# -\*- mode: ruby -\*-

# vi: set ft=ruby :

# All Vagrant configuration is done below. The "2" in Vagrant.configure

# configures the configuration version (we support older styles for

# backwards compatibility). Please don't change it unless you know what

# you're doing.

Vagrant.configure("2") do |config|

# The most common configuration options are documented and commented below.

# For a complete reference, please see the online documentation at

# https://docs.vagrantup.com.

# Every Vagrant development environment requires a box. You can search for

# boxes at https://vagrantcloud.com/search.

config.vm.box = "base"

# Disable automatic box update checking. If you disable this, then

# boxes will only be checked for updates when the user runs

# `vagrant box outdated`. This is not recommended.

# config.vm.box\_check\_update = false

# Create a forwarded port mapping which allows access to a specific port

# within the machine from a port on the host machine. In the example below,

# accessing "localhost:8080" will access port 80 on the guest machine.

# NOTE: This will enable public access to the opened port

# config.vm.network "forwarded\_port", guest: 80, host: 8080

# Create a forwarded port mapping which allows access to a specific port

# within the machine from a port on the host machine and only allow access

# via 127.0.0.1 to disable public access

# config.vm.network "forwarded\_port", guest: 80, host: 8080, host\_ip: "127.0.0.1"

# Create a private network, which allows host-only access to the machine

# using a specific IP.

# config.vm.network "private\_network", ip: "192.168.33.10"

# Create a public network, which generally matched to bridged network.

# Bridged networks make the machine appear as another physical device on

# your network.

# config.vm.network "public\_network"

# Share an additional folder to the guest VM. The first argument is

# the path on the host to the actual folder. The second argument is

# the path on the guest to mount the folder. And the optional third

# argument is a set of non-required options.

# config.vm.synced\_folder "../data", "/vagrant\_data"

# Provider-specific configuration so you can fine-tune various

# backing providers for Vagrant. These expose provider-specific options.

# Example for VirtualBox:

#

# config.vm.provider "virtualbox" do |vb|

# # Display the VirtualBox GUI when booting the machine

# vb.gui = true

#

# # Customize the amount of memory on the VM:

# vb.memory = "1024"

# end

#

# View the documentation for the provider you are using for more

# information on available options.

# Enable provisioning with a shell script. Additional provisioners such as

# Ansible, Chef, Docker, Puppet and Salt are also available. Please see the

# documentation for more information about their specific syntax and use.

# config.vm.provision "shell", inline: <<-SHELL

# apt-get update

# apt-get install -y apache2

# SHELL

End

Usage: vagrant [options] <command> [<args>]

-h, --help Print this help.

Common commands:

autocomplete manages autocomplete installation on host

box manages boxes: installation, removal, etc.

cloud manages everything related to Vagrant Cloud

destroy stops and deletes all traces of the vagrant machine

global-status outputs status Vagrant environments for this user

halt stops the vagrant machine

help shows the help for a subcommand

init initializes a new Vagrant environment by creating a Vagrantfile

login

package packages a running vagrant environment into a box

plugin manages plugins: install, uninstall, update, etc.

port displays information about guest port mappings

powershell connects to machine via powershell remoting

provision provisions the vagrant machine

push deploys code in this environment to a configured destination

rdp connects to machine via RDP

reload restarts vagrant machine, loads new Vagrantfile configuration

resume resume a suspended vagrant machine

snapshot manages snapshots: saving, restoring, etc.

ssh connects to machine via SSH

ssh-config outputs OpenSSH valid configuration to connect to the machine

status outputs status of the vagrant machine

suspend suspends the machine

up starts and provisions the vagrant environment

upload upload to machine via communicator

validate validates the Vagrantfile

vbguest plugin: vagrant-vbguest: install VirtualBox Guest Additions to the machine

version prints current and latest Vagrant version

winrm executes commands on a machine via WinRM

winrm-config outputs WinRM configuration to connect to the machine

For help on any individual command run `vagrant COMMAND -h`

Additional subcommands are available, but are either more advanced

or not commonly used. To see all subcommands, run the command

`vagrant list-commands`.

--[no-]color Enable or disable color output

--machine-readable Enable machine readable output

-v, --version Display Vagrant version

--debug Enable debug output

--timestamp Enable timestamps on log output

--debug-timestamp Enable debug output with timestamps

--no-tty Enable non-interactive output