

使用 WSL 和 VS Code

WSL (Windows Subsystem for Linux) 是一个很好用的工具。在 Windows 系统中使能了 WSL，就能在其下安装某个 Linux 发行版 (免费的哦!)。这样一来，就不要再安装类似于 VMware 的虚拟机了。

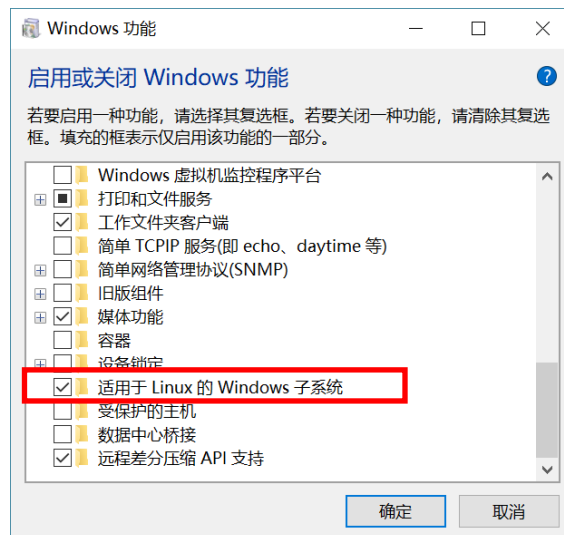
这里给大家介绍一下 WSL 的配置以及 Ubuntu 的安装。

注：如果你的 Windows 10 已经更新到了内部 2004 版，那么你可以安装 WSL2。你可以在设置->更新和安全->OS 内部版本信息->Windows 规格一栏中看到内部版本信息。

如果安装了 WSL2，那么推荐安装的 Linux 系统是 Ubuntu 20.04。这些版本都可以在 Windows Store 中搜索到。

一、启用 WSL/WSL2

1. 打开控制面板
2. 选择“程序和功能”
3. 选择“启用和关闭 Windows 功能”，勾选红框中的设置，然后确定。



注：如果是 2004 版，并且要使用 WSL2，那么在这个界面中还要勾选“虚拟机平台”。我这个界面版本较低，还没有这个选项。

4. 此后 Windows 要重新启动。启动完成后可以安装 Linux。

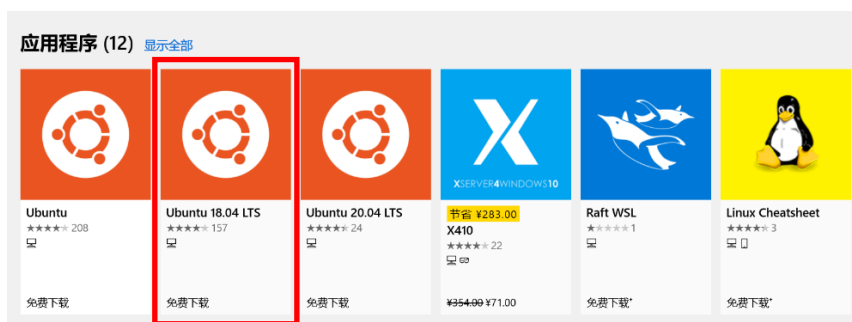
注：重启之后，在安装 Linux 之前，建议将你的 WSL 更新到 WSL2。这需要你的 Windows 10 的内部版本是 2004。如果不是，请忽略这个注释。

你的 Windows 默认的 WSL 的版本可能是 1。如果要启用 WSL2，那么你需要完成一些转换。请下载 wsl2_update_x64.msi，然后在启用了 WSL1 后安装这个升级程序。安装好后，用管理员身份运行 powershell，然后输入如下命令：

```
wsl --set-default-version 2
```

二、Ubuntu 的安装

1. 打开 Windows Store，搜索 Ubuntu，出现如下界面：



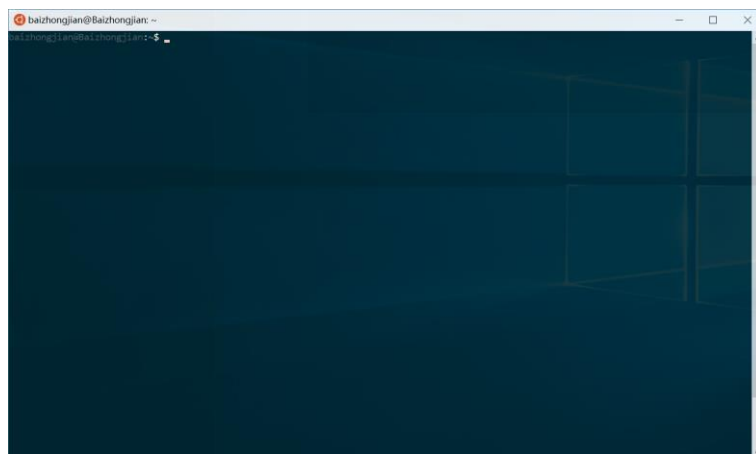
2. 点击 Ubuntu 18.04，其后的过程就是下载和安装。
3. 安装过程几乎不需要手工干预，唯一需要输入的是一个账户及其密码。此后使用 `sudo` 命令时需要上述设好的密码。

提示：

1. 这样安装的 Ubuntu 只有命令行，没有窗口界面。所以需要学习命令行使用方法。但如果是 WSL2 下的 20.04 版本，你可以有窗口界面哦！
2. Ubuntu 安装时，首先会让你输入账户名。你选一个合适的就行。注意，必须是全是小写字母。
3. 接下来是设置密码。密码需要输入两次以验证正确性。
4. 注意，输入密码时，屏幕上没有回显，即：你在敲键盘时，屏幕上没有显示。此时千万不要认为没有输进去哦！另外，请一定要记住你输入的密码！

三、配置 Ubuntu

1. 启动安装后的 Ubuntu 系统，打开的终端界面类似于：



提示：Ubuntu 和 Windows 互操作。

Windows 系统下的磁盘全部都挂接到了 Ubuntu 系统下。例如，C:盘对应的是 `/mnt/c`，其它盘以此类推。进入这些目录后就可以看到原来在 Windows 下的所有文档了。

2. 替换更新源列表文件 `sources.list`。
- 1) 在 Windows 中启动任意一种编辑器（例如 Sublime），将下面的内容复制到其中：

```
deb http://mirrors.aliyun.com/ubuntu/ bionic main restricted universe multiverse
deb http://mirrors.aliyun.com/ubuntu/ bionic-security main restricted universe multiverse
deb http://mirrors.aliyun.com/ubuntu/ bionic-updates main restricted universe multiverse
deb http://mirrors.aliyun.com/ubuntu/ bionic-backports main restricted universe multiverse
```

```
deb http://mirrors.aliyun.com/ubuntu/ bionic-proposed main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ bionic main restricted universe multiverse
deb-src http://mirrors.aliyun.com/ubuntu/ bionic-security main restricted universe multiverse
deb-src http://mirrors.aliyun.com/ubuntu/ bionic-updates main restricted universe multiverse
deb-src http://mirrors.aliyun.com/ubuntu/ bionic-backports main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ bionic-proposed main restricted universe multiverse

deb http://archive.canonical.com/ubuntu/ bionic partner
deb http://extras.ubuntu.com/ubuntu/ bionic main
```

注：bionic 是 Ubuntu 对应版本的发行号，它对应 18.04 版本。如果你用的是 20.04，那么请将 bionic 全部替换为 focal。

- 2) 将这个编辑好的文件保存到任意一个盘，例如 E:，并将文件命名为 sources.list。
- 3) 在 Ubuntu 终端中发出如下命令：

```
$ cd /etc/apt
$ sudo cp sources.list sources.list.bak
$ sudo cp /mnt/e/sources.list .
```

注：\$ 是 Linux 的命令行的提示符，不是命令的一部分。另外，如果你进入的 root 模式，那么提示符是#。

3. 更新 Ubuntu 和包。在终端中发出如下命令：

```
$ sudo apt update
```

命令要求输入 root 密码。此后可以看到更新信息。请忽略所有错误。
再发出如下命令：

```
$ sudo apt upgrade -y
```

然后会看到系统的很多包的更新信息。请忽略所有错误。

注：Ubuntu 的维护团队经常更新系统，所以你需要过几天就发出上述两条命令更新和升级 Ubuntu。

4. 安装 GCC 和 Make 工具。

在终端中发出如下命令，用来安装 GCC 8.3：

```
$ sudo apt install gcc-8
$ sudo apt install g++-8
$ sudo apt install make
```

注：20.04 可以安装 GCC 9.3。如果要安装，请将 gcc-8 替换为 gcc-9，g++-8 替换为 g++-9。以下同。

此后会看到安装信息。安装好后，这些编译器会被设置为默认的。

提示：如果安装过程提示原来的版本以不再使用，那么请观察提示信息，按提示移除原来的包。

此后，发出如下命令验证是否安装成功：

```
$ gcc-8 -v
```

```
$ g++-8 -v
```

如果看到版本信息，那么说明安装成功。

再后，创建符号链接，方便使用：

```
$ cd /usr/bin  
$ sudo ln -s gcc-8 gcc  
$ sudo ln -s g++-8 g++
```

以后就用 gcc 和 g++ 来启动编译器了。例如：

```
$ gcc f.c -o f -fsanitize=address  
$ g++ g.cpp -o g -std=c++17 -fsanitize=address
```

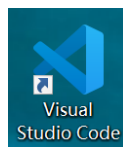
提示：所在的目录必须具有写权限。

四、VS code 的安装

Ubuntu 系统下没有图形界面，因此编码只能用 vim。如果你对这个工具不熟悉，那么建议你在 Windows 下用 VS code 来编码。

注：如果你需要在 Ubuntu 环境下编译代码，那么需要 VS code 远程连接到 Ubuntu。下面会讲到如何连接。

1. 下载 VS code，然后安装。



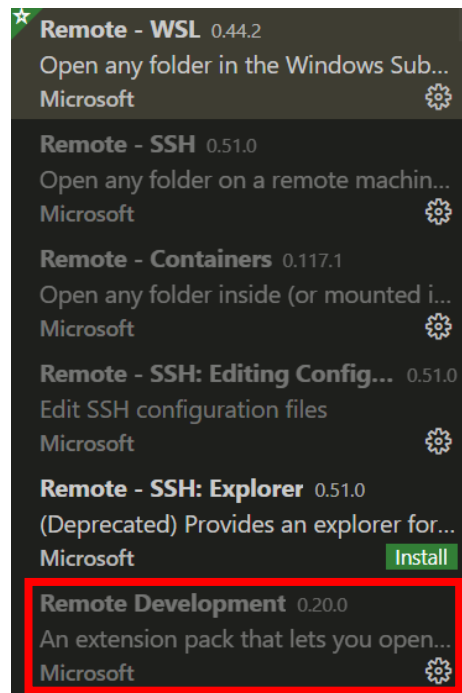
注：这是我安装好 VS Code 后，在 Windows 桌面上放置的快捷启动图标。

2. 安装插件。

启动 VS Code，点击左侧的插件管理图标。



在输入框中输入：remote development，出现如下界面。找到匹配的那个（红框），点击安装。



注：我的已经安装好了，所以这个插件是灰色的。

此后，你的 Code 的左侧栏就是这样的：



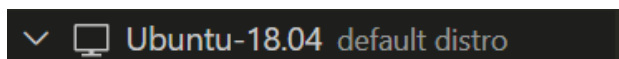
3. 在 Windows 中创建一个目录，该目录用于 Linux 程序的开发。

例如：建立 E:\test，专用于开发 Linux 下运行的 C/C++ 程序。这个 Windows 目录对应的 Linux 路径是：/mnt/e/test。

4. 用 Code 远程连接 Ubuntu

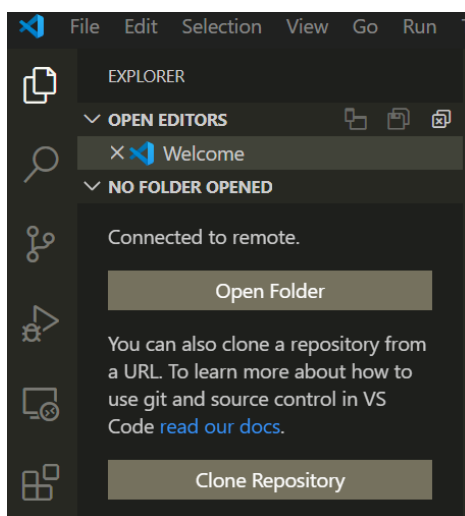
1) 点击远程连接图标，默认要连接到 WSL Targets

2) 在鼠标放到这个标题上：

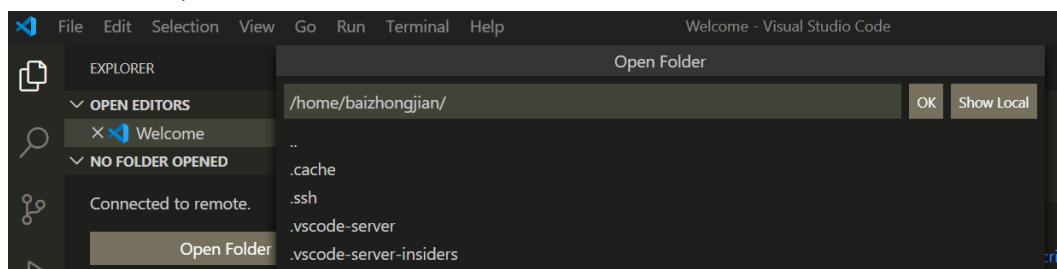


后面会出现一个有加号的目录图标。点击它。

3) 此后 Code 会打开一个新窗口。点击新窗口的打开目录图标，出现如下界面：



点击 Open Folder 按钮，出现如下界面：



在输入框中输入你的目录名，然后按 OK。

注：

1) 你输入的目录是在 WSL 中安装的 Ubuntu 系统下，从原则上讲，是一个远程系统中的目录。

2) 这个目录是 Linux 下的目录，因此路径名要符合 Linux 规范。

3) 如果你选择在 Windows 下建好的目录，例如：E:\test，那么这样输入：

/mnt/e/test

4) 此后，你指定的目录被打开。如果里面有文件，那么就会列在窗口左侧。

5) 现在，你就可以新建文件进行编码了。

注：

1) 在左侧的文件列表中，单击文件名可以打开文件，但一般情况下，文件名是斜体字。这说明这个编辑窗口要被下一个打开的文件窗口覆盖。此时，可以双击文件名，使之变成正的就可以了。

2) 可以双击文件名, 这样打开的窗口就不会被下一个覆盖了。

5. 编译链接

在编辑窗口中按“CTRL+`”, 会在下部弹出一个 Linux 终端, 并且进入了你打开的目录。

此时, 可以在终端中输入任何 Linux 命令, 例如:

编译链接源代码 f.c, 生成可执行代码 f:

```
$ gcc f.c -o f
```

运行可执行代码:

```
$ f
```

如果出现错误, 说: 找不到这个文件, 那么这么做:

```
$ ./f
```

如果你觉得这样做很麻烦, 那么请自行查阅: 如何将当前目录加入到系统搜索路径中。

注: 虽然看上去你是在 Windows 环境下编码, 但实际上你所做的工作都是在 WSL 中进行的。当然, Windows 和 WSL 的互操作非常方便, 你可以在 Windows 环境下看到你创建的源文件, 并可以用任何一种编辑软件打开它们。

如果在 WSL 中使用 sanitize 选项编译链接程序, 并在运行时出现下面的错误信息:

```
$ g++ f.cpp -o f -std=c++17 -fsanitize=address
$ f
=====021====Leaksanitizer has encountered a fatal error
...
```

但你又确信程序没有任何问题, 那么这个问题就是因为权限不够导致的。解决问题的方法是用 sudo 来运行程序:

```
$ sudo ./f
```