

pyOBD 项目安装和配置指南

pyobd

open source obd2 car diagnostics program - reuploaded

 项目地址: https://gitcode.com/gh_mirrors/py/pyobd

1. 项目基础介绍和主要编程语言

项目基础介绍

pyOBD 是一个开源的汽车诊断工具，符合 OBD-II 标准。它允许用户通过低成本的 ELM 32x OBD-II 诊断接口与车辆的 ECU（发动机控制单元）进行通信，从而显示故障代码、测量值、状态测试等。该项目最初由 Donour Sizemore 开发，现由 barracuda-fsh 维护和改进。

主要编程语言

pyOBD 项目主要使用 Python 编程语言开发。

2. 项目使用的关键技术和框架

关键技术

- **OBD-II 协议**：用于与车辆的 ECU 进行通信。
- **ELM 32x 接口**：低成本的 OBD-II 诊断接口，支持多种通信协议。
- **Python-OBD 库**：用于支持更多的 OBD-II 命令和自动检测车辆支持的功能。

框架

- **Python 3**：项目升级到 Python 3，以支持现代库和功能。

3. 项目安装和配置的准备工作和详细安装步骤

准备工作

1. 硬件要求：
 1. 一台支持 OBD-II 的车辆（美国 1996 年后，欧洲 2001 年后生产的车辆）。
 2. 一个 ELM327 适配器（推荐使用 USB 或蓝牙适配器）。
 3. 一台笔记本电脑或台式机。
2. 软件要求：
 1. Python 3.x 安装（建议使用最新版本）。
 2. 操作系统：Windows、Linux 或 MacOS。

详细安装步骤

1. 下载项目代码

首先，从 GitHub 仓库下载 pyOBD 项目的代码：

```
1 | git clone https://github.com/barracuda-fsh/pyobd.git
2 | cd pyobd
```

2. 安装 Python 3

确保你的系统上已经安装了 Python 3。如果没有安装，可以从 [Python 官方网站](#) 下载并安装。

3. 安装依赖库

在项目目录下，使用以下命令安装所需的 Python 依赖库：

```
pip install -r requirements.txt
```

4. 配置 ELM327 适配器

• **Windows：**

- 下载并安装 ELM327 适配器的驱动程序。
- 确保适配器通过 USB 连接到电脑，并识别为 COM 端口。

• **Linux：**

- 确保你的用户有权限访问 USB 和串口设备：

```
1 | sudo usermod -a -G dialout $USER
2 | sudo usermod -a -G tty $USER
```

- 如果使用蓝牙适配器，需要手动配对并连接：

```
sudo apt-get install bluetooth bluez-utils blueman
```

• **MacOS：**

- 确保你的用户有权限访问 USB 和串口设备：

```
1 | sudo usermod -a -G dialout $USER
2 | sudo usermod -a -G tty $USER
```

5. 运行 pyOBD

在项目目录下，运行以下命令启动 pyOBD：

```
python3 pyobd.py
```

6. 连接到车辆

- 打开车辆的点火开关（不需要启动发动机）。
- 在 pyOBD 应用程序中，选择正确的端口和波特率，然后点击“连接”。
- 如果一切正常，你将看到车辆的实时数据。

注意事项

- 确保 ELM327 适配器与车辆的 OBD-II 端口正确连接。
- 如果使用蓝牙适配器，在 Linux 上可能需要手动配对和连接。
- 如果遇到连接问题，尝试使用手动配置端口和波特率。

通过以上步骤，你应该能够成功安装和配置 pyOBD 项目，并开始使用它进行车辆诊断。