

1. Deploy a basic web application using Azure App Service.

- Choose a unique app name (e.g., "mywebapp").
- Select a runtime stack (e.g., Node.js, .NET).
- Configure other settings (region, operating system, etc.).
- Deploy a simple HTML or sample application.
- Access the web app's URL.

Solution:

1. Create an Azure Web App Using **Azure Portal**

- Go to the Azure Portal.
- Click on "Create a resource," then search for " App Services.", then click Create “Web app”
- Fill in the required details like App name, Subscription, Resource Group, etc.
- Choose the appropriate OS (Windows/Linux).

Microsoft Azure

Search resources, services, and docs (G+/I)

Home > App Services >

Create Web App

platform to perform infrastructure maintenance. [Learn more](#)

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * [?](#) Free Trial

Resource Group * [?](#) (New) webapprg [Create new](#)

Instance Details

Name * demowebapppguvi [.azurewebsites.net](#)

Publish * ☒ Code ☐ Docker Container ☐ Static Web App

Runtime stack * .NET 6 (LTS)

Operating System * ☐ Linux ☒ Windows

Region * East US

[?](#) Not finding your App Service Plan? Try a different region or select your App Service Environment.

Pricing plans

App Service plan pricing tier determines the location, features, cost and compute resources associated with your app. [Learn more](#)

Windows Plan (East US) * [?](#) (New) ASP-webapprg-9c4c [Create new](#)

Pricing plan Free F1 (Shared infrastructure)

Zone redundancy

An App Service plan can be deployed as a zone redundant service in the regions that support it. This is a deployment time only decision. You can't make an App Service plan zone redundant after it has been deployed. [Learn more](#)

[Review + create](#) [< Previous](#) [Next : Database >](#)

- Configure other settings as needed and click "Review + create" and then "Create."

The screenshot shows the 'Create Web App' wizard in the Microsoft Azure portal, specifically the 'Review + create' step. The interface includes a top navigation bar with the Microsoft Azure logo, a search bar, and user information. The main content area has a breadcrumb trail 'Home > App Services >' and a title 'Create Web App'. Below the title is a tabbed interface with tabs for 'Basics', 'Database', 'Deployment', 'Networking', 'Monitoring', 'Tags', and 'Review + create'. The 'Review + create' tab is active, showing a summary of the configuration. The summary includes a 'Web App by Microsoft' icon, a 'Free sku' with an 'Estimated price - Free', and a 'Summary' section. The 'Details' section lists the subscription ID, resource group, name, publish code, and runtime stack. The 'App Service Plan (New)' section lists the name, operating system, region, SKU, ACU, and memory. The 'Monitoring' section shows 'Application Insights' as 'Not enabled'. The 'Deployment' section shows 'Basic authentication' as 'Disabled' and 'Continuous deployment' as 'Not enabled / Set up after app creation'. At the bottom, there are buttons for 'Create', '< Previous', 'Next >', and a link to 'Download a template for automation'.

Microsoft Azure

Search resources, services, and docs (G+/I)

Home > App Services >

Create Web App

Basics Database Deployment Networking Monitoring Tags **Review + create**

Summary

Web App
by Microsoft

Free sku
Estimated price - Free

Details

Subscription	7bde43ec-4211-4bec-97d8-cc8f1520e309
Resource Group	webapprg
Name	demowebapppgvr
Publish	Code
Runtime stack	.NET 6 (LTS)

App Service Plan (New)

Name	ASP-webapprg-9c4c
Operating System	Windows
Region	East US
SKU	Free
ACU	Shared infrastructure
Memory	1 GB memory

Monitoring

Application Insights	Not enabled
----------------------	-------------

Deployment

Basic authentication	Disabled
Continuous deployment	Not enabled / Set up after app creation

Create < Previous Next > Download a template for automation

2. Configure Deployment Source using **GitHub Deployment**

- In the Azure Portal, go to your Web App.

Microsoft Azure | Search resources, services, and docs (G+/)

Home > Microsoft.Web-WebApp-Portal-707f1abf-b57d | Overview >

demowebappguvi

Web App

Search

[Browse](#)
[Stop](#)
[Swap](#)
[Restart](#)
[Delete](#)
[Refresh](#)
[Download publish profile](#)

Overview

- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Microsoft Defender for Cloud
- Events (preview)

Deployment

- Deployment slots
- Deployment Center

Settings

- Configuration
- Authentication
- Application Insights
- Identity
- Backups
- Custom domains
- Certificates
- Networking
- Scale up (App Service plan)
- Scale out (App Service plan)
- WebJobs
- MySQL In App
- Service Connector
- Properties

Essentials

Resource group [\(move\)](#)
[webapprg](#)

Status
Running

Location [\(move\)](#)
East US

Subscription [\(move\)](#)
[Free Trial](#)

Subscription ID
7bde43ec-4211-4bec-97d8-cc8f1520e309

Tags [\(add\)](#)
[Add tags](#)

Default domain
[demowebappguvi.azurewebsites.net](#)

App Service Plan
ASP-webapprg-9c4c

Operating System
Windows

Health Check
Cannot fetch health check data. Please try again later.

JSON View

Properties Monitoring Logs Capabilities Notifications Recommendations

Web app

Name
demowebappguvi

Publishing model
Code

Runtime Stack
Dotnet - v6.0

Domains

Default domain
[demowebappguvi.azurewebsites.net](#)

Custom domain
[Add custom domain](#)

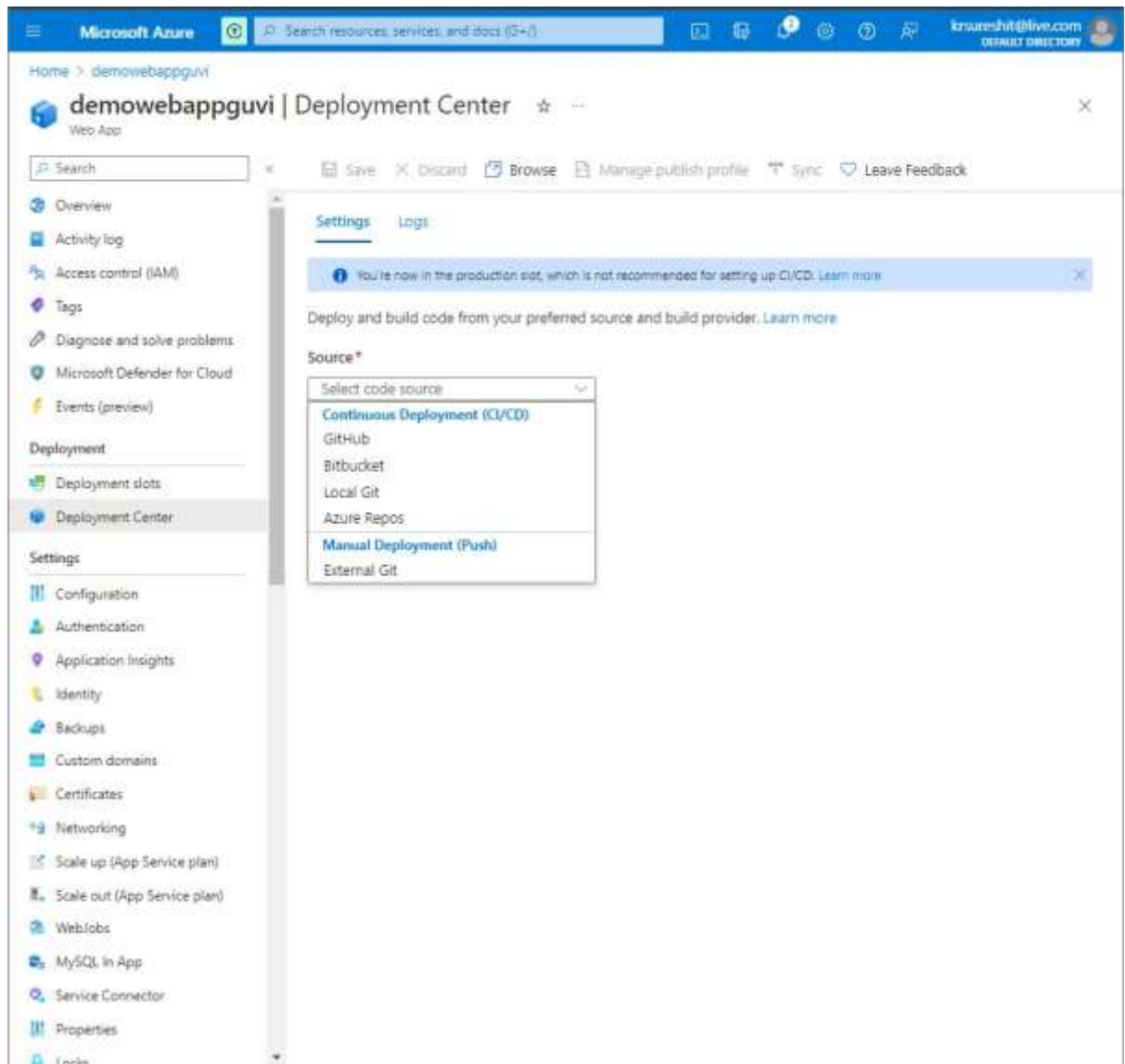
Hosting

Plan Type
App Service plan

Name
[ASP-webapprg-9c4c](#)

Operating System
Windows

- Under "Deployment Center," choose GitHub as the source.






- Authenticate and select your repository

The screenshot shows the Microsoft Azure portal interface. At the top, the header includes the Microsoft Azure logo, a search bar, and the user's email address 'ksureshit@live.com'. Below the header, the breadcrumb trail shows 'Home > demowebappguvi'. The main content area is titled 'demowebappguvi | Deployment Center' and includes a star icon and a close button. A left-hand navigation pane lists various services and settings, with 'Deployment Center' currently selected. The main pane displays the 'Settings' tab for the Deployment Center. It features a 'Source' dropdown menu set to 'GitHub', a 'Building with GitHub Actions' section with a 'Change provider' link, and a 'Signed in as' section showing the user 'yuvrajmanickam1' with a 'Change Account' link. Below this, there are dropdown menus for 'Organization' (set to 'yuvrajmanickam1'), 'Repository' (set to 'WebApplication1'), and 'Branch' (set to 'master'). The 'Build' section shows the 'Runtime stack' as '.NET'. A blue banner at the top of the settings pane states: 'You're now in the production slot, which is not recommended for setting up CI/CD. Learn more.'

- Configure settings and trigger a deployment.

Microsoft Azure

Search resources, services, and docs (G+)



ksureshit@live.com
DEFAULT DIRECTORY

Home > demowebappguvi

demowebappguvi | Deployment Center

Web App

Save

Discard

Browse

Merge publish profile

Sync

Leave Feedback

Settings

Logs

Refresh

Delete

	Time	Com...	Logs	Commit Author	Status
Monday, January 15, 2024 (1)					
	01/15/2024, 9:28:...	4e883c3	Build/Deploy La...	yusara/marictam1	In Progress...

Notifications

More events in the activity log →Dismiss all

Setting up deployment

Successfully setup GitHub Action build and deployment pipeline.

a few seconds ago

Deployment succeeded

Deployment 'Microsoft.Web-WebApp-Portal-70771abf-b57d' to resource group 'webapprg' was successful.

Go to res...

Pin to dash...

23 minutes ago

₹15,799.77 credit remaining

Subscription 'Free Trial' has a remaining credit of ₹15,799.77.

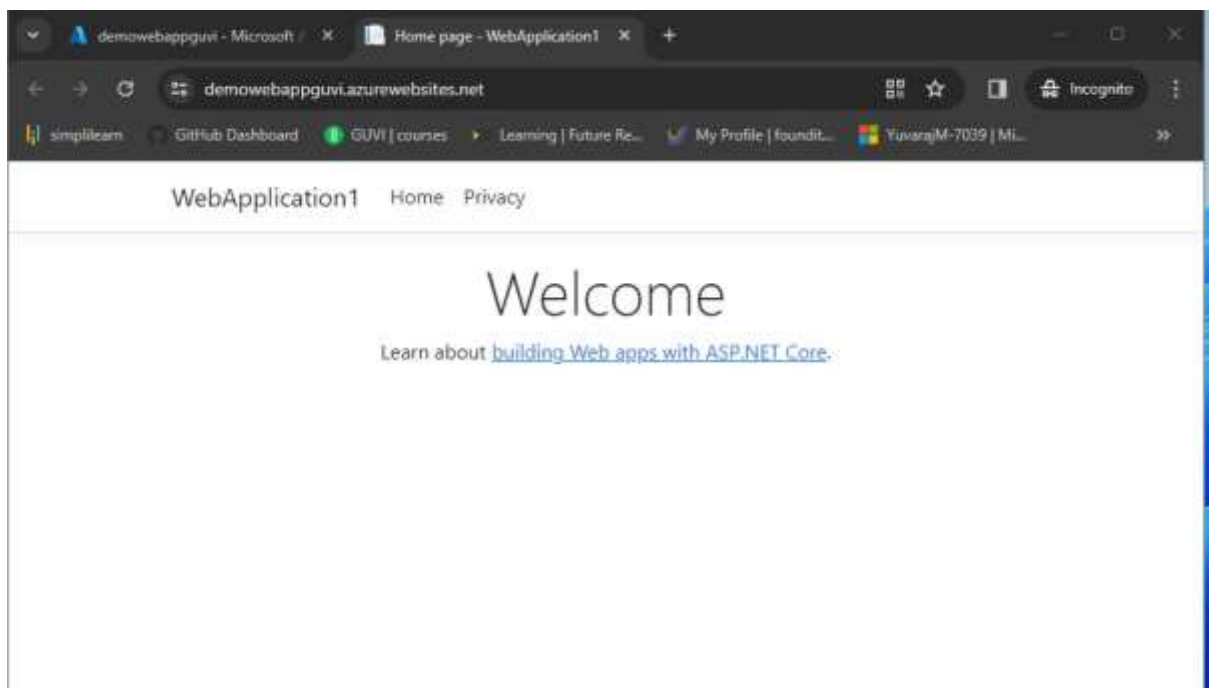
Upgrade to a Pay-As-You-Go subscription.

an hour ago

Time	Commit	Logs	Commit Author	Status	Message
Monday, January 15, 2024 (2)					
01/15/2024, 9:31:15 AM +05:30	921dc97	App Logs	N/A	Success (Active)	Add or update the Azure App Service build and deployment workflow config
01/15/2024, 9:28:43 AM +05:30	4e863c3	Build/Deploy Log	yuvrajmarickam1	In Progress...	Add or update the Azure App Service build and deployment workflow config

3. Verify Deployment:

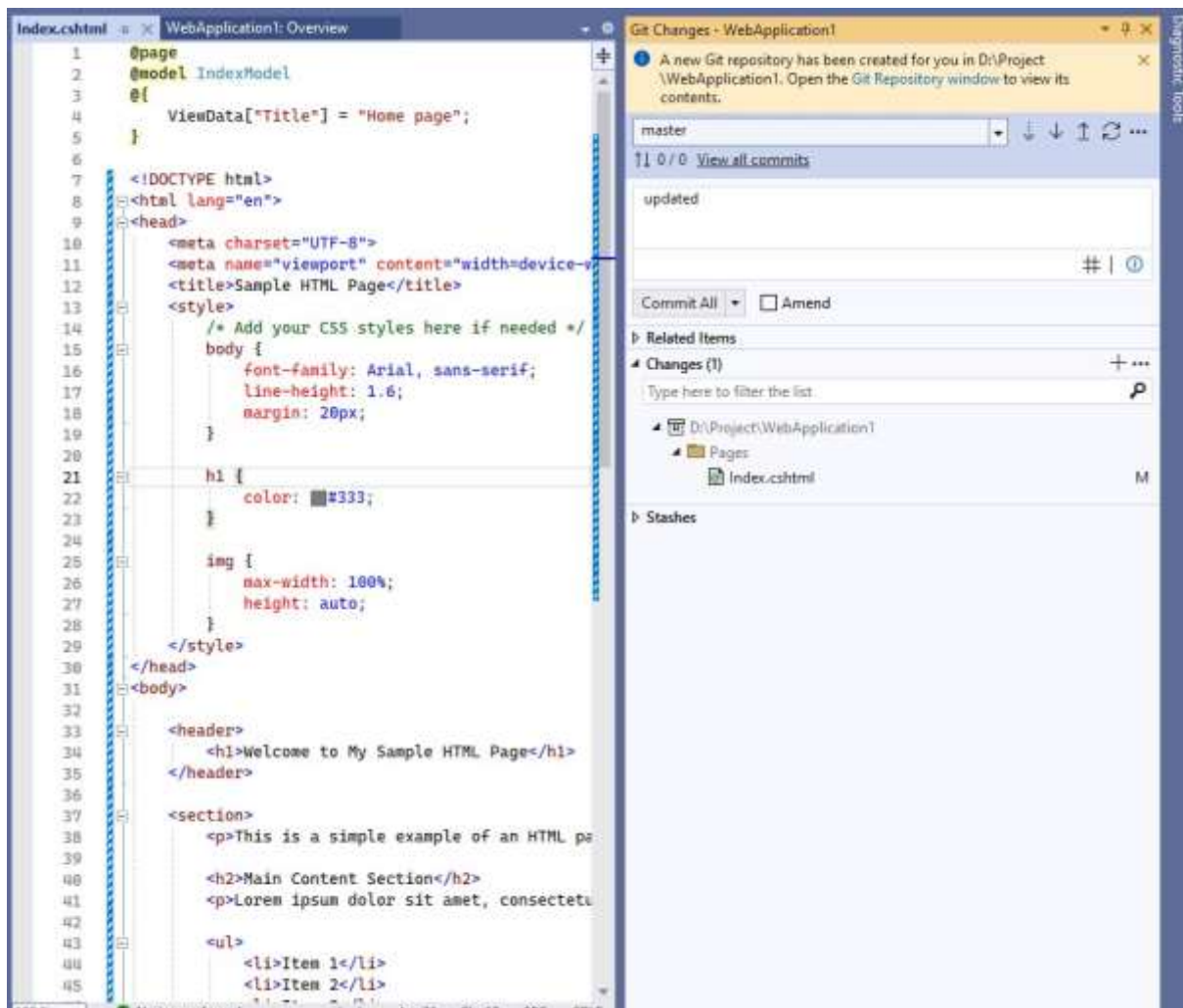
- Once the deployment is complete, you can access your web app using the URL <https://demowebappguvi.azurewebsites.net/> provided by Azure.



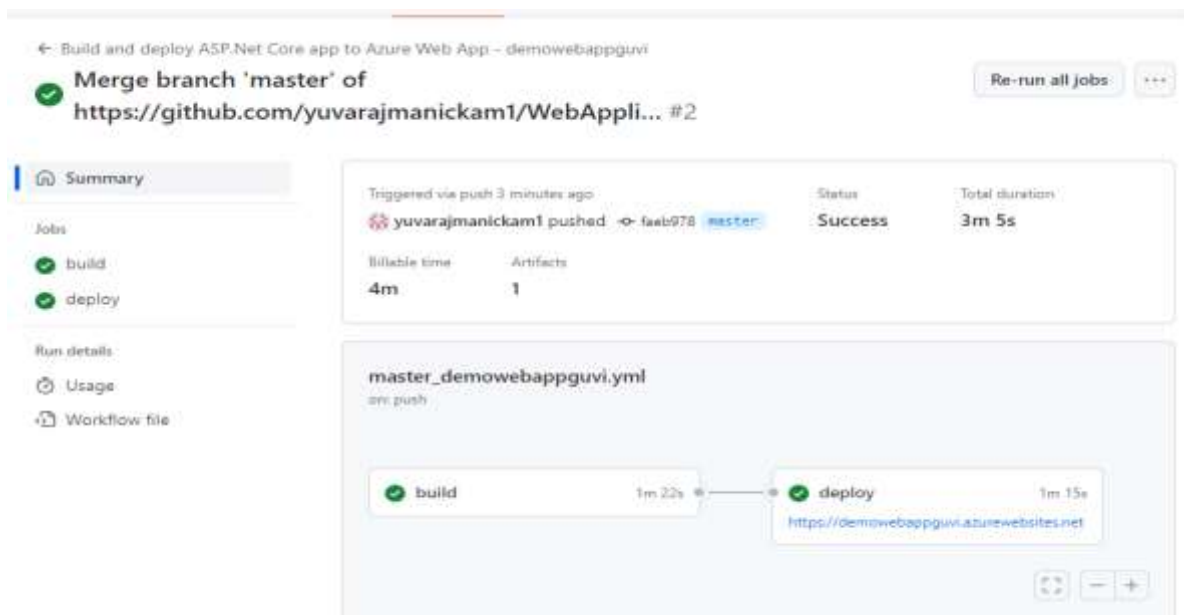
4. Continuous Deployment (Optional):

- Configure continuous deployment if you want automatic deployments when you push changes to your repository.

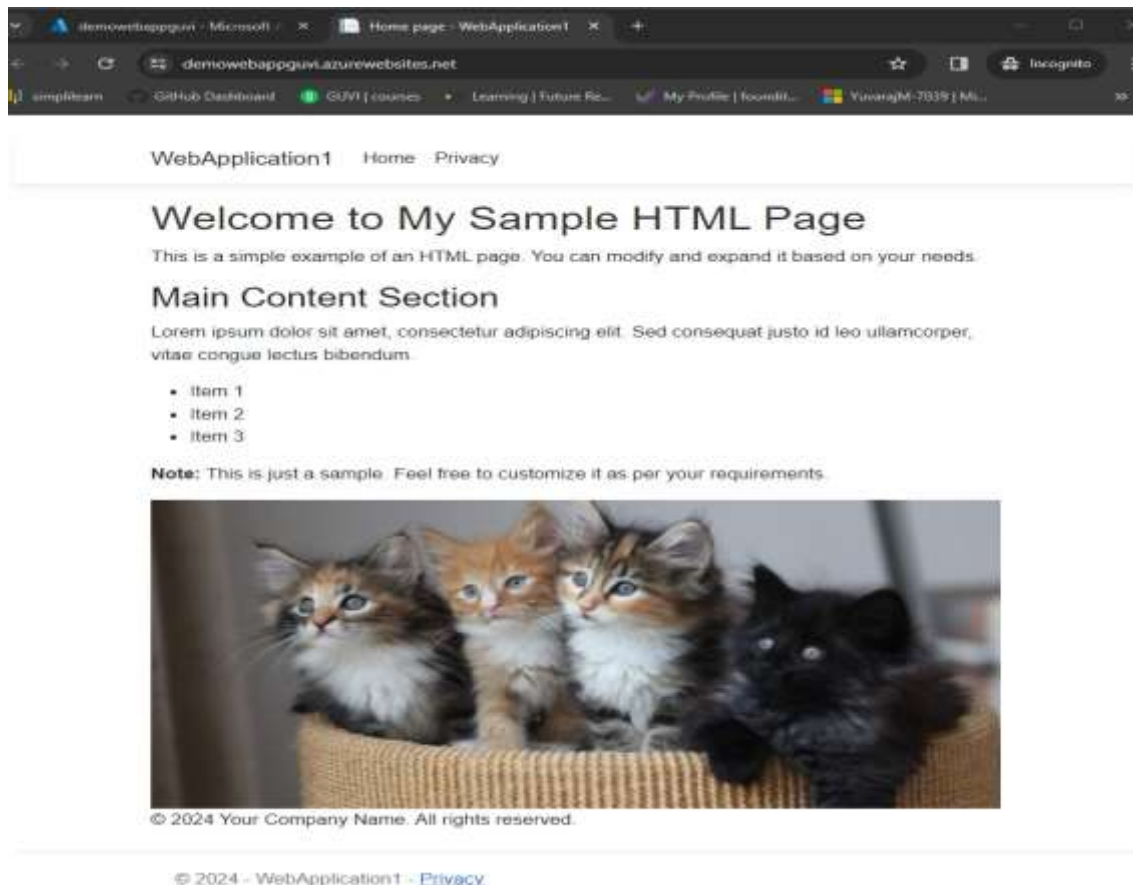
- Make changes to the GitHub repository, commit, and push the changes.



- It will trigger the deployment



Once the deployment is complete, you can access your web app using the URL



What are the benefits of using deployment slots, and how can they be utilized in a real-world scenario?

Azure deployment slots are a feature of Azure App Service that provides a way to host different versions of your web app or API in the same Azure App Service instance. Each deployment slot is like a separate instance of your app with its own host name and settings. Here are some benefits of using deployment slots and how they can be utilized in a real-world scenario:

Benefits of Deployment Slots:

1. Testing in Production-like Environment:

Deployment slots allow you to create a production-like environment for testing without affecting the live production environment. This helps in identifying and addressing issues that may arise only in a production setting.

2. Staging Environment:

Deployment slots are commonly used as staging environments where you can deploy a new version of your application and validate it before swapping it into production. This ensures a smoother release process.

3. Zero Downtime Deployment:

By deploying a new version of your app to a staging slot and swapping it with the production slot, you can achieve zero-downtime deployments. Users are seamlessly transitioned to the new version without experiencing downtime.

4. Rollback Capability:

If an issue is identified in the new version after swapping, it's easy to roll back to the previous version by swapping again or promoting the previous version from another slot.

5. Performance Testing:

Deployment slots can be used for performance testing. You can deploy a version of your application to a slot and test its performance under load before promoting it to production.

6. A/B Testing:

Slots can be utilized for A/B testing by deploying different versions of your application to different slots and directing a portion of your traffic to each version. This helps in evaluating the performance and user experience of different features.

Utilizing Deployment Slots in Real-world Scenarios:

1. Staging and Testing:

Use a deployment slot as a staging environment to validate changes and updates before promoting them to the production slot.

2. Blue-Green Deployments:

Implement a blue-green deployment strategy by having two deployment slots - blue and green. Deploy the new version to the green slot, test it thoroughly, and then swap the slots for the new version to go live.

3. Feature Branch Deployment:

If you are working with feature branches in your source control, deploy each feature branch to a separate slot for testing before merging it into the main branch and deploying to production.

4. Load Testing:

Conduct load testing in a deployment slot to ensure that your application can handle increased traffic and load without impacting the production environment.

5. Data Migration:

Use a deployment slot to perform data migration tasks or database schema changes without affecting the live application.

By leveraging deployment slots, you can enhance the reliability, availability, and efficiency of your deployment processes while minimizing the impact on end-users.