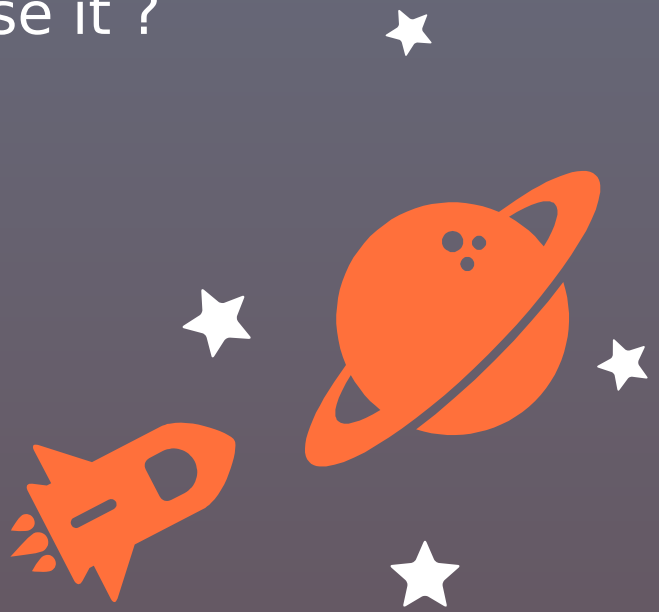
What are we going to see in this session?

- What is output and how to use it ?
- Types of execution
- Best Practice



What is output and how to use it?

- Terraform keeps attributes of all the resources you create.
- Those outputs can be queried and outputted.

- Example :
  - ✓ The [aws_instance] resource has the attribute of [public_ip] which can be extracted as part of your script.

```
output “ip_address” {
  value = “${aws_instance.example.public_ip}”
```

  - ✓ Resource type : aws_instance
  - ✓ Resource name : Example
  - ✓ Attribute name : public_ip

- This can be useful to output variable information or to feed information to external software.

Code : git-repo [file path : Terraform/Codes/E8_output

Types of Execution

- terraform apply : Which is going to print the output which you requested for.
- terraform output : You can do terraform output any point in time even after apply which will print the results.
- terraform output ip_address : You can also fetch outputs separately by calling the name.
- terraform output >> file.txt : You can extract the output to a file.

Let’s assume you are using Ansible for your application deployment, you can just extract these output to Ansible host file for further playbook deployments.

- ✓ [Stage 1 : Deploy the machine]
- ✓ [Stage 2 : Extract the IP’s of newly created machine to Ansible inventory file]
- ✓ [Stage 3 : Trigger your Ansible playbook]
- terraform output ip_address >> /etc/ansible/hosts : You can extract the IP address to ansible inventory file like this.

Best Practice

- One way is to keep all your output arguments in [main.tf] file
- Another option is to keep output arguments in separate file called [output.tf]
- Github link which has many examples for output.tf

End of this topic !

Any questions?