

Mãos à obra: Montando a função `converte_imagem`

Uma vez que criamos a função para varrer os diretórios, precisamos montar uma função que seja capaz de fazer as conversões das imagens. Abra o script que estamos trabalhando e logo acima da função `varrer_diretorio` crie uma função chamada de `converte_imagem`.

Essa função deverá como o próprio nome diz, converter as imagens passadas pela função `varrer_diretorio`, dessa forma, teremos que pegar esse parâmetro passado pela função `varrer_diretorio`. Para não ficarmos com vários `$1` espalhados no nosso código, vamos guardá-lo dentro de uma variável local chamada de `caminho_imagem`. Dentro da função `converte_imagem` coloque o seguinte trecho

```
local caminho_imagem=$1
```

Feito isso, deveremos remover a extensão `.jpg` desse arquivo de imagem para podermos fazer a conversão, vamos pedir ajuda do `awk` como fizemos nas etapas anteriores. Logo abaixo do trecho da atribuição da variável `caminho_imagem`, coloque o seguinte trecho na função `converte_imagem`

```
local imagem_sem_extensao=$(ls $caminho_imagem | awk -F. '{ print $1 }')
convert $imagem_sem_extensao.jpg $imagem_sem_extensao.png
```

Agora que montamos a função `converte_imagem`, devemos passar o parâmetro. Volte para a função `varrer_diretorio` e altere o comentário que havíamos feito *"Conversao jpg para png"* para o seguinte comando:

```
converte_imagem $caminho_arquivo
```

Com isso estamos passando o caminho completo do arquivo, que sabemos que é uma imagem porque está no bloco do `else`, para a função `converte_imagem`. Agora só falta nós chamarmos a função `varrer_diretorio` para iniciar o processo de varredura, faremos isso no próximo exercício.