

AWS Setup Steps and main.tf for assignment-terra-workspace

AWS Steps to Prepare for Terraform Backend:

1. Create an S3 Bucket:

- Go to AWS Management Console.
- Navigate to S3 service.
- Create a bucket named: assignment-terra-workspace-terraform-state
- Enable "Versioning" for the bucket.
- Keep other settings as default or as per your security needs.

2. Create a DynamoDB Table:

- Go to AWS Management Console.
- Navigate to DynamoDB service.
- Create a table named: assignment-terra-workspace-lock
- Set Partition key: LockID (Type: String).
- Leave other configurations as default.

Terraform main.tf Code

```
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 5.0"
    }
  }
}

backend "s3" {
  bucket         = "assignment-terra-workspace-terraform-state"
  key             = "envs/PLACEHOLDER/terraform.tfstate"
  region         = "us-east-1"
  dynamodb_table = "assignment-terra-workspace-lock"
  encrypt        = true
}

provider "aws" {
  region = "us-east-1"
}

resource "random_string" "suffix" {
```

```

length = 6
upper  = false
special = false
}

variable "instance_ami" {
  description = "AMI ID for EC2 instance"
  type        = string
}

variable "instance_type" {
  description = "EC2 instance type"
  type        = string
}

variable "bucket_prefix" {
  description = "Prefix for the S3 bucket"
  type        = string
}

locals {
  workspace      = terraform.workspace
  s3_bucket_name = "${var.bucket_prefix}-${random_string.suffix.result}"
}

resource "aws_s3_bucket" "workspace_bucket" {
  bucket = local.s3_bucket_name

  tags = {
    Name           = local.s3_bucket_name
    Environment    = local.workspace
  }
}

resource "aws_instance" "workspace_instance" {
  ami           = var.instance_ami
  instance_type = var.instance_type

  tags = {
    Name           = "${local.workspace}-assignment-terra-workspace-instance"
    Environment    = local.workspace
  }
}

```