

# Funções internas

- ▶ O PHP oferece uma série de funções internas, embutidas em seu código, para tratamento de números, *strings*, *arrays*, etc...
- ▶ A seguir, estudaremos algumas delas
- ▶ É essencial a leitura do manual do PHP para aprender a lidar com as funções e, também, para procurar por funções não mencionadas aqui
- ▶ A lista completa das funções do PHP pode ser obtida em:  
[http://php.net/manual/pt\\_BR/funcref.php](http://php.net/manual/pt_BR/funcref.php)

# Funções para variáveis

- isset

Retorna `TRUE` se a variável foi iniciada (mesmo que seja uma *string* vazia ou o valor zero) e `FALSE` caso contrário

- ▶ Exemplo

```
<?php
    $var = '';

    // Será interpretado como TRUE imprimindo o texto
    if (isset($var)) {
        echo "Essa variável existe.";
    }
?>
```

# Funções para variáveis (cont.)

- empty

Retorna TRUE se a variável possui conteúdo vazio (uma *string* vazia, o valor zero ou NULL) e FALSE caso contrário

- ▶ Exemplo

```
<?php
    $var = 0;

    //Retornará TRUE, porque $var é um valor vazio
    if (empty($var)) {
        echo 'o conteúdo de $var é: 0, vazio ou NULL';
    }

    //Retornará TRUE porque $var está 'setado'
    if (isset($var)) {
        echo '$var está "setado", apesar de vazio';
    }
?>
```

# Funções para variáveis (cont.)

- unset

Destrói uma variável, ou seja, libera a memória ocupada por ela, fazendo com que ela deixe de existir

- ▶ Exemplo

```
<?php
    $var = 'eu existo';

    // Será interpretado como TRUE imprimindo o texto
    if (isset($var)) {
        echo "Essa variável existe.";
    }

    unset($var);

    // Será FALSE, pois a variável $var foi destruída
    if (isset($var)) {
        echo "Será que ela continua existindo?";
    }
?>
```

# Funções para datas

## ○ date

Retorna a data e/ou hora: atual do servidor ou do *timestamp* informado

Possui diversos argumentos, que podem ser utilizados de acordo com o formato da data e/ou hora desejado

### ► Exemplo - Data no formato 31/12/2015

```
<?php
    $data = date("d/m/Y");
    echo "Hoje é: $data";
?>
```

### ► Exemplo - Data no formato 31/12/2015 22:05:00

```
<?php
    $data = date("d/m/Y H:i:s");
    echo "Hoje é: $data";
?>
```

Formato	Descrição
d	Dia do mês com 2 dígitos
D	Representação textual do dia
m	Representação numérica do mês
M	Representação textual do mês
Y	Representação do ano com 2 dígitos
Y	Representação do ano com 4 dígitos
l ( 'L' minúsculo)	Descrição do dia da semana
h	Formato em 12 horas
H	Formato em 24 horas
i	Minutos
s	Segundos

# Funções para datas (*continuação*)

- time

Retorna o *timestamp* Unix atual, ou seja, a quantidade de segundos desde 1º de Janeiro de 1970

Toma por base o relógio do servidor onde o PHP é executado

- ▶ Exemplo

```
<?php
    $timestamp = time();
    echo "Fazem $timestamp segundos desde 01/01/1970";
?>
```

# Funções para datas (*continuação*)

- mktime

Retorna o *timestamp* Unix para a data especificada

- ▶ **Formato**

`mktime(hora, minuto, segundo, mês, dia, ano)`

- ▶ **Exemplo**

```
<?php
//Gera o timestamp da data 09/02/1993 as 20:03:59
$timestamp = mktime(20, 3, 58, 2, 9, 1993);

// Exibe: 09/02 ~ 20:03
echo date('d/m ~ H:i', $timestamp);
?>
```

# Funções para datas (*continuação*)

- strtotime

Analisa a descrição em inglês e retorna o *timestamp* Unix

- ▶ Exemplo

```
<?php
//timestamp atual
echo strtotime("now");

//timestamp de 10 de setembro de 2000
echo strtotime("10 September 2000");

//timestamp de amanhã (daqui a 24h)
echo strtotime("+1 day");

//timestamp para daqui a uma semana, 2 dias, 4 horas e 2 segundos
echo strtotime("+1 week 2 days 4 hours 2 seconds");

//timestamp para a próxima quinta-feira
echo strtotime("next Thursday");

//timestamp da última segunda-feira
echo strtotime("last Monday");
?>
```



# Funções para datas (*continuação*)

- checkdate

Retorna TRUE caso a data informada seja válida ou FALSE, caso contrário

- ▶ **Formato**

`checkdate(mês, dia, ano)`

- ▶ **Exemplo**

```
<?php
    $dia = 29;
    $mes = 2;
    $ano = 2015;

    if(checkdate($mes, $dia, $ano)){
        echo "Data válida";
    }else{
        echo "Data inválida";
    }
?>
```

# Exercício 22

22) Incremente o arquivo `biblioteca.inc.php`, criado no exercício 21, incluindo:

- a) Uma função para verificar se uma data é válida, no formato:

```
verifica_data(dia, mês, ano)
```

Dica: utilize a função interna `checkdate`

- b) Uma função para receber um timestamp e retornar sua data equivalente por extenso, no formato:

```
data_extenso_timestamp(timestamp)
```

**Exemplo:**

```
data_extenso_timestamp(1451516400);  
//imprime: 31 de Dezembro de 2015
```

- c) Uma função para receber uma data e retorná-la escrita por extenso, no formato:

```
data_extenso(dia, mês, ano)
```

**Exemplo:**

```
data_extenso(31, 12, 2015);  
//imprime: 31 de Dezembro de 2015
```

Dica: utilize `mktime` e a função criada anteriormente (letra B)

## Importante para letra C

Verificar primeiramente se a data informada é válida.

Dica: utilize a função criada anteriormente (letra A).

Inclua o arquivo `biblioteca.inc.php` no arquivo `ex22.php` e faça pelo menos três chamadas das funções recentemente criadas