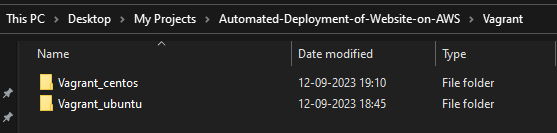
**Vagrant**

Vagrant is an open-source software product for building and managing virtualized development environments. It's designed to make it easier for developers to create, configure, and replicate development environments on their local machines, regardless of the underlying operating system. Vagrant is particularly popular among developers who work on projects that involve different dependencies, configurations, and software stacks.

**Pre-Requisites Tools:**

|  |
| --- |
| 1. Vagrant  2. VS code  3. Oracle VM |

[Reference Note : <https://www.youtube.com/watch?v=gL37hU-U88E&ab_channel=becloudready>]

**Steps to Set Up Vagrant for Project**:  
Create a Vagrant Directory and initialize it with the respective OS  


|  |  |  |
| --- | --- | --- |
| |  | | --- | | **For Centos:** 1. Create a sub folder Vagrant\_centos  2. vagrant init geerlingguy/centos7 | | **For Ubuntu:** 1. Create a sub folder Vagrant\_ubuntu  2. vagrant init ubuntu/bionic64 | |

Steps to Create a VM using vagrant file which got created after initializing

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | **Vagrant\_ubuntu: To do:** We will be creating Ubuntu VM for Nginx Load Balancer copy the below script inside ..\Vagrant\_ubuntu\vagrantfile   **Script:** Vagrant.configure("2") do |config|  ### Nginx VM ###  config.vm.define "web01" do |web01|  web01.vm.box = "ubuntu/bionic64"  web01.vm.hostname = "web01"  web01.vm.network "private\_network", ip: "192.168.56.11"  end  end | | **Vagrant\_centos: Execute 1 Script at a time**   |  | | --- | | **To do:** We will be creating Centos VM for DB copy the below script inside ..\Vagrant\_ubuntu\vagrantfile   **Script:** Vagrant.configure("2") do |config|  ### DB vm ####  config.vm.define "db01" do |db01|  db01.vm.box = "geerlingguy/centos7"  db01.vm.hostname = "db01"  db01.vm.network "private\_network", ip: "192.168.56.15"  end  end | | **To do:** We will be creating Centos VM for Memcache copy the below script inside ..\Vagrant\_ubuntu\vagrantfile  **Script:** Vagrant.configure("2") do |config|  ### Memcache vm ####  config.vm.define "mc01" do |mc01|  mc01.vm.box = "geerlingguy/centos7"  mc01.vm.hostname = "mc01"  mc01.vm.network "private\_network", ip: "192.168.56.14"  end  end | | **To do:** We will be creating Centos VM for RabbitMQ  copy the below script inside ..\Vagrant\_ubuntu\vagrantfile  **Script:** Vagrant.configure("2") do |config|  ### RabbitMQ vm ####  config.vm.define "rmq01" do |rmq01|  rmq01.vm.box = "geerlingguy/centos7"  rmq01.vm.hostname = "rmq01"  rmq01.vm.network "private\_network", ip: "192.168.56.16"  end  end | | **To do:** We will be creating Centos VM for Tomcat copy the below script inside ..\Vagrant\_ubuntu\vagrantfile  **Script:** Vagrant.configure("2") do |config|  ### tomcat vm ###  config.vm.define "app01" do |app01|  app01.vm.box = "geerlingguy/centos7"  app01.vm.hostname = "app01"  app01.vm.network "private\_network", ip: "192.168.56.12"  app01.vm.provider "virtualbox" do |vb|  vb.memory = "1024"  end  end  end | | |

**Vagrant Commands:**  
1. To initialize the Directory

|  |
| --- |
| vagrant init <OS> Ex: vagrant init ubuntu/bionic64 |

2. To run the Vagrant file [Make sure that you are at the same path where vagrant file is to execute the command]

|  |
| --- |
| vagrant up |

3. To delete the server

|  |
| --- |
| vagrant destroy |

4. To connect the server

|  |
| --- |
| vagrant ssh <Hostname or IP> Ex: vagrant ssh app01 |

**Vagrant Manuals:**

|  |
| --- |
| # -\*- 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 = "geerlingguy/centos7"  # 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"  # Disable the default share of the current code directory. Doing this  # provides improved isolation between the vagrant box and your host  # by making sure your Vagrantfile isn't accessable to the vagrant box.  # If you use this you may want to enable additional shared subfolders as  # shown above.  # config.vm.synced\_folder ".", "/vagrant", disabled: true  # 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 |