

配置本地运行开发环境

学习目的

当前市面上有很多Python运行的环境配置方案，经对比我们要求学员使用Anaconda搭建运行开发Python的环境。

根据 [Python 官方信息](#)，Python2 将在2020年不再被维护。我们要求学员统一学习 Python3，在本次学习过程中将不解答任何关于Python 2.x 相关的问题。

自学任务

1. 下载安装 Anaconda，并使用Anaconda安装 Visual Studio Code 和 Jupyter Notebook

2. 通过 [下载地址](#) 在本地电脑安装完成 Anaconda
3. 打开 Anaconda 的可视化操作界面，找到并安装 Visual Studio Code (vscode)，Jupyter Notebook 和 JupyterLab

2. 在 vscode 中配置 Python 环境

1. 阅读参考资料1 联系微信客服501863613进同步更新群
2. 在 vscode 中配置 Python 环境（有语法检查、代码提示等辅助开发的功能）

3. 编写 hello world python 程序

1. 请自行 Google 或其他搜索引擎学习如何编写一个 Python 的 helloworld 程序
2. 在本地仓库中（selfteaching-python-camp）自己专属的文件夹里，创建名字为 d2_exercise_hello_python.py 的 Python文件，编写一个可以打印出 hello world 的 Python程序
3. 将本地仓库关于本次作业的变更提交为一个 commit
4. 通过 Github 桌面客户端将本地电脑的变更推送到自己账户下的作业仓库
5. 回到 Github 页面，在自己账户下的作业仓库页面向远程公用作业仓库的 master 分支发起 Pull Request, 在提交的 Pull Request 中 @自己的教练 提醒他检查作业

4. 使用 Jupyter Notebook 阅读李笑来老师的新书《自学是门手艺》

1. 在 Github 上 fork [《自学是门手艺》by 李笑来](#)，然后克隆自己 fork 的仓库到本地电脑
2. 在 Anaconda 可视化界面中启动 JupyterLab，在 Files 目录下找到本地电脑新书的位置，开始阅读

- ## 目录

参考资料

1. [VSCode 官方配置 Python 环境教程](#)
2. [JupyterLab 官方关于操作界面的介绍](#)

拓展学习 (可选)

- ## 1. 了解 Python 的 Package 管理工具

