{cgi-lib}

3.1 LEI DE LAVOISIER

Na informática dizem:"nada se cria, tudo se copia..."

Estou falando isto porque vou apresentar vocês, nesse capítulo a um arquivo mágico. E ele é mágico justamente porque vocês (que estão começando a estudar PERL agora) provavelmente não entenderão muita coisa do que está dentro dele.

Mas tenham fé! E essa fé vai ajudar a gente a ir fazendo os exercícios de programação e acredito (aí a fé é minha) que quando chegarmos no fim deste curso vocês vão entender tudo o que está codificado neste arquivo. Então o mistério acaba.(Na última aula ensinamos como trabalhar com PERL sem este arquivo.)

3.2 O GURU E SUA MÁGICA

Um guru da programação de CGI com PERL é sem dúvida Steven E. Brenner que escreveu e tornou disponível pela Internet um programa chamado CGI-LIB.PL que auxilia bastante a interligação entre os controls que apresentamos no capítulo anterior e os programas colocados no servidor.

Este programa importantíssimo você deve copiar para um arquivo ASCII.

Como você fez na primeira aula, abra um arquivo usando um editor ASCII (o NotePad, por exemplo) e depois dê um COPY-AND_PASTE sobre todas as linhas do arquivo que você pode ver aqui. Salve seu arquivo ASCII com o nome de MAGIC.TXT. Nós vamos nos referenciar a ele em nossos exercícios.Faça isso agora! E volte com BACK.

Continuando:Esse arquivo deverá, sempre através de copy-and-paste ser colocado no mesmo arquivo onde você vai escrever seu programa PERL. Fazendo isso você vai poder usar várias funcões/procedures que ele contém, dentro do seu programa.Você vai colocá-lo DEPOIS do fim de seu programa.

Você não precisa transportar esse arquivo MAGIC,TXT para seu servidor. Como dissemos ele irá junto com cada programa seu; o que, apesar de aumentar o tamanho do arquivo a ser transportado via FTP vai dar uma melhora na performance de suas aplicações, pois não vai ser preciso um link com outro arquivo. E isso de performance é importante em CGI e PERL que são famosos pela baixa performance. Qualquer segundo ganho já é bom.

REPETINDO NA ÚLTIMA AULA DO CURSO ENSINAMOS COMO TRABALHAR COM CGI E PERL SEM USAR ESTE ARQUIVO. MAS VOCÊ PRECISA APRENDER MUITA COISA AINDA..

3.3 USANDO

Vamos então fazer uma aplicação simples já usando umas funções do Magic.txt.

Essa aplicação tem dois arquivos. O primeiro é um arquivo só com HTML. Você deve colocá-lo no diretório do seu servidor, onde coloca sempre os seus arquivos HTML.Você deve usar copy-and-paste para transportá-lo para um arquivo ASCII. Normalmente esses arquivos vão para o servidor via FTP em BINÁRIO.

O segundo arquivo é um programa em PERL.Vamos mostrar um arquivo só com o programa básico,digamos assim. Você deve usar copy-and-paste também e colocá-lo num outro arquivo ASCII

DEPOIS DISSO, COPIE O MAGIC.TXT E COLOQUE DEPOIS DO FIM DO PROGRAMA. Repetimos(isso vai acontecer com quase todos os programas/exercícios do curso):DEPOIS DE COPIAR O PROGRAMA-BASE, ACRESCENTE O MAGIC.TXT.

Então salve esse arquivo (composto dos dois outros) com o nome sugerido.

Esse arquivo de programa deve ser transportado para o servidor para aquele diretórip CGI-BIN ou CGI-LOCAL (consulte seu provedor) usando ASCII e não BINÁRIO.

Repetimos: USANDO ASCII E NÃO BINÁRIO (BINÁRIO é como você faz normalmente com arquivos HTML).

Repetimos (ô saco!): USE ASCII PARA TRANSPORTAR O ARQUIVO DE PROGRAMA (NÃO O DE HTML) PELO FTP!

Todo mundo vive se esquecendo disso e perde horas sem entender porque o programa não funciona.(Quer que a gente repita de novo? Não? Tudo bem...)

3.4 FINALMENTE OS PROGRAMAS.

Primeiro o arquivo HTML. Como em todos os arquivos desse livro, faça uma cópia já e coloque na tela numa outra janela junto com o browser onde você está lendo isso, para acompanhar melhor. O arquivo está aqui.

Vamos chamá-lo de prog2.html (Depois, o arquivo de programa vamos chamar de prog2.pl).E transportá-lo daqui a pouco para o servidor (para o diretório de HTML).

Vamos analisar esse arquivo.

A primeira linha que você não entende é a que começa com &ltFORM...

Essa linha e a que contém </FORM> têm que ser colocadas quando usamos os controls vistos no último capítulo.

O transporte daquelas DUPLAS com variáveis e seus valores para o programa PERL no servidor pode ser feita por um de dois modos: GET ou POST.

O método GET é um pouco limitado na quantidade de dados que pode transportar, dependendo do servidor HTTP de seu provedor. Apesar disso, por ser o mais antigo, é o default. Se não colocamos a "tag":&ltMETHOD="POST"> o método de transporte é GET.

Depois temos que colocar ACTION com o endereço do programa que vai receber os dados.Onde colocamos "seu-domínio" você deve colocar o domínio de seu provedor e sua designação (por exemplo :www.brasil.com).

Depois você coloca o nome da pasta para programas:cgi-bin,cgi-local ou o nome que seu provedor fornecer.

Na linha seguinte entramos com um texto e um EditBox (assunto já visto). Importante notar o NAME da variável que é:nome (realmente pouco original).

Depois temos a linha com:

&ltINPUT TYPE="SUBMIT" VALUE="Aperte">

Ela define um Botão que, quando clicado vai fazer com que os valores entrados sejam transportados para o servidor e o programa PERL disparado.

O que colocamos em VALUE é o que vai aparecer no Botão.

3.5 O PROGRAMA PERL

O programa PERL deve ser copiado para um arquivo ASCII a partir daqui

Depois você deve copiar o Magic.txt e colocar todas as suas linhas depois do programa.

No Magic.txt temos várias procedures/funçõe*s*, que vamos usar muito para facilitar nossa vida.

Uma se chama PrintHeader e retorna as duas linhas necessárias (como vimos na aula 1) para começar uma página on-the-fly:

"Content-type:text/html\n\n"

Dissemos DUAS porque temos A MISTERIOSA LINHA EM BRANCO, lembram-se?

Depois temos a procedure HtmlTop que aceita um título como parâmetro e vai retornar:

&lthtml>&lthead>

&lttitle>*título*</title>

</head>&ltbody>

Já a procedure HtmlBot encerra sempre uma página on-the-fly retornando:

</body></html>

Repare que se usa o elemento & antes do nome da procedure para chamá-la.

Repare também que essas procedures retornam aqueles strings entre aspas (digamos assim) mas precisamos printá-los e, para isso usamos: print. As outras três linhas do programa são mais dificeis de entender. Vamos estudar esses assuntos todos relativos a linguagem PERL mais em detalhes depois. Estamos montando esses primeiros exercícios bem práticos (mesmo sem vocês entendê-los bem), para motivá-los a seguir o resto do curso.

Em PERL existe uma coisa muito interessante chamada:Array Associativo.

Um Array Associativo contém duas colunas e várias linhas. Assim, em cada linha temos uma DUPLA!

Vocês se lembram que os inputs feitos nos controls são transportados em DUPLAS (o NAME e o valor entrado)? Todos os inputs entrados numa página vão para um Array Associativo que chamaremos de: input (muito pouco original).Esse nome é colocado como parâmetro da procedure ReadParse (como você pode ver no programa). Tem um asterisco misterioso mas nescessário...

Para acessar o valor entrado no EditBox com o NAME="nome"que está no Array Associativo input usamos:

$input{"nome"}

Jogamos esse valor numa variável que criamos chamada:$vNome.

Em PERL não precisamos especificar tipo de variável nem nada (como no Visual Basic e no FoxPro antigamente).

Finalmente temos a linha com:

```
print qq(Alô, $vNome)
```

em que mandamos printar Alô, seguido do que está na variável.

A função qq( ) seve para colocar aspas no string, o que é necessário no uso do print (como já vimos na aula 1).

Entendido (mais ou menos) o programa, você deve colocá-lo então no diretório CGI-BIN ou CGI-LOCAL do servidor transportando-o como ASCII.

3.6 EXECUTANDO

Recaptulando: temos dois arquivos.Um com tags HTML que deverá ser chamado como qualquer página HTML. Nessa página temos um control EditBox em que você vai escrever seu nome e um control Botão que deve ser clicado.

Nessa hora vai ser transportado para o servidor uma DUPLA com

nome=*seu-nome*

e vai ser disparado o programa PERL que vai pegar esse valor vindo na dupla e montar on-the-fly uma página que tem o texto:

Alô,*seu-nome*.

Então resta só executar. Você pode usar nosso programa que é visto aqui embaixo:

Entre com seu nome:

Aperte