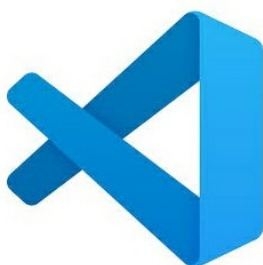
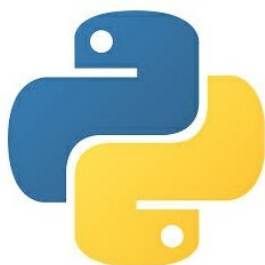


PyLab 1 : Raspberry Pi , 调试Python物联网应用程序

gloveboxes.github.io/PyLab-1-Debugging-a-Python-Internet-of-Things-Application

在Twitter [@dglover](#)上关注我



PDF实验指南

您可能会发现下载和遵循PDF版本的[调试Raspberry Pi物联网Flask App](#)动手实验室指南更容易。

PyLab内容

- [PyLab 1 : Raspberry Pi , 调试Python物联网应用程序](#)
- [PyLab 2 : Raspberry Pi , Azure物联网中心和Docker容器调试](#)

PyLab设置

- [多用户设置](#)
- [单用户设置](#)

作者 微软云开发者倡导者 [Dave Glover](#)

平台 Linux , macOS , Windows , Raspbian Buster

服务 [Azure物联网中心](#)

工具 [Visual Studio](#)[代码](#)

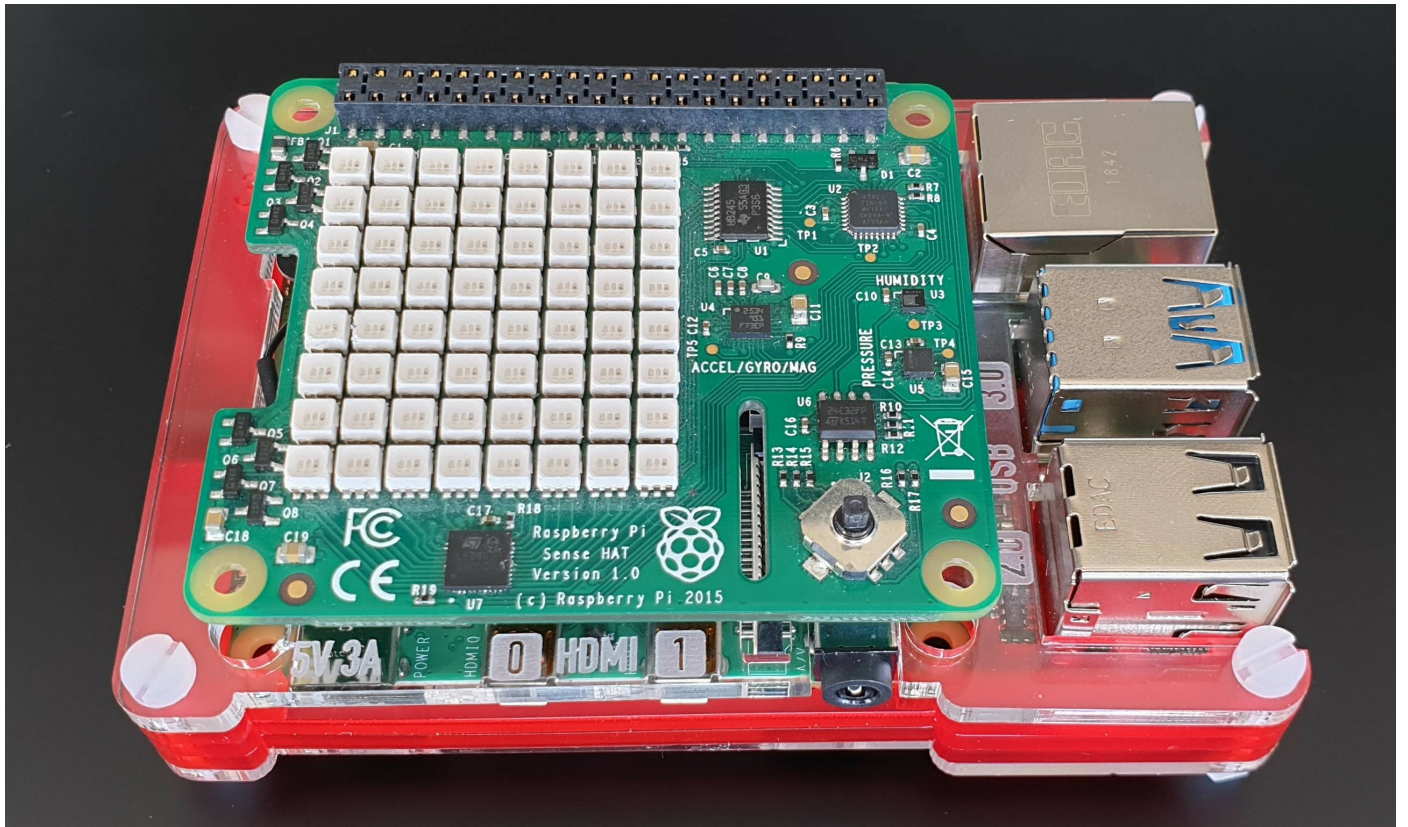
硬件 [Raspberry Pi](#) , [Raspberry Pi Sense HAT](#)

语言 蟒蛇

日期 2019年9月

介绍

在本动手实验中，您将学习如何使用[Visual Studio Code](#)和[Remote SSH](#)扩展在Raspberry Pi上创建和调试Python Web应用程序。Web应用程序将从连接到Raspberry Pi的传感器读取温度，湿度和气压遥测。



软件安装



这个动手实验室使用Visual Studio Code。Visual Studio Code是一个代码编辑器，是[GitHub](https://github.com)上最受欢迎的开源项目之一。它可以在Linux，macOS和Windows上运行。

安装Visual Studio代码

1. 安装[Visual Studio代码](#)

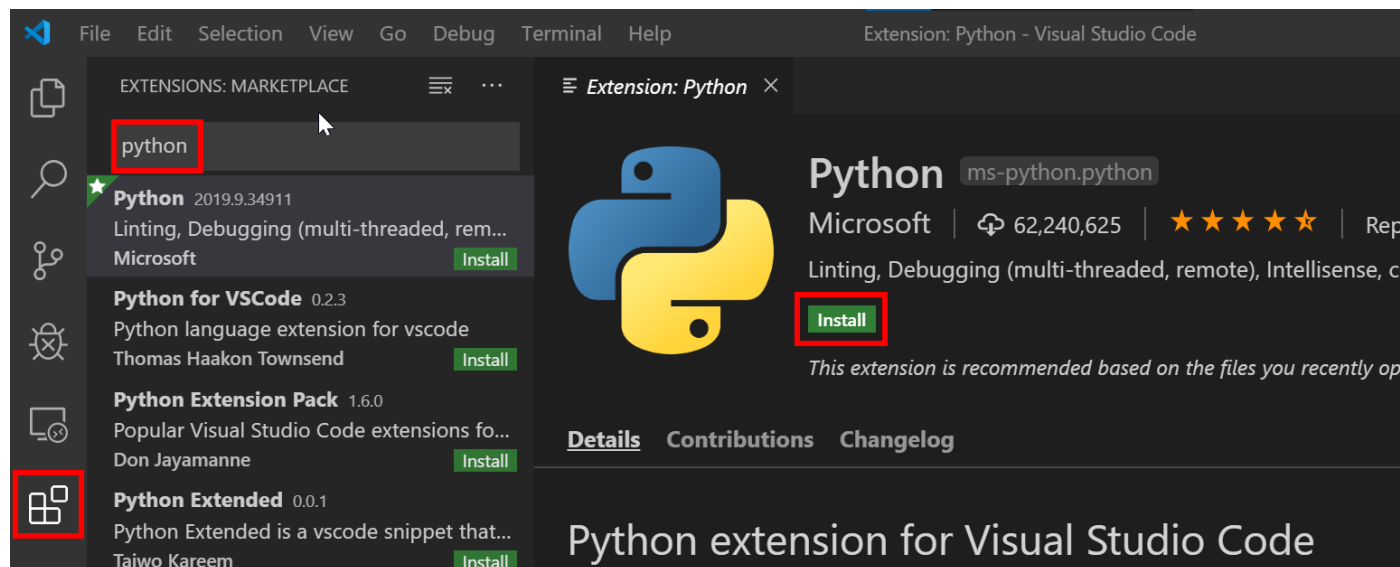
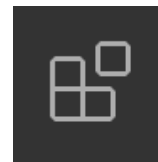
Visual Studio代码扩展

Visual Studio Code包含开箱即用的功能只是一个开始。VS代码扩展允许您为安装添加语言，调试器和工具，以支持您的开发工作流程。

浏览扩展程序

您可以从Visual Studio Code中搜索和安装扩展。从Visual Studio代码主菜单中打开Extensions视图，选择**View > Extensions**或单击Visual Studio Code一侧活动栏中的Extensions图标。

这将显示[VS Code Marketplace](#)上最受欢迎的VS Code扩展的列表。



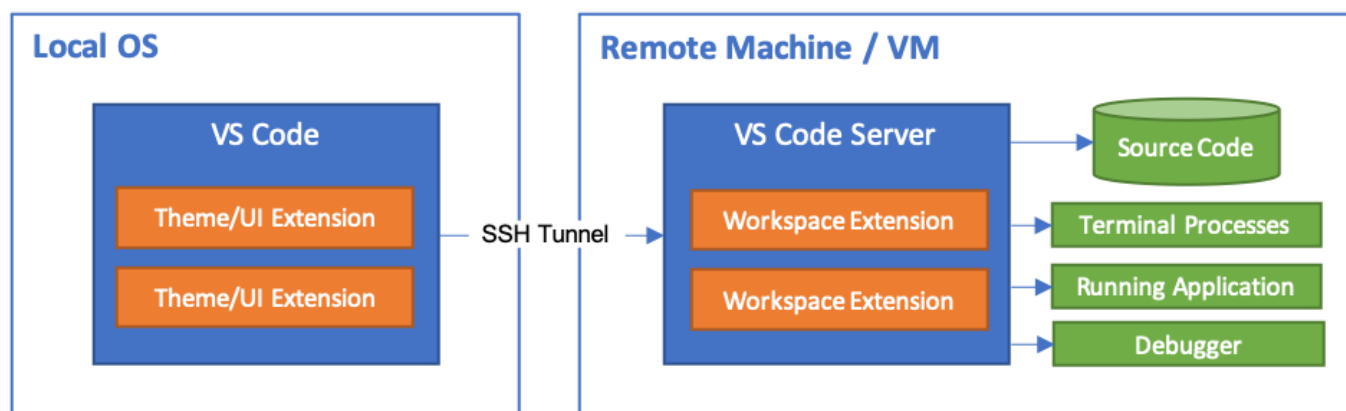
安装Python和远程SSH扩展

搜索并安装Microsoft发布的以下两个Visual Studio代码扩展。

1. [蟒蛇](#)
2. [远程 - SSH](#)

远程SSH开发

Visual Studio Code Remote - SSH扩展允许您使用正在运行的SSH服务器在任何远程计算机，虚拟机或容器上打开远程文件夹，并充分利用Visual Studio Code。



Raspberry Pi硬件

如果您正在参加研讨会，那么您可以使用共享的网络连接Raspberry Pi。您也可以使用自己的网络连接Raspberry Pi进行动手实验。

您将需要实验室指导员提供以下信息。

1. Raspberry Pi 的网络IP地址
2. 您指定的登录名和密码。

使用私钥/公钥进行SSH身份验证



为SSH身份验证设置公钥/私钥对是一种安全，快速的方法，可以从您的计算机进行身份验证到Raspberry Pi。建议用于此动手实验。

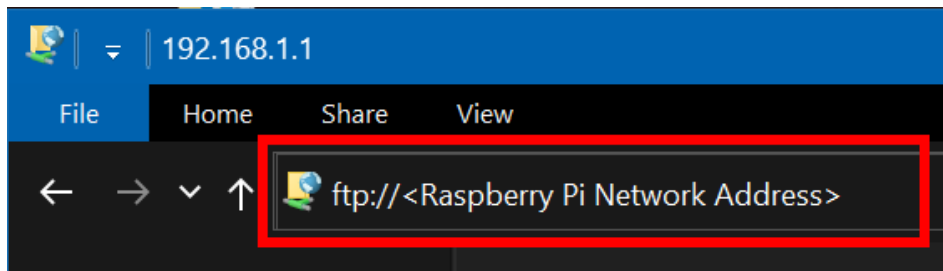
SSH为Windows用户设置

SSH实用程序将指导您完成成为Visual Studio Code和Raspberry Pi设置安全SSH通道的过程。

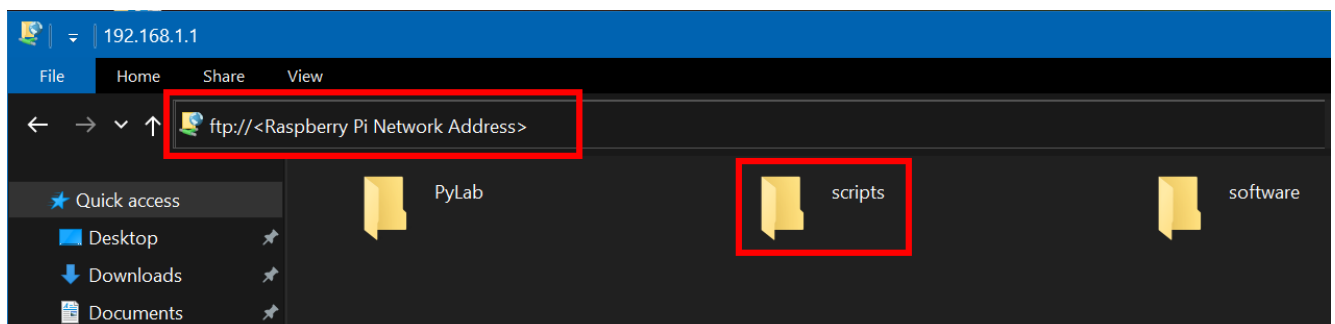
系统将提示您：

- Raspberry Pi网络IP地址，
- Raspberry Pi登录名和密码

1. 从Windows文件资源管理器中，打开ftp://<Raspberry Pi地址>



2. 将脚本目录复制到桌面



3. 打开复制到桌面的脚本文件夹
4. 双击windows-setup-ssh.cmd

SSH为Linux和macOS用户设置

SSH实用程序将指导您完成成为Visual Studio Code和Raspberry Pi设置安全SSH通道的过程

系统将提示您：

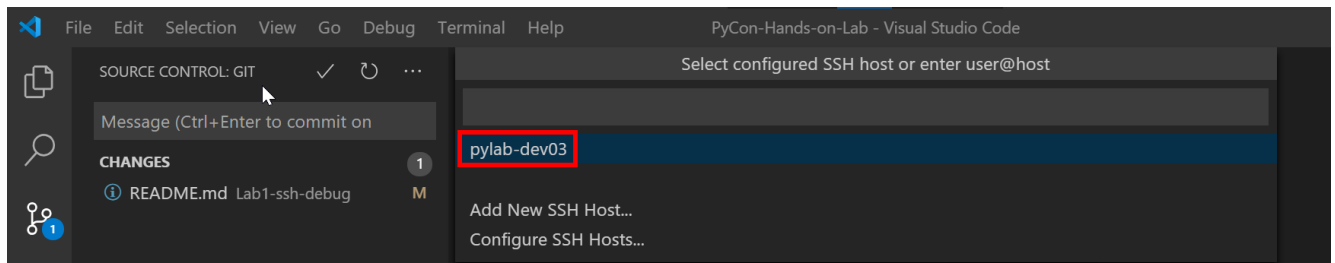
- Raspberry Pi网络IP地址，
- Raspberry Pi登录名和密码

1. 打开终端窗口
2. 复制并粘贴以下命令，然后按ENTER键

```
read -p "Enter the Raspberry Pi Address: " pyurl && \  
curl ftp://$pyurl/scripts/ssh-setup.sh | bash
```

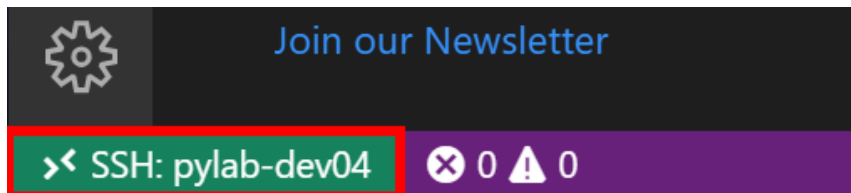
启动远程SSH连接

1. 启动Visual Studio代码
2. 按F1打开命令选项板，键入ssh connect并选择Remote-SSH：Connect to Host
3. 选择pylab-devnn配置



4. 检查远程SSH是否已连接。

连接需要一些时间，然后Visual Studio Code左下角的SSH Status将更改为> <SSH：pylab-devnn。devnn是你的Raspberry Pi登录名。

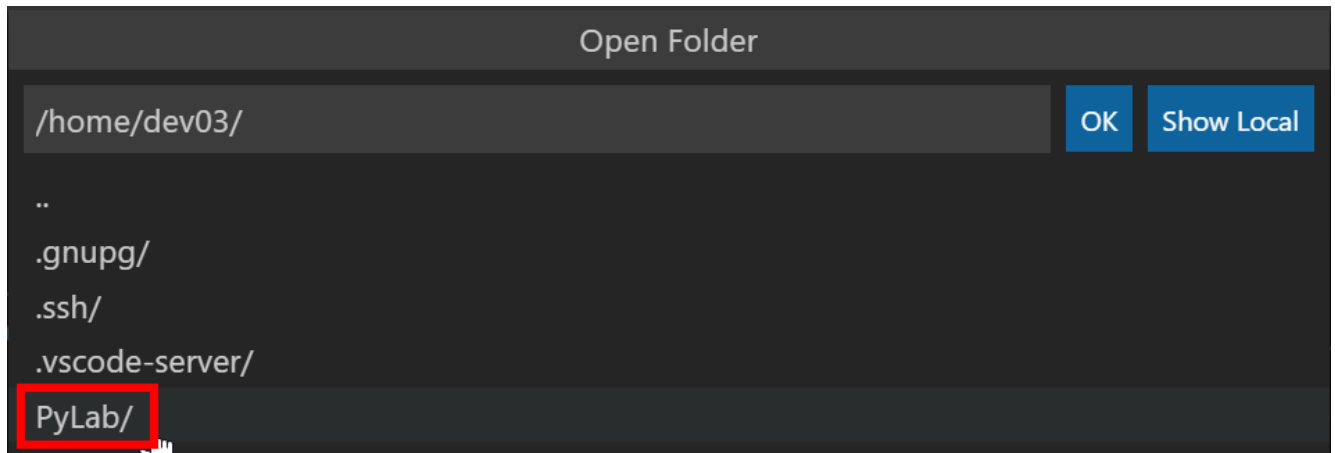


打开PyLab 1 Python调试项目

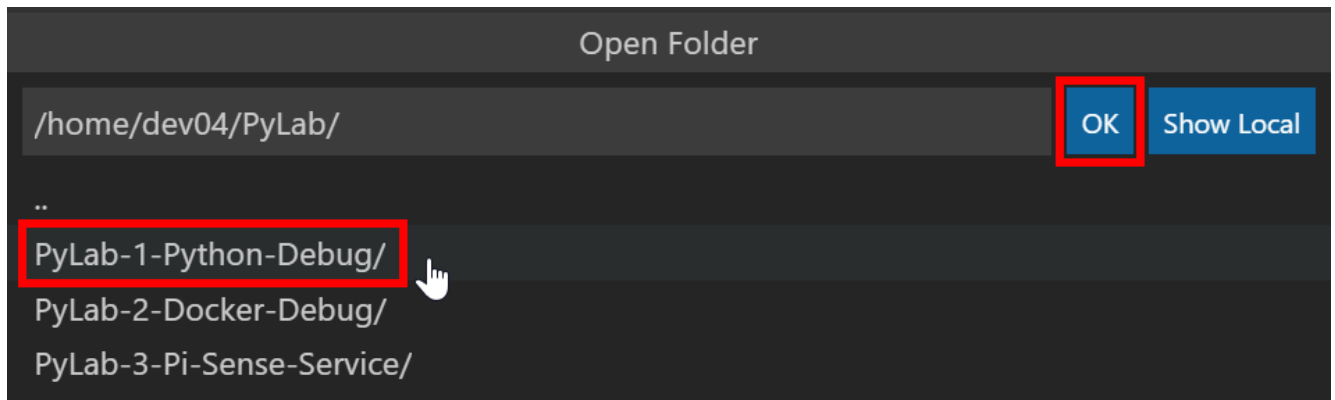
Python Flask Web应用程序

在本实验中，我们将启动并调试Flask应用程序，该应用程序读取连接到Raspberry Pi的传感器。Flask是一个流行的Python Web框架，功能强大，但对初学者也很容易。

1. 从Visual Studio代码主菜单：文件 > 打开文件夹
2. 选择PyLab目录

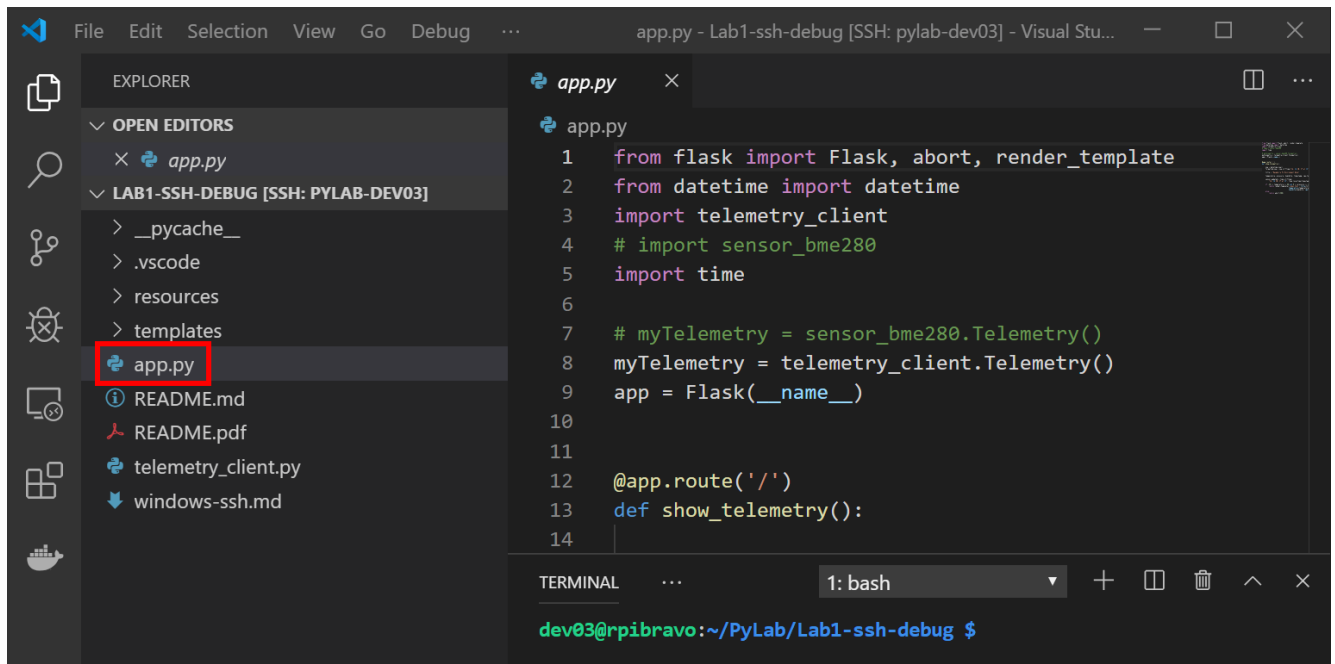


3. 接下来选择PyLab-1-Python-Debug目录



4. 单击“确定”以打开目录

5. 在资源管理器栏中，打开app.py文件并查看内容



花一点时间来查看Python Flask Web应用程序。

应用的.py

```
from flask import Flask, abort, render_template
from datetime import datetime
import telemetry_client
# import sensor_bme280
import time

# myTelemetry = sensor_bme280.Telemetry()
myTelemetry = telemetry_client.Telemetry()
app = Flask(__name__)

@app.route('/')
def show_telemetry():

    now = datetime.now()
    formatted_now = now.strftime("%A, %d %B, %Y at %X")

    title = "Raspberry Pi Environment Data"

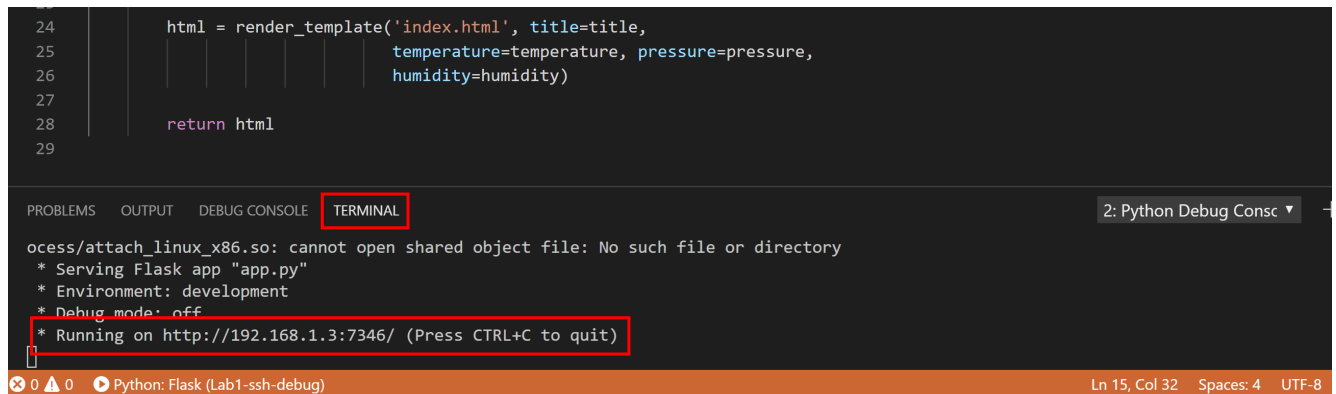
    temperature, pressure, humidity, timestamp, cpu_temperature = myTelemetry.measure()

    sensor_updated = time.strftime(
        "%A, %d %B, %Y at %X", time.localtime(timestamp))

    if -40 <= temperature <= 60 and 0 <= pressure <= 1500 and 0 <= humidity <= 100:
        return render_template('index.html', title=title,
                               temperature=temperature, pressure=pressure,
                               humidity=humidity, cputemperature=cpu_temperature)
    else:
        return abort(500)
```

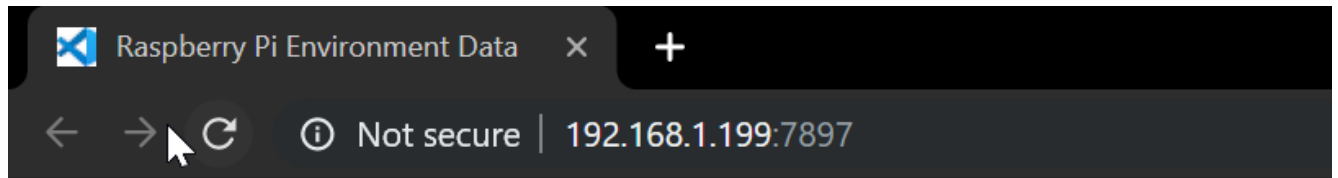
启动Python Flask App

1. 按F5启动Python Flask应用程序。
2. 在Visual Studio代码终端窗口中，单击在<http://...> web链接上运行。



The screenshot shows the Visual Studio Code interface. The top part displays the Python code for the Flask application, specifically the `show_telemetry` function. The bottom part shows the terminal output, which includes the message `* Running on http://192.168.1.3:7346/ (Press CTRL+C to quit)`. The terminal window is titled "2: Python Debug Consc" and the status bar at the bottom indicates "Python: Flask (Lab1-ssh-debug)".

3. 这将启动您的桌面Web浏览器。
 - Flask应用程序将读取连接Raspberry Pi 的传感器的温度，气压，湿度，并在Web浏览器中显示结果。



Raspberry Pi Environment Data

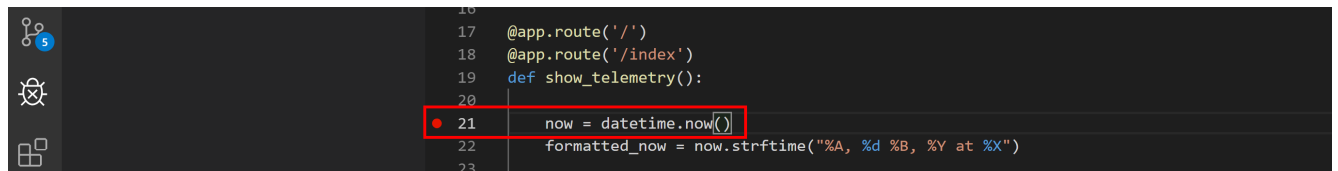
Raspberry Pi Sense HAT



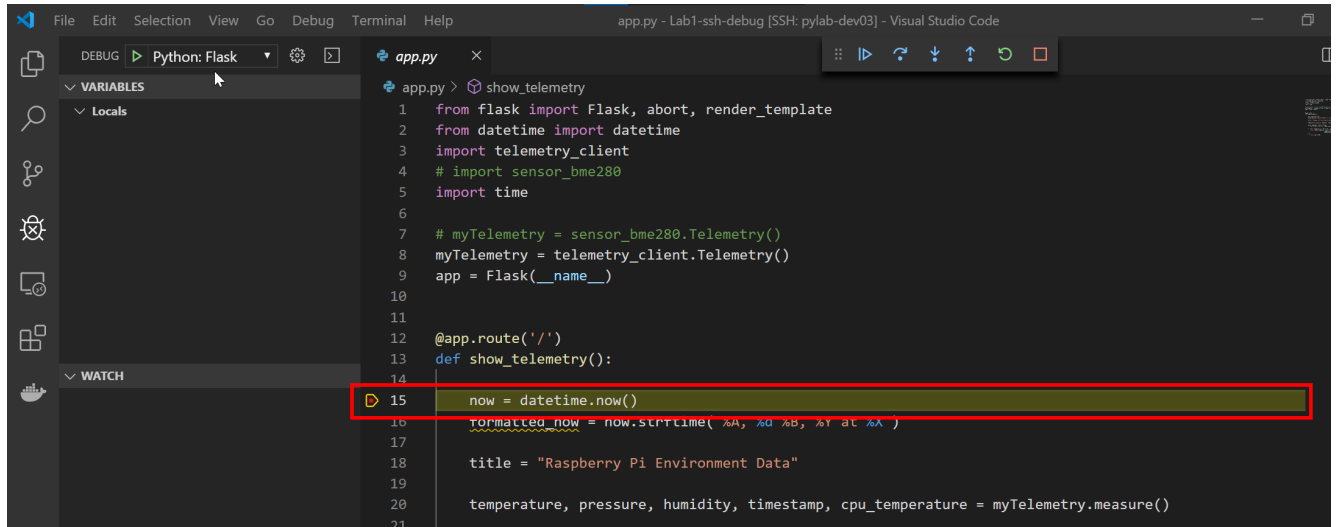
Telemetry	Value
Temperature	36.3 C
Humidity	22 %
Pressure	1020 hPa
CPU Temperature	57.9 C

使用断点进行调试

1. 切换回Visual Studio代码并确保app.py文件已打开。
2. 将光标放在现在读取的行= `datetime.now()`
3. 按F9设置断点。线上将出现一个红点，表示已设置断点。



4. 切换回Web浏览器并单击“刷新”。由于调试器已在您设置的断点处停止，因此网页不会响应。
5. 切换回Visual Studio代码。您将看到代码已在断点处停止运行。



调试器工具栏选项

调试会话启动时，“调试”工具栏将显示在编辑器窗口的顶部。

调试工具栏（如下所示）将出现在Visual Studio Code中。它有以下选项：

1. 暂停（或继续，F5），
2. 跳过（F10）
3. 步入（F11），
4. 走出去（Shift + F11），
5. 重启（Ctrl + Shift + F5），
6. 并停止（Shift + F5）。

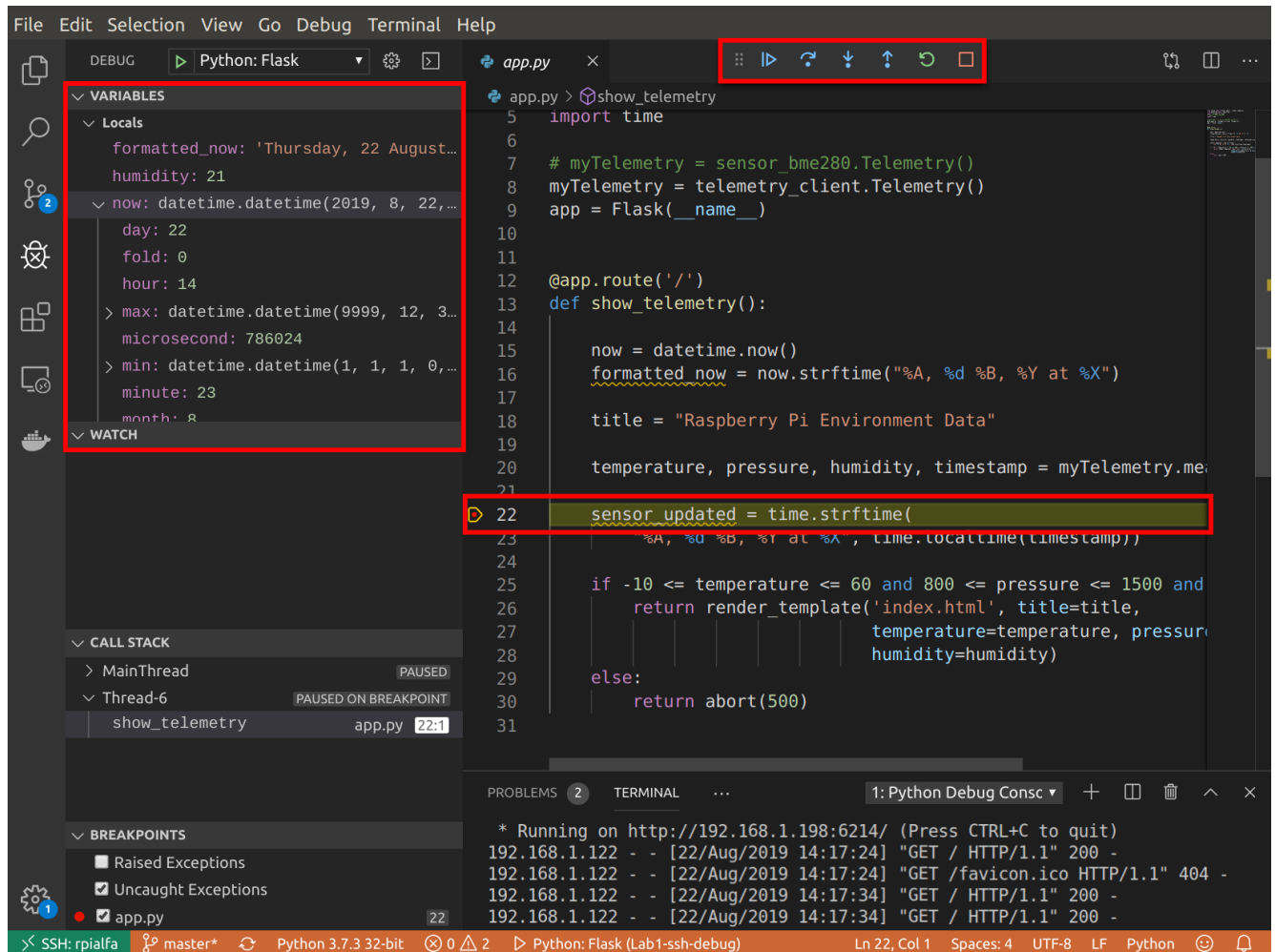


开始调试

1. 按（F10）或单击调试工具栏上的“跳过”逐步执行代码。
2. 重复按F10，直到到达读取的行，如果 $-40 \leq \text{温度} \leq 60$ 且 $0 \leq \text{压力} \leq 1500$ 且 $0 \leq \text{湿度} \leq 100$ ：
3. 您会注意到Python变量显示在变量窗口中。

如果看不到变量窗口，请单击活动栏中的“调试”。



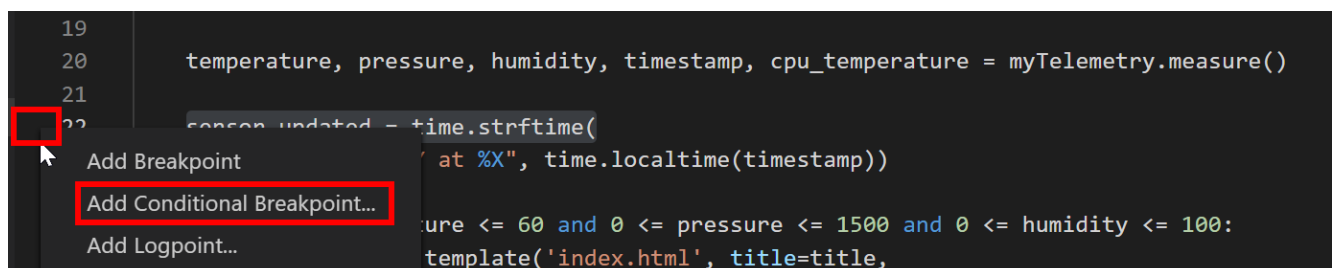


4. 尝试将温度变量更改为50。提示，右键单击温度变量并选择“设置值”，或双击温度变量。
5. 按F5恢复Flask应用程序，然后切换回Web浏览器，您将看到网页上显示的温度，湿度和压力传感器数据。

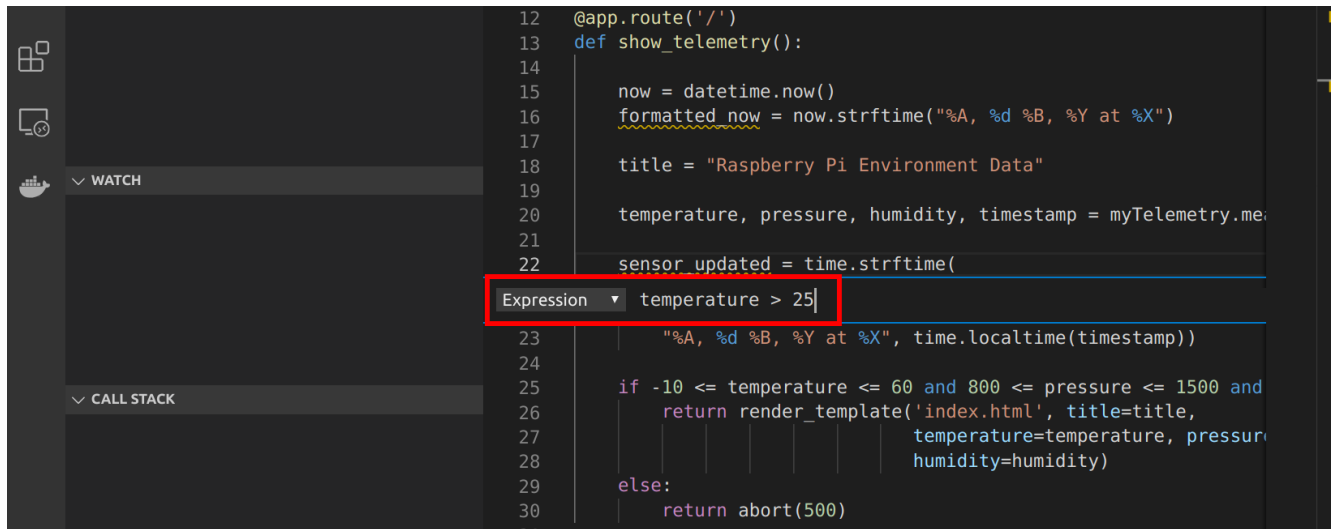
使用条件断点进行调试

尝试设置条件断点

1. 清除现有断点。从主菜单中选择Debug > Remove all breakpoints。
2. 确保打开app.py文件。
3. 右键单击第22行左侧的边距。



4. 选择添加条件断点...
5. 将条件设置为 > 25，然后按ENTER



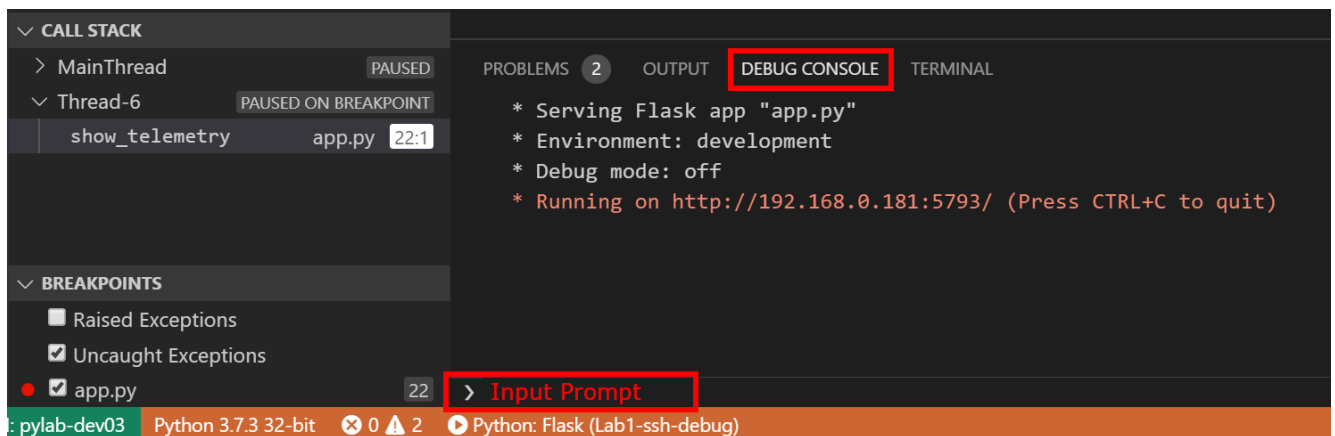
断点显示为红点，中间带有等号

6. 切换回Web浏览器并单击“刷新”。由于调试器已在您设置的断点处停止，因此网页不会响应。
7. 切换回Visual Studio代码，您将看到调试器已在条件断点处停止。
8. 按F5继续运行代码
9. 切换回Web浏览器以查看该页面。

交互式调试控制台

Visual Studio代码调试控制台将允许您访问[Python REPL](#)（读取，评估，打印循环）。

1. 切换回Web浏览器并单击“刷新”。由于调试器已停止Python代码，因此网页不会响应。
2. 切换回Visual Studio代码
3. 代码将在您先前设置的条件断点处停止。
4. 选择Visual Studio 调试控制台窗口。



5. 按Enter键以执行您键入的Python代码。
6. 尝试从输入提示符运行以下Python代码段。

```

temperature = 24
import random
random.randrange(100, 1000)

```

7. 按F5继续执行Python代码。

8. 切换回您的Web浏览器以查看更新的页面。

实验室挑战

实验室挑战1：更新Flask模板

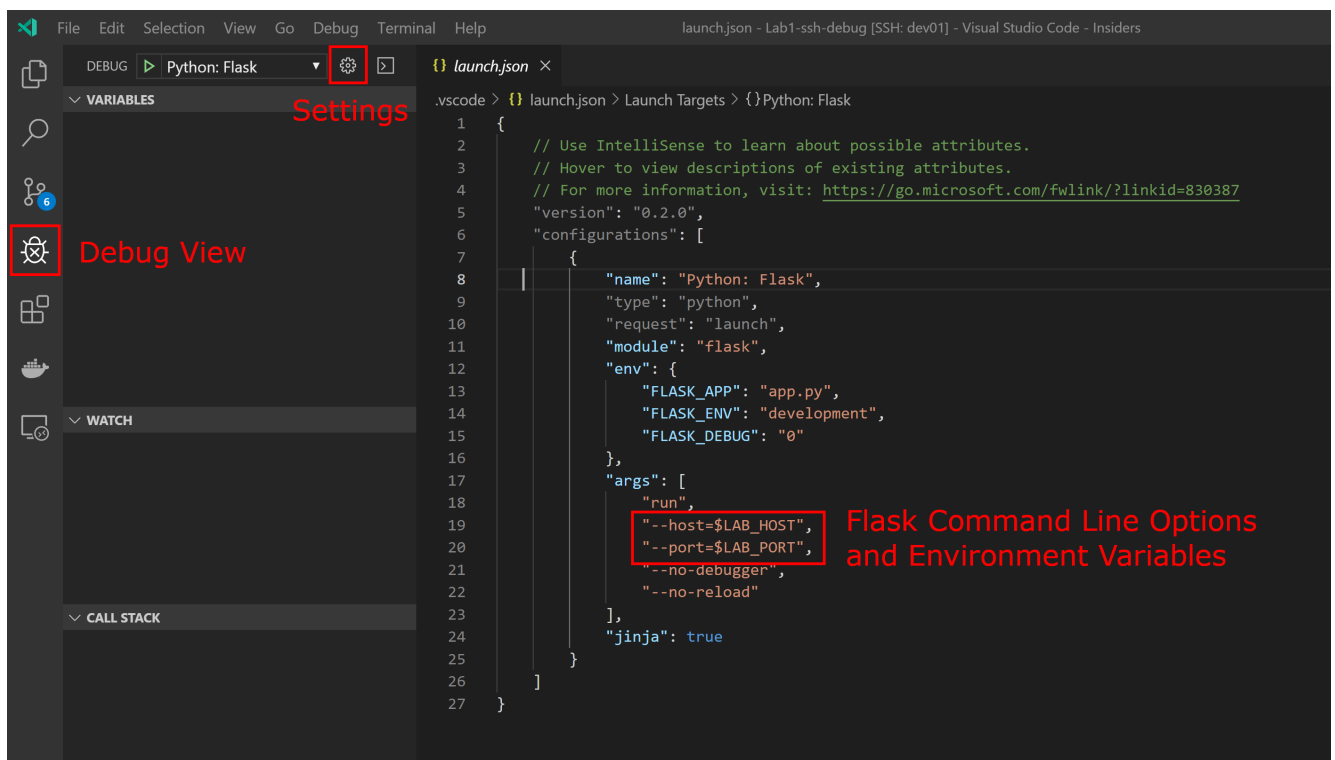
1. 更新**templates**文件夹中的Flask **index.html**模板以显示当前日期和时间。
2. 重新运行Flask应用程序。

实验室挑战2：使用调试器选项进行实验

要尝试的事情：

查看调试启动设置

1. 在Visual Studio代码中切换到Debug视图（使用左侧活动栏）。



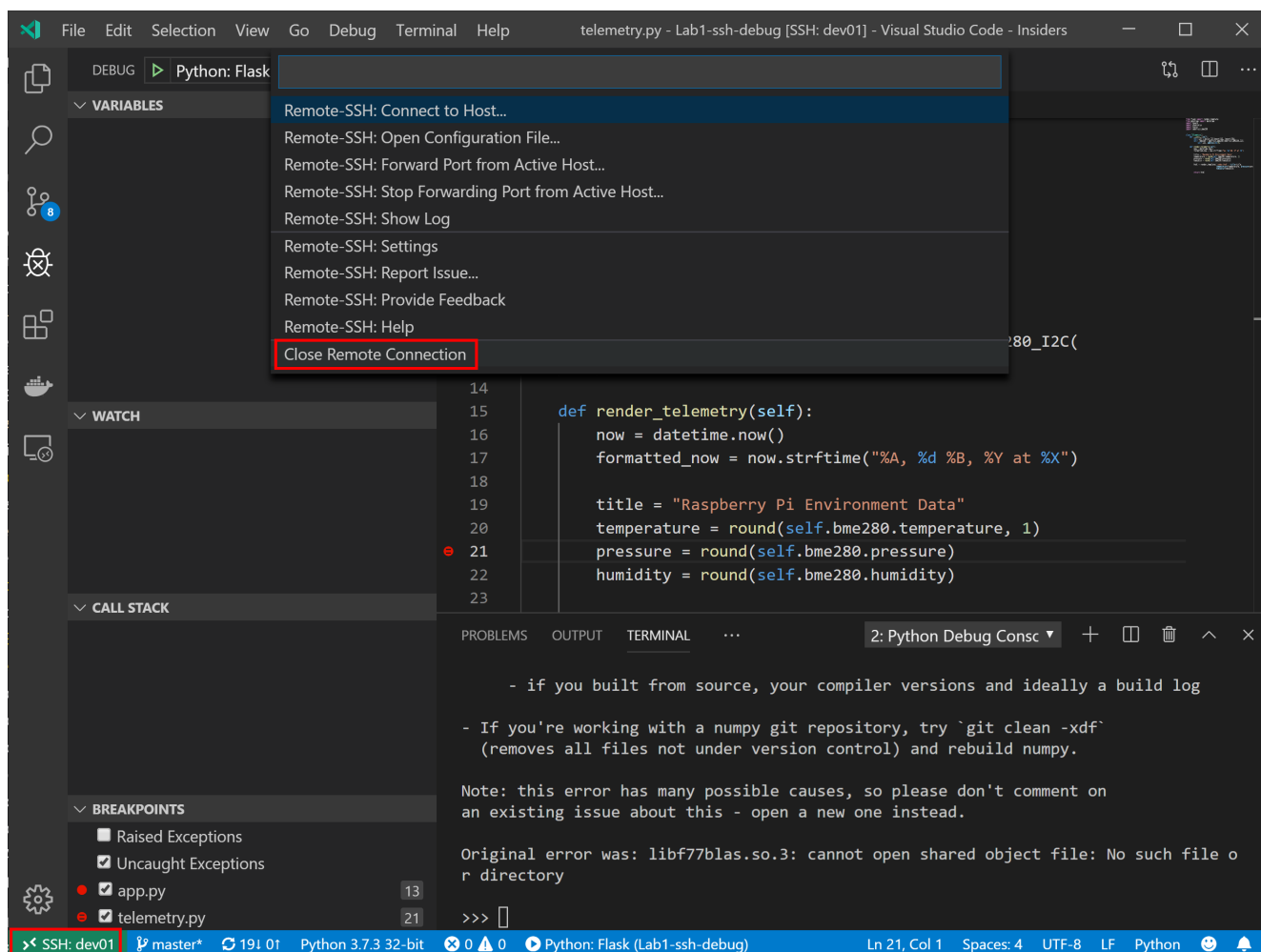
2. 单击“设置”按钮，将打开**launch.json**文件。
3. 该**launch.json**文件定义了应用程序将如何开始，什么**命令行**参数在启动时通过。

`launch.json`文件中使用了两个环境变量。这些是**LAB_HOST**（它是Raspberry Pi的IP地址）和**LAB_PORT**（5000到8000之间的随机TCP / IP端口号）。这些环境变量由**bashrc**脚本设置，该脚本在使用Visual Studio远程SSH连接到Raspberry Pi时运行。

关闭远程SSH会话

从Visual Studio代码，关闭远程连接。

1. 单击左下角的**Remote SSH**按钮，然后从下拉列表中选择**Close Remote Connection**。



成品



参考

故障排除SSH客户端安装

- [使用SSH进行远程开发](#)
- [安装支持的SSH客户端](#)

