

LINGUAGEM SHELL SCRIPT

Jonis Nogueira dos Santos, José Adelar Souza da Silva

Faculdades de Informática de Taquara – Faculdades de Taquara (FACCAT)

Caixa Postal 84 – 95600-000 – Taquara – RS - Brasil

jonis@faccat.br, adelar@faccat.br

Resumo Este artigo mostrará um breve histórico sobre a Linguagem SHELL SCRIPT, juntamente com suas características, aplicabilidade e paradigma.

1.Histórico

A linguagem Shell Script é a linha de comando Linux (Unix), comandos armazenados em um arquivo texto que são executados sequencialmente. Esta linguagem interpreta a linha de comandos digitados pelo usuário no terminal e chama os programas desejados. Além de executar comandos do sistema, o Shell também tem seus próprios comandos, e também possui variáveis e funções. Existem diversos tipos de shell: bash, csh, ksh, ash, etc. O mais utilizado atualmente é o bash (GNU Bourne-Again SHell). A primeira linha de todo Shell script deve começar com algo do tipo: `#!/bin/bash`, a qual indica com qual Shell deverá ser executado o script.

2.Paradigma

Shell Script é uma linguagem Interpretada e Imperativo.

3.Características

Tendo as características de uma linguagem de programação, o Shell é uma ferramenta muito poderosa para desenvolver Scripts e programinhas rápidos, para automatizar tarefas do dia-a-dia. Podemos escrever programas elaborados em Shell Script, substituindo aplicativos que, em uma linguagem mais elaborada demorariam muito mais tempo para ser escritos . Seus scripts podem possuir interfaces simples de modo texto , ou possuir interface gráfica elaborada escrita usando o kommander e funções do kdialog. Para quem está acostumado com o MSDOS, o Shell é como um Batch (dos arquivos .BAT).Além de servir para fazer pequenos scripts o Shell é completo o suficiente para fazer grandes programas.

Entre as características do Shell é que nele pode ser feito Interfaces Interativas com o usuário, programas de cálculos, CGI, instaladores de software, manipulação de banco de dados, rotinas de backup etc . Uma das vantagens dos Shell Scripts é que eles não precisam ser compilados, para dar a um arquivo uma definição de "Shell script" temos que incluir uma linha no começo do arquivo (`#!/bin/bash`) e torná-lo "executável", utilizando o comando `chmod`.

Alguns comandos do Shell : IF, FOR e WHILE.

Exemplos de alguns caracteres usados:

Testes em variáveis

-eq se é igual
!= se é diferente
-gt se é maior
-lt se é menor

Testes em arquivos

-o Ou lógico
-d Se for um diretório
-f Se é um arquivo normal
-r Se o arquivo tem permissão de leitura
-w Se o arquivo tem permissão de escrita
-x Se o arquivo pode ser executado

Exemplos de alguns comandos utilizados:

echo Imprime texto na tela
read Captura dados do usuário e coloca numa variável
exit Finaliza o script
sleep Dá uma pause de segundos no script
clear Limpa a tela
stty Configura o terminal temporariamente
tput Altera o modo de exibição
if Controle de fluxo que testa uma ou mais expressões
case Controle de fluxo que testa várias expressões ao mesmo tempo
for Controle de fluxo que testa uma ou mais expressões
while Controle de fluxo que testa uma ou mais expressões

4.Aplicabilidade

Shell, é o programa utilizado para “pegar” os comandos do usuário, interpretá-los e executar a tarefa solicitada, logo após o login . Esta ferramenta é muito poderosa para desenvolver Scripts e pequenos programas, mas também é possível fazer grandes programas utilizando Shell .

5.Exemplo de Programa em Shell Script

```
#!/bin/bash
clear
echo
echo "Boa noite meu amigo" $USER.
echo
echo -e '\033[00;37m'
echo "escolha uma das opcoes abaixo"
echo
echo "[1] Soma dois valores"
echo "[2] Multiplica dois valores"
echo "[3] Sair"
echo
echo -n "Digite a opcao: "
read opcao
if test "$opcao" == 1
then
    echo -e '\033[30;34m'
    echo "Digite um valor para soma:"
    read valsoma1
    echo
    echo "Digite um outro valor para soma:"
    read valsoma2
    echo
    echo "O resultado da soma eh:"
    soma=$((valsoma1+valsoma2))
    echo $soma
    echo
    echo -e '\033[00;37m'
fi
if test "$opcao" == 2
then
    echo -e '\033[30;34m'
    echo "Digite um valor para multiplicacao:"
    read valmult1
    echo
    echo "Digite um outro valor para multiplicacao:"
    read valmult2
    echo
    echo "O resultado da multiplicacao eh:"
    multiplicacao=$((valmult1*valmult2))
    echo $multiplicacao
    echo
    echo -e '\033[00;37m'
```

```
fi
if test "$opcao" == 3
then
    clear
    exit
fi
```

O script acima, dá ao usuário a chance de escolher se ele quer efetuar uma soma ou uma multiplicação entre dois números, depois que o usuário informa qual opção escolheu, o script altera a cor da fonte para azul e no fim da execução a fonte volta a cor normal.

6.Referências

<http://www.aurelio.net/shell>

<http://vivaolinux.com.br>

<http://sh.underlinux.com.br/doc/programacaosh.htm>

http://www.devin.com.br/eitch/shell_script/