

Utilizando laço de repetição

Transcrição

Até a etapa anterior, o script era capaz de receber dois parâmetros passados pelo usuário. Para pegar o conteúdo desses parâmetros, usamos o símbolo `$`.

E se o usuário pudesse passar um **terceiro**, ou até mesmo um **quarto** parâmetro? Seria necessário repetir o código, e com certeza, essa não é a solução mais elegante.

Então, *para cada parâmetro* passado pelo usuário, *é executada a conversão*. Com isso, é necessário implementar uma **estrutura de repetição** falada no curso de [Lógica de Programação](https://cursos.alura.com.br/course/logica-programacao-javascript-html?preRequirementFrom=shellscripting) (<https://cursos.alura.com.br/course/logica-programacao-javascript-html?preRequirementFrom=shellscripting>).

"Para cada variável `imagem` dentro do conjunto de parâmetros passados pelo usuário, englobamos todos eles usando `$@`".

Com a afirmação acima, conseguimos criar o nosso **laço** e definir o que ele irá executar!

```
#!/bin/bash

CAMINHO_IMAGENS=~Downloads/imagens-livros

for imagem in $@
do
    convert $CAMINHO_IMAGENS/$imagem.jpg $CAMINHO_IMAGENS/$imagem.png
done
```

Como estamos criando um laço de repetição, não tem necessidade de ter duas linhas de comando realizando a mesma tarefa. Retiramos uma.

Queremos converter o conteúdo da variável `imagem` que vai receber cada parâmetro passado pelo nosso usuário, por isso utilizamos `$imagem` no lugar de `$1`.

Uma vez que esse laço `for` foi concluído, precisamos dizer que a tarefa dele foi finalizada e com o `done`, conseguimos esse resultado.

Para sair e salvar, utilizamos "Ctrl + x" e "y".

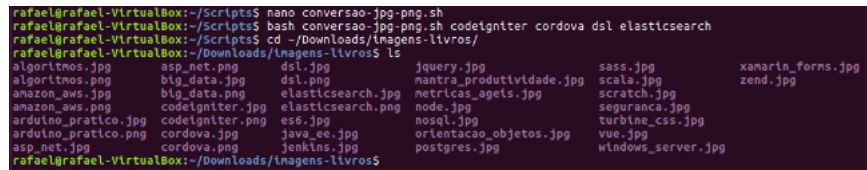
Vamos executar esse script passando 4 parâmetros:

```
$ bash conversao-jpg-png.sh codeigniter cordova dsl elasticsearch
```

Aparentemente, o script rodou sem nenhum problema. Vamos ver se os arquivos de conversão estão no diretório "imagens-livros".

```
$ cd ~/Downloads/imagens-livros/  
$ ls
```

Observe a imagem abaixo:



```
rafaelgrafael-VirtualBox:~/Scripts$ nano conversao-jpg-png.sh  
rafaelgrafael-VirtualBox:~/Scripts$ bash conversao-jpg-png.sh codeigniter cordova dsl elasticsearch  
rafaelgrafael-VirtualBox:~/Scripts$ cd ~/Downloads/imagens-livros/  
rafaelgrafael-VirtualBox:~/Downloads/imagens-livros$ ls  
algoritmos.jpg      asp_net.png      dsl.jpg           jquery.jpg        sass.jpg          xamarin_forms.jpg  
algoritmos.png      big_data.jpg     dsl.png          mantra_produtividade.jpg  scale.jpg         zend.jpg  
amazon_aws.jpg      big_data.png     elasticsearch.jpg  metricas_apels.jpg  scratch.jpg  
amazon_aws.png      codeigniter.jpg  elasticsearch.png  node.jpg          segurancs.jpg  
arduino_pratico.jpg  codeigniter.png  es6.jpg           nosql.jpg         turbine_css.jpg  
arduino_pratico.png  cordova.jpg      java_ee.jpg       orientacao_objetos.jpg  vue.jpg  
asp_net.jpg         cordova.png      jenkins.jpg       postgres.jpg       windows_server.jpg  
rafaelgrafael-VirtualBox:~/Downloads/imagens-livros$
```

Como podemos ver, o script foi executado com sucesso!

Entendemos que com o `$@` pegamos todos os parâmetros que o usuário passar, independente da quantidade.

Os diretores da *Multillidae*, nos passaram a missão de realizar a **conversão automática** de todo o conteúdo do diretório "imagens-livros".

Na próxima aula, veremos como podemos melhorar o script.