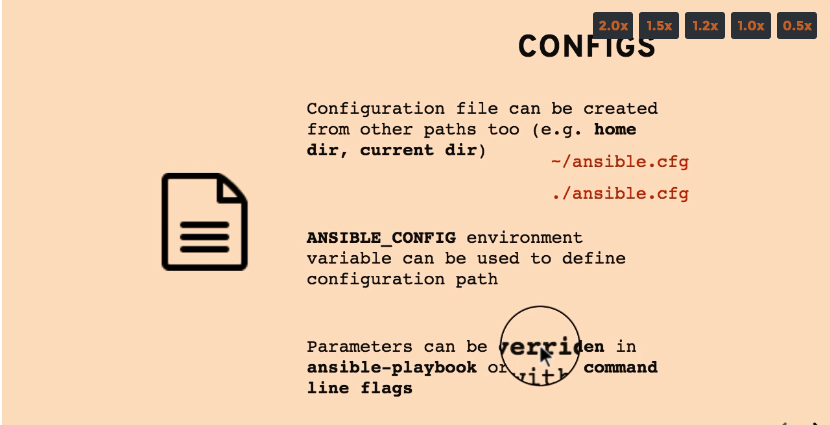
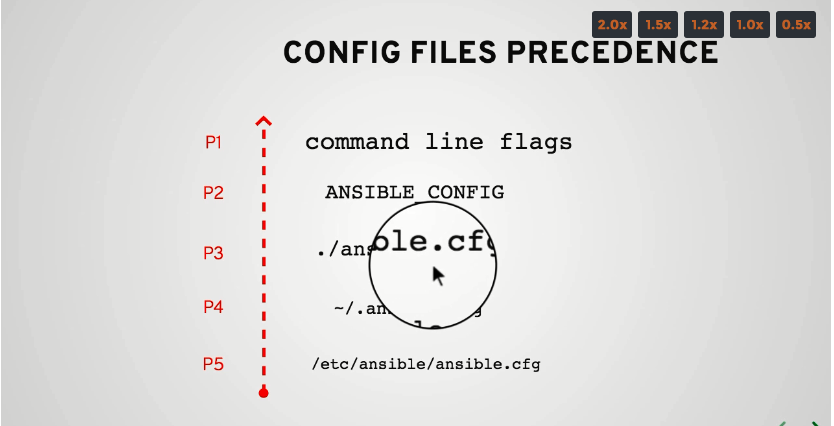
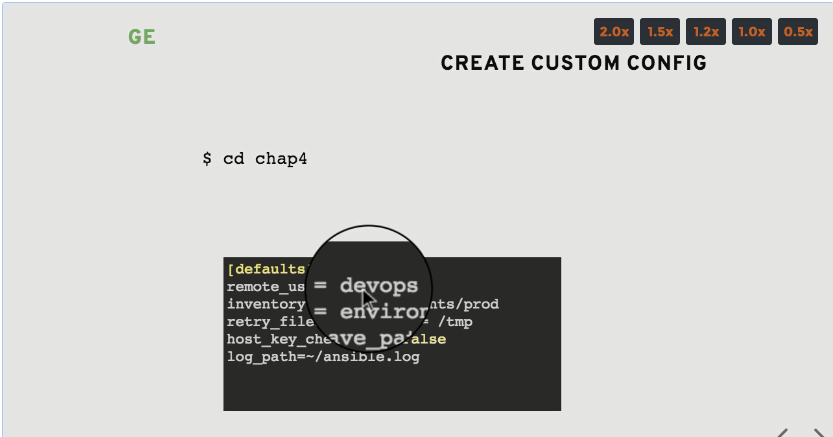
Configuring Ansible

The default location for the ansible configuration file is /etc/ansible/ansible.cfg. Basically it is an ini file as below

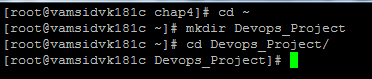




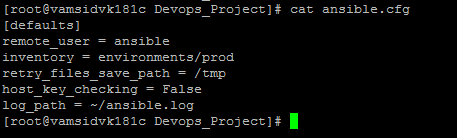




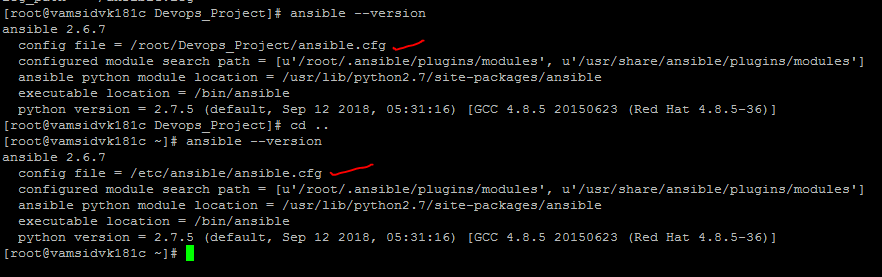
Above ansible.cfg we are going to create in the project repo we gonna create. Ans I will use ansible\_user=ansible in my cfg file.



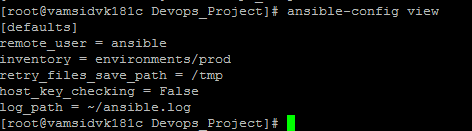
Below is the custom cfg file I’m using in this project

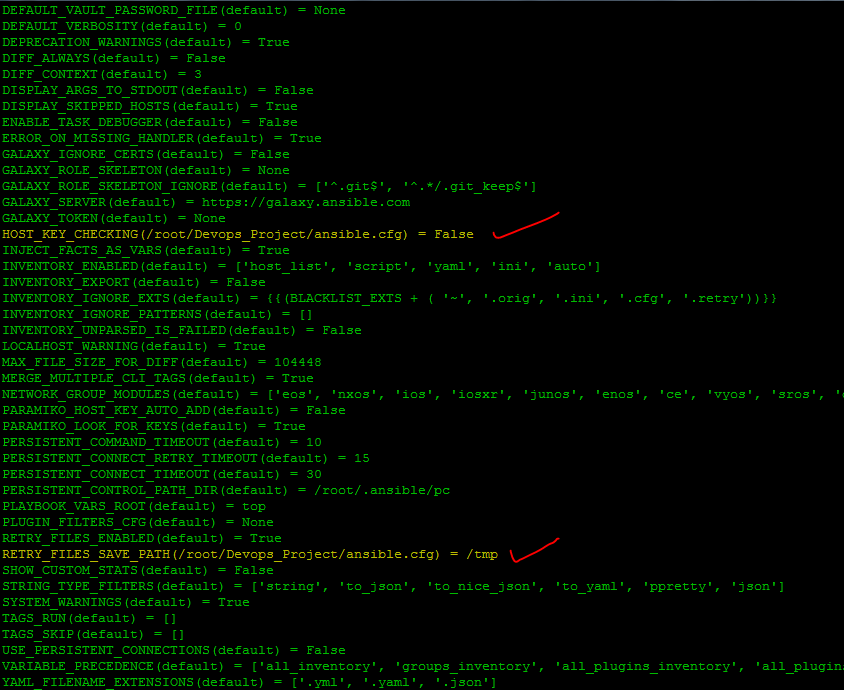


To validate the current configuration we are using in the project. We can validate it using ansible utility as below



Alternatively we can use ansible-config utility as below



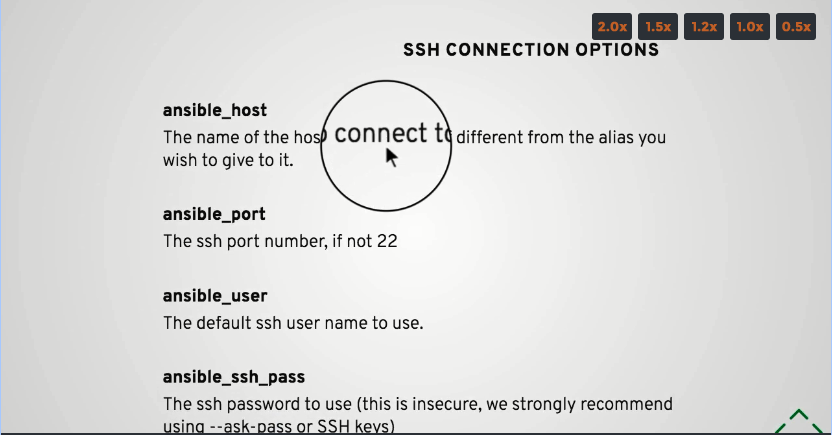
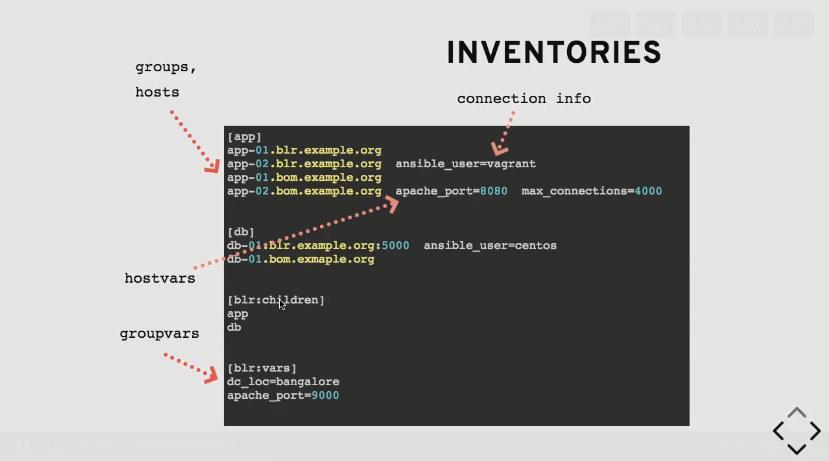


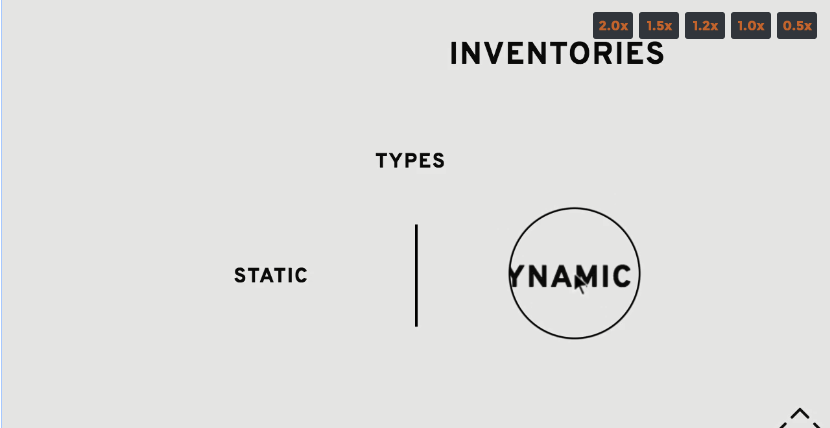
We can also use ansible-config dump which would list out all the config and current project config in orange color

Defining a static Inventory

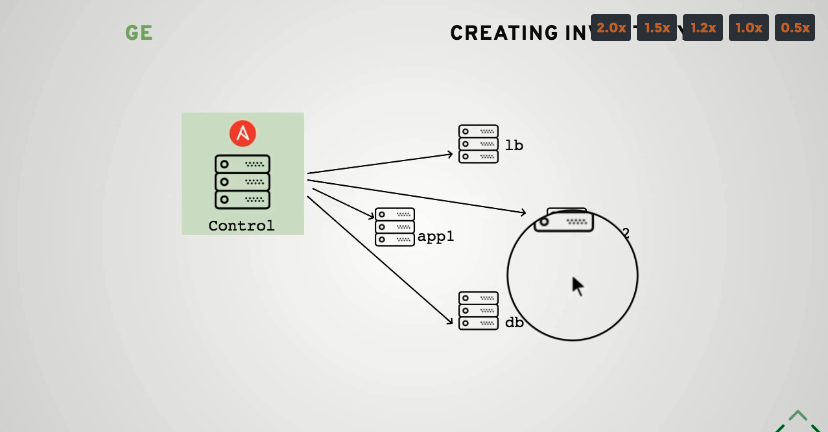


Inventory nothing but list of servers that we want to manage using ansible . As an example of above picture

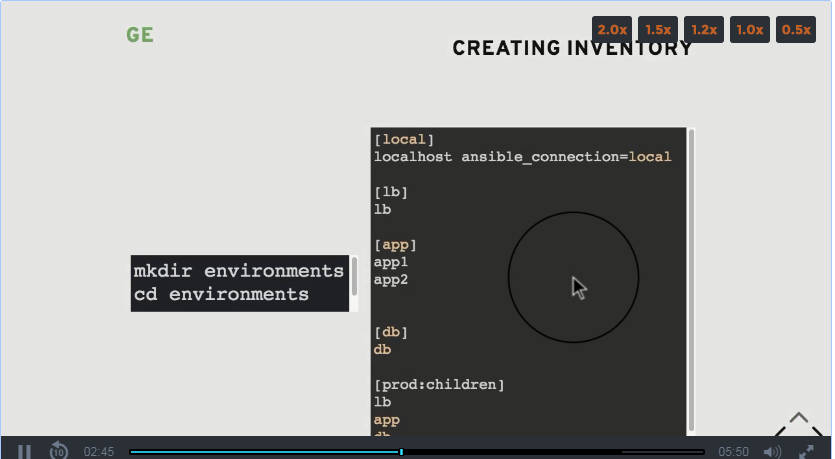




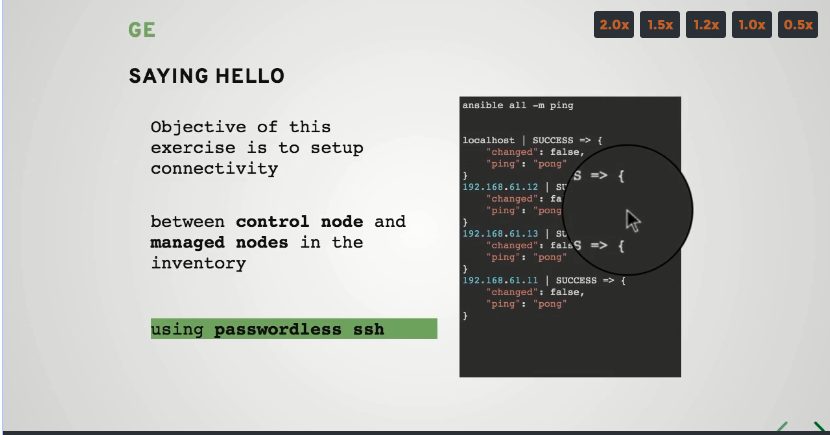
Inventory can be static and dynamic . Static we define them based on the file on local filesystem .Dynamic are such for the cloud providers which would have script which would generate the script



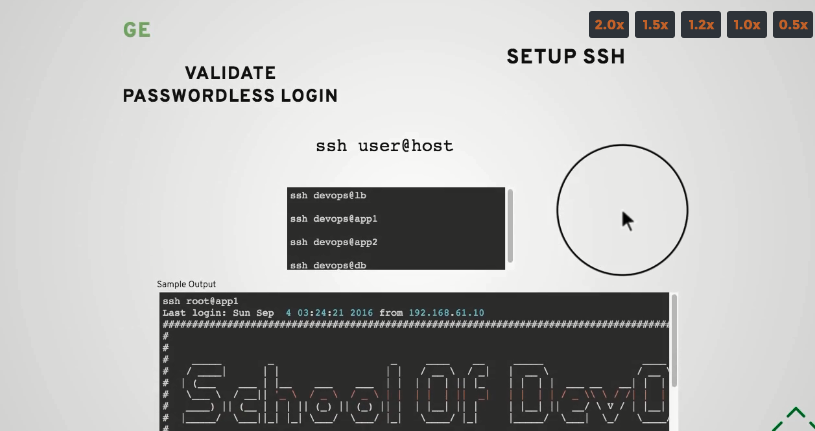
This is the diagrammatic representation our inventory should look like





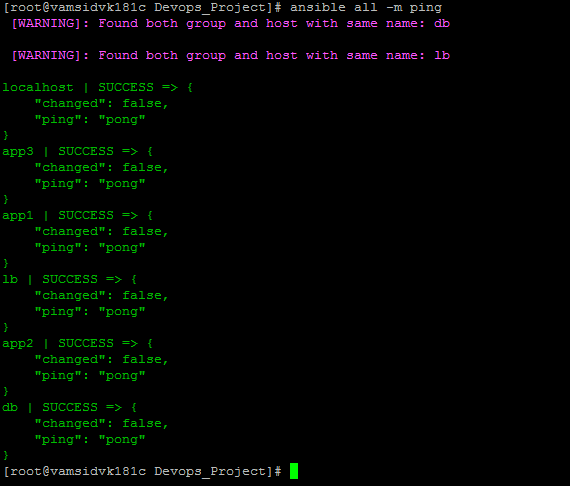




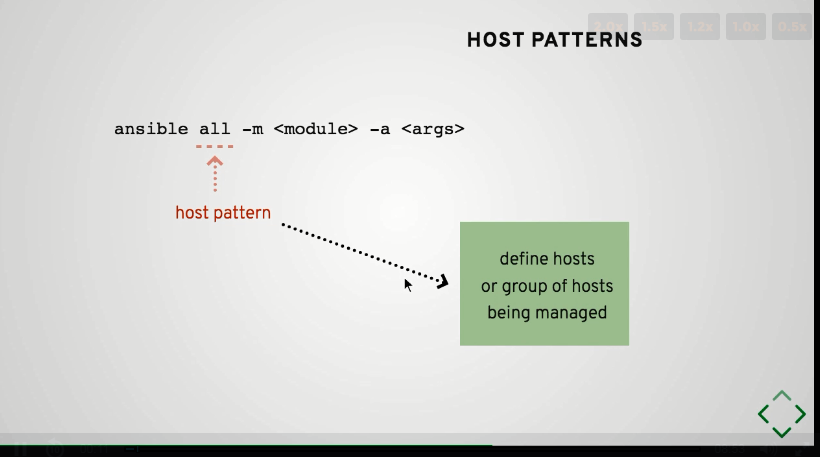


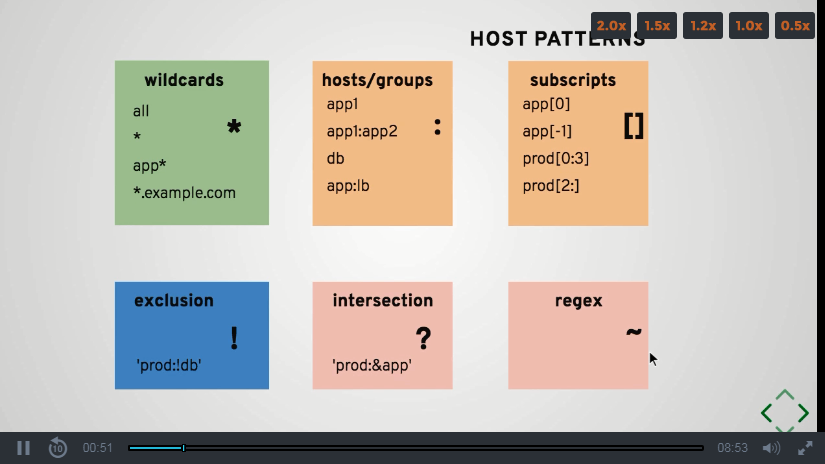


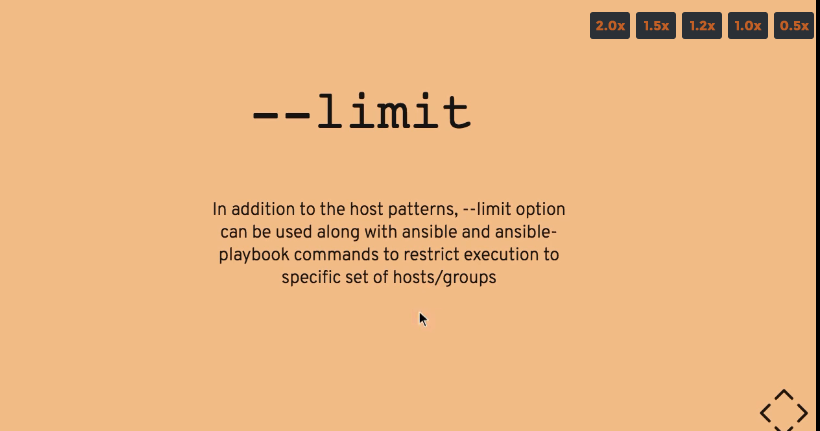
Let’s run sample ansible adhoc command to test connection

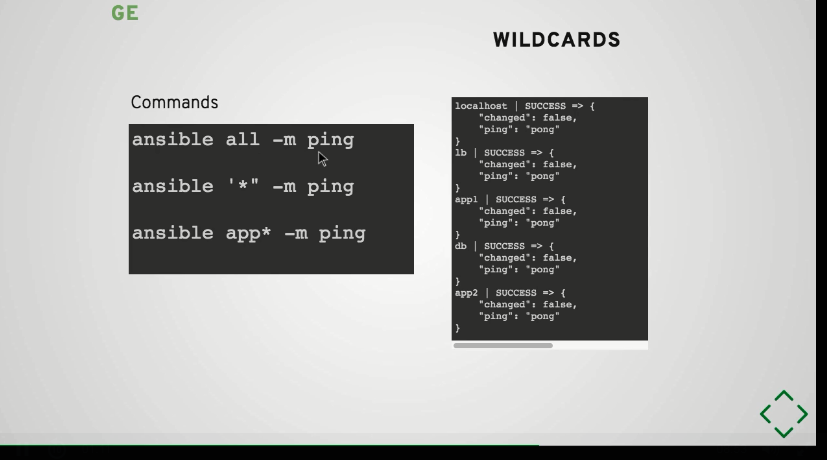


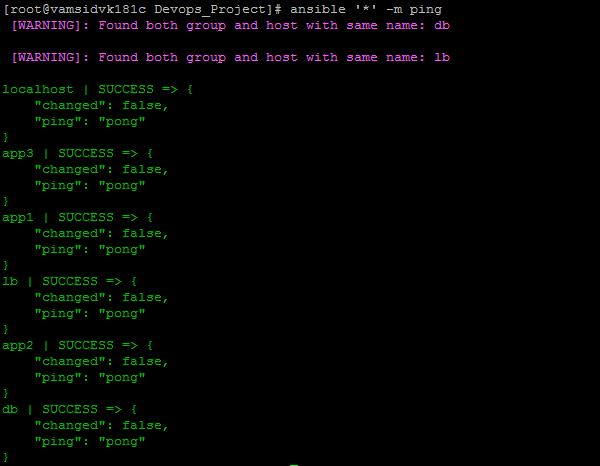
## Host Patterns and Ad Hoc Commands



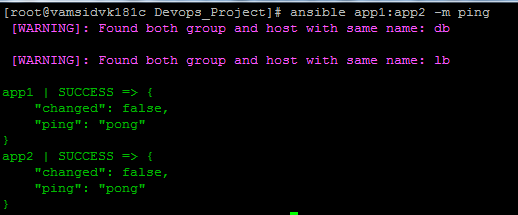


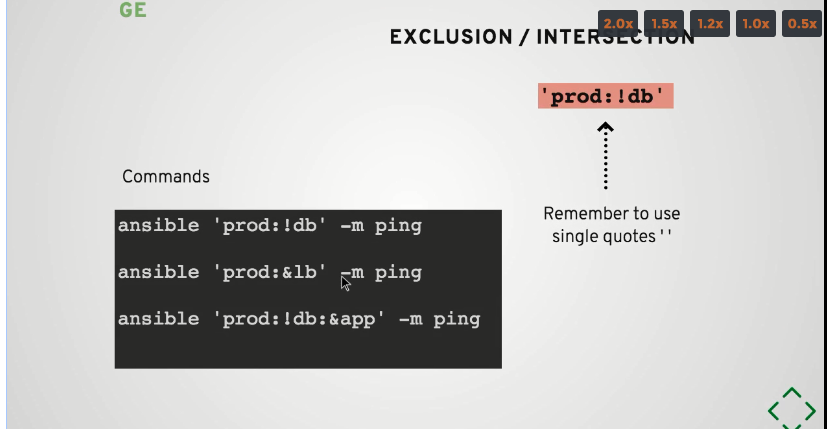
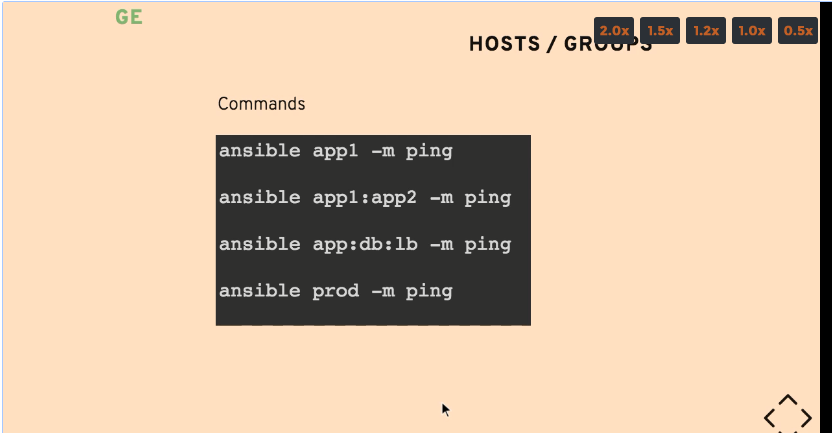






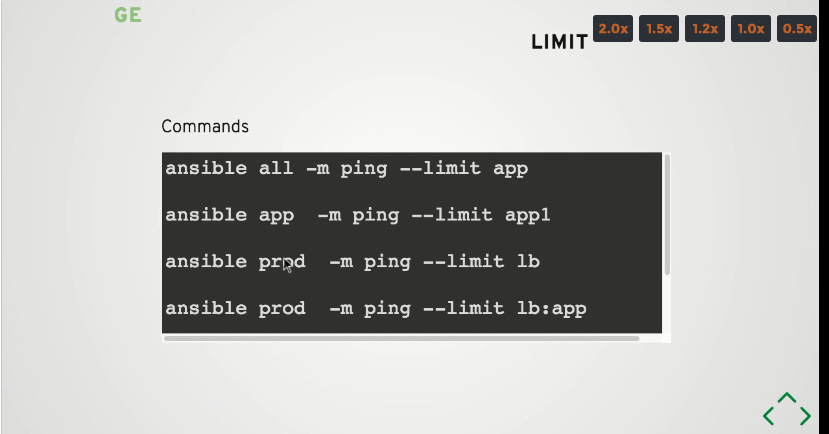




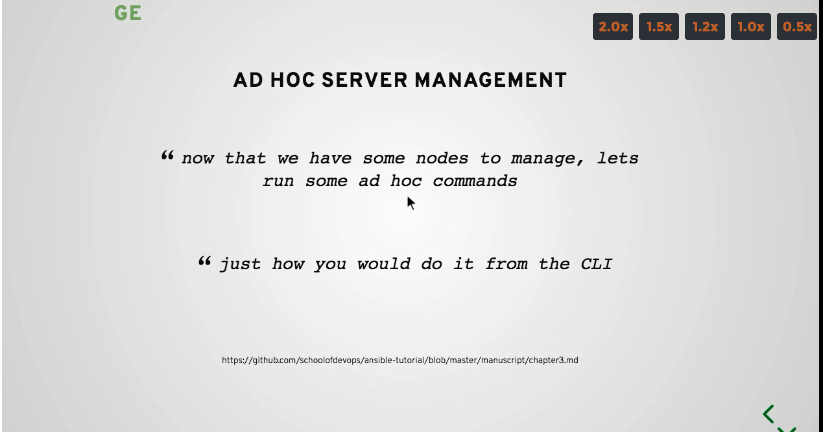




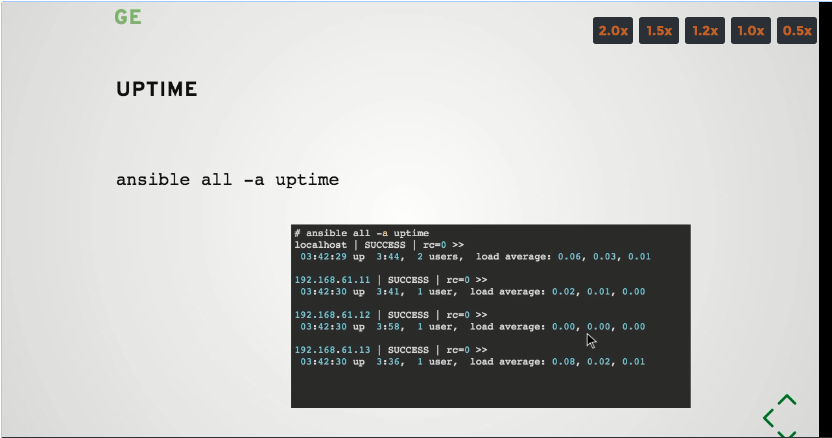


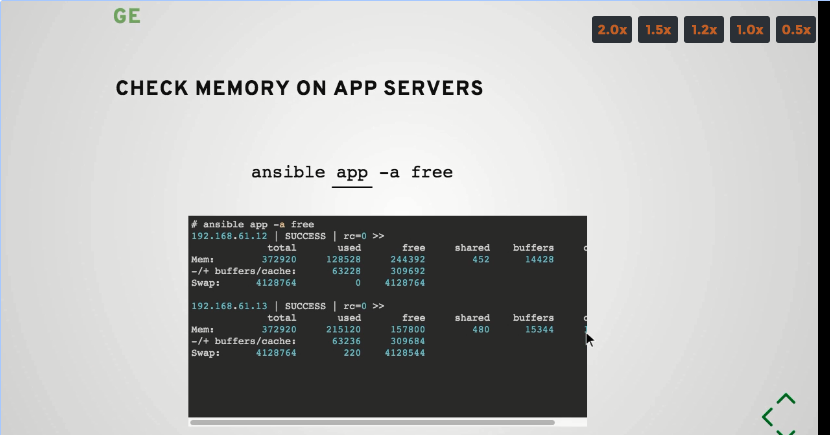


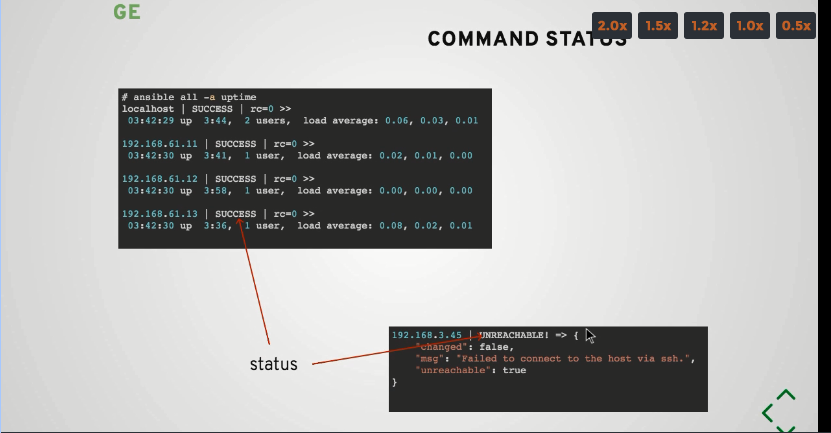


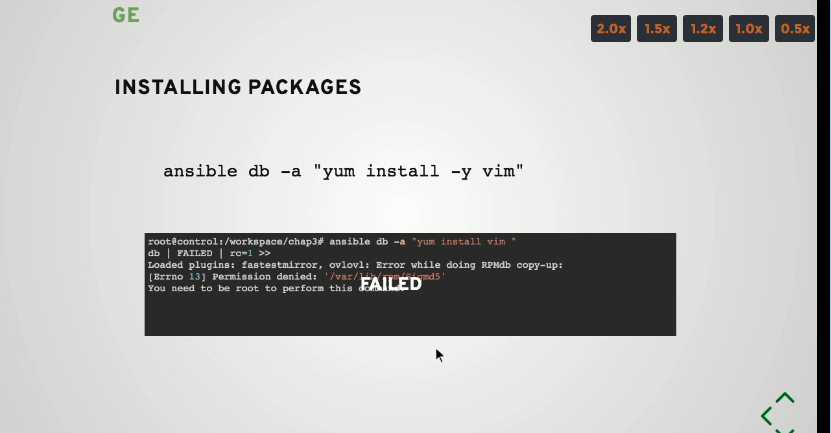






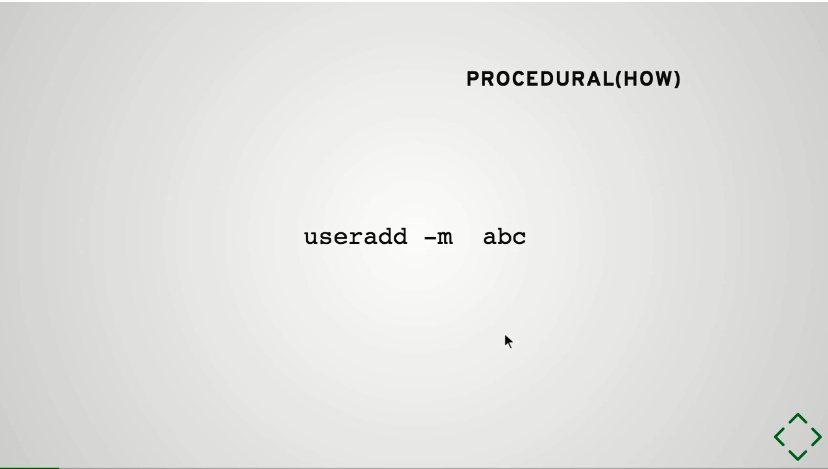




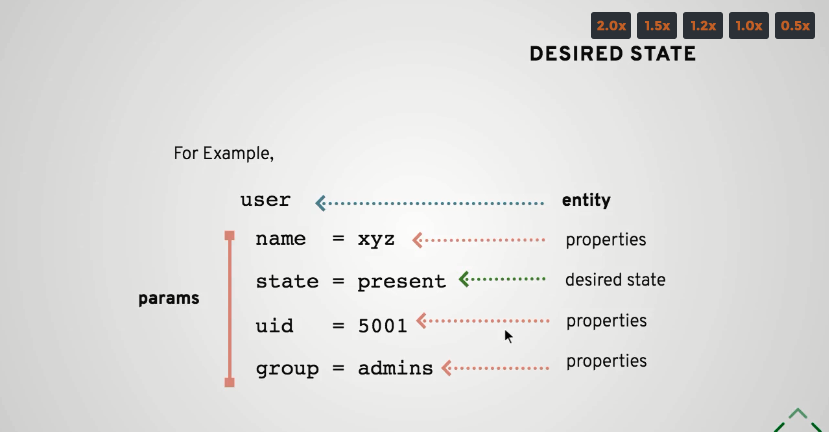
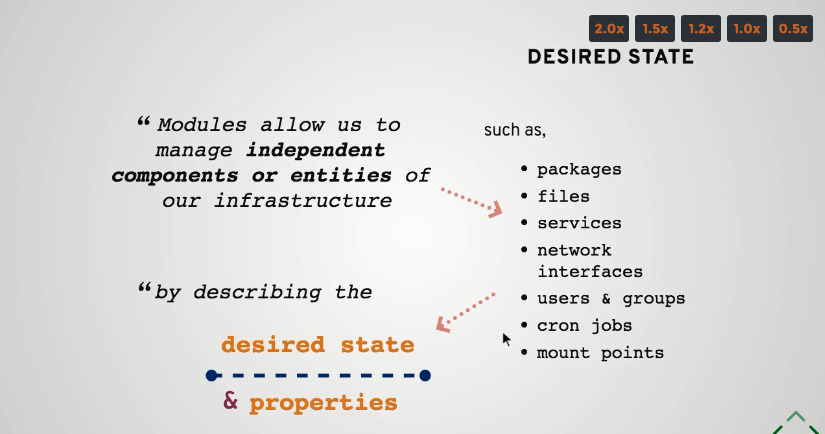


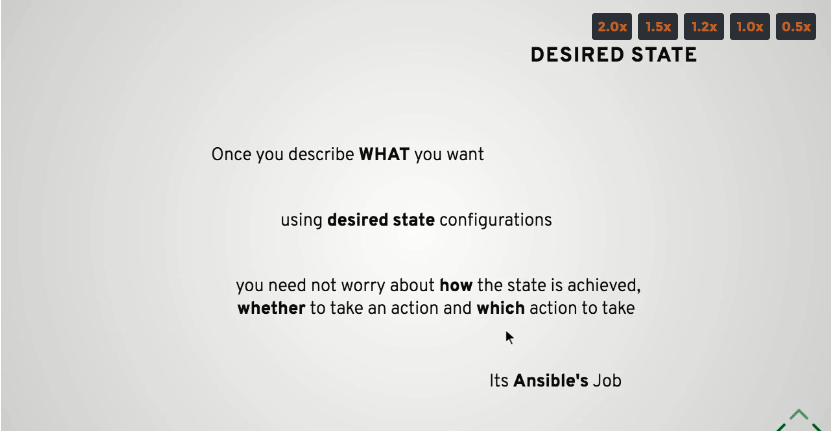
## Modules and Desired State Configuration

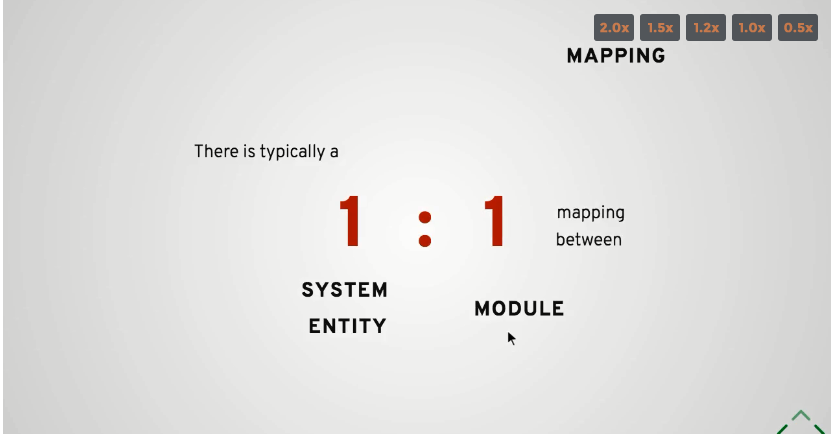


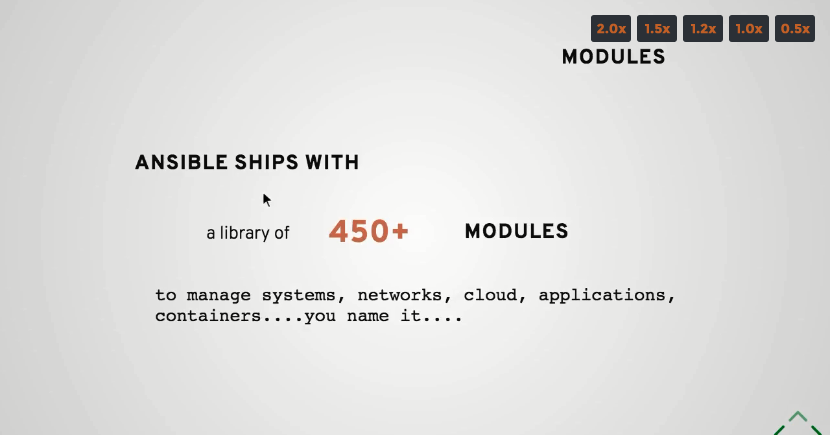


If we use the above command again & again it would fail because most of the linux system command are not idempotent. To overcome this that’s where ansible modules comes into play

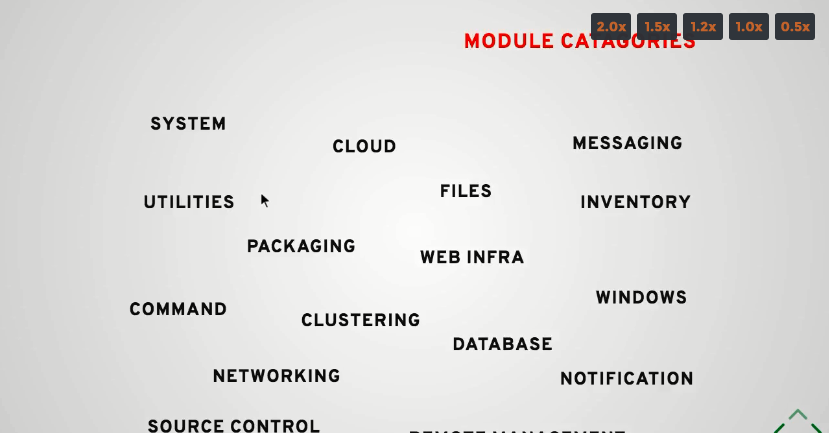








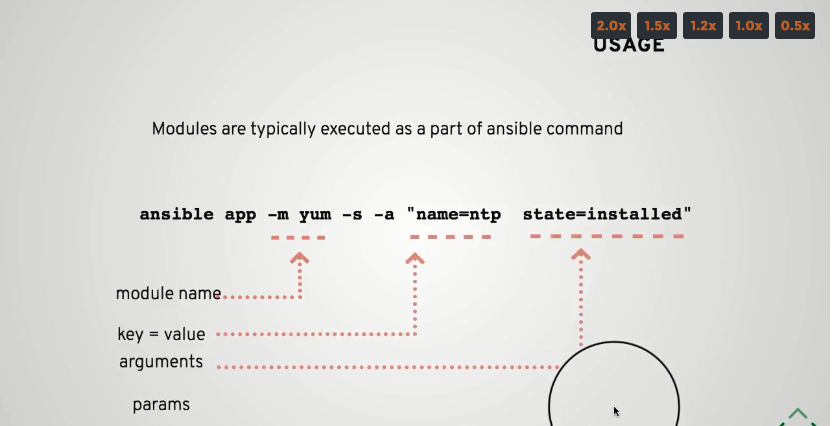
Actually it is not 450 currently it is 1350 + modules for different type of operations as below

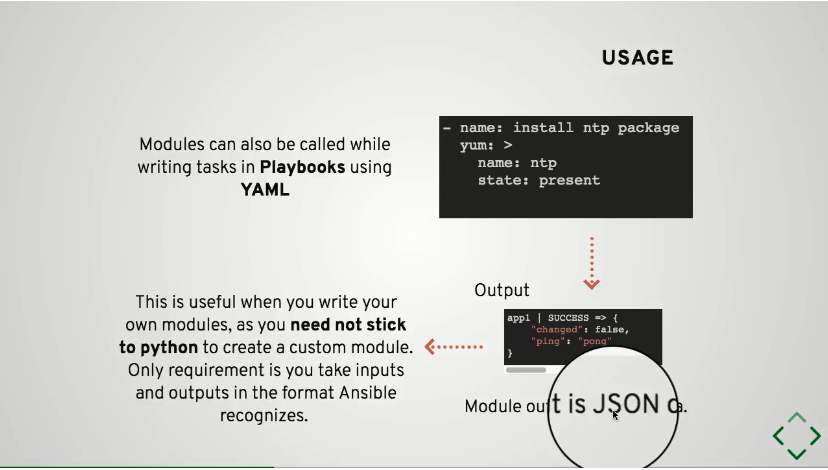


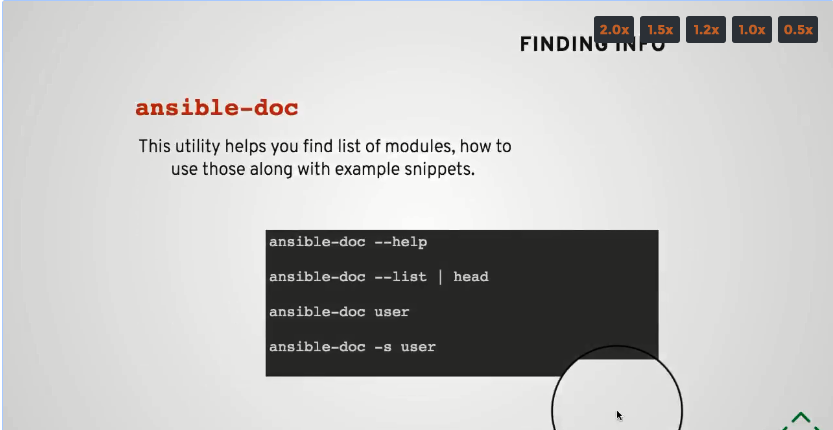


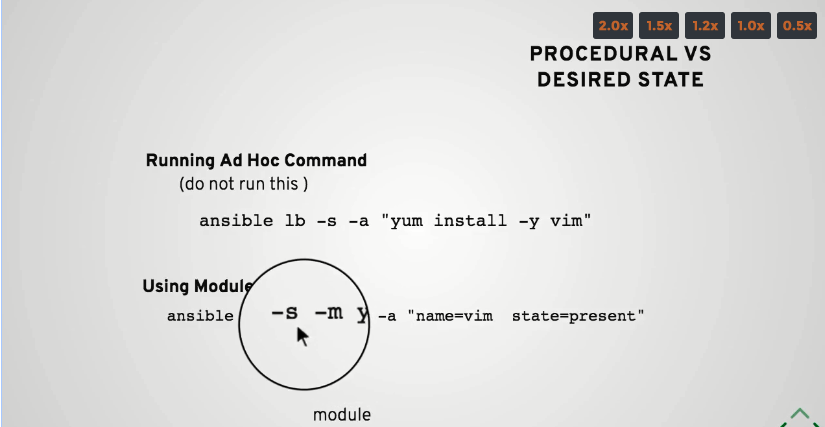
Above are the no.of modules as of 2.6.7 version

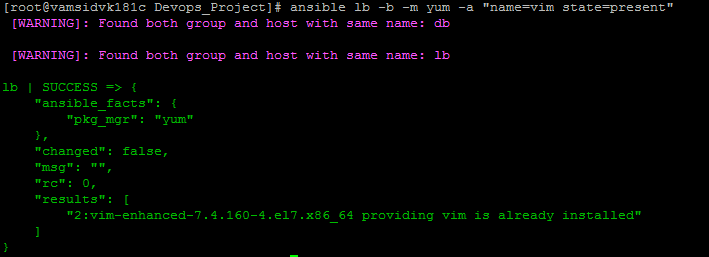
## Invoking Modules, Idempotence

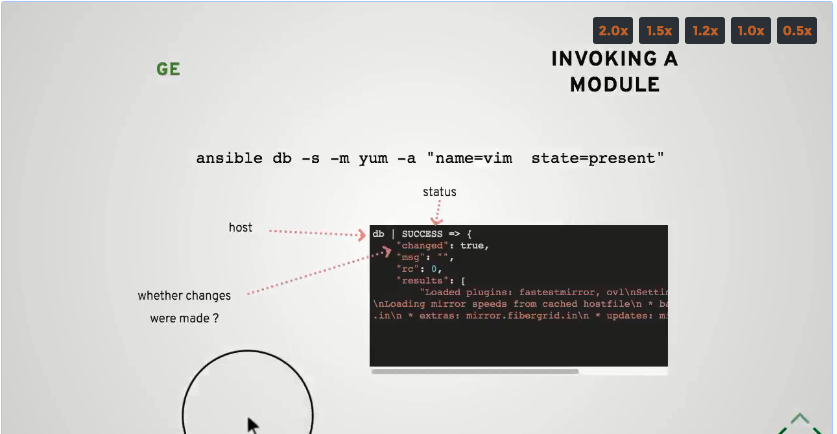


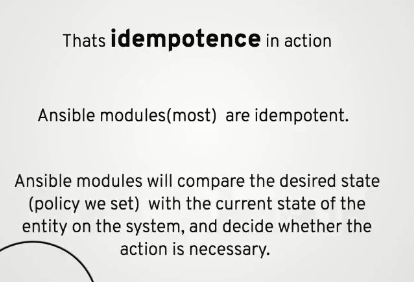




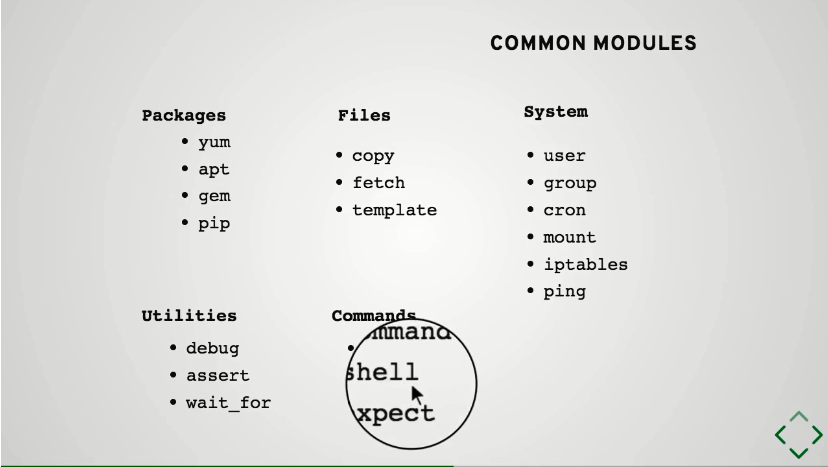


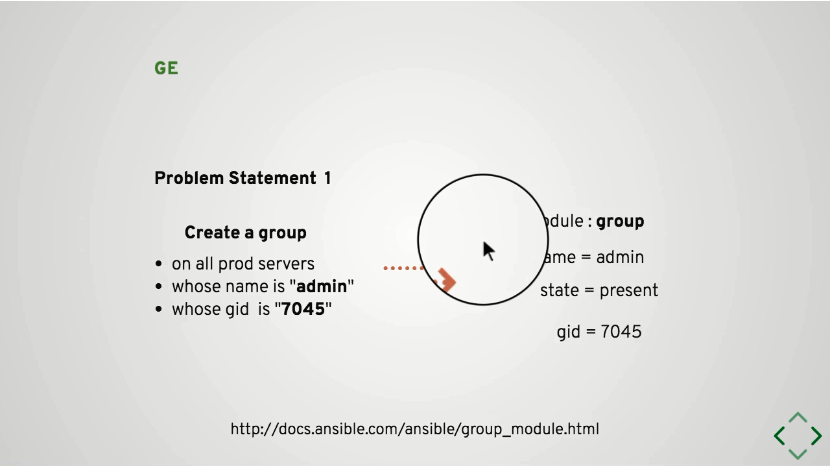


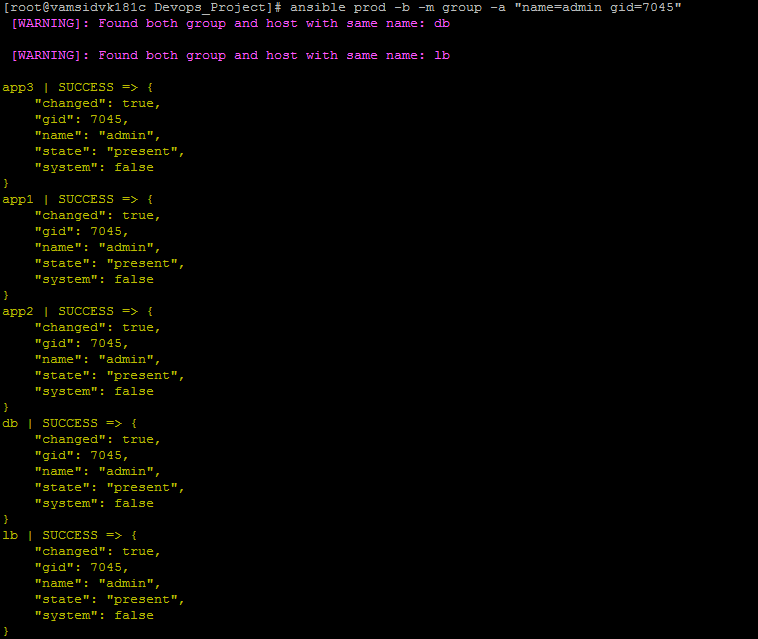


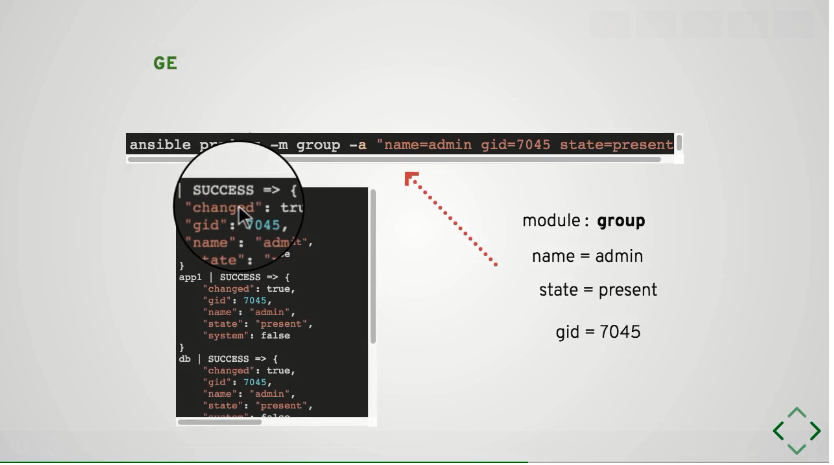


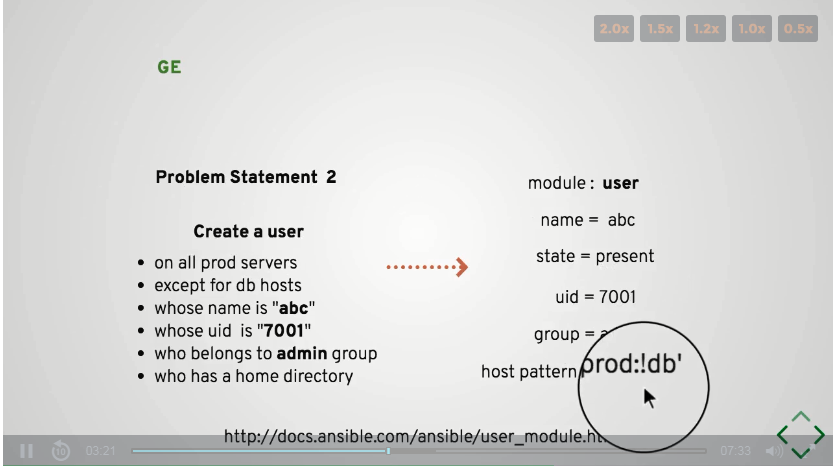
## Using common modules

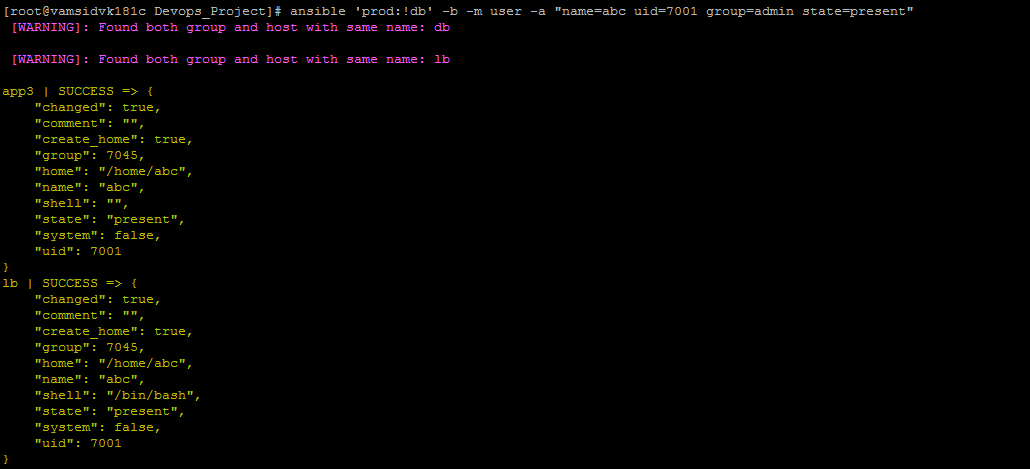


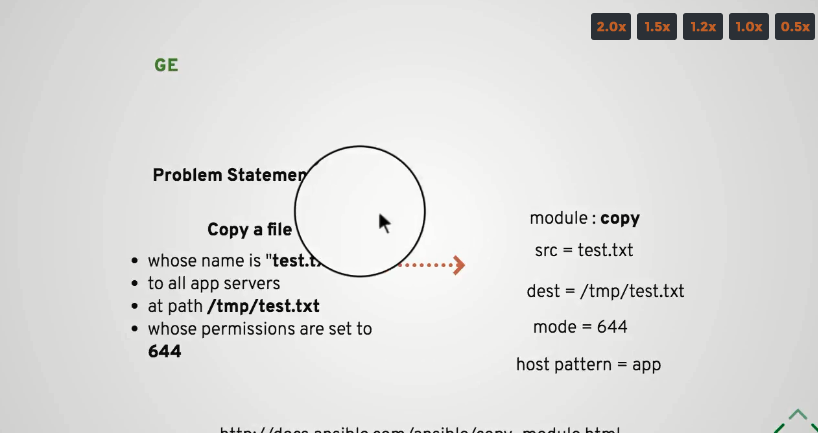






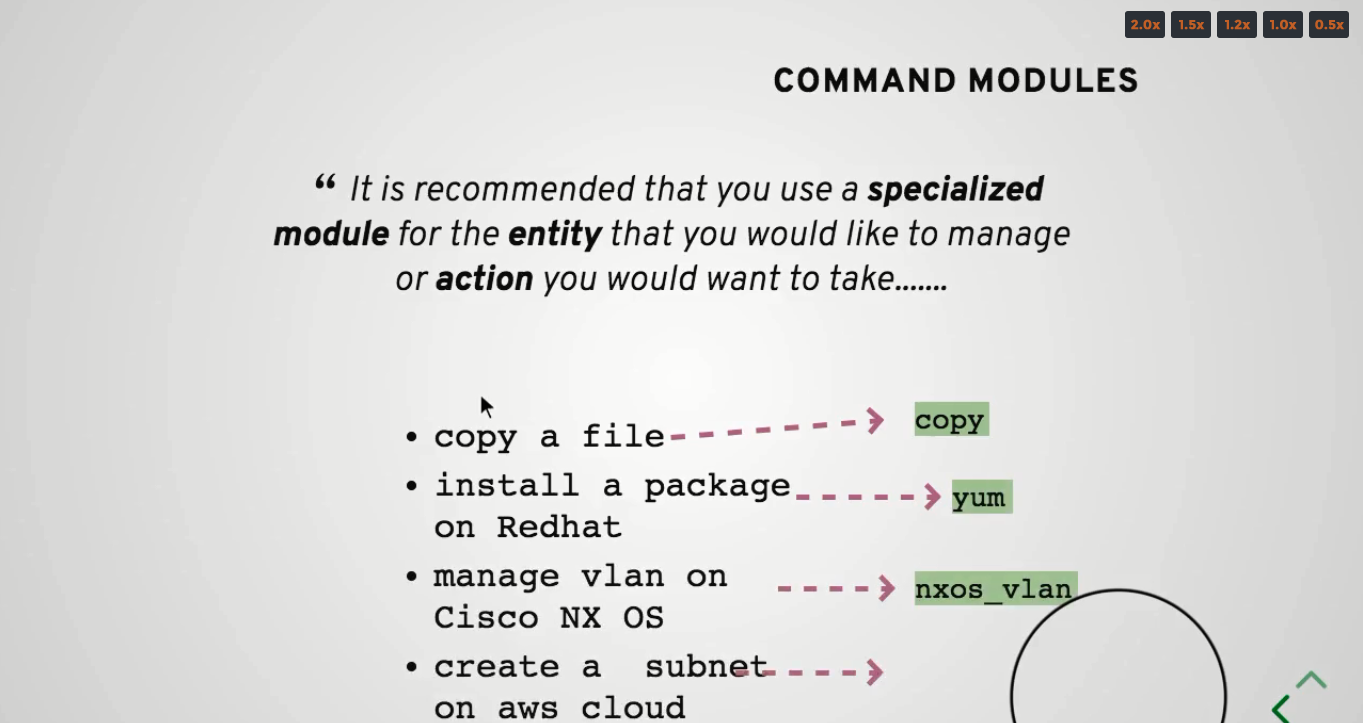


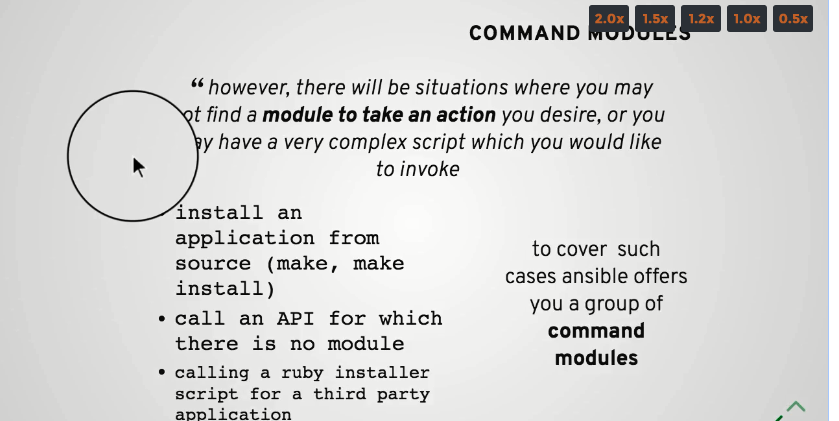


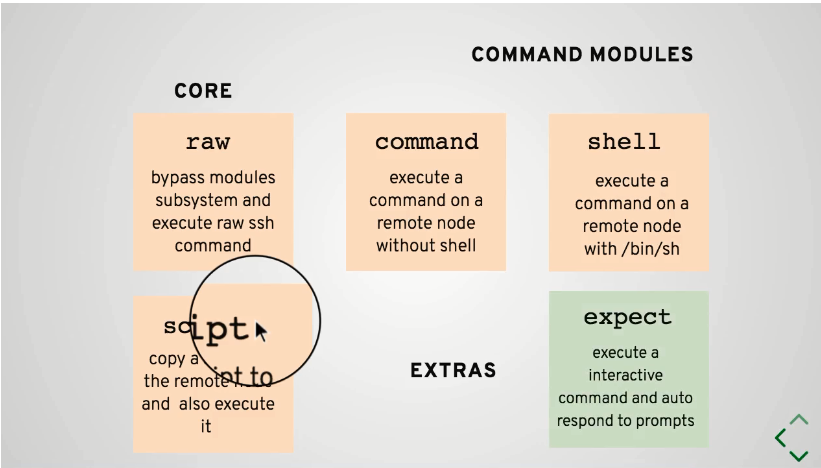




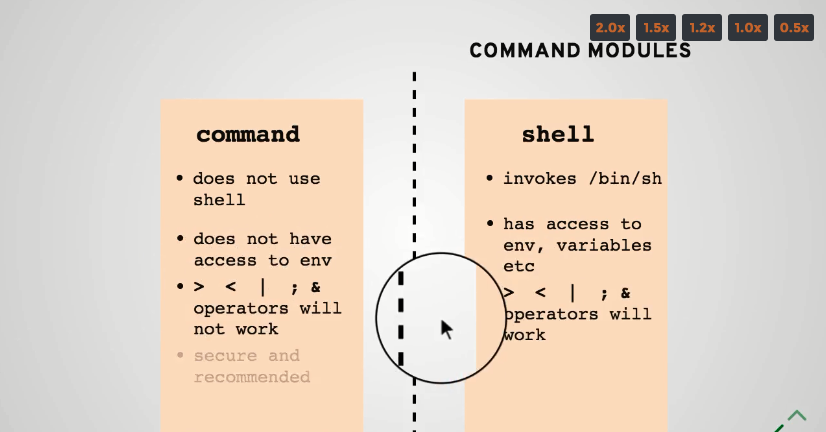
## Command Modules and Idempotence



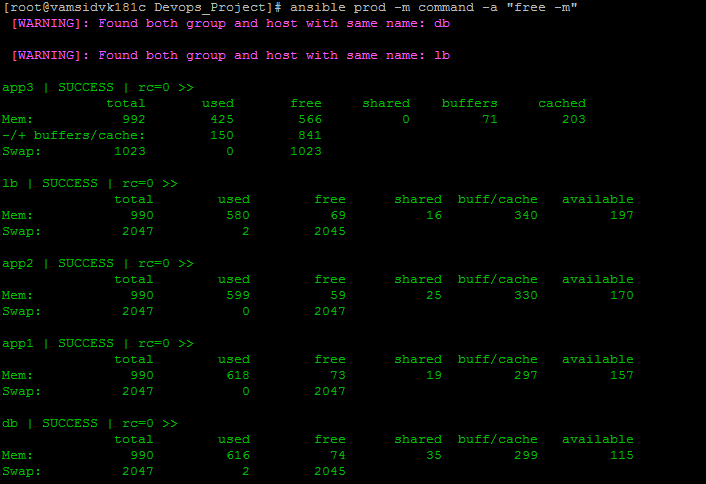


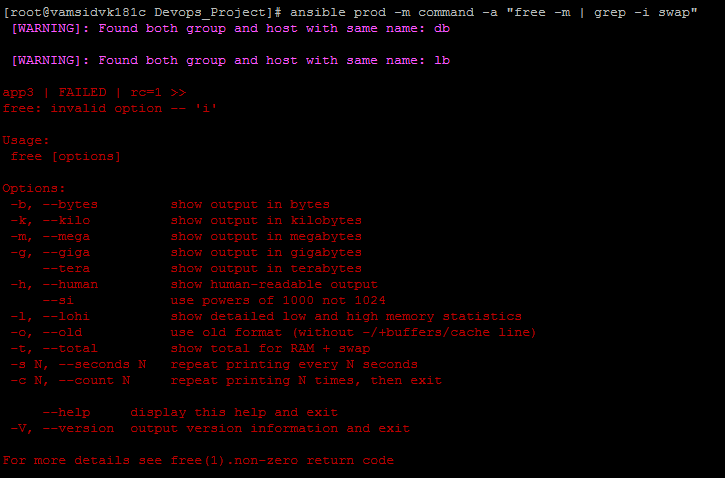


Raw module is useful when you don’t have python installed on the remote host EX: network devices

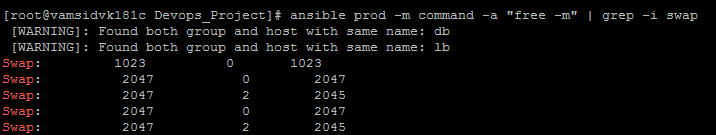




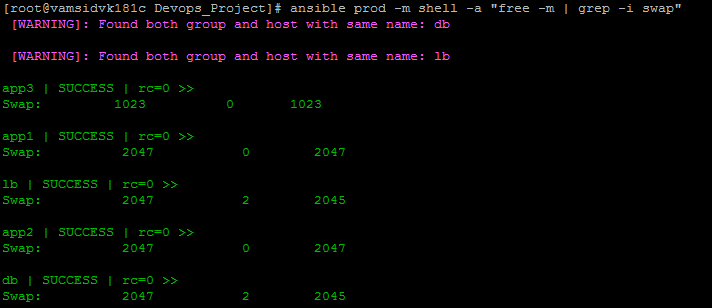




Above pic states that | (piping) will not work in the command modules. To overcome below shows it



Grep is happening on control node



You can also use shell module to grep the result on the remote node rather control node

.

