**Terraform**

1. What all resources have you created using terraform?
2. How to deal with already created resources?

We use **terraform import** for the same.

<https://chatgpt.com/share/66fa1bdc-08a0-800c-8c52-0b661565d602>

1. Where to store state file?

By default, it is stored locally. It is recommended to store it in remote location (remote backend). We can use terraform cloud as well which is built-in feature.

1. What if we lost terraform state file?

We can recover the resources using “**terraform import**”. Also, terraform cloud is automatically taking backup of the state so we can retrieve state file from there.

1. Explain command: **terraform validate**

<https://chatgpt.com/share/66fa20d7-9f58-800c-83dd-1959fec58109>

1. Explain terraform lifecycle block:

<https://chatgpt.com/share/66fa21ab-afe8-800c-a122-e01a4ba97a9e>

1. How to use terraform destroy to destroy a specific resource?

We can use - **terraform destroy -target=RESOURCE\_TYPE.RESOURCE\_NAME**

<https://chatgpt.com/share/66fa324c-14a0-800c-a699-867fa621849fs>

1. Explain terraform profile (for aws) –

<https://chatgpt.com/share/66fa3468-5770-800c-9858-815e143a4ce5>

1. Explain terraform modules:

<https://chatgpt.com/share/66fa36f1-3f14-800c-be84-c7a7bac0e059>

1. Explain terraform workspaces:

<https://chatgpt.com/share/66fa3758-5c4c-800c-bb5b-3bc518346a93>

1. Explain terraform variables:

<https://chatgpt.com/share/66fa3823-aa88-800c-b55d-b8aea87fd98d>

1. Provisioners in terraform: