

Project Title

Deployment of Website on AWS S3 using Terraform

Project Overview:

This project is to deploy a website on Amazon Web Services (AWS) S3 using Terraform. Terraform, an Infrastructure as Code (IaC) tool, will enable consistent and repeatable infrastructure deployment, making the process efficient and scalable. This project will leverage AWS S3's robust, scalable, and highly available object storage service to host and deliver web content.

Step 1: Provider Configuration & Bucket Creation:

The screenshot shows the VS Code interface with the main.tf file open in the editor. The code defines an AWS provider and creates two S3 objects: 'index.html' and 'style.css'. The segments for creating these objects are highlighted with red boxes.

```
1 terraform {  
2   required_providers {  
3     aws = {  
4       source = "hashicorp/aws"  
5       version = "5.59.0"  
6     }  
7   }  
8 }  
9  
10 provider "aws" {  
11   region = "ap-south-1"  
12 }  
13  
14 resource "random_id" "rand_id" {  
15   byte_length = 8  
16 }  
17  
18 resource "aws_s3_bucket" "mywebapp-bucket" {  
19   bucket = "mywebapp-bucket-${random_id.rand_id.hex}"  
20 }  
21  
22 resource "aws_s3_object" "index_html" {  
23   bucket = aws_s3_bucket.mywebapp-bucket.bucket  
24   source = "./index.html"  
25   key = "index.html"  
26 }  
27  
28 resource "aws_s3_object" "style_css" {  
29   bucket = aws_s3_bucket.mywebapp-bucket.bucket  
30   source = "./style.css"  
31   key = "style.css"  
32 }  
33  
34 output "name" {  
35   value = random_id.rand_id.hex  
36 }  
37
```

The screenshot shows the terminal tab in VS Code displaying the output of the 'terraform init' command. It shows the initialization process, including the download of provider plugins.

```
PS C:\02-terraform\TF-AWS\Project-1> terraform init  
Initializing the backend...  
Initializing provider plugins...  
- Finding hashicorp/aws versions matching "5.59.0"...  
- Finding latest version of hashicorp/random...  
- Installing hashicorp/random v3.6.2...  
- Installed hashicorp/random v3.6.2 (signed by HashiCorp)  
- Installing hashicorp/aws v5.59.0...
```

The screenshot shows the terminal tab in VS Code displaying the output of the 'terraform validate' command. It confirms that the configuration is valid.

```
27  
28   resource "aws_s3_object" "style_css" {  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\02-terraform\TF-AWS\Project-1> terraform validate  
Success! The configuration is valid.  
PS C:\02-terraform\TF-AWS\Project-1>
```

```

 20 }
 21
 22 resource "aws_s3_object" "index_html" {
 23   bucket = aws_s3_bucket.mywebapp-bucket.bucket
 24   source = "./index.html"
 25   key = "index.html"
 26 }
 27
 28 resource "aws_s3_object" "style_css" {

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Plan: 4 to add, 0 to change, 0 to destroy.

Changes to Outputs:

- + name = (known after apply)

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

random_id.rand_id: Creating...
random_id.rand_id: Creation complete after 0s [id=WGy_id88dPU]
aws_s3_bucket.mywebapp-bucket: Creating...
aws_s3_bucket.mywebapp-bucket: Creation complete after 2s [id=mywebapp-bucket-586cbf89df3c74f5]
aws_s3_object.style_css: Creating...
aws_s3_object.index_html: Creating...
aws_s3_object.index_html: Creation complete after 0s [id=index.html]
aws_s3_object.style_css: Creation complete after 0s [id=style.css]

Apply complete! Resources: 4 added, 0 changed, 0 destroyed.

Outputs:

Amazon S3

Account snapshot - updated every 24 hours All AWS Regions

Storage lens provides visibility into storage usage and activity trends. [Learn more](#)

[View Storage Lens dashboard](#)

[General purpose buckets](#) [Directory buckets](#)

General purpose buckets (1) [Info](#) All AWS Regions

are containers for data stored in S3.

[Find buckets by name](#)

Name	AWS Region	IAM Access Analyzer	Creation date
mywebapp-bucket-586cbf8	Asia Pacific (Mumbai) ap-south-1	View analyzer for ap-south-1	July 28, 2024, 14:12:27 (UTC+05:30)

Bucket Creation and Successfully uploaded Website Files

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

Objects (2) [Info](#)

[C](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions ▾](#)

[Create folder](#)

Objects are the fundamental units of data in S3. To upload objects, you'll need to provide a key, content, and optional metadata.

[Find objects](#)

You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to grant them permission to do so.

Name	Type	Last modified	Size	Storage class
index.html	html	July 28, 2024, 14:12:29 (UTC+05:30)	3.5 KB	Standard
style.css	css	July 28, 2024, 14:12:29 (UTC+05:30)	4.1 KB	Standard

Here is the HTML & CSS Files

Open the index.html file

S for

X index.html [Info](#)

[Copy S3 URI](#) [Download](#) [Open](#) [Object actions ▾](#)

[Properties](#) [Permissions](#) [Versions](#)

Object overview

Owner a4bae9be903ca86ecef6b44fb24cff8a5a8e7e779c806c3e6c90658e1 7efe62e	S3 URI s3://mywebapp-bucket-586cbf89.../index.html
AWS Region Asia Pacific (Mumbai) ap-south-1	Amazon Resource Name (ARN) arn:aws:s3:::mywebapp-bucket-586cbf89.../index.html
Last modified July 28, 2024, 14:12:29 (UTC+05:30)	Entity tag (Etag) 3fcda4230b8199f86ef1f1320dfddaba
Size 3.5 KB	Object URL https://mywebapp-bucket-586cbf89...s3.ap-south-1.amazonaws.com/index.html
Type html	

“Click on website URL”

Acess Denied because the Public access block is ON.

← → G mywebapp-bucket-586cbf89...s3.ap-south-1.amazonaws.com/index.html

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied</Message>
  <RequestId>2CK1ZC4J9YHJ0QQZ</RequestId>
  <HostId>Ye3RaQ3ngyPIoxw06dd593X1jNMh3SjJ5Mvuo8k8QZXikefiL1A+FSP3/9V80LUMkkRdoc5pgdU=</HostId>
</Error>
```

Amazon S3 X

Buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3

Block Public Access settings for this account

▼ Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings

Permissions overview

Access finding
Access findings are provided by IAM external access analyzers. Learn more about [How IAM analyzer finds external access](#) [View analyzer for eu-north-1](#)

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access grants, and so on. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. If you want to grant public access only to this bucket and its access points, turn on Block public access at the bucket level. AWS recommends that you turn on Block all public access, but be aware that this setting prevents users from accessing your buckets or objects without proper authentication. If you require some flexibility, you can customize the individual settings below to suit your specific storage needs. [more ▾](#)

Block all public access

On

▶ Individual Block Public Access settings for this bucket

Bucket policy

Step 2: Public Access:

- Configures public access to the bucket.

54 matching results

- aws_s3_bucket_metric
- aws_s3_bucket_notification
- aws_s3_bucket_object
- aws_s3_bucket_object_lock_configuration
- aws_s3_bucket_ownership_controls
- aws_s3_bucket_policy
- aws_s3_bucket_public_access_block
- aws_s3_bucket_replication_configuration
- aws_s3_bucket_request_payment_configuration
- aws_s3_bucket_server_side_encryption_configuration
- aws_s3_bucket_versioning
- aws_s3_bucket_website_configuration
- aws_s3_directory_bucket

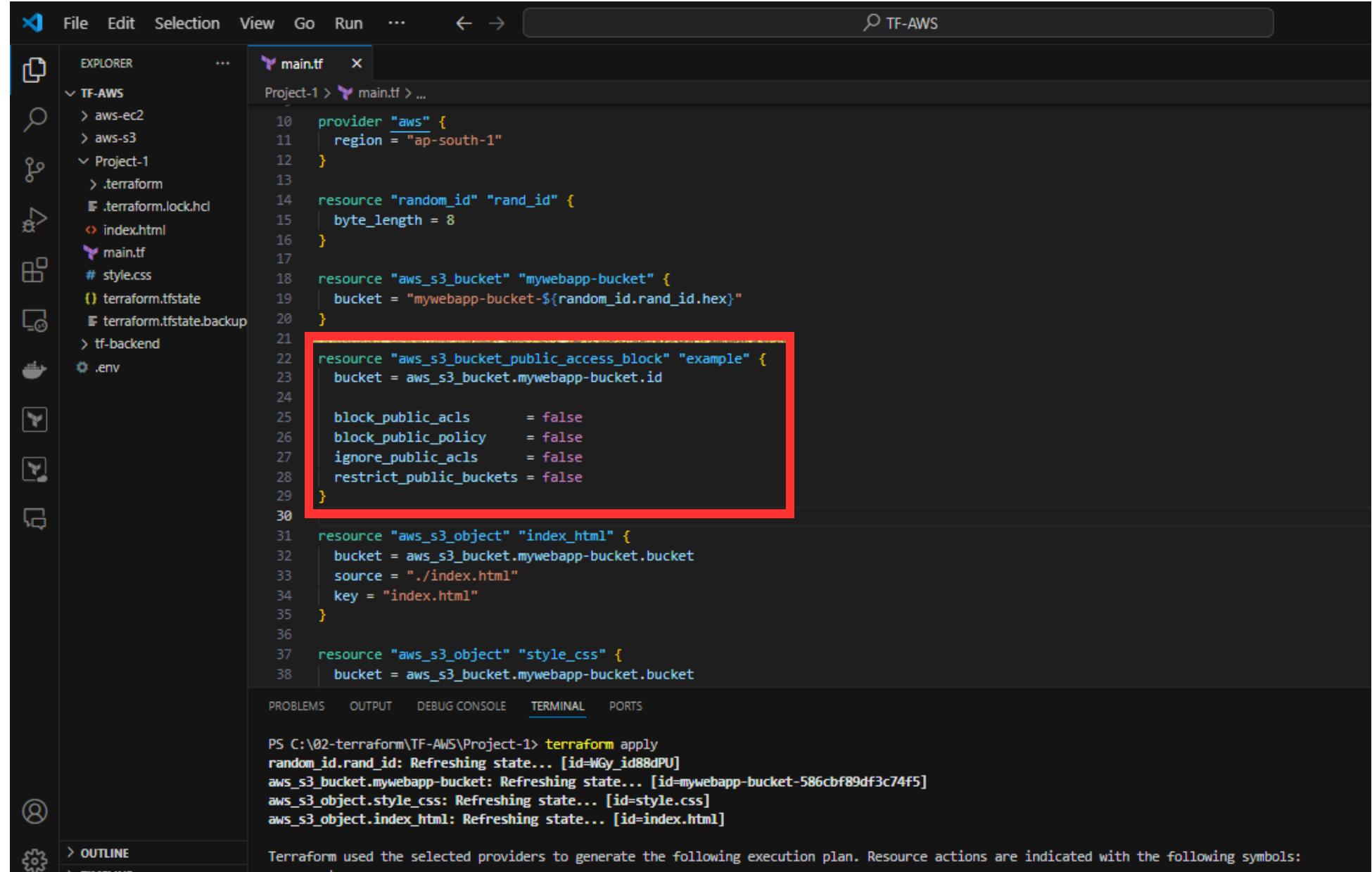
Example Usage

```
resource "aws_s3_bucket" "example" {
  bucket = "example"
}

resource "aws_s3_bucket_public_access_block" "example" {
  bucket = aws_s3_bucket.example.id

  block_public_acls      = true
  block_public_policy     = true
  ignore_public_acls     = true
  restrict_public_buckets = true
}
```

Argument Reference



```
provider "aws" {
  region = "ap-south-1"
}

resource "random_id" "rand_id" {
  byte_length = 8
}

resource "aws_s3_bucket" "mywebapp-bucket" {
  bucket = "mywebapp-bucket-${random_id.rand_id.hex}"
}

resource "aws_s3_bucket_public_access_block" "example" {
  bucket = aws_s3_bucket.mywebapp-bucket.id

  block_public_acls      = false
  block_public_policy     = false
  ignore_public_acls     = false
  restrict_public_buckets = false
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\02-terraform\TF-AWS\Project-1> terraform apply
random_id.rand_id: Refreshing state... [id=WGy_id88dPU]
aws_s3_bucket.mywebapp-bucket: Refreshing state... [id=mywebapp-bucket-586cbf89df3c74f5]
aws_s3_object.style_css: Refreshing state... [id=style.css]
aws_s3_object.index_html: Refreshing state... [id=index.html]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create
```

Permissions overview

Access finding

Access findings are provided by IAM external access analyzers. Learn more about [How IAM analyzer findings work](#)

[View analyzer for ap-south-1](#)

Block public access (bucket settings)

Edit

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

⚠ Off

▶ Individual Block Public Access settings for this bucket

Step 3: Bucket Policy:

AWS DOCUMENTATION

s3

54 matching results

- aws_s3_bucket_metric
- aws_s3_bucket_notification
- aws_s3_bucket_object
- aws_s3_bucket_object_lock_configuration
- aws_s3_bucket_ownership_controls
- aws_s3_bucket_policy**
- aws_s3_bucket_public_access_block
- aws_s3_bucket_replication_configuration
- aws_s3_bucket_request_payment_configuration
- aws_s3_bucket_server_side_encryption_configuration

Resource: aws_s3_bucket_policy

Attaches a policy to an S3 bucket resource.

Note

Policies can be attached to both S3 general purpose buckets and S3 directory buckets.

Example Usage

Basic Usage

```
resource "aws_s3_bucket" "example" {  
    bucket = "my-tf-test-bucket"  
}
```

View Go Run ... ← → ⌂ TF-AWS

main.tf

```
Project-1 > main.tf > resource "aws_s3_bucket_policy" "mywebapp" > policy  
22 resource "aws_s3_bucket_public_access_block" "example" {  
23     block_public_acls      = false  
24     block_public_policy     = false  
25     ignore_public_acls     = false  
26     restrict_public_buckets = false  
27 }  
28  
29  
30  
31 resource "aws_s3_bucket_policy" "mywebapp" {  
32     bucket = aws_s3_bucket.mywebapp-bucket.id  
33     policy = jsonencode(  
34         {  
35             Version = "2012-10-17",  
36             Statement = [  
37                 {  
38                     Sid = "PublicReadGetObject",  
39                     Effect = "Allow",  
40                     Principal = "*",  
41                     Action = "s3:GetObject",  
42                     Resource = "arn:aws:s3:::${aws_s3_bucket.mywebapp-bucket.id}/*"  
43                 }  
44             ]  
45         }  
46     )  
47 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Enter a value: yes  
aws_s3_bucket_policy.mywebapp: Creating...  
aws_s3_bucket_policy.mywebapp: Creation complete after 1s [id=mywebapp-bucket-586cbf89df3c74f5]  
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```

Step 4: Website Configuration:

- Sets up the bucket for static website hosting.

AWS DOCUMENTATION

s3

54 matching results

- aws_s3_bucket_request_payment_configuration
- aws_s3_bucket_server_side_encryption_configuration
- aws_s3_bucket_versioning
- aws_s3_bucket_website_configuration**
- aws_s3_directory_bucket
- aws_s3_object
- aws_s3_object_copy

✓ Data Sources

- aws_canonical_user_id
- aws_s3_bucket
- aws_s3_bucket_object

Type here to search

Resource: aws_s3_bucket_website_configuration

Provides an S3 bucket website configuration resource. For more information, see [Hosting Websites on S3](#).

Note

This resource cannot be used with S3 directory buckets.

Example Usage

With `routing_rule` configured

```
resource "aws_s3_bucket_website_configuration" "example" {
    bucket = aws_s3_bucket.example.id
```

Copy

```
45 }
46 )
47 }
48
49 resource "aws_s3_bucket_website_configuration" "mywebapp" {
50     bucket = aws_s3_bucket.mywebapp-bucket.id
51
52     index_document {
53         suffix = "index.html"
54     }
55 }
56
57 resource "aws_s3_object" "index_html" {
58     bucket = aws_s3_bucket.mywebapp-bucket.bucket
59     source = "./index.html"
60     key = "index.html"
61 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Enter a value: yes

```
aws_s3_bucket_website_configuration.mywebapp: Creating...
aws_s3_bucket_website_configuration.mywebapp: Creation complete after 0s [id=mywebapp-bucket-586cbf89df3c74f5]
```

```
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
```

mywebapp-bucket-586cbf89 [REDACTED] Info

Objects Properties Permissions Metrics Management Access Points

Bucket overview

AWS Region
Asia Pacific (Mumbai) ap-south-1

Amazon Resource Name (ARN)
arn:aws:s3:::mywebapp-bucket-586cbf89 [REDACTED]

Creation date
July 28, 2024, 14:12:27 (UTC+05:30)

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#) [REDACTED]

Edit

Requester pays

When enabled, the requester pays for requests and data transfer costs, and anonymous access to this bucket is disabled. [Learn more](#) [REDACTED]

Edit

Requester pays
Disabled

Static website hosting

Use this bucket to host a website or redirect requests. [Learn more](#) [REDACTED]

Edit

Static website hosting
Enabled

Hosting type
Bucket hosting

Bucket website endpoint copied [REDACTED]

Configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. [Learn more](#) [REDACTED]

[http://mywebapp-bucket-e7ac8e1912\[REDACTED\]-website.ap-south-1.amazonaws.com](http://mywebapp-bucket-e7ac8e1912[REDACTED]-website.ap-south-1.amazonaws.com) [REDACTED]

Industrial Tools Store

Home

About

Products

Cart

Contact

Welcome to Our Store

Find the best products here!

Step 5: Website Endpoint:

- Outputs the URL of the static website.

The screenshot shows the VS Code interface with the main.tf file open. The code defines two S3 objects and an output block:

```
resource "aws_s3_object" "index_html" {
  bucket = aws_s3_bucket.mywebapp.bucket
  source = "./index.html"
  key = "index.html"
  content_type = "text/html"
}

resource "aws_s3_object" "style_css" {
  bucket = aws_s3_bucket.mywebapp.bucket
  source = "./style.css"
  key = "style.css"
  content_type = "text/css"
}

output "name" {
  value = aws_s3_bucket_website_configuration.mywebapp.website_endpoint
}
```

A red box highlights the output block. Below the code editor is a terminal window showing the command `terraform apply`.

The screenshot shows the terminal window with the following output:

```
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:
name = "mywebapp-bucket-e7ac8e19.s3-website.ap-south-1.amazonaws.com"
```

Step 6: Access the website:

The screenshot shows a web browser window with the URL "mywebapp-bucket-e7ac8e1.s3-website.ap-south-1.amazonaws.com". The page title is "Industrial Tools Store". The navigation menu includes "Home", "About", "Products", "Cart", and "Contact". The main content area features a heading "Welcome to Our Store" and a subtext "Find the best products here!". A footer bar at the bottom contains the copyright notice "© 2024 My E-Commerce Store".

The screenshot shows a web browser window with the URL "mywebapp-bucket-e7ac8e1.s3-website.ap-south-1.amazonaws.com". The page title is "Products". The main content area displays eight product cards arranged in two rows of four. Each card contains a thumbnail image, the product name, price, and an "Add to Cart" button. The products are: Product 1 (\$10.00), Product 2 (\$20.00), Product 3 (\$20.00), Product 4 (\$20.00), Product 5 (\$20.00), Product 6 (\$20.00), Product 7 (\$20.00), and Product 8 (\$20.00). A footer bar at the bottom contains the copyright notice "© 2024 My E-Commerce Store".

The screenshot shows a web browser window with the URL "mywebapp-bucket-e7ac8e1.s3-website.ap-south-1.amazonaws.com". The page title is "Tools Hub". The main content area features a heading "About Us" and two paragraphs: "We are a leading e-commerce store dedicated to providing the best products at the most affordable prices." and "Our mission is to make online shopping easy and accessible for everyone." Below this is a "Shopping Cart" section stating "Your cart is empty". Further down is a "Contact Us" section with a "Name:" input field and a footer bar containing the copyright notice "© 2024 My E-Commerce Store".