GitHub Repository

Create a new repository
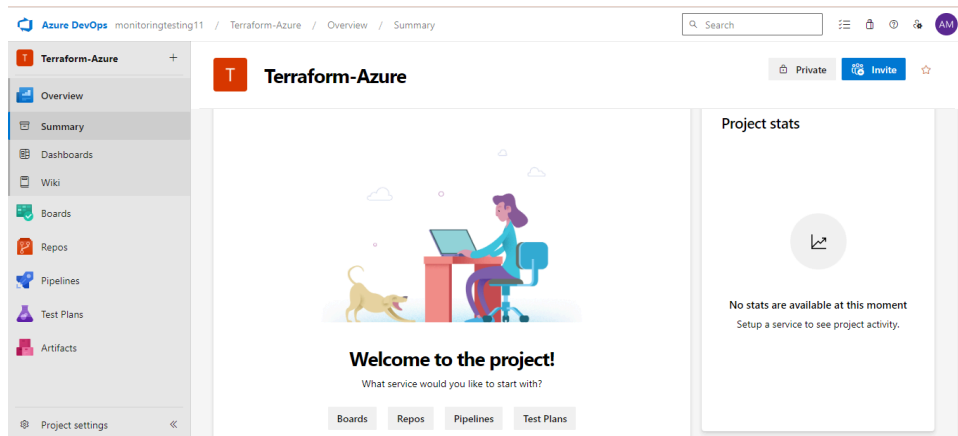
Add your main.tf file there.

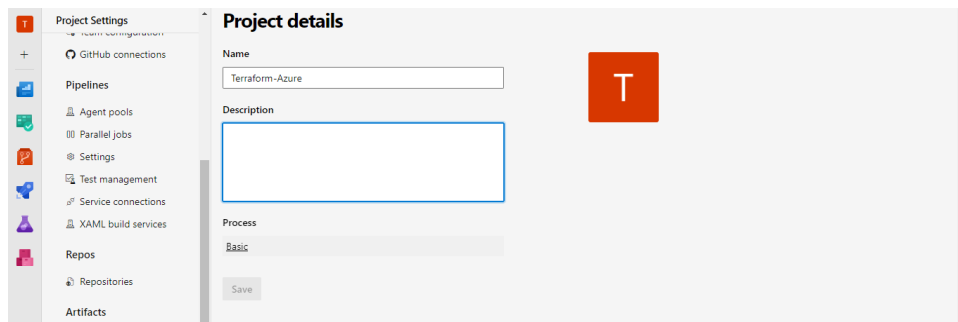-------------------------

----
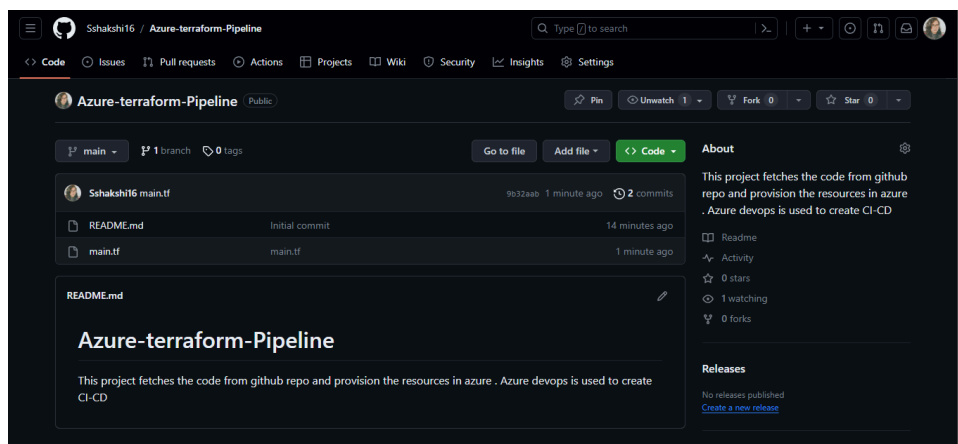
Azure DevOps Pipeline

Firstly, we need to create a Service Principal Name (SPN) to allow our Azure DevOps Organization project to deploy our environment.
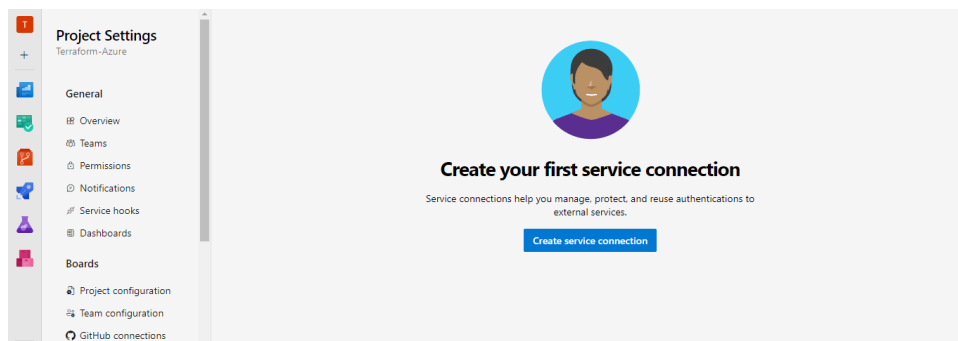
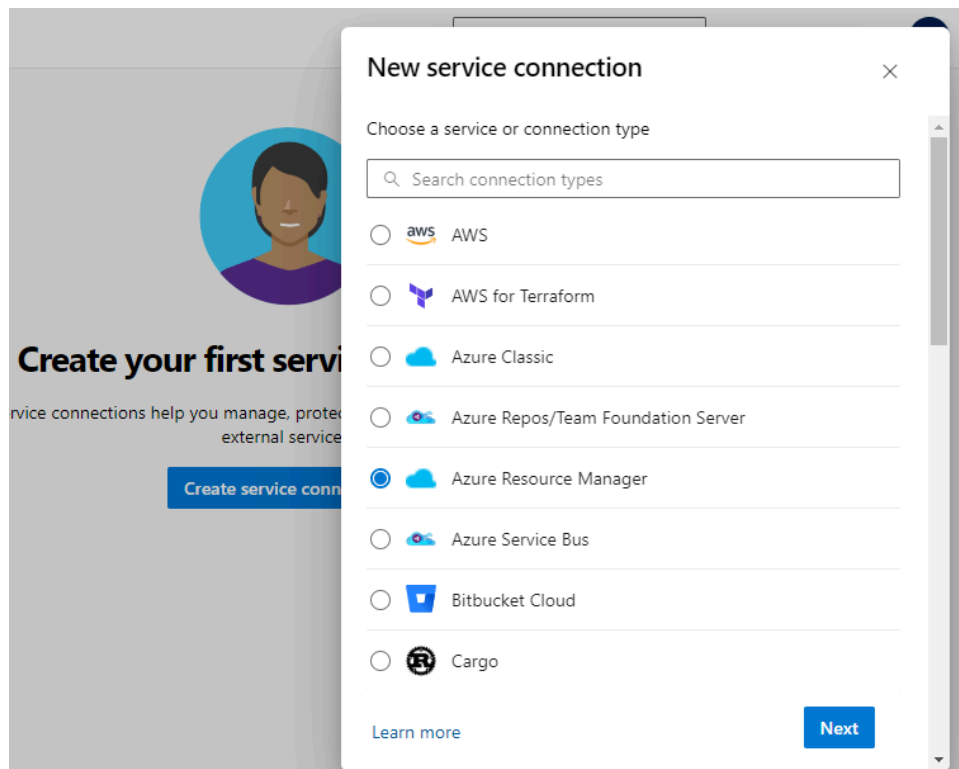Within our Azure DevOps Project, navigate to Project Settings
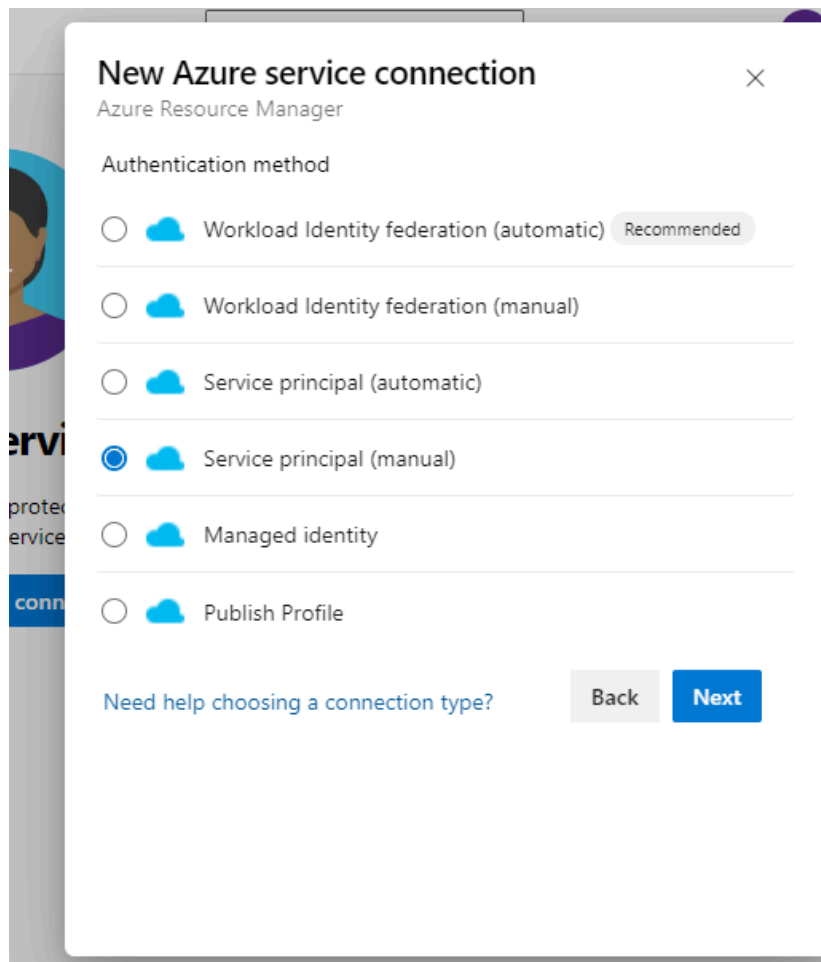
Click on   Service Connections.

Click on create your first service connection

Click on 'Azure Resource Manager' -> 'Next'.

Then select 'Service principal (automatic)' -> 'Next'.

These are the scope settings used for my SPN:

Select scope level -> Subscription

Resource group -> Select resource group according to your choice

Service connection name -> Give any suitable name to the connection
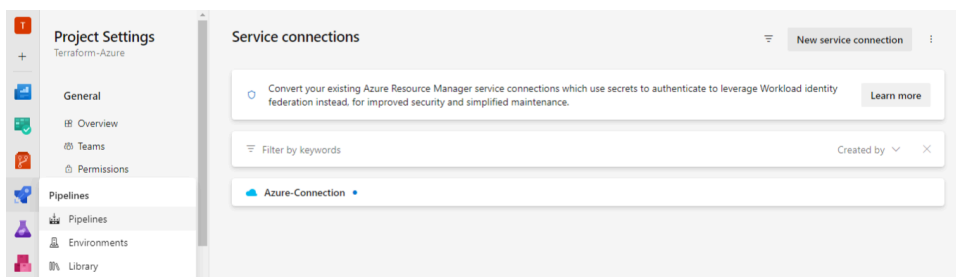
Check the Grant access permission to all pipelines
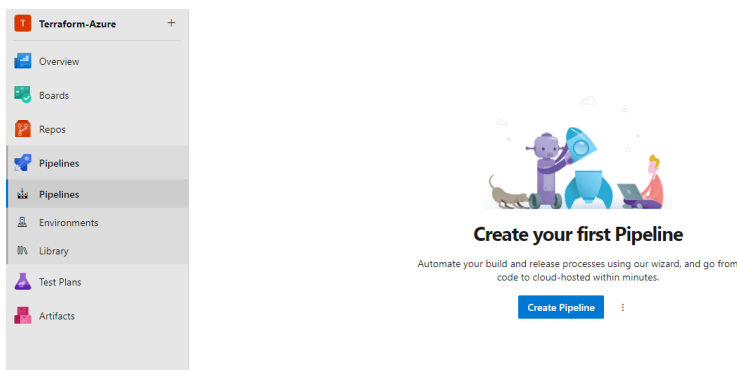Click Save

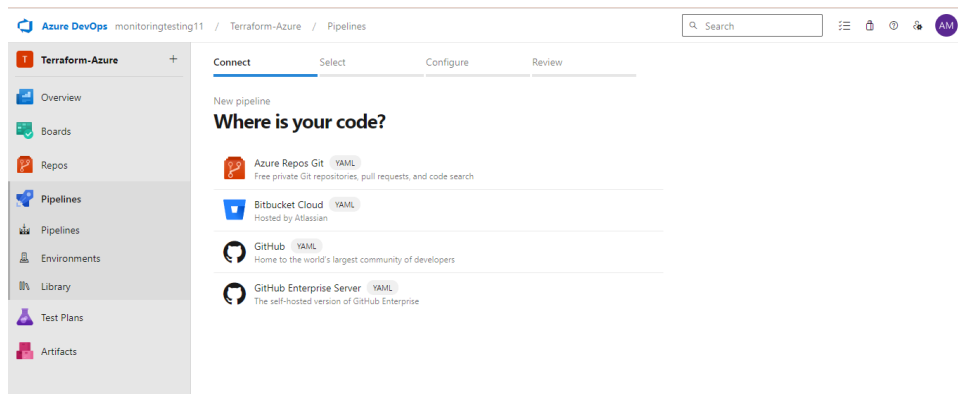Connection created successfully. Our connection name is Azure-Connection
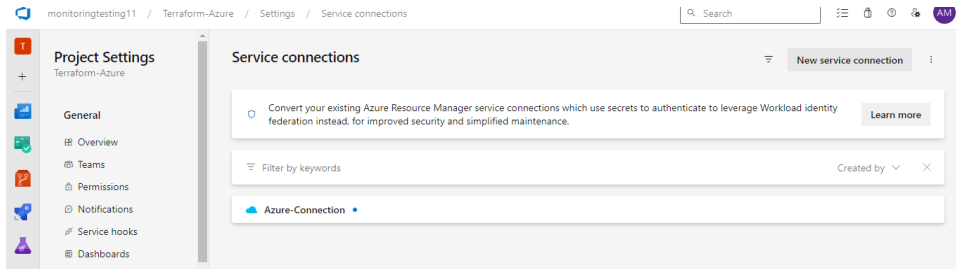
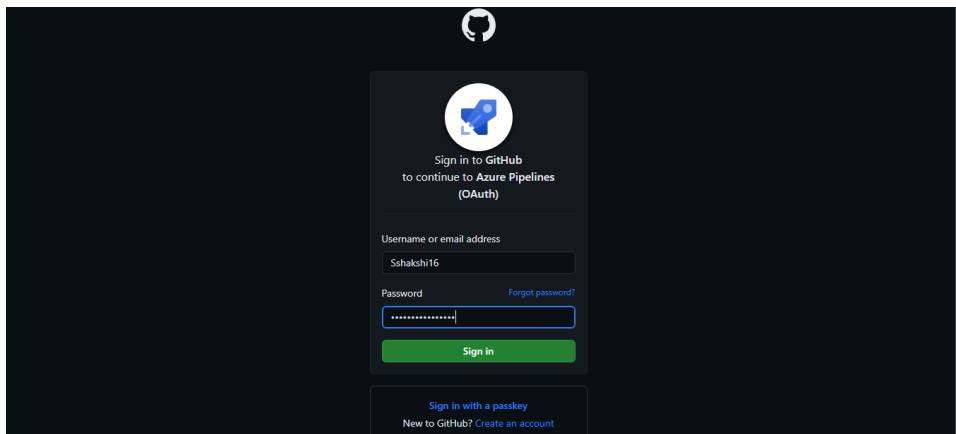Selecting 'Pipelines' -> 'Pipelines'.

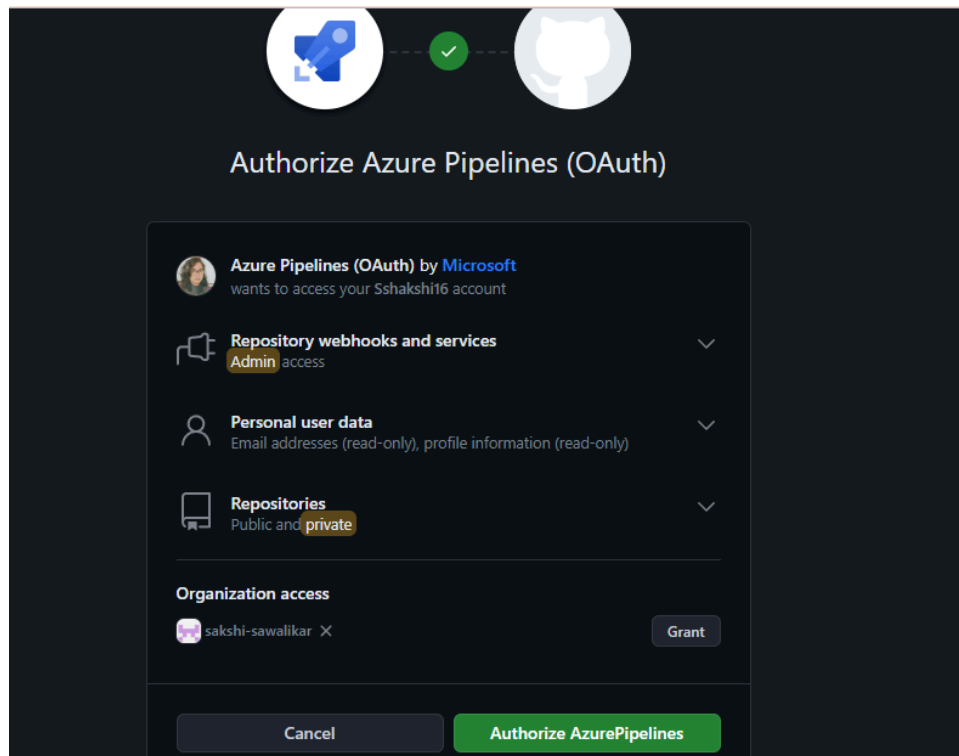Click create pipeline

Selecting GitHub

Provide credentials -> Press Sign in
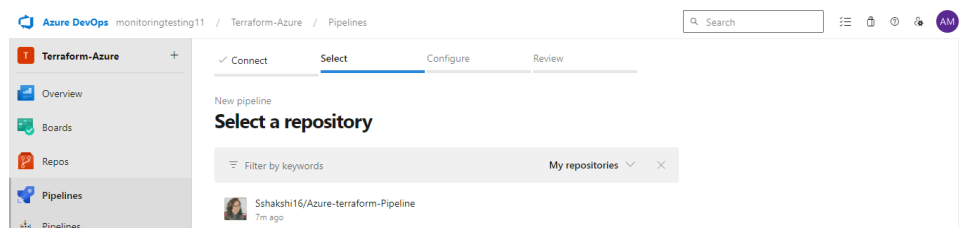

Click Authorise AzurePipelines

Now after authentication one will see all the GitHub repos.

Select Azure-terraform-pipeline as our main.tf file is present in that repository.

Scroll down to 'Repository Access' and selecting the repo, then clicking 'Approve and Install'.
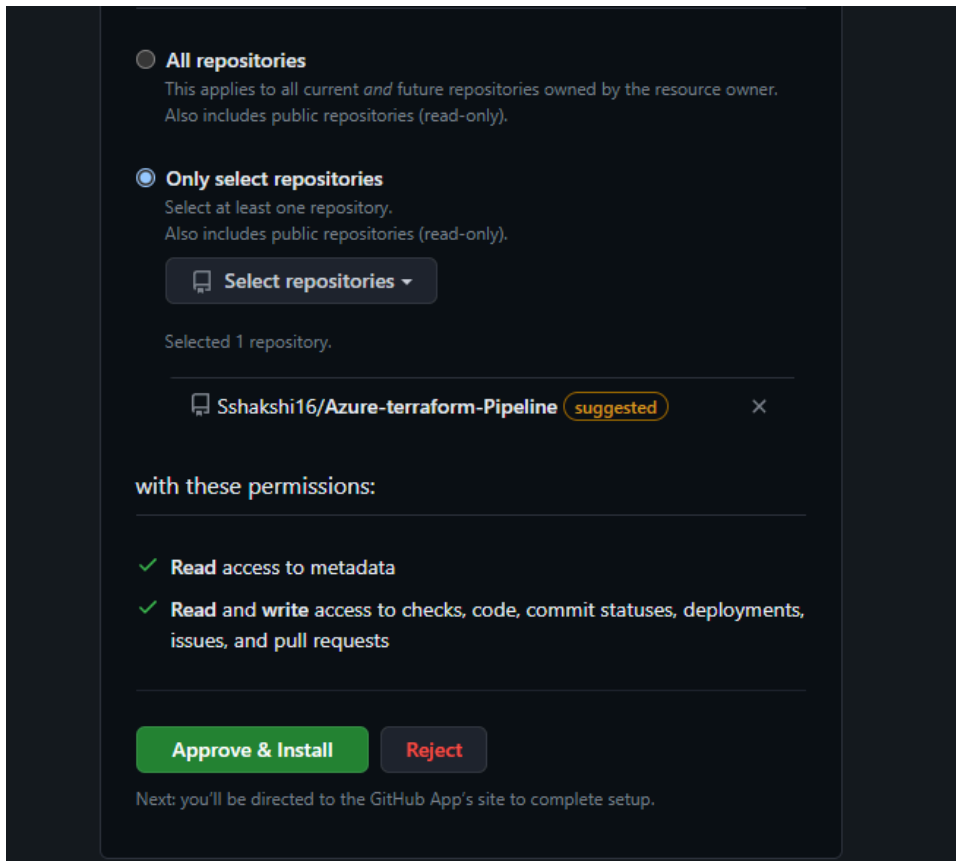
Click starter pipeline
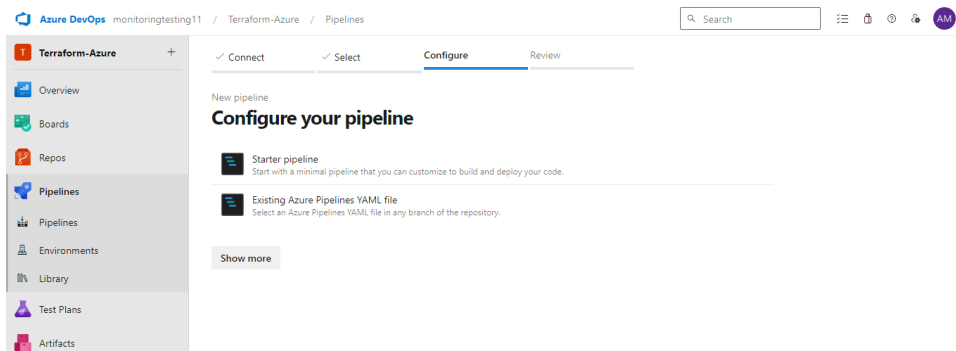
Now clicking show assistant

Searching terraform installer

Click add

Now selecting terraform CLI

Selecting init command -> Click add

Again selecting terraform CLI

Selecting plan

Again selecting terraform CLI

Selecting apply

azure-pipelines.yml

Starter pipeline

```yaml
# Start with a minimal pipeline that you can customise to build
and deploy your code.

# Add steps that build, run tests, deploy, and more:

# https://aka.ms/yaml

trigger:
- main

pool:
  vmImage: ubuntu-latest

steps:
- script: echo Hello, world!
  displayName: 'Run a one-line script'
- task: JasonBJohnson.azure-pipelines-tasks-terraform.azure-pipelines-tasks-terraform-installer.TerraformInstaller@0
  displayName: 'Install Terraform'
  inputs:
        terraformVersion: 'latest'

- task: TerraformCLI@2
  inputs:
        command: 'version'
        allowTelemetryCollection: true
- task: TerraformCLI@2
  inputs:
        command: 'init'
        allowTelemetryCollection: true
- task: TerraformCLI@2
```

```yaml
    inputs:
        command: 'plan'
        environmentServiceName: 'Azure-connection-recom'
        runAzLogin: true
        allowTelemetryCollection: true
- task: TerraformCLI@2
  inputs:
        command: 'apply'
        environmentServiceName: 'Free
Trial(ec88c61c-3230-434f-9bf2-a6eff5fe3e48)'
        runAzLogin: true
        allowTelemetryCollection: true


- script: |
        echo Add other tasks to build, test, and deploy your project.
        echo See https://aka.ms/yaml
  displayName: 'Run a multi-line script'
```
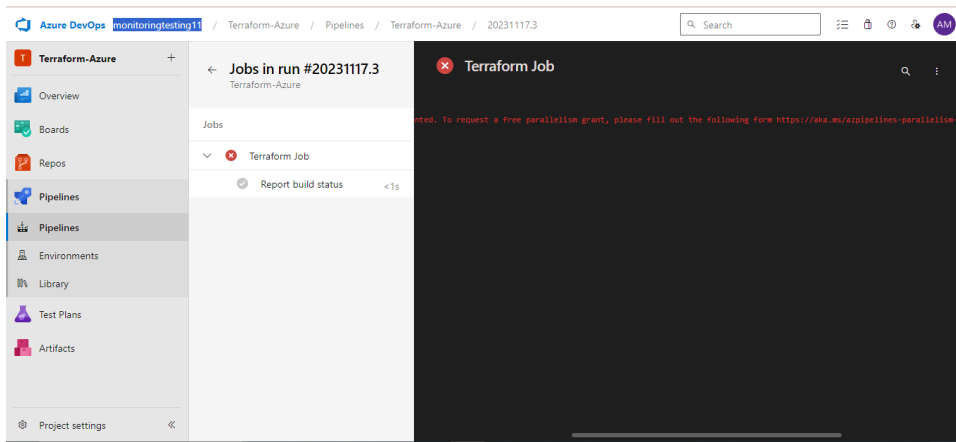
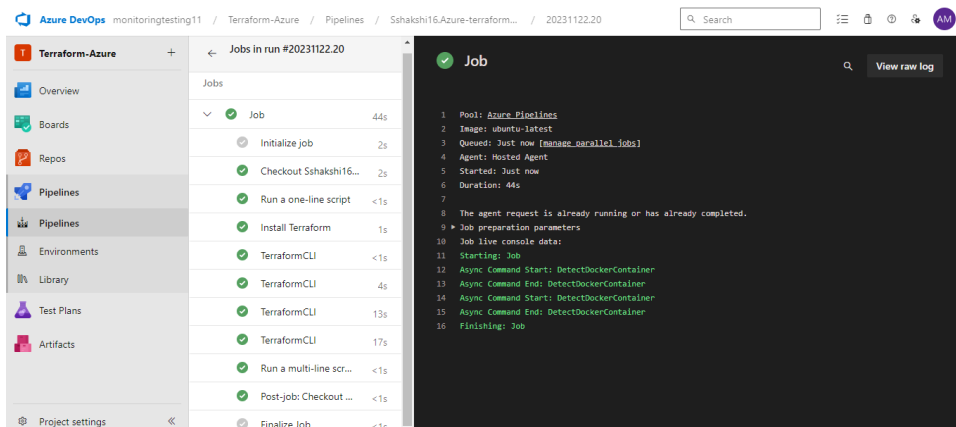---------------

Click save and run

Error

Filling the form by clicking on the link , it took me 2-3 business days to get approval .

After that running the pipeline again

Here resource group created successfully

Resource groups
Default Directory (monitoringtesting11outlook.onmicrosoft.com)

+ Create　⚙ Manage view ∨　🔄 Refresh　⬇ Export to CSV　🔗 Open query　|　🏷 Assign tags

| Filter for any field... | Subscription equals **all** | Location equals **all** ✕ | + Add filter |

Showing 1 to 4 of 4 records.

No grouping ∨　　List view ∨

☑ Name ↑↓	Subscription ↑↓	Location ↑↓	
☐ cloud-shell-storage-centralindia	Free Trial	Central India	···
☐ example-resourcegroup	Free Trial	West Europe	···
☑ example-resources	Free Trial	West Europe	···
☐ NetworkWatcherRG	Free Trial	West Europe	···