

Perl

{ substituições }

13.1 ENCONTRA E TROCA

Se você ainda não desistiu de estudar PERL depois do que viu na última aula, parabéns! Vamos continuar...

Vimos no último capítulo uma forma de encontrar (ou não) um substring (molde) em um string.

A linguagem PERL tem uma maneira de, encontrando um molde, substituí-lo por outro substring. É assim:

```
$x = "a+cdefde";  
$x =~ s/de/mn/;
```

Isso fará com que em **\$x** tenhamos: a+cmnfde.

Se quisermos manter o valor em **\$x** temos que primeiro igualá-lo a outra variável e fazer a operação com ela.

Essa operação de substituição só acontece na primeira ocorrência do molde. Para a substituição em todas as ocorrências temos que colocar um **g**e depois da última barra:

```
$x =~ s/de/mn/ge;
```

Essas trocas valem para todo tipo de encontro de moldes que vimos no último capítulo, inclusive aqueles que usam (.) e \1 para criar uma memória dos caracteres. Assim:

```
$x =~ s/d(.)f/x\1y/;
```

fará com que tenhamos: a+cxeyde.

13.2 TROCA DE UM CARACTERE SÓ

Nos casos em que queremos a troca de um caractere só, podemos usar:

```
$x =~ tr/f/x/;
```

Este uso é requerido nos casos em que queremos trocar um caractere do tipo **+** por **branco**

```
$x =~ tr/+/ /;
```

Nestes casos o **s//** não funciona.

13.3 DECIMAIS BRASILEIROS

Uma utilização interessante dessa técnica de substituição é para trocar o separador brasileiro **vírgula** em entradas numéricas com decimais, para o separador americano **ponto** que é o único válido em operações matemáticas no PERL.

Lembre-se que, quando um usuário digita um número (preço, por exemplo) num control de um form, a sua entrada é um string. Temos, por exemplo então:

```
$z = "29,40";  
$z =~ s/,/./;
```

para em \$z termos o string 29.40 que será transformado automaticamente em número se executarmos uma operação matemática com ele (característica do PERL que já foi comentada).

Para termos a operação inversa, isto é, tendo um número separado por ponto e o passarmos para um string com o separador vírgula a coisa é mais complicada e vamos ver no próximo capítulo.

13.3 SUBSTITUIÇÃO POR POSIÇÃO

Diferentemente do "encontra e troca" que vimos aí em cima, podemos provocar a substituição de um substring dentro de um string indicando o início e tamanho do substring, usando a função **substr**.

Veja:

```
$v = "abcdef";  
substr($v,2,3)="xxxx";
```

O primeiro número indica o dígito inicial do substring (a origem é ZERO!).

O segundo número indica a quantidade de dígitos.

Como aparece no exemplo, podemos substituir um substring por outro de tamanho diferente. No caso teremos em \$v: abxxxxf.

Se usarmos "=" o substring é substituído por nada (não é branco, no caso), donde será extraído.