

[PHP] Regex com PHP

Transcrição

No PHP devemos definir uma regex dentro do caracteres `~regex aqui~`, por exemplo:

```
$regexp = '~(\d\d)(\w)~';
```

Para executar e encontrar o primeiro *match*, usamos a função `preg_match`:

```
$alvo = '12a34b56c';  
$achou = preg_match($regexp, $alvo, $match);
```

A função devolve um booleano (na verdade 0 ou 1) para indicar o sucesso ou não. Os resultados estão guardados na variável `$match`. Assim podemos imprimir o primeiro *match*:

```
> echo $match[0];  
12a  
> echo $match[1];  
12  
> echo $match[2];  
a
```

Caso desejemos receber os índices, devemos passar mais um parâmetro na função `preg_match`:

```
$achou = preg_match($regexp, $alvo, $match, PREG_OFFSET_CAPTURE);
```

Imprimindo:

```
print_r($match);  
Array  
(  
  [0] => Array  
  (  
    [0] => 12a  
    [1] => 0  
  )  
  [1] => Array  
  (  
    [0] => 12
```

```
[1] => 0

)

[2] => Array

(

[0] => a

[1] => 2

)

)
```

Para, por exemplo, acessar o índice do segundo grupo usamos:

```
> echo $match[2][1]; //inicio do segundo grupo a
2
```

Igualmente também tem um método que busca todos os resultados da regex dentro do alvo:

```
$achou = preg_match_all($regexp, $alvo, $matches);
```

A função `preg_match_all` cria um array bidimensional. Na primeira posição tem todos os resultados do *match* inteiro:

```
> echo $matches[0][0];
12a
> echo $matches[0][1];
34b
> echo $matches[0][2];
56c
```

A segunda posição possui todos os *matches* do primeiro grupo, e a terceira do segundo grupo:

```
> echo $matches[1][0];
12
> echo $matches[2][0];
a
```

Na função `preg_match_all`, também existe o parâmetro `PREG_OFFSET_CAPTURE`, para receber os índices. Nesse caso será criado um array de 3 dimensões.

