

WSL2でPHPを体験してみた話

shizuoka.php online #0 LT

佐野浩士



お前誰よ

佐野浩士 twitter@hrs_sano645

- 静岡県富士市在住
- 佐野設計事務所（家業） ITなんでも担当
- Pythonの勉強会とかによくいます
 - shizuoka.py
 - Unagi.py
 - Python駿河（よくいます）

Python駿河/Unagi.py は20日です。
ぜひご参加を！

Unagi.py 勉強会30枚目～オンラインもくもく読書
会～ - connpass

もくもく会なのでぶっちゃけ
Python関係無くてもよいですw
もくもくしたい人は一緒にもくもく
しましょう。

宣伝終わり 



本題



4Kディスプレイを自室（仕事部屋）に導入







MBP2016だと4Kディスプレイでの作業きつすぎる。。

YouTube聞くとコマ送りだし。。

MBP 16インチつよつよ構成は高いし...
(400k Yen超えるし)



グラフィックス

あなたにぴったりのグラフィックスのオプションは？

AMD Radeon Pro 5500M(4GB
GDDR6メモリ搭載)

- 70,000円 (税別)

AMD Radeon Pro 5500M(8GB
GDDR6メモリ搭載)

- 60,000円 (税別)

AMD Radeon Pro 5600M(8GB
HBM2メモリ搭載)

返品できます

398,800円 (税別)

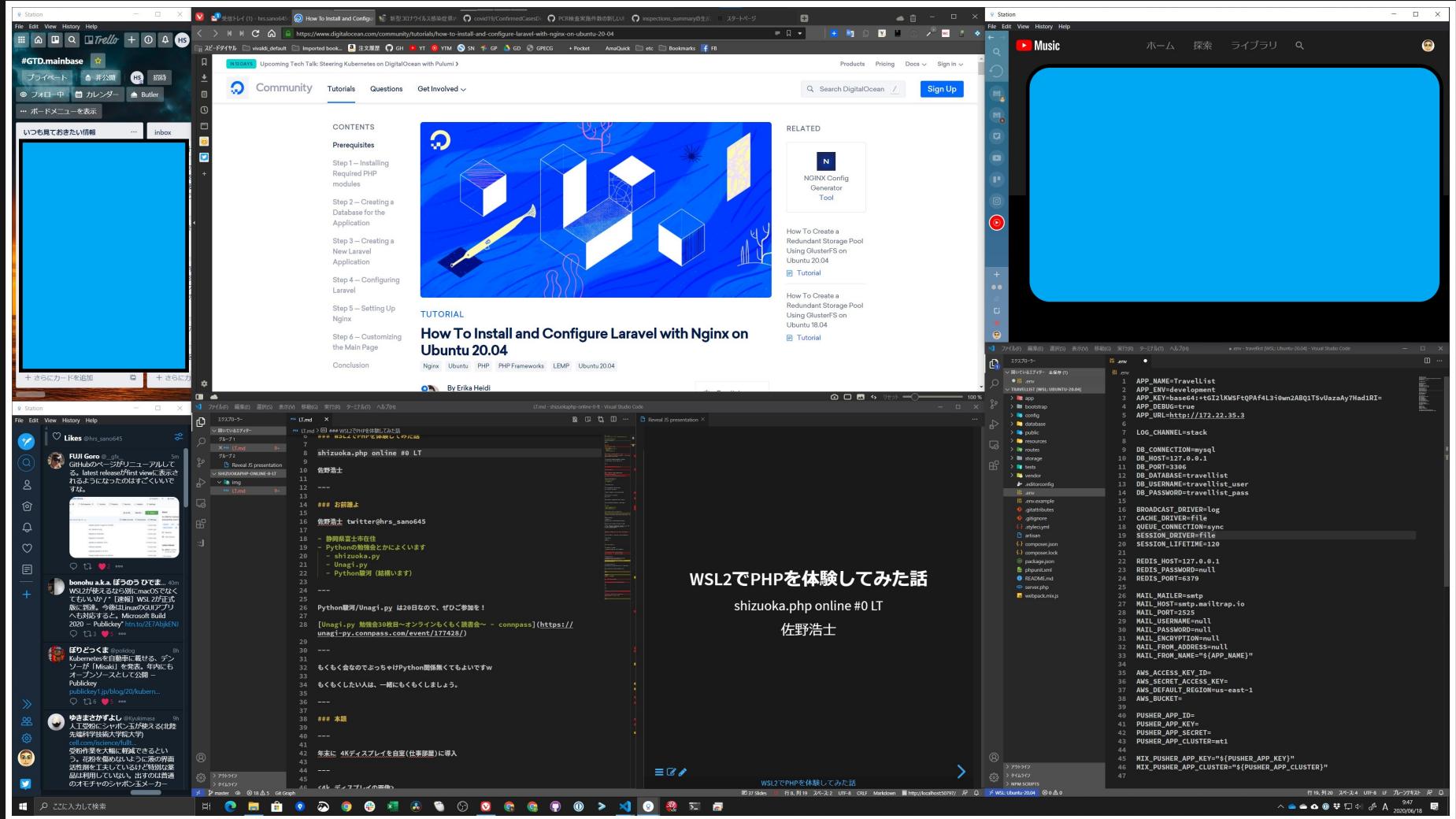
24回払い18,278円(税込)/月~

バッグに追加



そこでWinのデスクマシンを活用！

- CPU: Core i5 6200ぐらい
- メモリ: 16GB
- GPU: Radeon RX480



WSL2でPHPを体験してみた話

とっても快適！4Kディスプレイおススメです！

問題として: macOSでやってたUnixな作業がしづらい。。。。

Windows Subsystem for Linux2 (WSL2)

正式リリース



WSL2が正式版になったので
インストールして使ってみよう



shizuoka.phpなので
laravelのインストールまでやってみよう



やった作業

- Windows 10 2004へアップデートが前提
- WSL2のセットアップ
- LEMP, Laravelのセットアップ

WSL2のセットアップ:ここ見れば普通にできます。

-> [Windows Subsystem for Linux \(WSL\) を Windows 10 にインストールする | Microsoft Docs](#)

さらにわかりやすい-> [WSL2導入 | WinアップデートからWSL2を既定にするまでのスクショ - Qiita](#)

今後はカーネルの提供がWindows Update経由になる話があるので、ちょっと変わるかも？

OSはUbuntu 20.04を利用

MSストアから手に入ります（MSアカウント必要）



概要

システム必要条件

レビュー

関連するセクション

対応プラットフォーム

回 PC



ディストリビューションのエクスポート/インポートもできる

```
# インストール済みのディストリビューション一覧  
wsl --list -v  
  
NAME          STATE      VERSION  
* Ubuntu-20.04  Running    2  
  Legacy        Stopped    1  
  
wsl --export (ディストリビューション名) [ファイル名]  
# tarで固まったファイルができます  
# wsl --export Ubuntu-20.04 ub2004-wsl2-20200618
```



LEMP（と最近言うらしい） Linux, nginx, mysql, php セットアップ

[How To Install Linux, Nginx, MySQL, PHP \(LEMP stack\)
on Ubuntu 20.04 | DigitalOcean](#)

Step3まで進めました



Laravelのセットアップ -> composerを入れて、
Laravelのインストール

[How To Install and Use Composer on Ubuntu 20.04 |
DigitalOcean Step2まで](#)

[How To Install and Configure Laravel with Nginx on
Ubuntu 20.04 | DigitalOcean](#)

ここまでやってみた結果



Linuxマシンとほぼ変わらねえ！

(資料はUbuntuベースのものを見てましたが、ほぼ同じでした。)

TIPS:NGINXが動くIP

WSL2のターミナルから探します

ubuntu(debian系)なら ip a でeth0がNICっぽいです

```
$ ip a
# ~~前略~
5: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq s
  link/ether 00:15:5d:b0:18:d9 brd ff:ff:ff:ff:ff:ff
    inet 172.22.54.15/20 brd 172.22.63.255 scope global eth0
      valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:feb0:18d9/64 scope link
      valid_lft forever preferred_lft forever
```

TIPS:WSL2のサービス設定

- systemdではないっぽい: systemctlとか
journalctlではない
- init.dっぽい: serviceコマンドが通りました

(ググるとWSLの時代からsystemdは動かない状態
らしい)

-> <https://github.com/microsoft/WSL/issues/994>)

TIPS: WINDOWS TERMINALも便利

WSL(2)でインストールしたディストリビューションも一覧で出してくれる



--verbose, -v
すべてのう

--set-default, -s <デ
ディストリビューション>

--set-default-version
新しいディストリビュ

--set-version <ディス
指定したディストリビ

--shutdown

実行中のすべてのディストリビューションと WSL 2 ライトウェイト ユーティリティ仮想マシンを直ちに終了します。

--terminate, -t <ディストリビューション>

指定したディストリビューションを終了します。

--unregister <ディストリビューション>

ディストリビューションの登録を解除します。

--help

使用法を表示します。

PS C:\Users\hiroshi> wsl --list

Linux 用 Windows サブシステム ディストリビューション:

Ubuntu-20.04 (既定)

Legacy

PS C:\Users\hiroshi> wsl --list -v

NAME	STATE	VERSION
* Ubuntu-20.04	Running	2
Legacy	Stopped	1

PS C:\Users\hiroshi>

Windows PowerShell Ctrl+Shift+1

コマンドプロンプト Ctrl+Shift+2

Legacy Ctrl+Shift+3

Azure Cloud Shell Ctrl+Shift+4

Ubuntu-20.04 Ctrl+Shift+5

設定 Ctrl+,

フィードバック

? バージョン情報

の詳細情報を表示します。

ルバージョンを変更します。

>
更します。

まとめ



Windows10のWSL2でターミナル環境を作つて
快適なUnix生活を送ろう！



以上



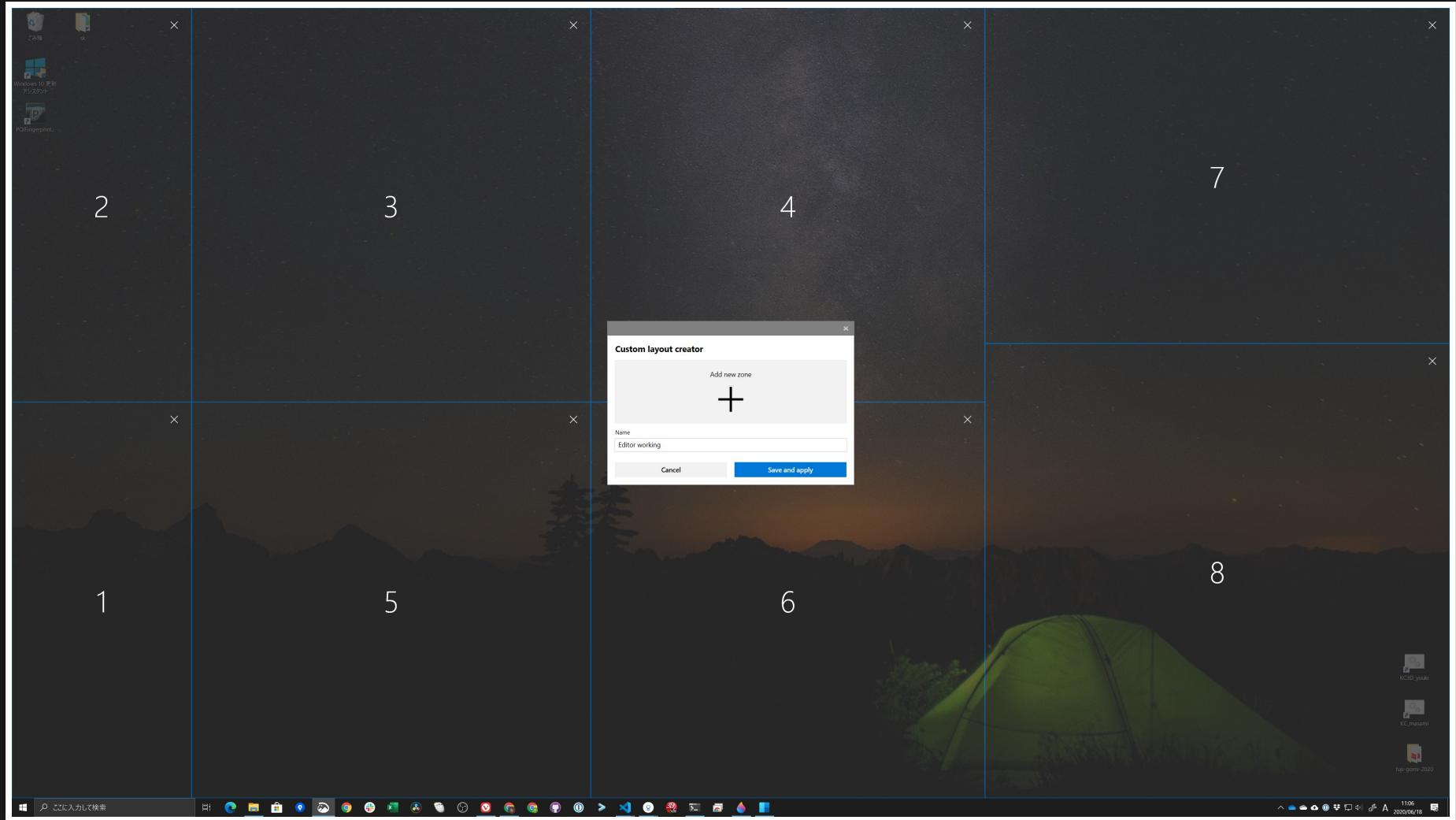
おまけ

もし余ったら / お茶濁し用

大きい画面を手に入れたら、PowerToysの
FancyZoneを設定しよう！

[microsoft/PowerToys: Windows system utilities to
maximize productivity](#)





おまけ2



WSL2 + VS Code の連携もやってみた



WSL2とVS Codeの連携は Remote-WSL拡張機能で実現できます

Remote - WSL - Visual Studio Marketplace



拡張機能: マーケットプレース

wsl

Remote - WSL 0.44.3

Open any folder in the Windows Subsystem for Linux

Microsoft

WSL workspaceFolder 1.1.2

Returns `workspaceFolder` as a WSL path

Ifurzewdaddock

Linux/Unix/WSL paths 0.1.0

Provides commands to have unix-like paths

Krzysztof Grzywocz

Save and Run with wsl 0.0.182

Run commands when a file is saved in ...

rais cui

OCaml and Reason IDE - for WSL 1.0.39

OCaml and Reason language support, f...

rais cui

Remote Development 0.20.0

An extension pack that lets you open a...

Microsoft

xsysfdx 0.1.7

A rapid development tool for Salesforce...

Exia.Huang

Easy C++ projects 2.0.0

Create C++ projects easily for GCC, WS...

Alejandro Charte Luque

Copy uPath 0.2.1

Allows to copy path in Unix style on Wi...

kira-dev

VSCode Terminal For Ubuntu 0.0.2

Integrates the Ubuntu WSL into VS Code

Docter60

Windows Terminal Utility 2.0.0

Windows Terminal Utility for VS Code.

wraith13

xycode 0.0.8

a lightweight and fast command execu...

Exia.Huang

Remote - WSL - ms-vscode

Microsoft | 2,756,574 | ★

Open any folder in the Windows Subsystem for Linux

無効にする アインストール この拡張機能はグローバルにインストールされています。

Remote - WSL がインストールされているため、この拡張機能をグローバルにインストールするには、この拡張機能をアンインストールする必要があります。

詳細 機能のコンピューション 変更ログ

Remote - WSL: New Window

Remote - WSL: New Window using Distro...

Remote - WSL: Getting Started

Remote - SSH: Connect to Host...

Remote - SSH: Open Configuration File...

Remote - SSH: Getting Started

Remote - Containers: Open Folder in Container...

Remote - Containers: Open Workspace in Container...

Remote - Containers: Open Repository in Container...

Remote - Containers: Attach to Running Container...

Remote - Containers: Try a Sample...

Remote - Containers: Getting Started

utils

 __init__.py

 config.py

 db.sqlite3

 routes.py

 scripts

 static

 venv

 deployment

 .gitignore

 LICENSE.md

 manage.py

 package-lock.json

 package.json

 README.md

 requirements.txt

@app.route('/', methods=['GET'])

def index():

 return render_template('index.html')

@app.route('/<path:path>', methods=['GET'])

def any_root_path(path):

 return render_template('index.html')

@app.route("/api/user", methods=["GET"])

@requires_auth

def get_user():

 return jsonify(result=g.current_user,

 tasks=Task.get_latest_tasks())

@app.route("/api/create_user", methods=["POST"])

def create_user():

 incoming = request.get_json()

 success = User.create_user(incoming)

 if not success:

 return jsonify(message="User with that email already exists"), 409

 new_user = User.query.filter_by(email=incoming["email"]).first()

 return jsonify(id=new_user.id)

WSL master 01:21 Python 3.6.7 64-bit (venv: venv) 0 0 ▲ 0 Python: Launch (Flask-Read-Postgres)

Type here to search

Installation

- Install [VS Code](#) or [VS Code Insiders](#) and this extension.

Note: When prompted to **Select Additional Tasks** during installation, be sure to check the **Add to** folder in WSL using the **code** or **code-insiders** command.

- Install the [Windows Subsystem for Linux](#) along with your preferred Linux distribution.

Note: WSL2 support is **experimental**. Also, extensions installed in Alpine Linux may not work due to limitations inside the extension. See the [Remote Development and Linux](#) article for details.

Working with Git? Here are two tips to consider:

Win10開発環境にがっつり使えると思う！
WSL2とVS Codeで快適開発環境を目指そう！

ちなみにVS Code + Remote-SSHも便利でおススメ！
(Remote-Containerも便利だと思われます)

終わり

