CHEAT SHEETS HERO

# Vagrant Cheatsheet

A quick reference guide for Vagrant, covering essential commands, configuration options, and best practices for managing virtual development environments.

## Vagrant Essentials

### Basic Commands

| | |
|---|---|
| `vagrant init` | Initializes a new Vagrant environment by creating a `Vagrantfile` in the current directory. |
| `vagrant up` | Starts the Vagrant environment. Downloads the specified box (if not already present) and provisions the virtual machine. |
| `vagrant ssh` | Connects to the Vagrant environment via SSH. |
| `vagrant halt` | Stops the Vagrant environment gracefully. |
| `vagrant suspend` | Pauses the Vagrant environment, saving its current state. |
| `vagrant resume` | Resumes a suspended Vagrant environment. |
| `vagrant destroy` | Stops and deletes all traces of the Vagrant environment. |
| `vagrant status` | Displays the current status of the Vagrant environment. |

### Vagrantfile Configuration

The `Vagrantfile` defines the configuration for your Vagrant environment. It is written in Ruby.

**Example:**

```
Vagrant.configure("2") do |config|
  config.vm.box = "ubuntu/focal64"
  config.vm.network "forwarded_port", guest: 80, host: 8080
  config.vm.provider "virtualbox" do |vb|
    vb.memory = "2048"
  end
end
```

`config.vm.box` - Specifies the base box for the virtual machine.

`config.vm.network` - Configures network settings, like forwarded ports or private networks.

`config.vm.provider` - Configures provider-specific settings (e.g., VirtualBox, VMware).

## Networking and Provisioning

### Networking Options

| | |
|---|---|
| `forwarded_port` | Forwards a port from the host machine to the guest machine.<br>**Example:**<br>`config.vm.network "forwarded_port", guest: 80, host: 8080` |
| `private_network` | Creates a private network accessible only from the host machine.<br>**Example:**<br>`config.vm.network "private_network", ip: "192.168.33.10"` |
| `public_network` | Creates a network bridged to your host’s network interface, making the VM accessible from the external network.<br>**Example:**<br>`config.vm.network "public_network"` |

### Provisioning Methods

Vagrant can provision the VM using shell scripts, Chef, Puppet, Ansible, or Docker.

**Shell Provisioning Example:**

```
config.vm.provision "shell", inline: <<-SHELL
  apt-get update
  apt-get install -y nginx
SHELL
```

**Ansible Provisioning Example:**

```
config.vm.provision "ansible" do |ansible|
  ansible.playbook = "provisioning/playbook.yml"
end
```

## Advanced Configuration

### Box Management

| | |
|---|---|
| `vagrant box list` | Lists all installed boxes. |
| `vagrant box add <name> <url>` | Adds a box from a URL or local file. |
| `vagrant box remove <name>` | Removes a box from the system. |
| `vagrant box update` | Updates installed boxes to the latest version. |

### Synced Folders

Synced folders allow you to share files between your host machine and the Vagrant environment.

**Example:**

```
config.vm.synced_folder "./data", "/var/www/data"
```

The first argument is the path on the host machine, and the second argument is the path on the guest machine.

## Multi-Machine Environments

### Defining Multiple Machines

Vagrant supports defining multiple machines in a single `Vagrantfile`.

**Example:**

```ruby
Vagrant.configure("2") do |config|
  config.vm.define "web" do |web_config|
    web_config.vm.box = "ubuntu/focal64"
    web_config.vm.network "forwarded_port", guest: 80, host: 8080
  end

  config.vm.define "db" do |db_config|
    db_config.vm.box = "ubuntu/focal64"
    db_config.vm.network "private_network", ip: "192.168.33.20"
  end
end
```

In this example, we define two machines: `web` and `db`.

### Managing Multiple Machines

| | |
|---|---|
| `vagrant up <machine>` | Starts a specific machine. |
| `vagrant halt <machine>` | Stops a specific machine. |
| `vagrant ssh <machine>` | Connects to a specific machine via SSH. |
| `vagrant destroy <machine>` | Destroys a specific machine. |
| `vagrant status` | Shows the status of all machines defined in the Vagrantfile. |