

Introdução ao php

O php é uma linguagem de script usada para o desenvolvimento da web e é livre e gratuita sendo executada pelo terminal.

O nome originalmente significava Personal Home Page, mas atualmente é conhecido como Hypertext Preprocessor.

Case-sensitive: utiliza o padrão CamelCase que diferencia letras maiúsculas e minúsculas.

Deve-se usar a extensão **.php** e iniciar um bloco com **<?php**.

As variáveis são declaradas pelo caractere **\$** e para atribuir um valor para a variável se usa o **=**.

Ao término de cada instrução se utiliza o **;**.

Utilize as **{ }** para o início e fim de um bloco. Caso exista apenas uma linha para ser executada as **{ }** são opcionais.

Nas strings utiliza o **" "** ou **' '** e podem ser concatenadas pelo uso do **.** (ponto).

Comandos

IF (se)

■ Para declarar condições.

ELSE (senão)

■ Para condições contrárias.

SWITCH (escolha)

■ Condição com casos fixos de uma única variável

WHILE (enquanto)

■ Repetições controladas por condições.

FOR (para)

■ Repetições considerando um número específico/predefinido.

Operadores relacionais

== igual

=== idêntico /igual e mesmo tipo/

!= diferente

!== não idêntico

> maior

< menor

>= maior ou igual

<= menor ou igual

Operadores lógicos

E : and, **&&**

Ou: or, **||**

Ou exclusivo: xor

Not/negação: **!**

EXEMPLOS:

```
if.php
1  <?php
2  $numero = 10;
3
4  if ($numero > 5) {
5      echo "O número é maior que 6.\n";
6  } elseif ($numero == 6) {
7      echo "O número é igual a 6.\n";
8  } else {
9      echo "O número é menor que 6.\n";
10 }
```

if and else/elseif

```
1  <?php
2  $nota = 'B';
3
4  switch ($nota) {
5      case 'A':
6          echo "Excelente!\n";
7          break;
8      case 'B':
9          echo "Bom!\n";
10         break;
11     case 'C':
12         echo "Regular!\n";
13         break;
14     case 'D':
15         echo "Precisa comparecer nos atendimentos!\n";
16         break;
17     default:
18         echo "Nota inválida!\n";
19 }
```

switch

No switch o default funciona como um “else”

```
while.php
1  <?php
2  $gatinhos = ["Miau", "Ronron", "Peludinho", "Bigodes", "Soneca"];
3  $posicao = 0;
4
5  while ($posicao < count($gatinhos)) {
6      echo "Gatinho: " . $gatinhos[$posicao] . "\n";
7      $posicao++;
8  }
```

while

```

for.php
1  <?php
2  $gatinhos = ["Soneca", "Miau", "Ronron"];
3
4  for ($i = 0; $i < count($gatinhos); $i++) {
5      echo "Gatinho: " . $gatinhos[$i] . "\n";
6  }

```

for

Leitura de dados

A função readline retorna para o que o usuário digita no terminal, serve para interagir com o usuário e é uma string.

```

readline.php
1  <?php
2
3  echo "Digite o nome do seu gatinho: ";
4  $nomeDoGatinho = readline();
5
6  echo "O nome do seu gatinho é $nomeDoGatinho.\n";
7

```

function readline

As funções servem para organizar o código e é útil na realização do código . Para chamar uma função basta digitar ela e funções podem ter um retorno.

```

function.php > ...
1  <?php
   1 reference
2  function definirNomeGato($nome) {
3      return $nome;
4  }
   1 reference
5  function exibirNomeGato($nomeDoGato) {
6      if ($nomeDoGato) {
7          echo "O nome do seu gato é $nomeDoGato.\n";
8      } else {
9          echo "O nome do gato ainda não foi definido.\n";
10     }
11 }
12 //programa principal
13 $nomeDoGato = definirNomeGato("Ronron");
14
15 exibirNomeGato($nomeDoGato);

```

Saída: O nome do seu gato é Ronron.

Array

Arrays em PHP são como caixas grandes onde você pode guardar várias coisas juntas. Elas são muito flexíveis e podem conter diferentes tipos de dados, como números e palavras. Isso ajuda a organizar e usar essas informações de forma eficiente no seu programa.

Arrays Indexados: Usam índices numéricos e são úteis para listas ordenadas.

Arrays Associativos: Usam chaves nomeadas e são úteis para dados com rótulos específicos.

Arrays Multidimensionais: Contêm arrays dentro de arrays, permitindo estruturas de dados mais complexas.

```
array.php
1  <?php
2  $gatinhos = ["Soneca", "Griffith", "Flynn"];
3
4  foreach ($gatinhos as $gatinho) {
5      echo "O nome do gatinho é $gatinho.\n";
6  }
7
8  $gatinhos[] = "Nietz";
9
10 echo "\nAdicionando um novo gatinho:\n";
11 foreach ($gatinhos as $gatinho) {
12     echo "O nome do gatinho é $gatinho.\n";
13 }
```

array associativo