1. What will happen if you will start the vagrant machine without $vagrant init. Why?

$vagrant init: Initializes the current directory by creating an initial Vagrantfile that contains configuration for your virtual machine(s) instance. Vagrant is meant to run with one Vagrantfile per project, so if there is one file previously created will happen nothing otherwise the progam will not run.

2. To bring up the Vagrant virtual environment, you can use $vagrant up. What will happen after the command?

$vagrant up: Creates and configures guest machines according to your Vagrantfile, so the Vagrant machine is created.

3. How do you access (login into) virtual machine created with vagrant?

$vagrant ssh: give you access to a shell of your virtual machine. Then you can use vagrant login if you are accessing to a protected box.

4. What is a BOX in Vagrant?

Boxes are the package format for Vagrant environments. In other words the base image to quickly clone a virtual machine.

5. What is Provider in Vagrant?

A provider, type of machine to manage. Vagrant ships out of the box with support for VirtualBox, Hyper-V, and Docker, Vagrant also has the ability to manage other types of machines. This is done using other providers with Vagrant. Alternative providers may offer different features that make the most sense for your use case.

6. What is Provisioner in Vagrant?

Provisioners in Vagrant allow you to automatically install software, alter configurations, and more on the machine.

7. What is Vagrantfile?

The Vagrantfile is a Ruby file used to configure Vagrant on a per-project basis. The main function of the Vagrantfile is to described the virtual machines required for a project as well as how to configure and provision these machines.

8. What are Synced Folders in Vagrant?

Synced folders enable Vagrant to sync a folder on the host machine to the guest machine, allowing you to continue working on your project's files on your host machine, but use the resources in the guest machine to compile or run your project.

9. What is Multi-Machine environment in Vagrant?

It is the ability to define and control multiple guest machines per Vagrantfile.

10. How do you define multiple machines in Vagrant?

Vagrant allows us to define more virtual machines by wrapping them in blocks like the following:

```
config.vm.define "<vmName>" do |<vmName>|
  <snip>
end
```

11. What does vagrant push do?

Deploy or "push" application code in the same directory as your Vagrantfile to a remote.

12. What does vagrant box list do?

Lists all the boxes that are installed into Vagrant.

13. What does vagrant box outdated do?

Tells you whether or not the box you are using in your current Vagrant environment is outdated.

14. What does vagrant box prune do?

Removes old versions of installed boxes

15. What does vagrant box remove do?

$vagrant box remove NAME: Removes a box from Vagrant that matches the given name.

16. What does vagrant box repackage do ?

Takes the files from Vagrant's internal storage ~/.vagrant.d/... and re-packages them back into a box in the current directory called package.box. to save changes.

17. What does vagrant box update do?

Updates the box for the current Vagrant environment if there are updates available.

18. What does vagrant connect do?

Gives the connecting person a static IP they can use to communicate to the shared Vagrant environment.

19. What does vagrant destroy do?

Stops the running machine Vagrant is managing and destroys all resources that were created during the machine creation process