

- 学前准备——课程代码
  - 克隆课程到本地（1 个教学代码，1 个完整演示项目）
  - 检查课程代码是否就绪
    - 配置 API 秘钥
    - 利用 VS Code Terminal 和 pip 安装依赖包
    - 运行 Hello World

## 学前准备——课程代码

---

### 克隆课程到本地（1 个教学代码，1 个完整演示项目）

---

1. 打开 GitHub Desktop。
2. 点击左上角菜单 **File > Clone Repository...**。
3. 在弹出的窗口中选择 **URL** 选项卡。
4. 输入第一个项目的仓库地址：

```
https://github.com/MartinChen1973/111m
```

5. 在 **Local Path** 中选择保存项目的本地路径。
6. 点击 **Clone** 按钮开始克隆项目。

重复上述步骤，输入第二个项目的仓库地址：

```
https://github.com/MartinChen1973/LangStart
```

完成克隆。

### 检查课程代码是否就绪

---

在拿到课程代码后，请做以下检查活动。

# 配置 API 秘钥

在 src 目录下有一个.env\_ 文件，将其命名为.env（没有文件名，只有后缀），双击打开后，里边应该有类似如下的内容：

```
OPENAI_API_BASE="https://api.agicto.cn/v1"
OPENAI_API_KEY="sk-一个公开共享但也因此经常欠费甚至失效的秘钥（也可以替换成自己的秘钥）"
TAVILY_API_KEY="tvly-请输入自己前面注册Tavily获得的KEY"
```

## 利用 VS Code Terminal 和 pip 安装依赖包

为运行项目或课程中的示例代码，可能需要安装特定的依赖包。可以使用 `pip` 命令安装这些包。在安装依赖包前，请确保已经在 VS Code 的 Terminal 中启动虚拟环境（`venv`），以避免对全局 Python 环境造成影响。

- **启用虚拟环境：** 在 VS Code Terminal 中，确保在当前项目根目录下（一般有一个 README.md）。运行

```
python -m venv venv
.\venv\Scripts\activate
pip install -r requirements.txt
```

此时会发现提示行的最前面有一个蓝色的（venv），则表示成功。 **⚠ 注意 ⚠**：每次启动项目都需要重新运行第二行 `.\venv\Scripts\activate`。

- **安装单个依赖包（选）：** 如果运行项目时提示某个模块找不到（例如 `langchain`），可以单独安装该模块：

```
pip install langchain
```

## 运行 Hello World

- 双击进入第一课的 lesson01\_Get\_Started/hello\_world.py
- 点击顶端 tab 页右侧的三角符号，运行代码。
- 如果能看到类似“Hello World!”的输出，即表明环境正常。

- 如果收到欠费错误（**402 Payment Required** 或其他类似信息），请按其他教程准备秘钥。
- 如果出现 **ModuleNotFoundError: No module named 'dotenv'** 之类的错误，可能是：
  - 没有所需依赖包，请参考 [预装依赖包](#)
  - 配置了错误的 python 解释器，请参考[配置-python-解释器](#)