/método dentro e fora do escopo da classe
  - **protected** (protegido):
    - símbolo: #
    - permite acesso a um atributo/método apenas dentro do escopo da classe ou no escopo das classes filhas (herança)

Como utilizar o encapsulamento

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- **Regras** para aplicar o encapsulamento:
  - **Atributos:**
    - Devem sempre ser **privados**
      - Acesso concedido pelos métodos GET, SET ou construtor
  - **Métodos:**
    - Podem ser **privados** ou **públicos**

Encapsulamento em PHP

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Classe com encapsulamento em PHP

```php
class Cachorro {

    private string $raca;
    private float $peso;

    public function latir() {
        echo "latindo...\n";
    }

    //GET e SET atributo raca
    public function getRaca(): string {
        return $this->raca;
    }

    public function setRaca(
                string $raca): self {
        $this->raca = $raca;
        return $this;
    } //Continua.....

    //GET e SET atributo peso
    public function getPeso(): float {
        return $this->peso;
    }

    public function setPeso(
            float $peso): self {
        $this->peso = $peso;
        return $this;
    }

} //Fim classe Cachorro
```

Atributos privados

Método público

return: retorna um valor ao chamar o método

Encapsulamento em PHP

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Instanciar objetos com encapsulamento:

```php
//Programa principal
$cao = new Cachorro();

$cao->setRaca("Beagle");
$cao->setPeso(15.6);

$cao->latir();

echo $cao->getRaca() . "\n";
echo $cao->getPeso() . "\n";
```

Cria/instancia o objeto: operador **new**

Seta os atributos pelos métodos SET

Chamada do método

Acessa o valor dos atributos usando os métodos GETs

Exercícios

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- **1-** Faça um programa que crie a classe Pessoa com:
  - Atributos: nome, endereço, cidade, UF e altura (todos privados)
  - Métodos: getApresentacao -> deve retornar uma String no formato:
    - “Olá mundo, sou <nome>, resido no endereço <endereco>, <cidade>-<UF> e possuo uma altura de <altura>!”;
  - Crie 2 objetos a partir da classe Pessoa, sete seus atributos, chame o método getApresentacao e imprima o seu retorno.
- **2-** Faça um programa que possua uma classe Aluno com os atributos nome, nota 1 e nota 2, além de um método para calcular e retornar a média. Após isso, crie e leia os atributos para 3 objetos Alunos, imprimindo a média de cada um deles.
- **3-** Faça um programa que crie uma classe Livro com os atributos título, autor, gênero e número de páginas. Depois, crie 3 objetos do tipo livro e leia todos os seus atributos. Após isso, imprima os dados do livro com maior número de páginas.