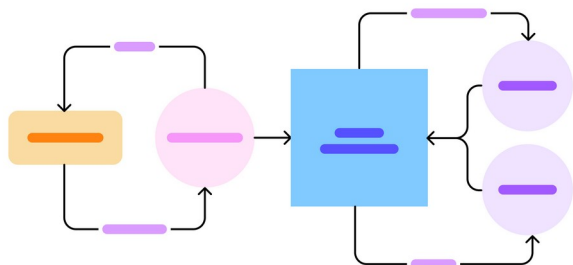


# Linguagem PHP

Prof. Daniel Di Domenico

## Comandos de fluxo



# PHP: comandos de fluxo

- A linguagem PHP possui **comandos** para controlar o fluxo de execução do código, sendo:
  - Comando **IF** (se) para condições
    - Comando **ELSE** (senão) para condições compostas
  - Comando **SWITCH** (escolha) para condições com casos fixos com uma única entrada
  - Comando **WHILE** (enquanto) para repetições controladas por condições
  - Comando **FOR** (para) para repetições considerando um intervalo numérico pré-definido

# IF e ELSE (se e senão)

- Se a **condição** for verdadeira, o bloco é executado
  - ELSE é o caso contrário, sendo executado se as demais condições testadas nos IFs forem falsas

//IF simples

```
$n1 = 1;
```

```
$n2 = 2;
```

```
if($n1 == 1 && $n2 >= 2) {  
    echo "n1 = 1 e n2 >= 2";  
}
```

Condição

Em PHP, as **chaves** para abrir e fechar o bloco são opcionais caso exista apenas uma linha a ser executada

//IF composto

```
$n1 = 1;
```

```
if($n1 == 1) {  
    echo "n1 = 1";  
} else if($n1 == 2) {  
    echo "n1 = 2";  
} else {  
    echo "n1 != 1 e n1 != 2";  
}
```

Pode utilizar  
**else if** ou  
**elseif**

# PHP: operadores

- Operadores relacionais:
  - == (igual)
  - === (idêntico – igual e de mesmo tipo)
  - !=, <> (diferente)
  - !== (não identico)
  - > (maior)
  - < (menor)
  - >= (maior ou igual)
  - <= (menor ou igual)
- Operadores lógicos
  - **and**, && (E)
  - **or**, || (OU)
  - **xor** (OU EXCLUSIVO)
  - **!** (NEGAÇÃO - not)

# SWITCH (escolha)

- Se a **variável** for igual ao **caso**, são executados os comandos do **bloco**
  - DEFAULT é o caso contrário. O uso do BREAK é obrigatório para finalizar o fluxo do comando SWITCH

Variável de  
teste para os  
casos

```
$n1 = 1;  
switch($n1) {  
    case 1:  
        echo "n1 = 1";  
        break;  
    case 2:  
        echo "n1 = 2";  
        break;  
    default:  
        echo "n1 != 1 e n1 != 2";  
}
```

# WHILE (enquanto)

- **Repete** a execução de um bloco de comandos enquanto a condição for **verdadeira**

```
$n1 = 1;
```

*//while com condição avaliada no INÍCIO*

```
while($n1 <= 5) {  
    echo $n1 . "<br>";  
    $n1 = $n1 + 1;  
}
```

**Condição**

Executa **ao menos uma vez**,  
mesmo a condição sendo  
inicialmente FALSA

```
$n1 = 1;
```

*//while com condição avaliada no FINAL*

```
do {  
    echo $n1 . "<br>";  
    $n1 = $n1 + 1;  
} while($n1 <= 5);
```

**Condição**

Saída na tela:

1  
2  
3  
4  
5

# FOR (para)

- **Repete** a execução de um bloco de comandos enquanto a condição for **verdadeira**
  - Após cada iteração, ocorre a execução de uma expressão de incremento

**2- Condição de parada da repetição**

$n1++$  equivale a  
 $n1 = n1 + 1$

```
for( $n1=1; $n1<=5; $n1++ ) {  
    echo $n1 . "<br>";  
}
```

**1- Declaração da variável de controle da repetição**

Saída na tela:

1  
2  
3  
4  
5

**3- Incremento da variável de controle**