

第一天 环境部署和基本语法

一、下载安装

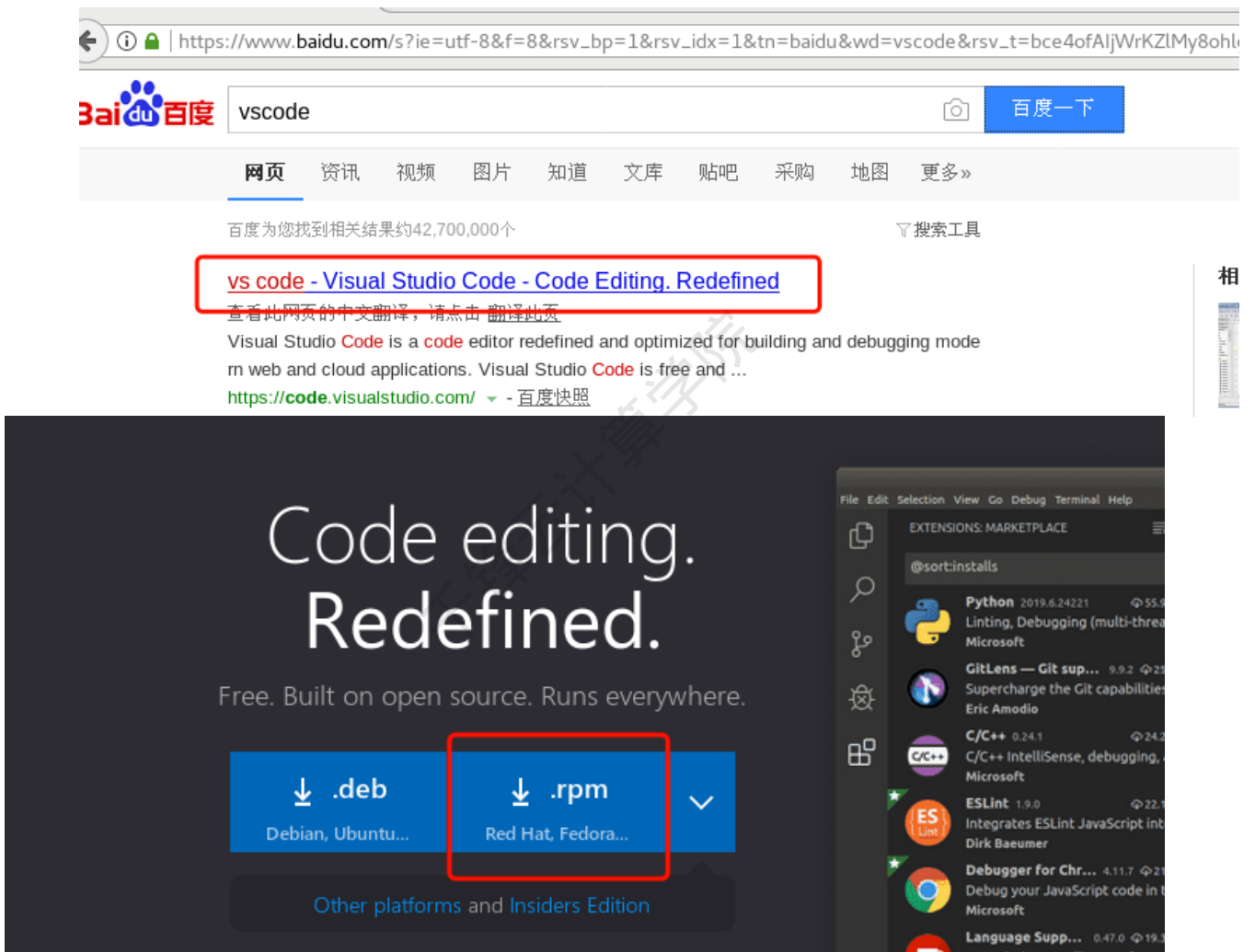
1. 打开浏览器



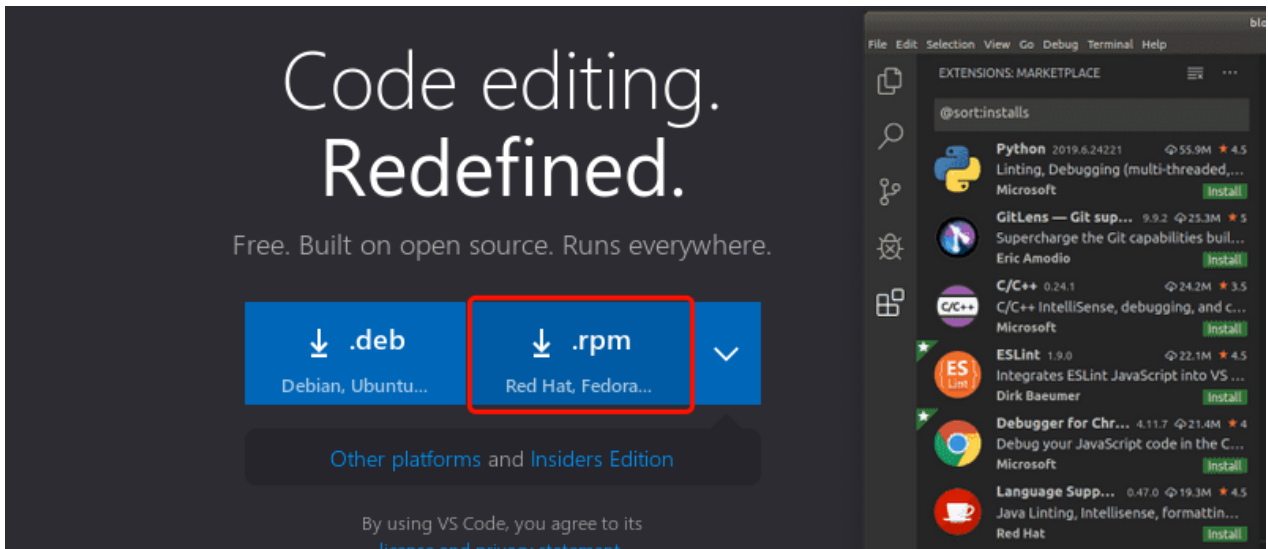
2. 输入百度的网址并搜索 `vscode`



3. 打开官网



4. 点击



5. 按照提示安装即可

RHEL, Fedora, and CentOS based distributions

We currently ship the stable 64-bit VS Code in a yum repository, the following script will install the key and repository:

```
sudo rpm --import https://packages.microsoft.com/keys/microsoft.asc
sudo sh -c 'echo -e "[code]\nname=Visual Studio Code\nbaseurl=https://packages.microsoft.com/yumrepos/vscode\nenabled=1\nngpgcheck=1\nngpgkey=https://packages.microsoft.com/keys/microsoft.asc" > /etc/yum.repos.d/vscode.repo'
```

Then update the package cache and install the package using **dnf** (Fedora 22 and above):

```
sudo dnf check-update
sudo dnf install code
```

Or on older versions using **yum**:

```
yum check-update
sudo yum install code
```

Due to the manual signing process and the system we use to publish, the yum repo may lag behind and not get the latest version of VS Code immediately.

也可以右键 打开终端，输入如下命令进行安装

```

1 [root@qfedu.com ~] sudo rpm --import
  https://packages.microsoft.com/keys/microsoft.asc
2 [root@qfedu.com ~]# sh -c 'echo -e "
  [code]\nname=Visual Studio
  Code\nbaseurl=https://packages.microsoft.com\
3 /yumrepos/vscode\nenabled=1\nngpgcheck=1\nngpgkey=\
4 https://packages.microsoft.com/keys/microsoft.asc" >
  /etc/yum.repos.d/vscode.repo'
5 [root@qfedu.com ~]# yum check-update
6 [root@qfedu.com ~]# yum install code

```

6. 添加桌面快捷方式

在终端中输入如下命令

```

1 [root@qfedu.com ~]# cp
  /usr/share/applications/code.desktop ~/桌面/
2 或者
3 [root@qfedu.com ~]# cp
  /usr/share/applications/code.desktop ~/Desktop/
4

```

7. 接下来双击桌面快捷方式，再点击 标记为信任

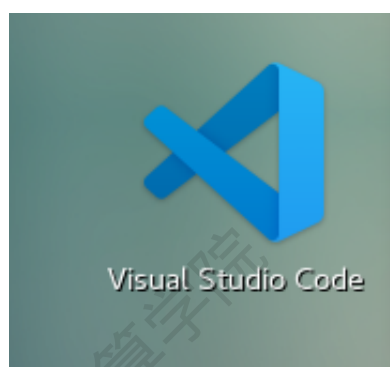


8. 完成

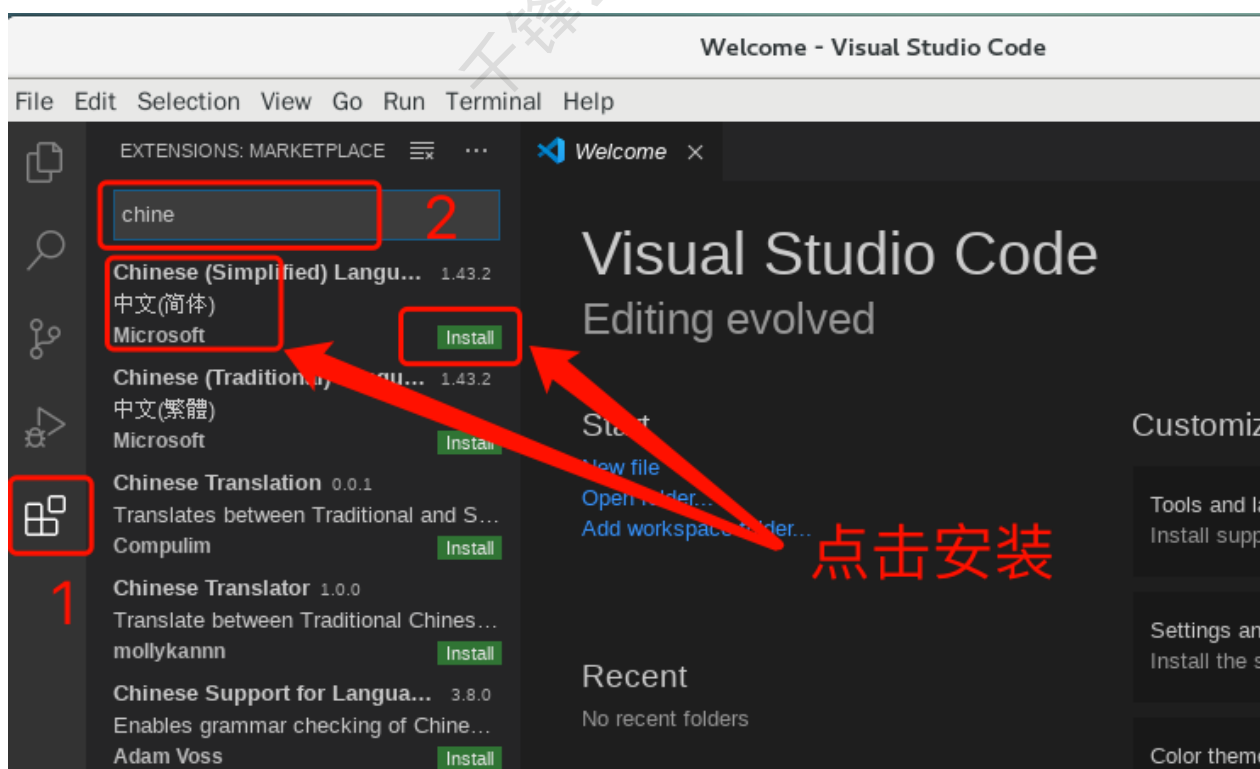


二、安装插件

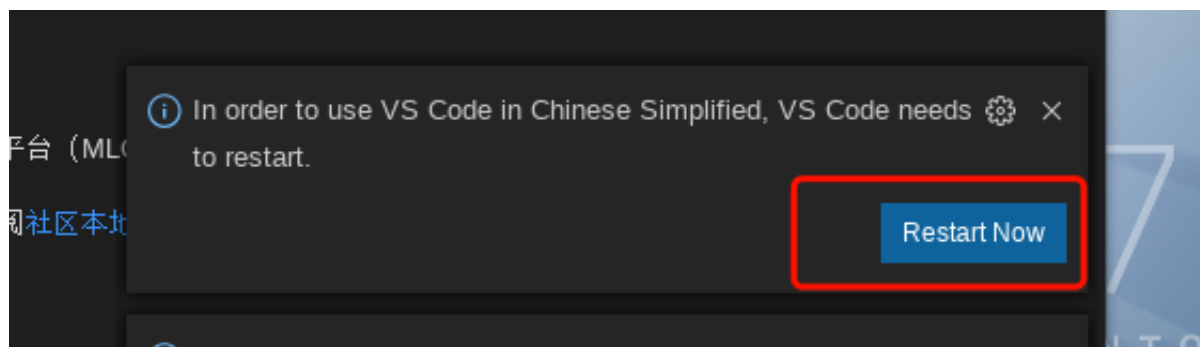
1. 双击桌面图标打开软件



2. 打开商店，安装中文简体插件



3. 重启软件生效



汉化完成



4. 安装文档图标

5. 接下来会自动提示设置图标主题

三、基本设置

打开配置文件方式

添加如下内容到 `settings.json` 配置文件中,并按下 `Control + S` 保存

```
1 {
2     "editor.fontSize": 30,
3     "debug.console.fontSize": 30,
4     "markdown.preview.fontSize": 26,
5     "terminal.integrated.fontSize": 30,
6     // 编辑的文件 每 1 秒自动保存一次
7     "files.autoSave": "afterDelay"
8 }
```

完成后的样子

鼠标悬停到某一行，会自动显示当前配置项的意义

四、创建一个 python 文件

创建的文件需要有对应的扩展名，有了对应的扩展名，VSCode 就会自动识别，并提示安装对应的插件进行支持。

比如下面创建一个 python 的文件的步骤

1. 可以在当前用户的家目录下，创建一个文件夹 `python_code`

```
1 [root@qfedu.com ~]# mkdir python_code
```

2. 接下来在 VSCode 中打开这个文件夹

目前打开的这个目录就是一个项目的主目录，项目所有的代码都应该放到这个目录下，可以在这个目录下再创建其他目录（稍后介绍）。

重要：在VSCode 看来，这个目录就是当前的工作目录，假设在 `vscode` 中执行的代码的相对路径，就是相对于这个目录的，而不会是这个目录下面的其他目录。

3. 在项目目录中创建目录和文件

鼠标点击项目目录会出现创建文档的图标

4. 点击创建的目录，选择创建文件

5. 安装对应的插件

VSCode 会自动提示安装对应的插件

6. 配置默认的 python 解释器

vscode 默认选择的python 解释器是 python2.7，我们需要的是 python3

7. 接下来会提示安装用于 python 语法检测的插件 `pylint`

这个插件会突出显示了Python源代码中的语法和风格问题，通常可以帮助您识别和纠正细微的编程错误或可能导致错误的非常规编码实践。例如，检测未使用或未定义的变量的使用，对未定义函数的调用，缺少括号以及甚至更细微的问题，例如尝试重新定义内置类型或函数。因此，与[Formatting](#)不同，因为这个插件分析代码的运行方式并检测错误，而[Formatting](#)仅重组代码的显示方式。

五、编辑代码并运行

运行的输出结果

六、安装 Code Runner

刚刚的执行方式有点小缺点，当你重复执行这个文件的时候，之前的输出在终端中不能清除，这可能会影响观看。使用 `code runner` 插件可以解决，并且它支持多重语言。

1. 安装 Code Runner

2. 配置 Run Code

再次打开配置文件 `settings.json` 文件添加如下内容，并按下 `Control + s` 保存

```
1      "code-runner.clearPreviousOutput": true,  
2      "code-runner.runInTerminal": true,  
3      // 每次运行文件之前，保存当前文件  
4      "code-runner.saveFileBeforeRun": true,  
5      // 默认的语言  
6      "code-runner.defaultLanguage": "python",  
7      // 每次运行文件前 保存所有的文件  
8      "code-runner.saveAllFilesBeforeRun": true,  
9      // 设置 python3 解释器路径  
10     "code-runner.executorMap": {  
11         "python": "/usr/local/bin/python3"  
12     }
```

七、最终的配置文件

```
1  {  
2      "workbench.iconTheme": "vscode-icons",  
3      "editor.fontSize": 30,  
4      "debug.console.fontSize": 30,
```

```
5     "markdown.preview.fontSize": 26,  
6     "terminal.integrated.fontSize": 30,  
7     // 编辑的文件 每 1 秒自动保存一次  
8     "files.autoSave": "afterDelay",  
9     "code-runner.clearPreviousOutput": true,  
10    "code-runner.runInTerminal": true,  
11    "code-runner.saveFileBeforeRun": true,  
12    "code-runner.defaultLanguage": "python",  
13    "code-runner.saveAllFilesBeforeRun": true,  
14    // set python path  
15    "code-runner.executorMap": {  
16        "python": "/usr/local/bin/python3"  
17    }  
18 }
```

八、部分快捷键

快速添加注释

鼠标点击某一行，同时按下 `Ctrl` 和 `?`，重复按下取消注释

快速复制一行

通用鼠标选中需要复制的行，之后先按住 `Ctrl` 和 `Shift` 不放，再按下方向键，会向下自动复制被选中的内容。