

# Instructions for the demo

---

Create a directory backend and save the resource.tf, providers.tf and variables.tf file

First run and it will create the s3 bucket and dynamodb table and terraform state is stored locally.

```
terraform init
terraform validate
terraform plan -out "backend.tfplan"
terraform apply "backend.tfplan"
terraform show
```

Second run - Add the backend configuration to the resource\_s3.tf file (resource\_s3.tf.rename)

```
// terraform backend
terraform {
  backend "s3" {
    bucket      = "sample-st-bucket-april"
    key         = "globalstate/s3/terraform.tfstate"
    region      = "eu-west-1"
    # Replace this with your DynamoDB table name!
    dynamodb_table = "my-first-db-locks"
    encrypt      = true
  }
}
```

**To tell terraform to use s3 for storing the state file, execute the following commands again.**

```
terraform init
```

**Note:** Terraform will automatically detect that already there is a state file locally and prompt you to copy it to the new S3 backend. If you type in "yes," backend "s3" will get configured.

**Add the outputs.tf file in the backend directory and execute the below commands.**

```
terraform plan -out "backend.tfplan"
terraform apply "backend.tfplan"
```

**Note:** Points to highlight

- Explain terraform state in detail and discuss how to ensure that state file will be not be corrupted.
- Discuss how to share state file with multiple team members or use it for different environments like dev, QA, pre-prod etc.
- What is remote state storage or backends ? Refer to <https://www.terraform.io/docs/backends/index.html> to understand backends
- How can s3 be used as remote state backend? Refer to <https://www.terraform.io/docs/backends/types/s3.html> to understand s3 backend.
- Explain what are workspaces and the pros and cons of it. Execute the commands for terraform workspace. Refer to <https://www.terraform.io/docs/state/workspaces.html> to understand workspaces