

【Other Apps】 備忘録

目次

- はじめに
- Visual Studio Code
- OneDrive
- PowePoint
- Power Query
- VirtualBox
- Vagrant
- Docker (【Docker】全容)
- Kubernetes (【Kubernetes】基本)
- Amazon ECS (【AWS】 備忘録 Amazon ECS)

はじめに

■はじめに

- ▶ ※ このページでは、OSを問わず利用可能な各種アプリケーションの使い方について書く。

Visual Studio Code

■備忘録

- ▶ ☆ Python ライブラリについてサジェストさせる
- ▶ ☆ PCにインストールされているコマンドラインシェルを開く (に接続する?)
- ▶ ☆ Mac OSにて、コマンドラインから呼び出せるようにする
- ▶ ☆ ターミナルにてVSCode特有のショートカットを発動させないようにする

OneDrive

■Windowsのエクスプローラーにおいて

- ▶ ☆ 中身は同期済みなのにあるフォルダーが同期保留中のまま変わらない場合

【Other Apps】 備忘録

目次

- はじめに
- Visual Studio Code
- OneDrive
- PowePoint
- Power Query
- VirtualBox
- Vagrant
- Docker (【Docker】全容)
- Kubernetes (【Kubernetes】基本)
- Amazon ECS (【AWS】 備忘録 Amazon ECS)

はじめに

■はじめに

- ▶ ※ このページでは、OSを問わず利用可能な各種アプリケーションの使い方について書く。

Visual Studio Code

■備忘録

- ▶ ☆ Python ライブラリについてサジェストさせる
- ▶ ☆ PCにインストールされているコマンドラインシェルを開く (に接続する?)
- ▶ ☆ Mac OSにて、コマンドラインから呼び出せるようにする
- ▶ ☆ ターミナルにてVSCode特有のショートカットを発動させないようにする

OneDrive

■Windowsのエクスプローラーにおいて

- ▶ ☆ 中身は同期済みなのにあるフォルダーが同期保留中のまま変わらない場合

PowerPoint

- 注意
- ▶ ※ 動画をエクスポートする際は、[画面切り替え] > [画面切り替えのタイミング] > [自動] にチェックがついており、ちゃんと時間が設定されていることを確認しよう。

Power Query

- 備忘録
- ▶ 現在時刻

VirtualBox

- 用語
- 仮想マシン（仮想PC） ↔ 物理マシン（ベアメタル）

• ゲストOS ↔ ホストOS

- はじめに
- ▶ VirtualBoxとは

▶ ☆ インストール

▶ ☆ ダウングレード

▶ ☆ ~/VirtualBox VMs ディレクトリを別の場所へ移動

Vagrant

- はじめに
- ▶ Vagrantとは

▶ ☆ インストール

▶ バージョン確認

PowerPoint

- 注意
- ▶ ※ 動画をエクスポートする際は、[画面切り替え] > [画面切り替えのタイミング] > [自動] にチェックがついており、ちゃんと時間が設定されていることを確認しよう。

Power Query

- 備忘録
- ▶ 現在時刻

DateTime.ToText(DateTime.LocalNow(), [Format="yyyy-MM-dd HH:mm:ss"])

VirtualBox

- 用語
- 仮想マシン（仮想PC） ↔ 物理マシン（ベアメタル）

• ゲストOS ↔ ホストOS

- はじめに
- ▶ VirtualBoxとは

▶ ☆ インストール

▶ ☆ ダウングレード

▶ ☆ ~/VirtualBox VMs ディレクトリを別の場所へ移動

Vagrant

- はじめに
- ▶ Vagrantとは

▶ ☆ インストール

▶ バージョン確認

VirtualBox の操作を簡単なコマンドで扱えるようにしてくれるソフト

\$ vagrant --version

- ▶ ☆ アップデート
- ▶ ☆ `~/.vagrant.d` ディレクトリを別の場所へ移動
- ▶ ☆ プロジェクトディレクトリを別の場所へ移動

■Boxの管理

- ▶ ※ Boxとは仮想マシンのベースとなるもので、これをもとにして仮想マシンが作成される。Boxは様々なOSのものが Vagrant Cloud で公開されており、ここからローカルの Vagrant に追加して利用する。
- ▶ Boxをダウンロード
- ▶ ローカルにあるBoxの一覧
- ▶ 全Boxを更新
- ▶ Boxを削除

■仮想マシンを作成する手順

- ▶ 1. プロジェクトディレクトリへ移動
- ▶ 2. プロディにVagrantfileを作成
- ▶ ☆ 3. Vagrantfileを編集
- ▶ 4. 仮想マシンを作成して起動

■こうしたのならVagrantfile編集時にこれを追記

- ▶ ※ 前提として、`Vagrant.configure("2") do |config| .. end` の間に追記すること。
- ▶ 仮想マシンのホスト名
- ▶ 共有フォルダを設定

ネットワーク

- ▶ ※ 以下の設定はいずれか1つしか設定できない。
- ▶ ポート転送
- ▶ 固定のIP割り当て
- ▶ 動的なIP割り当て
- ▶ ※ `:private_network` の代わりに `:public_network` にすると、ホストOS以外にも、同じLAN内のPCからアクセスできるようになる。ただ一般的にはセキュリティ面から Private Network を使う（`:private_network` にする）。

一般・システム

- ▶ ☆ アップデート
- ▶ ☆ `~/.vagrant.d` ディレクトリを別の場所へ移動
- ▶ ☆ プロジェクトディレクトリを別の場所へ移動

■Boxの管理

- ▶ ※ Boxとは仮想マシンのベースとなるもので、これをもとにして仮想マシンが作成される。Boxは様々なOSのものが Vagrant Cloud で公開されており、ここからローカルの Vagrant に追加して利用する。
- ▶ Boxをダウンロード `$ vagrant box add box※ ※ ubuntu/bionic64 など`
- ▶ ローカルにあるBoxの一覧 `$ vagrant box list`
- ▶ 全Boxを更新 `$ vagrant box update`
- ▶ Boxを削除 `$ vagrant box remove box`
や `$ vagrant box remove box --box-version version`

■仮想マシンを作成する手順

- ▶ 1. プロジェクトディレクトリへ移動 `$ mkdir dirName $ cd dir`
- ▶ 2. プロディにVagrantfileを作成 `$ vagrant init box`
- ▶ ☆ 3. Vagrantfileを編集
- ▶ 4. 仮想マシンを作成して起動 `$ vagrant up` ※うまくいかないこともある！

■こうしたのならVagrantfile編集時にこれを追記

- ▶ ※ 前提として、`Vagrant.configure("2") do |config| .. end` の間に追記すること。
- ▶ 仮想マシンのホスト名 `config.vm.hostname = "ホスト名"`
- ▶ 共有フォルダ `config.vm.synced_folder "ディのパス※1", "マウントポイント"`
を設定 ※¹ 物理マシン上のディ。 `./app/` でプロディ直下のappディに。

ネットワーク

- ▶ ※ 以下の設定はいずれか1つしか設定できない。
- ▶ ポート転送 `config.vm.network :forwarded_port, guest: guestPort, host: hostPort`
- ▶ 固定のIP割り当て `config.vm.network :private_network, ip: "privateIPAddress"`
- ▶ 動的なIP割り当て `config.vm.network :private_network, type: "dhcp"` ※要エラー対策
- ▶ ※ `:private_network` の代わりに `:public_network` にすると、ホストOS以外にも、同じLAN内のPCからアクセスできるようになる。ただ一般的にはセキュリティ面から Private Network を使う（`:private_network` にする）。

一般・システム

▶ ☆ ディスク容量

▶ ※ 以下はさらにネストして、`config.vm.provider :virtualbox do |vb| .. end` の間に追記すること。

▶ 仮想マシンの名前	
▶ 使用するメモリサイズ	
▶ 使用するプロセッサの数	

同時に複数のマシンを立てる

▶ ☆ 同時に複数のマシンを立てるようなVagrantfileの書き方

■仮想マシンの管理

▶ ※ 作成済み仮想マシンの2回目以降の起動は、VirutalBoxマネージャで行うのが確実。

▶ 仮想マシンにSSH接続	

▶ 仮想マシンを終了	
▶ 仮想マシンの一覧	

▶ 仮想マシンを削除	
▶ Vagrantfileの更新を反映	
▶ ポート転送の設定を確認	

▶ ☆ ディスク容量

▶ ※ 以下はさらにネストして、`config.vm.provider :virtualbox do |vb| .. end` の間に追記すること。

▶ 仮想マシンの名前	vb.name = "マシン名※" ※ <code>Ubuntu 20 April 2023</code> とか
▶ 使用するメモリサイズ	vb.memory = 2048 ※MB単位。
▶ 使用するプロセッサの数	vb.cpus = 4

同時に複数のマシンを立てる

▶ ☆ 同時に複数のマシンを立てるようなVagrantfileの書き方

■仮想マシンの管理

▶ ※ 作成済み仮想マシンの2回目以降の起動は、VirutalBoxマネージャで行うのが確実。

▶ 仮想マシンにSSH接続 マシンが起動中のうえで、プロディにて \$ vagrant ssh
※同時に複数マシンを立てた場合は `$ " machineName` 。

▶ 仮想マシンを終了 マシンからログアウトし、プロディにて \$ vagrant halt

▶ 仮想マシンの一覧 \$ vagrant global-status
※同時に立てた複数マシンの一覧なら `$ vagrant status` 。

▶ 仮想マシンを削除 \$ vagrant destroy *machineID* ※プロディは削除されない

▶ Vagrantfileの更新を反映 プロディにて \$ vagrant reload ※**再作成ではない**！

▶ ポート転送の設定を確認 プロディにて \$ vagrant port