



8 - Noção de Script

★★★★★ Ainda sem avaliações

Gerar um site estático para documentar o progresso no desenvolvimento de software é uma abordagem eficaz. Neste artigo, criaremos o Gerweb, uma ferramenta que permite criar e atualizar facilmente um site para acompanhar seu crescimento nessa área.


Derivado do projeto "Blogático" do Professor Silvio Rhatto, apresentado em aulas ministradas de "Boas Práticas de Desenvolvimento de Software" na plataforma da Universidade de Campinas -SP (MOOC-GGTE UNICAMP), apenas mudei seu nome para Gerweb prevendo eventuais mudanças em suas linhas de código.

Ele é um software que oferece uma solução prática para compartilhar suas atualizações em um site. Veremos como configurar o ambiente, definir parâmetros importantes e utilizar templates para obter um site minimamente estruturado .

Ao longo do artigo, você aprenderá a criar e atualizar seu site, ele permitirá , como veremos em artigos posteriores, compartilhar seu conhecimento de forma clara e organizada.



Vamos criar o arquivo do projeto com o comando touch:



```
leandro@leandro-VirtualBox: ~/projetos/viagem_ao_mundo_do_software
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ touch gerweb
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$
```

Após criar o arquivo, dentro dele devemos escrever uma linha especial iniciando com o shebang (`#!/`) "forquilha e o sinal de exclamação" para indicar ao interpretador shell qual interpretador deve ser chamado para interpretar o script. Nesse caso, usaremos o `#!/usr/bin/env bash` desta forma:



```
UBUNTU [Executando] - Oracle VM VirtualBox
Activities
Text Editor
*gerweb
~/projetos/viagem_ao_mundo_do_software
Save
Open
readme.mdwn
*gerweb
#!/usr/bin/env bash
```

Nas próximas linhas um comentário básico apresentando o projeto será de grande relevância:



```
Open
*gerweb
~/projetos/viagem_ao_mundo_do_software
Save
readme.mdwn
*gerweb
#!/usr/bin/env bash

Gerweb é um gerador de conteúdo estático baseado no Blogático do professor Silvio Rhatto
É um conteúdo alterado por Leandro Souza durante as aulas de "Boas práticas em Desenvolvimento
de Software" - Universidade de Campinas - SP (MOOC-GGTE Unicamp).
```

Repare que a forquilha `"#"` indica um comentário. O compilador ignorará esta linha, com exceção da primeira linha do script.

Logo depois serão detalhados os parâmetros :


```

Open  [icon]  *gerweb  ~/projetos/viagem_ao_mundo_do_software  Save  [icon]  [icon]  [icon]
*gerweb  Novo Documento de Texto.txt
usr/bin/env bash

Gerweb é um gerador de conteúdo estático baseado no Blogático do professor Silvio Rhatto
É um conteúdo alterado por Leandro Souza durante as aulas de "Boas práticas em Desenvolvimento
de Software" - Universidade de Campinas - SP (MOOC-GGTE Unicamp).

Parâmetros:

-> chama uma subshell que é o cifrão e um comando dentro de parênteses que indica em qual pasta o software está rodando:
$SEDIR "$(pwd)"
-> indica qual é a pasta dos meus arquivos fonte do blog:
TA="$BASEDIR/data"
-> Pasta de saída onde o software vai deixar os arquivos do blog gerados:
OUTPUT="$BASEDIR/output"

```

- *`BASEDIR "$(pwd)"` --> chama uma subshell que é o cifrão e um comando dentro de parênteses que indica em qual pasta o software está rodando;*
- *`DATA= "$BASEDIR/data"` --> indica qual é a pasta dos meus arquivos fonte do blog;*
- *`OUTPUT= "$BASEDIR/output"` --> Pasta de saída onde o software vai deixar os arquivos do blog gerados.*

```

Open  [icon]  *gerweb  ~/projetos/viagem_ao_mundo_do_software  Save  [icon]  [icon]  [icon]
*gerweb  Novo Documento de Texto.txt
usr/bin/env bash

Gerweb é um gerador de conteúdo estático baseado no Blogático do professor Silvio Rhatto
É um conteúdo alterado por Leandro Souza durante as aulas de "Boas práticas em Desenvolvimento
de Software" - Universidade de Campinas - SP (MOOC-GGTE Unicamp).

Parâmetros:

-> chama uma subshell que é o cifrão e um comando dentro de parênteses que indica em qual pasta o software está rodando:
$SEDIR "$(pwd)"
-> indica qual é a pasta dos meus arquivos fonte do blog:
TA="$BASEDIR/data"
-> Pasta de saída onde o software vai deixar os arquivos do blog gerados:
OUTPUT="$BASEDIR/output"

Em seguida deixaremos uma mensagem para o usuário (para quem irá rodar o software):
ho "Limando a versão anterior do blog ..."

```

Em seguida deixaremos uma mensagem para o usuário (para quem irá rodar o software)
echo "Limando a versão anterior do blog ...", pois a primeira coisa que queremos ao rodar o software é que ele apague a pasta de saída para garantir que quando formos criar um blog, ele seja gerado em uma pasta limpa, sem que esteja poluído por uma versão anterior e também, sempre que a pasta de dados for alterada, tenha-se uma versão atualizada do blog gerado:

rm -rf \$OUTPUT || exit 1

```

echo "Limando a versão anterior do blog ."
rm -rf $OUTPUT || exit 1

```

Observe os pipes "||" e o comando "exit 1" que servem como controle de erros. Caso o bloco de comandos à esquerda dos pipes falhe, o script como um todo será encerrado com um erro.

Isso é uma forma de garantir um controle de qualidade, pois ao criar um script que testa o software, podemos verificar se alguns comandos falham, ajudando o usuário a compreender por que o programa não teve êxito.

Em seguida, vamos garantir a existência das pastas necessárias para executar o programa e copiar a estrutura de pastas para a pasta de saída antes de compilar o código.

```

Open  [icon]  *gerweb  Save  [icon]  [icon]  [icon]
~/projetos/viagem_ao_mundo_do_software

usr/bin/env bash

Gerweb é um gerador de conteúdo estático baseado no Blogático do professor #Silvio Rhatto
É um conteúdo alterado por Leandro Souza durante as aulas de "Boas práticas em #Desenvolvimento
de Software" - Universidade de Campinas - SP (MOOC-GGTE Unicamp).
#####
chamar uma subshell que é o cifrão e um comando dentro de parênteses que indica #em qual pasta o software está rodando;
BASEDIR=$(pwd)
indicar qual é a pasta dos meus arquivos fonte do blog;
DATA=$BASEDIR/data
Pasta de saída onde o software vai deixar os arquivos do blog gerados.
OUTPUT=$BASEDIR/output
Em seguida deixaremos uma mensagem para o usuário (para quem irá rodar o software)
cho "Limpando a versão anterior do blog ..."
rm -rf $OUTPUT || exit 1
cho "Garantindo a existência de pastas necessárias..."
mkdir -p $OUTPUT || exit 1
cho "Copiando a estrutura de pastas..."
rsync -qav $DATA/ $OUTPUT/ || exit 1
cho "Convertendo todos os arquivos Markdown para HTML..."
find $OUTPUT -name '*.mdwn' -exec pandoc -o {} .html {} \; || exit 1
cho "Corrigindo o nome dos arquivos gerados..."

```

O comando "find" será usado para procurar todos os arquivos com extensão ".mdwn". Em seguida, o programa "pandoc" será usado para converter esses arquivos de formato ".mdwn" para ".html".

No entanto, o "pandoc" não remove a extensão ".mdwn" do nome do arquivo convertido para ".html". Para resolver isso, podemos usar o comando a seguir:

```

cho "Corrigindo o nome dos arquivos gerados..."
find $OUTPUT -name '*.html' -exec rename 's/\.mdwn.html$/ .html/' {} \; || exit 1

```

E assim terminaremos com uma mensagem agradável:

```

usr/bin/env bash

Gerweb é um gerador de conteúdo estático baseado no Blogático do professor Silvio Rhatto
É um conteúdo alterado por Leandro Souza durante as aulas de "Boas práticas em Desenvolvimento
de Software" - Universidade de Campinas - SP (MOOC-GGTE Unicamp).
#####
chamar uma subshell que é o cifrão e um comando dentro de parênteses que indica em qual pasta o software está rodando;

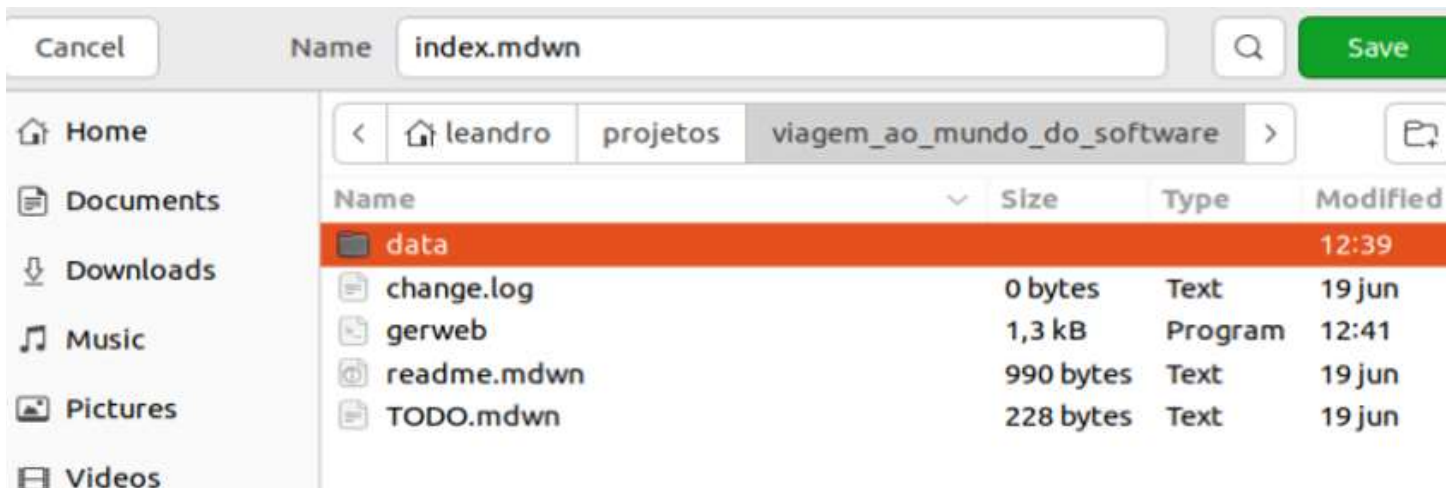
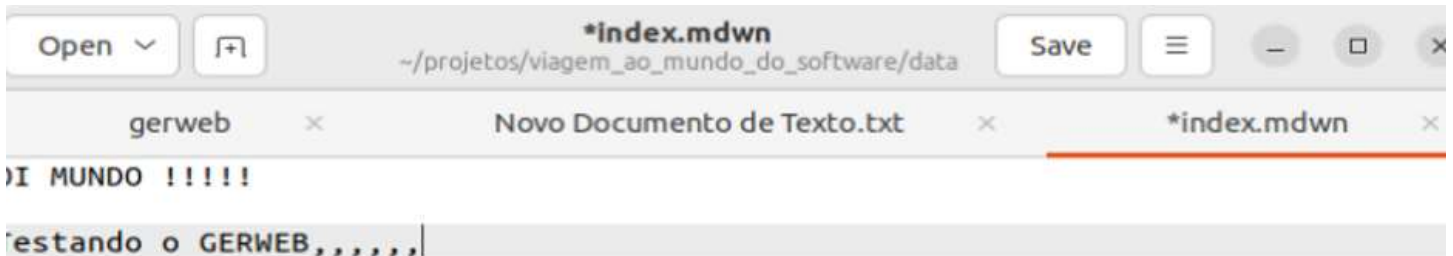
BASEDIR=$(pwd)
indicar qual é a pasta dos meus arquivos fonte do blog;
DATA=$BASEDIR/data
Pasta de saída onde o software vai deixar os arquivos do blog gerados.
OUTPUT=$BASEDIR/output
Em seguida deixaremos uma mensagem para o usuário (para quem irá rodar o software)
cho "Limpando a versão anterior do blog ..."
rm -rf $OUTPUT || exit 1
cho "Garantindo a existência de pastas necessárias..."
mkdir $OUTPUT || exit 1
cho "Convertendo todos os arquivos Markdown para HTML..."
find $OUTPUT -name '*.mdwn' -exec pandoc -o {} .html {} \; || exit 1
cho "Corrigindo o nome dos arquivos gerados..."
find $OUTPUT -name '*.html' -exec rename 's/\.mdwn.html$/ .html/' {} \; || exit 1
cho "Fim!!! Tudo bem dia!!!"

```

Agora, vamos salvar o script e iniciar o teste do mesmo, criando uma versão inicial de teste para o blog.

Primeiro, crie uma pasta chamada "data" e dentro dela crie um arquivo chamado "index.mdwn":

```
leandro@leandro-VirtualBox: ~/projetos/viagem_ao_mundo_do_software$ mkdir data
leandro@leandro-VirtualBox: ~/projetos/viagem_ao_mundo_do_software$
```



Agora iremos executar o Gerweb e para isso devemos dar permissão de execução do arquivo:

`chmod +x gerweb`

O "chmod" (change mode) muda o modo do arquivo para que ele tenha permissão de execução.

```
leandro@leandro-VirtualBox: ~/projetos/viagem_ao_mundo_do_software$ chmod +x gerweb
leandro@leandro-VirtualBox: ~/projetos/viagem_ao_mundo_do_software$
```

Por curiosidade essas são as variações comuns do comando "chmod +x":

- "chmod +x arquivo": Concede permissão de execução ao proprietário, grupo e outros usuários do arquivo. Esse comando adiciona o bit de execução para todos os usuários, mantendo as outras permissões inalteradas.

- *"chmod u+x arquivo": Concede permissão de execução exclusivamente ao proprietário do arquivo. Apenas o proprietário terá permissão de execução após a execução desse comando, enquanto as permissões do grupo e outros usuários permanecem inalteradas.*
- *"chmod g+x arquivo": Concede permissão de execução exclusivamente ao grupo proprietário do arquivo. Apenas os membros do grupo proprietário terão permissão de execução após a execução desse comando, enquanto as permissões do proprietário e outros usuários permanecem inalteradas.*
- *"chmod o+x arquivo": Concede permissão de execução exclusivamente a outros usuários (não proprietários e não membros do grupo proprietário). Apenas os usuários que não são proprietários nem membros do grupo proprietário terão permissão de execução após a execução desse comando, enquanto as permissões do proprietário e grupo permanecem inalteradas.*
- *"chmod a+x arquivo": Concede permissão de execução a todos os usuários (proprietário, grupo e outros). Esse comando adiciona o bit de execução para todos os usuários, sem alterar as outras permissões.*
- *"chmod -x arquivo": Remove a permissão de execução de todos os usuários. Esse comando remove o bit de execução, negando a permissão de execução para todos os usuários.*
- *"chmod ug+x arquivo": Concede permissão de execução ao proprietário e ao grupo proprietário do arquivo, mantendo as outras permissões inalteradas.*
- *"chmod u+x,g+x arquivo": Concede permissão de execução ao proprietário e ao grupo proprietário do arquivo, sem alterar as outras permissões.*

Agora devemos instalar o programa "pandoc" de conversão dos arquivos ".mdwn" para ".html" com o comando "sudo apt-get install pandoc" e o "rename" (sudo apt-get install rename) para renomear os arquivos restantes na pasta OUTPUT criada...

```
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ sudo apt-get
install pandoc
[sudo] password for leandro:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libcbmark-gfm-extensions0.29.0.gfm.3 libcbmark-gfm0.29.0.gfm.3 pandoc-data
Suggested packages:
  texlive-latex-recommended texlive-xetex texlive-luatex pandoc-citeproc
  texlive-latex-extra context wkhtmltopdf librsvg2-bin groff ghc nodejs php
  python ruby r-base-core libjs-mathjax libjs-katex
  citation-style-language-styles
The following NEW packages will be installed:
  libcbmark-gfm-extensions0.29.0.gfm.3 libcbmark-gfm0.29.0.gfm.3 pandoc
  pandoc-data
0 upgraded, 4 newly installed, 0 to remove and 115 not upgraded.
Need to get 20,6 MB of archives.
After this operation, 156 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://br.archive.ubuntu.com/ubuntu jammy/universe amd64 libcbmark-gfm0.29
  0.gfm.3 amd64 0.29.0.gfm.3-3 [115 kB]
```

```
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ sudo apt-get install rename
[sudo] password for leandro:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  rename
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 20,2 kB of archives.
After this operation, 55,3 kB of additional disk space will be used.
Get:1 http://br.archive.ubuntu.com/ubuntu jammy/universe amd64 rename all 1.30-1 [20,2 kB]
Fetched 20,2 kB in 2s (13,1 kB/s)
Selecting previously unselected package rename.
Reading database ... 203757 files and directories currently installed.)
Preparing to unpack .../archives/rename_1.30-1_all.deb ...
Unpacking rename (1.30-1) ...
Setting up rename (1.30-1) ...
update-alternatives: using /usr/bin/file-rename to provide /usr/bin/rename (rename) in auto mode
Processing triggers for man-db (2.10.2-1) ...
```

... e executar o programa gerweb com o ponto barra "/" antes do nome do arquivo. É uma sintaxe básica de execução de scripts:

```
leandro@leandro-VirtualBox: ~/projetos/viagem_ao_mundo_do_soft...
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ chmod +x gerweb
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ ./gerweb
bash: ./gerweb: usr/bin/env: bad interpreter: No such file or directory
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$
```

Notamos então um erro inesperado que identificamos na imagem acima. Na sintaxe "usr/bin/env" falta uma barra "/" no início do arquivo do programa:



```
Open  ▾  [icon]  gerweb  ~/projetos/viagem_ao_mundo_do_software  Save  [icon]  [icon]  [icon]
gerweb  x  Novo Documento de Texto.txt  x  index.mdwn  x
usr/bin/env bash

Gerweb é um gerador de conteúdo estático baseado no Blogático do professor #Silvio Rhatto
É um conteúdo alterado por Leandro Souza durante as aulas de "Boas práticas em #Desenvolvimento
de Software" - Universidade de Campinas - SP (MOOC-GGTE Unicamp).
#####
chamar uma subshell que é o cifrão e um comando dentro de parênteses que indica #em qual pasta o
software está rodando;

ASEDIR "$(pwd)"
indicar qual é a pasta dos meus arquivos fonte do blog;
ATA="$BASEDIR/data"
Pasta de saída onde o software vai deixar os arquivos do blog gerados.
UTPUT="$BASEDIR/output"
Em seguida deixaremos uma mensagem para o usuário (para quem irá rodar o software)
cho "Limpando a versão anterior do blog ..."
n -rf $OUTPUT || exit 1
cho "Garantindo a existência de pastas necessárias..."
kdir $OUTPUT || exit 1
cho "Convertendo todos os arquivos Markdown para HTML..."
lnd $OUTPUT -name '*.mdwn' -exec pandoc -o {} .html {} \; || exit 1
cho "Corrigindo o nome dos arquivos gerados,,"
lnd $OUTPUT -name '*.html' -exec rename 's/\.mdwn.html$/ .html/' {} \; || exit 1
cho "Então... Tudo em bom dia!!!"
```

O correto é:



Agora o programa iniciou, porém um erro de pasta não criada aconteceu:

```
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ ./gerweb
/gerweb: line 11: BASEDIR: command not found
 limpando a versão anterior do blog ...
garantindo a existência de pastas necessárias...
mkdir: cannot create directory '/output': Permission denied
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$
```

A cada execução de './gerweb', um erro foi corrigido, como a remoção de pontos ou vírgulas indevidos e a exclusão de linhas desnecessárias:

```
leandro@leandro-VirtualBox: ~/projetos/viagem_ao_mundo...
corrigindo o nome dos arquivos gerados,,,
/gerweb: line 24: unexpected EOF while looking for matching `"'
/gerweb: line 25: syntax error: unexpected end of file
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ ./gerweb
 limpando a versão anterior do blog ...
garantindo a existência de pastas necessárias...
convertendo todos os arquivos Markdown para HTML...
corrigindo o nome dos arquivos gerados,,,
/gerweb: line 24: unexpected EOF while looking for matching `"'
/gerweb: line 25: syntax error: unexpected end of file
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ ./gerweb
 limpando a versão anterior do blog ...
garantindo a existência de pastas necessárias...
convertendo todos os arquivos Markdown para HTML...
corrigindo o nome dos arquivos gerados,,,
/gerweb: line 24: unexpected EOF while looking for matching `"'
/gerweb: line 25: syntax error: unexpected end of file
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ ./gerweb
```

Até que o programa rodou e obtivemos nosso arquivo final revisado:

```
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ ./gerweb
 limpando a versão anterior do blog ...
garantindo a existência de pastas necessárias...
convertendo todos os arquivos Markdown para HTML...
corrigindo o nome dos arquivos gerados,,,
feito! Tenha um bom dia!
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$
```



```

Open [icon] gerweb ~/projetos/viagem_ao_mundo_do_software Save [icon] [icon] [icon]
gerweb x Novo Documento de Texto3.txt x
/usr/bin/env bash

Gerweb é um gerador de conteúdo estático baseado no Blogático do professor #Silvio Rhatto
É um conteúdo alterado por Leandro Souza durante as aulas de "Boas práticas em #Desenvolvimento
de Software" - Universidade de Campinas - SP (MOOC-GGTE Unicamp).
#####
hamar uma subshell que é o cifrão e um comando dentro de parênteses que Indica #em qual pasta o software está rodando;
SEDIR="$(pwd)"
ndicar qual é a pasta dos meus arquivos fonte do blog;
TA="$BASEDIR/data"
asta de saída onde o software vai deixar os arquivos do blog gerados.
TPUT="$BASEDIR/output"
m seguida deixaremos uma mensagem para o usuário (para quem irá rodar o software)
ho "Limpando a versão anterior do blog ..."
-rf $OUTPUT || exit 1
ho "Garantindo a existência de pastas necessárias..."
dir -p $OUTPUT || exit 1
ho "Copiando a estrutura de pastas..."
ync -qav $DATA/ $OUTPUT/ || exit 1
ho "Convertendo todos os arquivos Markdown para HTML..."
nd $OUTPUT -name '*.mdwn' -exec pandoc -o {}.html {} \; || exit 1
ho "Corrigindo o nome dos arquivos gerados..."
nd $OUTPUT -name '*.html' -exec rename 's/\.mdwn.html$/\.html/' {} \; || exit 1
ho "Feito! Tenha um bom dia!"

```

Ao listar o conteúdo da pasta 'OUTPUT', podemos observar que, quando ela foi criada, o arquivo 'index.mdwn' foi copiado para seu interior e convertido para 'index.html', como demonstrado abaixo:

```

eandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ ls output/
index.html  index.mdwn
eandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$

```

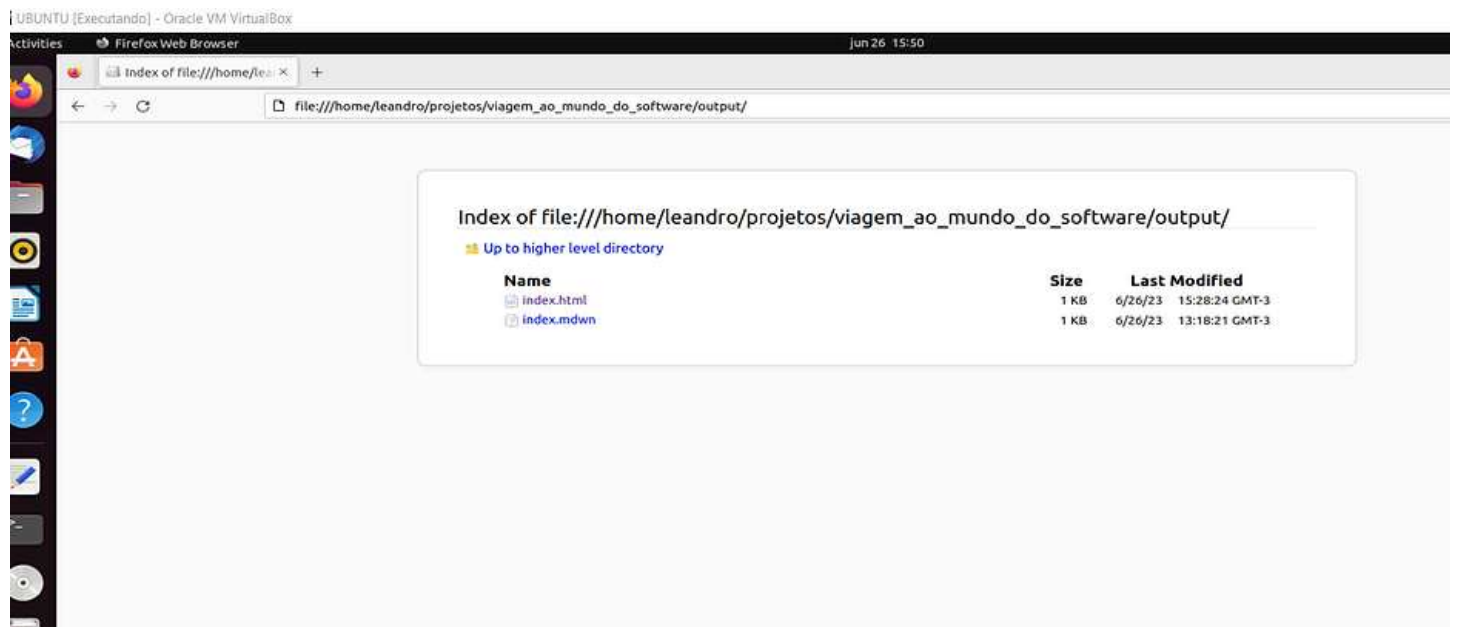
Podemos mostrar o conteúdo do arquivo "index.html" com o comando de concatenação "cat":

```

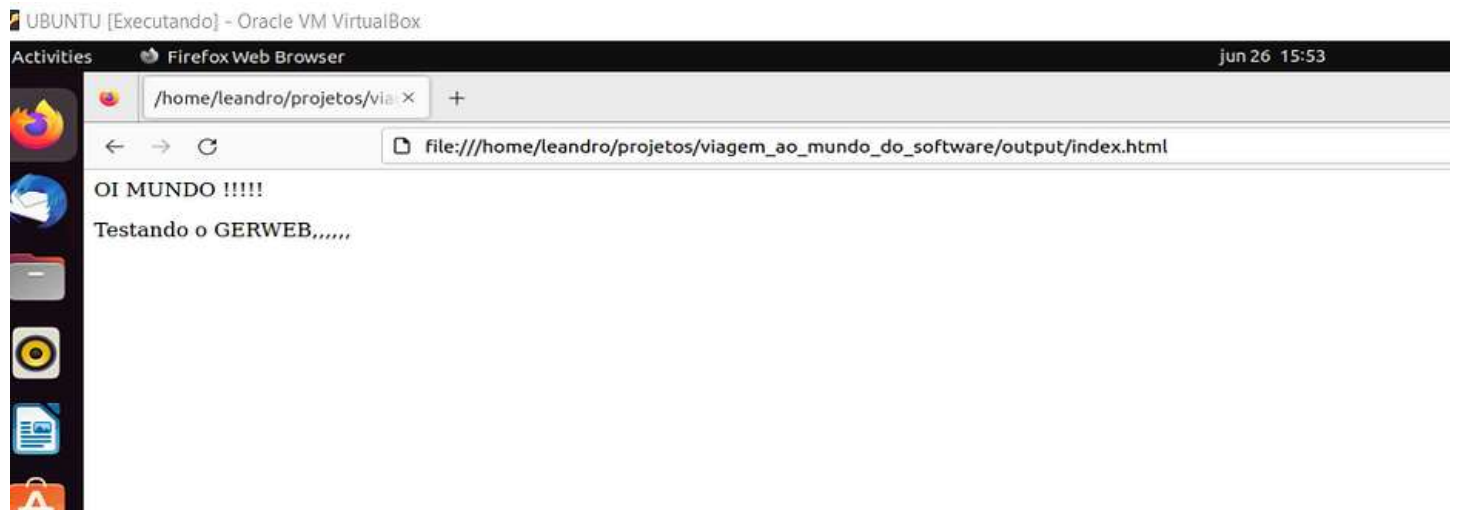
eandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software/output$ cat index.html
p>OI MUNDO !!!!!</p>
p>Testando o GERWEB, , , , ,,</p>
eandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software/output$

```

Para testar basta irmos ao navegador e indicar o caminho do arquivo:



A página aparecerá com nosso arquivo de exemplo bem simples:



Faremos agora uma pequena alteração no Gerweb para que ele gere um "html" completo com todas as tags necessárias para um sitio de verdade .

Iremos implementar uma opção para o programa "pandoc" no arquivo do software, assim , ele irá usar um template de "html" completo para a constituição do site.


```

gerweb
~/projetos/viagem_ao_mundo_do_software

Novo Documento de Texto3.txt

Gerweb é um gerador de conteúdo estático baseado no Blogático do professor #Silvio Rhatto
É um conteúdo alterado por Leandro Souza durante as aulas de "Boas práticas em #Desenvolvimento
de Software" - Universidade de Campinas - SP (MOOC-GGTE Unicamp).
#####
chamar uma subshell que é o cifrão e um comando dentro de parênteses que Indica #em qual pasta o software es
$BASEDIR=$(pwd)
indicar qual é a pasta dos meus arquivos fonte do blog;
ATA="$BASEDIR/data"
Pasta de saída onde o software vai deixar os arquivos do blog gerados.
OUTPUT="$BASEDIR/output"
PANDOC_OPTS="--template=default.html5"
Em seguida deixaremos uma mensagem para o usuário (para quem irá rodar o software)
cho "Limpendo a versão anterior do blog ..."
rm -rf $OUTPUT || exit 1
cho "Garantindo a existência de pastas necessárias..."
mkdir -p $OUTPUT || exit 1
cho "Copiando a estrutura de pastas..."
sync -qav $DATA/ $OUTPUT/ || exit 1
cho "Convertendo todos os arquivos Markdown para HTML..."
find $OUTPUT -name '*.mdwn' -exec pandoc $PANDOC_OPTS {} {} .html {} \; || exit 1
cho "Corrigindo o nome dos arquivos gerados..."
find $OUTPUT -name '*.html' -exec rename 's/\.mdwn.html$/\.html/' {} \; || exit 1
cho "Feito! Tenha um bom dia!"

```

Agora novamente executando o programa e concatenando para que mostre seu conteúdo, veremos que o arquivo será bem mais complexo que o anterior devido a utilização do template.

Um template de site é uma estrutura predefinida para criar um site. Ele define o layout e a aparência geral. Os templates agilizam o processo de criação, economizando tempo e fornecendo uma base para personalizar o conteúdo.

```

leandro@leandro-VirtualBox: ~/projetos/viagem_ao_mundo_do_software

leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ ./gerweb
Limpendo a versão anterior do blog ...
Garantindo a existência de pastas necessárias...
Copiando a estrutura de pastas...
Convertendo todos os arquivos Markdown para HTML...
[WARNING] Could not deduce format from file extension .mdwn
Defaulting to markdown
[WARNING] This document format requires a nonempty <title> element.
Defaulting to 'index' as the title.
To specify a title, use 'title' in metadata or --metadata title="...".
Corrigindo o nome dos arquivos gerados...
Feito! Tenha um bom dia!
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ cat index.html
cat: index.html: No such file or directory
leandro@leandro-VirtualBox:~/projetos/viagem_ao_mundo_do_software$ cat output/index.html
DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" lang="" xml:lang="">
<head>
<meta charset="utf-8" />
<meta name="generator" content="pandoc" />
<meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=yes" />
<title>index</title>
<style>
code{white-space: pre-wrap;}
span.smallcaps{font-variant: small-caps;}
span.underline{text-decoration: underline;}
div.column{display: inline-block; vertical-align: top; width: 50%;}
div.hanging-indent{margin-left: 1.5em; text-indent: -1.5em;}
ul.task-list{list-style: none;}
</style>
<body>
<div>
<p>OI MUNDO !!!!!</p>
<p>Testando o GERWEB,,,,,</p>
</div>
</body>
</html>

```

Em resumo, neste post aprendemos sobre a criação de um site estático para documentar o progresso do conhecimento em desenvolvimento de software.

Exploramos a importância dos templates na criação do site e conhecemos os passos necessários para gerar e atualizar o site com novas informações.

Com o conhecimento adquirido, agora temos os recursos necessários para criar nosso próprio site de acompanhamento e compartilhar nosso progresso.

