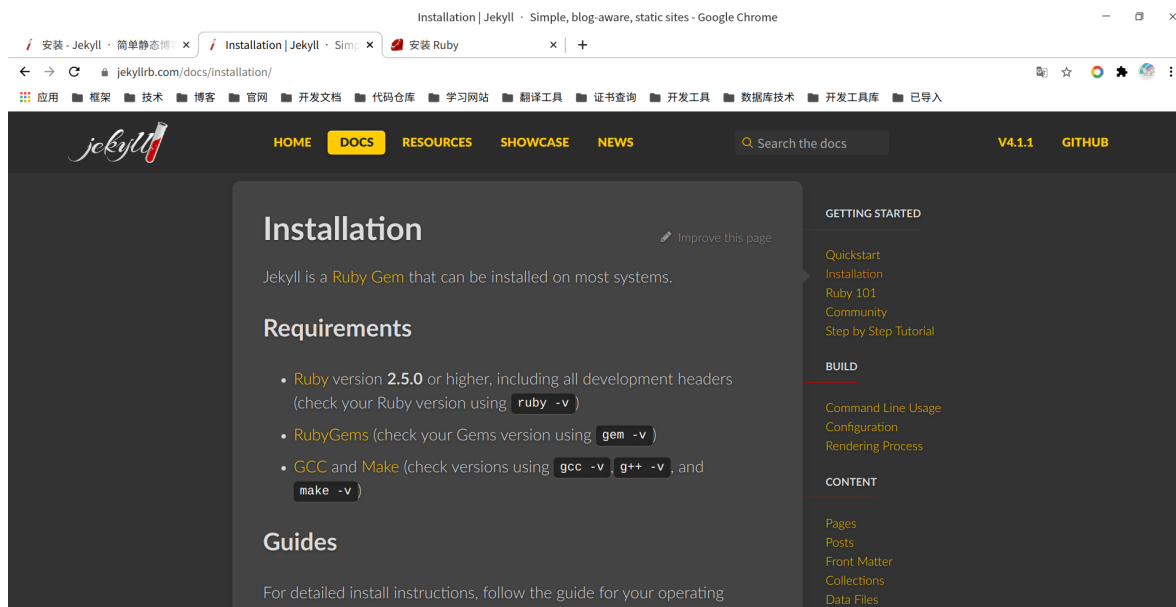


Deeping 安装jekyll

参考官网地址：<https://jekyllrb.com/docs/> 或 <http://jekyllcn.com/>

参考 <https://jekyllrb.com/docs/> 进行安装：



备注：安装好 Ruby 后根据上述图片提示，逐次使用版本查看指令查看。

一、安装Ruby

1.1、安装指令

```
sudo apt-get install ruby-full
```

1.2、查看指令

```
ruby -v
```

1.3、操作截图

```
joshua@joshua-PC:~$ sudo apt-get install ruby-full
请输入密码
[sudo] joshua 的密码：
验证成功
正在读取软件包列表... 完成
正在分析软件包的依赖关系树
正在读取状态信息... 完成
将会同时安装下列软件：
  ri
下列【新】软件包将被安装：
  ri ruby-full
升级了 0 个软件包，新安装了 2 个软件包，要卸载 0 个软件包，有 0 个软件包未被升级。
需要下载 20.1 kB 的归档。
解压缩后会消耗 36.9 kB 的额外空间。
您希望继续执行吗？ [Y/n] y
获取:1 https://community-packages.deepin.com/deepin apricot/main amd64 ri all 1:2.5.1 [10.1 kB]
获取:2 https://community-packages.deepin.com/deepin apricot/main amd64 ruby-full all 1:2.5.1 [10.0 kB]
已下载 20.1 kB，耗时 1秒 (36.7 kB/s)
正在选中未选择的软件包 ri。
(正在读取数据库 ... 系统当前共安装有 233118 个文件和目录。)
准备解压 .../archives/ri_1%3a2.5.1_all.deb ...
正在解压 ri (1:2.5.1) ...
正在选中未选择的软件包 ruby-full。
准备解压 .../ruby-full_1%3a2.5.1_all.deb ...
正在解压 ruby-full (1:2.5.1) ...
正在设置 ri (1:2.5.1) ...
正在设置 ruby-full (1:2.5.1) ...
joshua@joshua-PC:~$ ruby -v
ruby 2.5.5p157 (2019-03-15 revision 67260) [x86_64-linux-gnu]
joshua@joshua-PC:~$
```

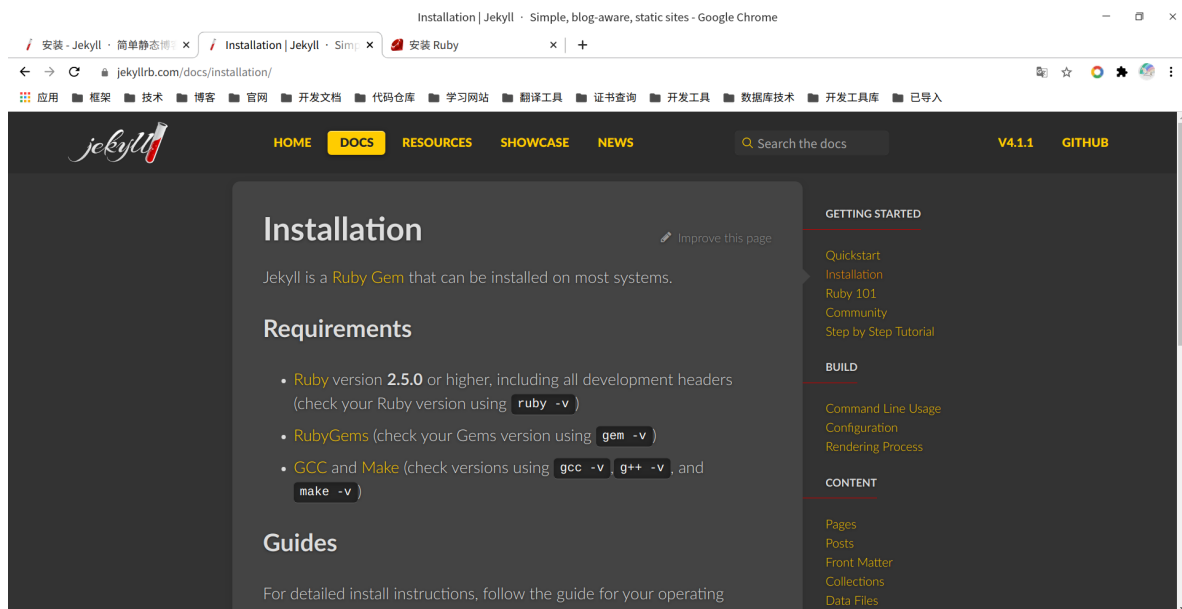
1.4、参考指南

https://www.ruby-lang.org/zh_cn/documentation/installation/#apt



二、安装g++

安装好 Ruby 后，根据官网的提示，逐一使用下列指令查看：



```
joshua@joshua-PC:~$ ruby -v
ruby 2.5.5p157 (2019-03-15 revision 67260) [x86_64-linux-gnu]
joshua@joshua-PC:~$ gem -v
2.7.6.2
joshua@joshua-PC:~$ gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/lib/gcc/x86_64-linux-gnu/8/lto-wrapper
OFFLOAD_TARGET_NAMES=nvptx-none
OFFLOAD_TARGET_DEFAULT=1
Target: x86_64-linux-gnu
Configured with: ../src/configure -v --with-pkgversion='Uos 8.3.0.3-3+rebuild' --with-bugurl=file:///usr/share/doc/gcc-8/README.Bugs --enable-languages=c,ada,c++,go,brig,d,fortran,objc,obj-c++ --prefix=/usr --with-gcc-major-version-only --program-suffix=-8 --program-prefix=x86_64-linux-gnu- --enable-shared --enable-linker-build-id --libexecdir=/usr/lib --without-included-gettext --enable-threads=posix --libdir=/usr/lib --enable-nls --enable-bootstrap --enable-clocale=gnu --enable-libstdc++-debug --enable-libstdc++-time=yes --with-default-libstdc++-abi=new --enable-gnu-unique-object --disable-vtable-verify --enable-libmpx --enable-plugin --with-system-zlib --with-target-system-zlib --enable-objc-gc=auto --enable-multiarch --disable-werror --with-arch-32=i686 --with-abi=m64 --with-multilib-list=m32,m64,mx32 --enable-multilib --with-tune=generic --enable-offload-targets=nvptx-none --without-cuda-driver --enable-checking=release --build=x86_64-linux-gnu --host=x86_64-linux-gnu --target=x86_64-linux-gnu
Thread model: posix
gcc version 8.3.0 (Uos 8.3.0.3-3+rebuild)
joshua@joshua-PC:~$ g++ -v
bash: g++: 未找到命令
joshua@joshua-PC:~$ make -v
GNU Make 4.2.1
为 x86_64-pc-linux-gnu 编译
Copyright (C) 1988-2016 Free Software Foundation, Inc.
许可证: GPLv3+: GNU 通用公共许可证第 3 版或更新版本 <http://gnu.org/licenses/gpl.html>。
本软件是自由软件: 您可以自由修改和重新发布它。
在法律允许的范围内有其他保证。
joshua@joshua-PC:~$
```

根据提示信息，需要安装g++。

2.1、安装指令

```
sudo apt-get install g++
```

2.2、查看指令

```
g++ -v
```

2.3、操作截图

```
joshua@joshua-PC:~$ sudo apt-get install g++
正在读取软件包列表... 完成
正在分析软件包的依赖关系树
正在读取状态信息... 完成
建议安装：
  g++-multilib
下列【新】软件包将被安装：
  g++
升级了 0 个软件包，新安装了 1 个软件包，要卸载 0 个软件包，有 0 个软件包未被升级。
需要下载 0 B/1,644 B 的归档。
解压后会消耗 15.4 kB 的额外空间。
正在选中未选择的软件包 g++。
(正在读取数据库... 系统当前共安装有 237253 个文件和目录。)
准备解压 .../g++_4%3a8.3.0-1_amd64.deb ...
正在解压 g++ (4:8.3.0-1) ...
正在设置 g++ (4:8.3.0-1) ...
update-alternatives: 使用 /usr/bin/g++ 来在自动模式中提供 /usr/bin/c++ (c++)
joshua@joshua-PC:~$ g++ -v
Using built-in specs.
COLLECT_GCC=g++
COLLECT_LTO_WRAPPER=/usr/lib/gcc/x86_64-linux-gnu/8/lto-wrapper
OFFLOAD_TARGET_NAMES=nvptx-none
OFFLOAD_TARGET_DEFAULT=1
Target: x86_64-linux-gnu
Configured with: ../src/configure -v --with-pkgversion='Uos 8.3.0.3+rebuild' --with-bugurl=file:///usr/share/doc/gcc-8/README.Bugs --enable-languages=c,ada,c++,go,brig,d,fortran,objc,obj-c++ --prefix=/usr --with-gcc-major-version-only --program-suffix=-8 --program-prefix=x86_64-linux-gnu- --enable-shared --enable-linker-build-id --libexecdir=/usr/lib --without-included-gettext --enable-threads=posix --libdir=/usr/lib --enable-nls --enable-bootstrap --enable-clocale=gnu --enable-libstdcxx-debug --enable-libstdcxx-timeyes --with-default-libstdcxx-abi=new --enable-gnu-unique-object --disable-vtable-verify --enable-libmpx --enable-plugin --with-system-zlib --with-target-system-zlib --enable-objc-gc=auto --enable-multiarch --disable-werror --with-arch=32=i686 --with-abi=m64 --with-multilib-list=m32,m64,mx32 --enable-multilib --with-tune=generic --enable-offload-targets=nvptx-none --without-cuda-driver --enable-checking=release --build=x86_64-linux-gnu --host=x86_64-linux-gnu --target=x86_64-linux-gnu
Thread model: posix
gcc version 8.3.0 (Uos 8.3.0.3+rebuild)
joshua@joshua-PC:~$
```

二、安装jekyll

3.1、安装

安装指令：

```
sudo gem install jekyll bundler
```

操作截图：

```
joshua@joshua-PC:~$ sudo gem install jekyll bundler
Successfully installed jekyll-4.1.1
Parsing documentation for jekyll-4.1.1
Done installing documentation for jekyll after 0 seconds
Fetching: bundler-2.1.4.gem (100%)
Successfully installed bundler-2.1.4
Parsing documentation for bundler-2.1.4
Installing ri documentation for bundler-2.1.4
Done installing documentation for bundler after 2 seconds
2 gems installed
```

3.2、创建

安装指令：

```
jekyll new myblog
```

操作截图：

```
joshua@joshua-PC:~$ ls
插件 Desktop Documents Downloads Gemfile Gemfile.lock Music Pictures _site Videos
joshua@joshua-PC:~$ jekyll new myblog
Running bundle install in /home/joshua/myblog...
  Bundler: Bundle complete! 0 Gemfile dependencies, 1 gem now installed.
  Bundler: Use `bundle info [gemname]` to see where a bundled gem is installed.The Gemfile specifies no dependencies
New jekyll site installed in /home/joshua/myblog.
joshua@joshua-PC:~$ ls
插件 Desktop Documents Downloads Gemfile Gemfile.lock Music myblog Pictures _site Videos
joshua@joshua-PC:~$
```

3.3、启动服务

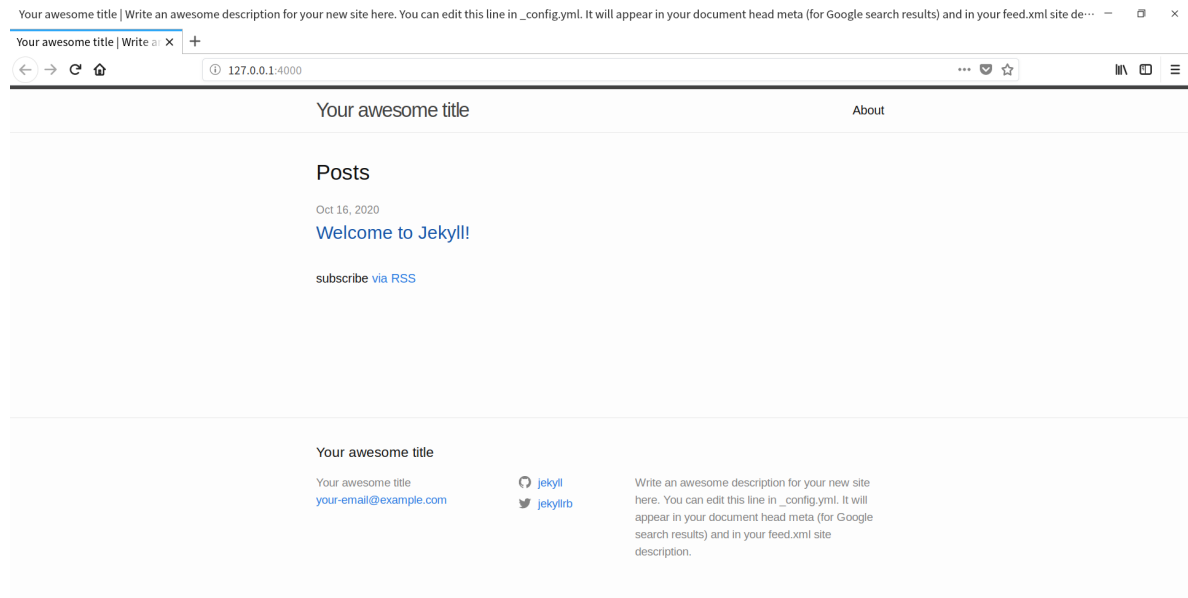
进入myblog:

```
cd myblog
```

启动服务:

```
bundle exec jekyll serve
```

访问端口:



操作截图:

