

Dash+環境センサーで家環境ダッシュボードを作った話

PyCharity LT 2021/02/20






Hiroshi Sano

お前誰よ

Hiroshi Sano @hrs_sano645

:静岡の  見えるところ

Community

-  : shizuoka.py, unagi.py, Python駿河
-  : PyCon mini Shizuokaスタッフ
- : PyCon JP 2020 チュートリアル講師

宣伝

静岡Pythonコミュニティの勉強会 Python駿河

2/27（土）です！来週！

「Pythonプロフェッショナルプログラミング 第3版 Chapter 05 課題管理とレビュー」

つまみ食い読書します。ぜひ遊びに来てください！

参加方法はconnpass検索、#pycharityに流します

PyCon mini Shizuoka 2021

開催日まだ未定ですが、年内にできたらで

スタッフ募集中です！🙏

#pycharity 二回目 🎉

開催めでたい

LTラストバッター

めっちゃ緊張

そのLTですが

1週間前にリハーサルしてから

前日から作ろうと思っていたものの

今日の朝から作りました

このスライドの作り方は

高橋メソッドです（もはや懐かしい？）

今日言いたいことは

1. 可視化のアプリを作って思ったこと

2. Plotly Dashというダッシュボードアプリ フレームワーク便利

3. IoTとの連携はPythonが手軽

まず趣味プロジェクト、homeenvdashの紹介

家環境ダッシュボード

場所:

祖母部屋

气温: 24.5°C

湿度: 26.7%

気圧: 1004.3hPa

更新時間 : 2021-02-20 13:30:12

環境グラフ

1日

温度

湿度

気圧

温

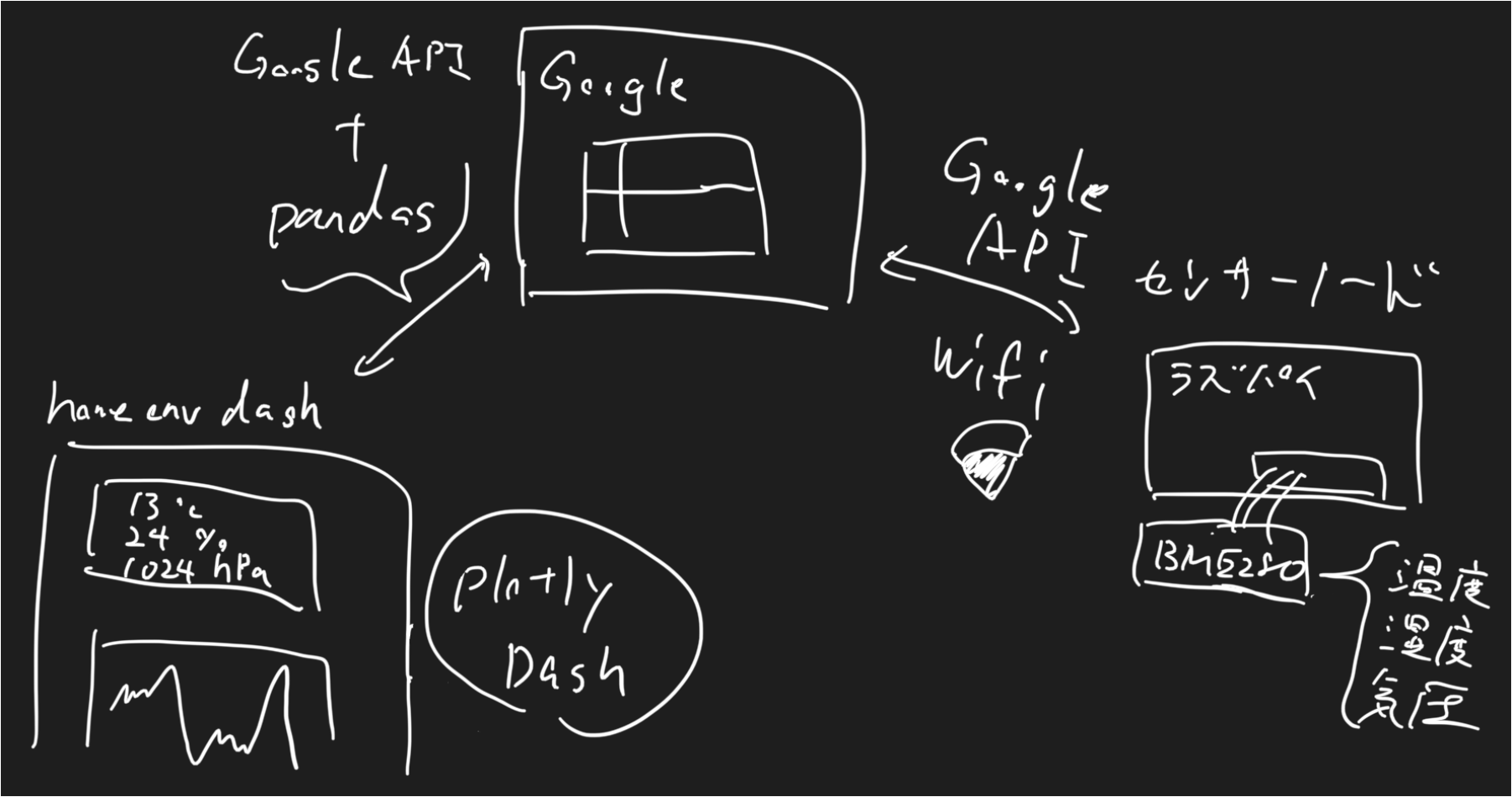


一言でいうと

家の環境をセンシングしてダッシュボード化する

まあよくあるプロジェクトと思う

中身はこんな感じで



やってみて思うことですが

1. 可視化アプリ作って思ったこと

センシングとデータ可視化は結構よくあるネタ

元々は家の環境を手軽に見ることを解決したくて始める

気圧見れるのがとても便利

低気圧にとっても弱いので状況判断しやすい

みなさんもストレートネックなど首回りにはお気をつけください

実家では祖母の介護で見守りにも使う

- ちゃんとエアコンついてるか
- 暑すぎないか寒すぎないか
- 乾燥してないか

そこから思うこととして

何か見たい=何かを知りたいから

気圧見たい -> 不調の原因は気圧に関係しているらしいから調べてみる

祖母の様子見守りたい -> 祖母の体調の心配

断片的な情報は混乱しか生まない

自分の調子悪い -> 何が原因だろう？を探したくなる

**ある人の体調が気になる -> 精神的につらい、いつも
気にする**

ITのシステムにも同じことは言える

見ることができる（測定できる）環境を大事にする

2. データ分析 -> アプリ化にはPlotly Dash便利

pandasを使えるならすぐにグラフ化出来る

環境グラフ

1日

温度

湿度

気圧



```
df = [[日付],[温度]の行列]

fig1 = px.line(df, x="Time", y="Temperature", title="温度")

dcc.Graph(id="tempature", figure=fig1), label="温度"
```


使っていて感じたこと

?. デザイン面倒

WEBアプリなのでCSSとかカラムとか考えないといけない

A. bootstrapを使えるコンポーネントがある

Dash Bootstrap Components

?. コンポーネントツリーが見つからない問題

webアプリだけど、html書かなくていい👍

htmlの構造表現をPythonで行う😞

```
return dbc.Container(  
    [  
        dcc.Location(id="url", refresh=False),  
        html.H2(config.TITLE),  
        html.Hr(),  
        dbc.Row(  
            [  
                dbc.Col(  
                    [dbc.Label("場所: "), location_dd, latest_view],  
                    md=4,  
                    id="sidebar",  
                ),  
                dbc.Col(main_view, md=8),  
            ],  
        ),  
    ],  
)
```

アプリの構造が複雑になるとネストも増える

必要なブロックを名前つけて管理



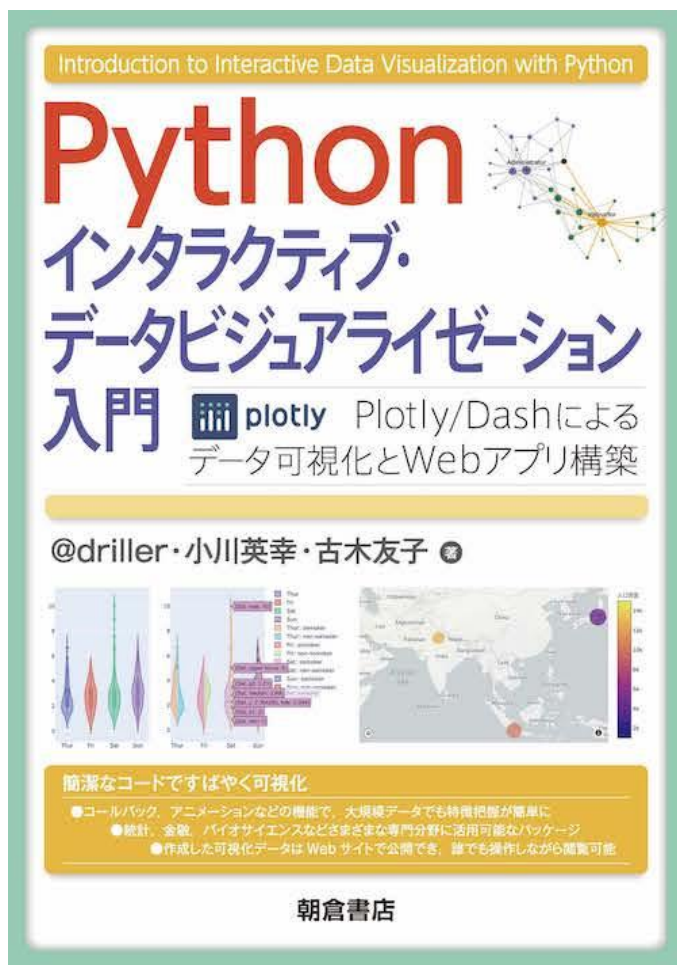
```
date_dd = [日付の種類]
graph_tab = [グラフのタブをまとめたもの]

# グラフとタブのビュー
main_view = html.Div(
    [
        html.H4("環境グラフ", style={"textAlign": "center"}),
        date_dd,
        graph_tab,
    ]
)
```

スッキリ

日本語情報少ない

書籍買おう！



とっても丁寧な説明とリファレンスにも使える

3. IoTはPythonとの親和性が良すぎる

IoTをPythonで動かす環境

ラズパイとかMicropythonとかで使える

ラズパイ: Raspberry Pi 小型のLinux開発ボード

micropython : マイクロコントローラー向けのPython環境

ライブラリそろってる = めっちゃいい環境

adafruitのCurcitPython向けのライブラリ

adafruit: オープンハードウェアなIoTキットや電子部品を販売する英国の会社

CurcitPython: Micropythonのフォーク版

少ないコードでセンサー情報取れる

```
import board
import busio
import digitalio
import adafruit_bme280

spi = busio.SPI(board.SCK, MOSI=board.MOSI, MISO=board.MISO)
# D5は任意のGPIOピン
cs = digitalio.DigitalInOut(board.D5)
bme280 = adafruit_bme280.Adafruit_BME280_SPI(spi, cs)

print("\nTemperature: %0.1f C" % bme280.temperature)
print("Humidity: %0.1f %" % bme280.humidity)
print("Pressure: %0.1f hPa" % bme280.pressure)
```

まとめ

可視化をしたい目的/動機があればぜひやろう

Pythonなら可視化しやすいよ！

Thanks !

時間あったらしゃべる↓

今後もっと使いやすくしたい

センサー対応を増やしたい

CO2とか

照度とか

スマートカメラの画像がみれたりすると防犯向けにも使える

設定ベースでできるようにする

センサーに対応するために毎回コード書くの面倒
決まった組み合わせでつなげばすぐに使えるようにしたい

データは外部に置かないでなるべくローカル

Google スプレッドシート is クラウド

センサーは個人な情報も入ってしまうので、出来るだけ内部にしたい

センサーノード側はFastAPIあたりでjsonで出し続けて
ダッシュボード側はセンサーノードからの情報をプルしてデータをためて可視化

進捗出たらどこかで発表を目標です。CfP頑張る。

Thanks!