

Automating Web Application Deployment with GitHub Integration



Presented by
Debrik Chakraborty

4th Year, 8th Semester

**Department of Computer Science and Engineering
Dr. B.C. ROY ENGINEERING COLLEGE**

Guided by
Mr. Apratim Guha

**Sr. Technical Trainer
AEM Technologies**

Introduction:

In today's fast-paced software development environment, the ability to release features quickly and reliably has become a core requirement. This is where **CI/CD (Continuous Integration and Continuous Deployment)** plays a critical role. CI/CD is a modern DevOps practice that automates the process of building, testing, and deploying code, enabling faster development cycles and higher software quality.

Traditionally, deploying a web application involved numerous manual steps — setting up servers, transferring files, restarting services — all of which were time-consuming and error-prone. CI/CD eliminates these bottlenecks by introducing automation at every stage of the development pipeline.

Take **Flipkart**, one of India's largest e-commerce platforms. During high-traffic events like *The Big Billion Days*, any minor bug or delay in deployment can result in significant revenue loss. Flipkart uses CI/CD pipelines to push updates, fix bugs, and roll out features quickly and reliably — sometimes multiple times a day — all without affecting user experience or causing downtime. Without automation, such frequent and large-scale changes would be unmanageable.

This project aims to simulate that level of automation by integrating **GitHub with Azure App Services**, showcasing how developers can automatically deploy a .NET web application with every code change. The implementation of CI/CD not only enhances development speed and consistency but also aligns with industry best practices for modern cloud-based application deployment.

Objective

The primary objective of this project is to automate the deployment of a web application using GitHub integration with Azure App Service. The key steps involved in achieving this objective are:

- **Develop a Web Application:** Build and set up a web application using .NET 8.0 in Visual Studio Code.
- **Initialize a GitHub Repository:** Create and configure a GitHub repository for version control.
- **Push Code to GitHub:** Upload the web application code to GitHub to enable tracking and collaboration.
- **Set Up Azure App Service:** Create an Azure App Service Plan and deploy a web application.
- **Integrate GitHub with Azure:** Connect the GitHub repository to Azure App Service for automated deployment.
- **Verify Deployment:** Ensure the web application is successfully deployed and running on Azure.

Prerequisites

Tools Required:

- Git
- GitHub
- Visual Studio Code (VS Code)

Technologies Used:

- .NET 8.0 SDK
- Azure App Service
- GitHub Actions (for CI/CD Integration)

Azure App Services and App Service Plan

Azure App Service

- A cloud computing platform by Microsoft for hosting and managing web applications.
- Supports continuous deployment via GitHub integration for automated web application deployment.
- Provides scalability, security, and performance optimization for web applications.

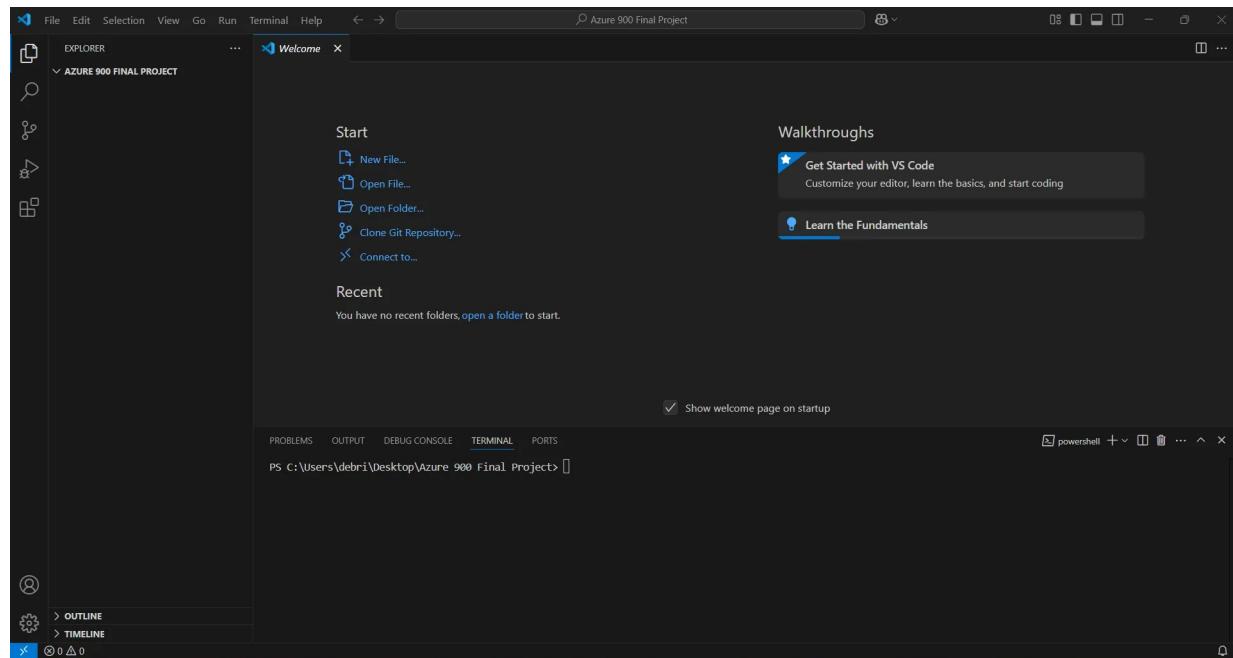
Azure Service Plan

- A scalable cloud service that defines the infrastructure for Azure App Service applications.
- Determines performance, scaling, and pricing options for deployed applications.
- Offers multiple pricing tiers based on application needs, including Free, Basic, Standard, and Premium plans.

Project Steps

1. Building the Web Application in VS Code

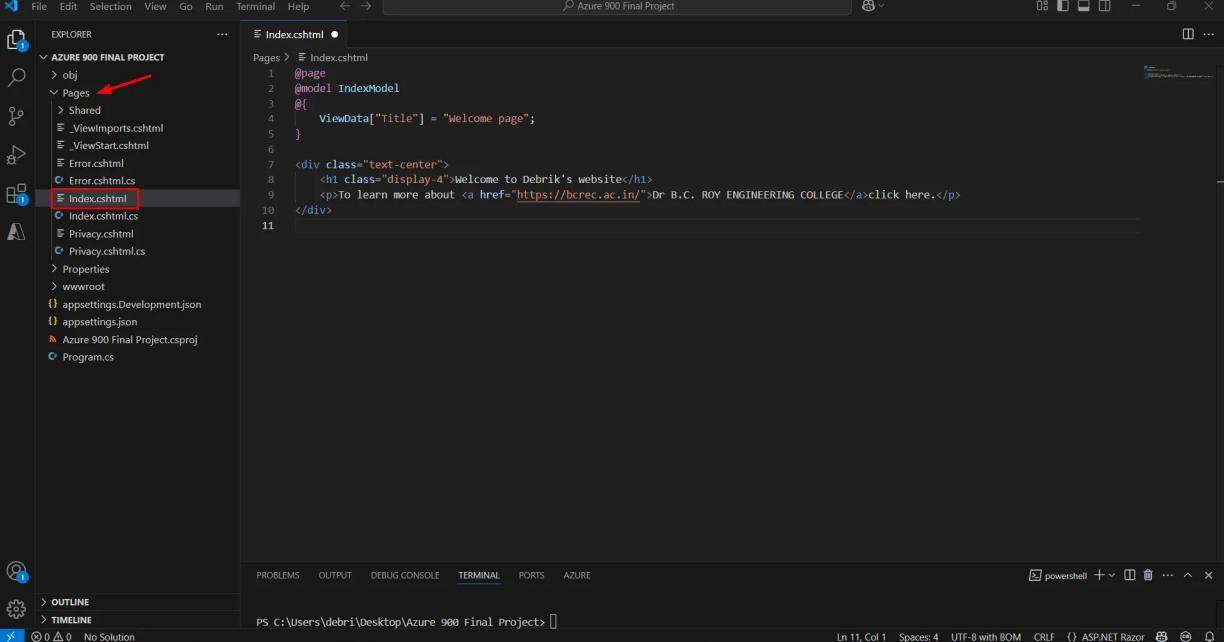
1. Open VS Code and navigate to the desired project directory.



2. Open the terminal and run the following command to create a web application using .NET 8.0:

A screenshot of the Visual Studio Code interface. The title bar says "Azure 900 Final Project". The left sidebar shows the "EXPLORER" view with a folder named "AZURE 900 FINAL PROJECT" expanded, displaying subfolders like ".vscode", "obj", "Pages", "Properties", and "wwwroot". The main area is titled "TERMINAL" with the command "dotnet new webapp --framework net8.0" highlighted. The terminal output shows: "PS C:\Users\debra\Desktop\Azure 900 Final Project> dotnet new webapp --framework net8.0" followed by "The template "ASP.NET Core Web App (Razor Pages)" was created successfully." and "This template contains technologies from parties other than Microsoft, see https://aka.ms/aspnetcore/8.0-third-party-notices for details." Below that, it says "Processing post-creation actions..." and "Restore succeeded." The bottom status bar shows the terminal output: "PS C:\Users\debra\Desktop\Azure 900 Final Project>".

3. Navigate to the Pages directory and edit index.cshtml as needed.



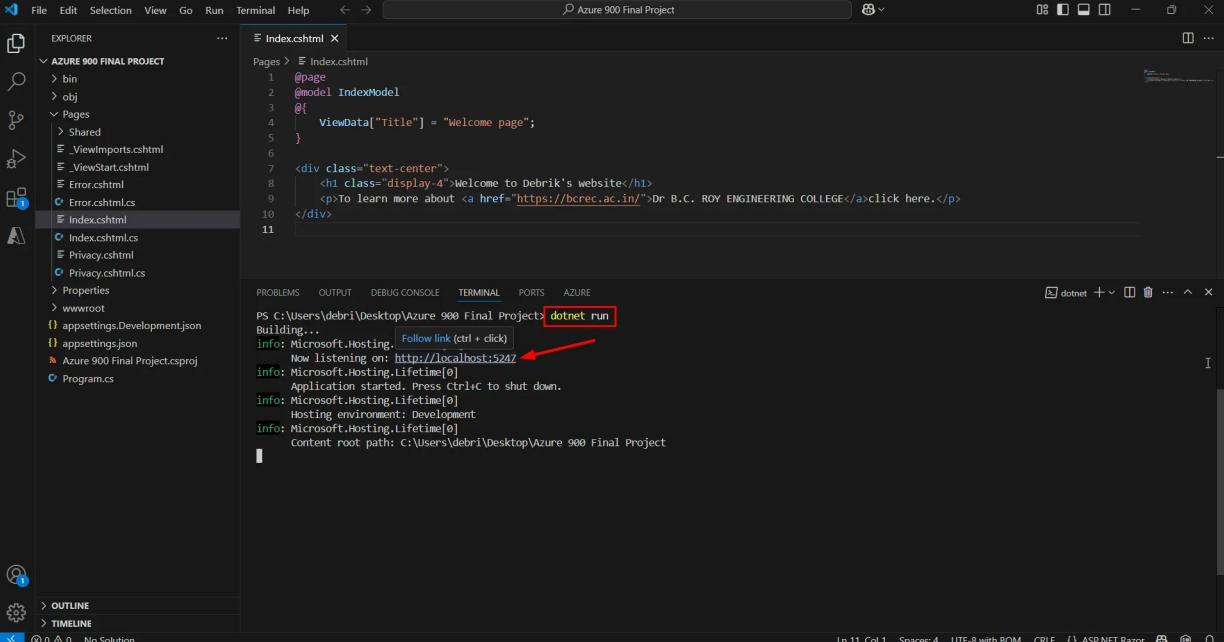
```
File Edit Selection View Go Run Terminal Help 🔍 → Azure 900 Final Project ⚙️ ... EXPLORER AZURE 900 FINAL PROJECT > obj > Pages (highlighted) > Shared & _ViewImports.cshtml & _ViewStart.cshtml & Error.cshtml & Error.cshtml.cs & Index.cshtml (highlighted) & Index.cshtml.cs & Privacy.cshtml & Privacy.cshtml.cs > Properties > wwwroot { appsettings.Development.json { appsettings.json Azure 900 Final Project.csproj Program.cs Pages > Index.cshtml 1 @page 2 @model IndexModel 3 @{ 4 ViewData["Title"] = "Welcome page"; 5 } 6 7 <div class="text-center"> 8 <h1 class="display-4">Welcome to Debrik's website</h1> 9 <p>To learn more about <a href="https://bcrec.ac.in/">Dr B.C. ROY ENGINEERING COLLEGE</a><a href="#">click here.</a></p> 10 </div> 11
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE

PS C:\Users\debri\Desktop\Azure 900 Final Project

Ln 11, Col 1 Spaces: 4 UTF-8 with BOM CRLF {} ASP.NET Razor

4. Save the changes and test the application using:



```
File Edit Selection View Go Run Terminal Help 🔍 → Azure 900 Final Project ⚙️ ... EXPLORER AZURE 900 FINAL PROJECT > bin > obj > Pages (highlighted) > Shared & _ViewImports.cshtml & _ViewStart.cshtml & Error.cshtml & Error.cshtml.cs & Index.cshtml (highlighted) & Index.cshtml.cs & Privacy.cshtml & Privacy.cshtml.cs > Properties > wwwroot { appsettings.Development.json { appsettings.json Azure 900 Final Project.csproj Program.cs Pages > Index.cshtml 1 @page 2 @model IndexModel 3 @{ 4 ViewData["Title"] = "Welcome page"; 5 } 6 7 <div class="text-center"> 8 <h1 class="display-4">Welcome to Debrik's website</h1> 9 <p>To learn more about <a href="https://bcrec.ac.in/">Dr B.C. ROY ENGINEERING COLLEGE</a><a href="#">click here.</a></p> 10 </div> 11
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE

PS C:\Users\debri\Desktop\Azure 900 Final Project dotnet run

Building...
info: Microsoft.Hosting [Follow link (ctrl + click)]
Now listening on: http://localhost:5247
info: Microsoft.Hosting.Lifetime[0]
Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
Content root path: C:\Users\debri\Desktop\Azure 900 Final Project

Ln 11, Col 1 Spaces: 4 UTF-8 with BOM CRLF {} ASP.NET Razor

5. Follow the provided link to verify that the website is running correctly.



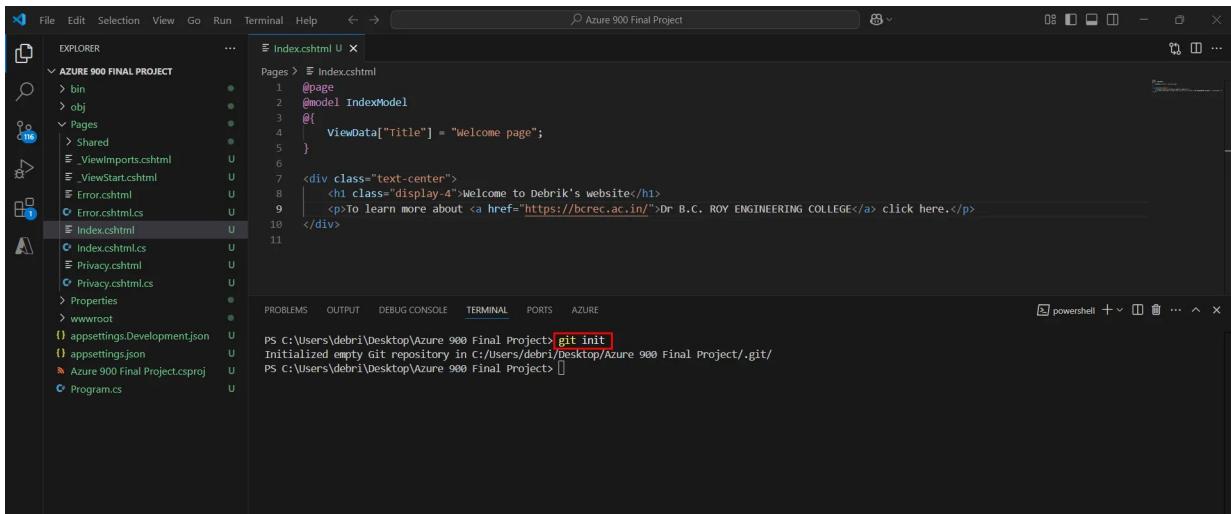
Welcome to Debrik's website

To learn more about [Dr B.C. ROY ENGINEERING COLLEGE](https://bcrec.ac.in/) click here.

© 2025 - Azure_900_Final_Project - [Privacy](#)

2. Creating a Local Repository

1. Initialize a Git repository in the project directory:



Azure 900 Final Project

File Edit Selection View Go Run Terminal Help

EXPLORER AZURE 900 FINAL PROJECT

Pages Index.cshtml

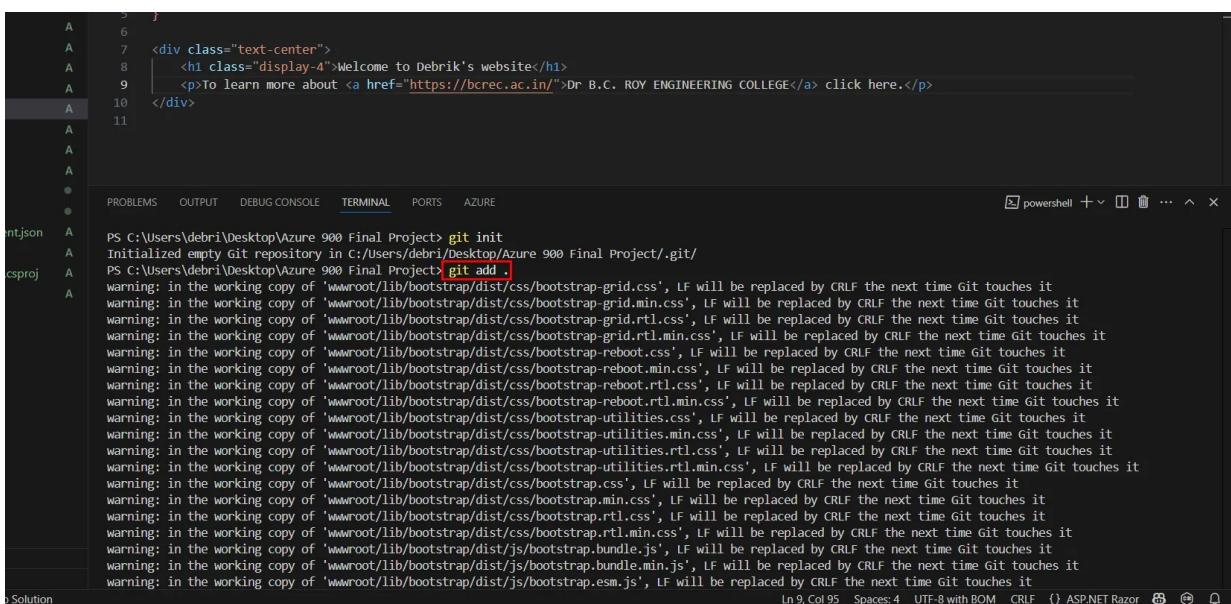
```
1 <@page
2 @model IndexModel
3 @{
4     ViewData["Title"] = "Welcome page";
5 }
6
7 <div class="text-center">
8     <h1 class="display-4">Welcome to Debrick's website</h1>
9     <p>To learn more about <a href="https://bcrec.ac.in/">Dr B.C. ROY ENGINEERING COLLEGE</a> click here.</p>
10 </div>
```

TERMINAL

```
PS C:\Users\debri\Desktop\Azure 900 Final Project> git init
Initialized empty Git repository in C:/users/debri/Desktop/Azure 900 Final Project/.git/
PS C:\Users\debri\Desktop\Azure 900 Final Project>
```

powerShell

2. Add the project files to the repository:



Index.cshtml

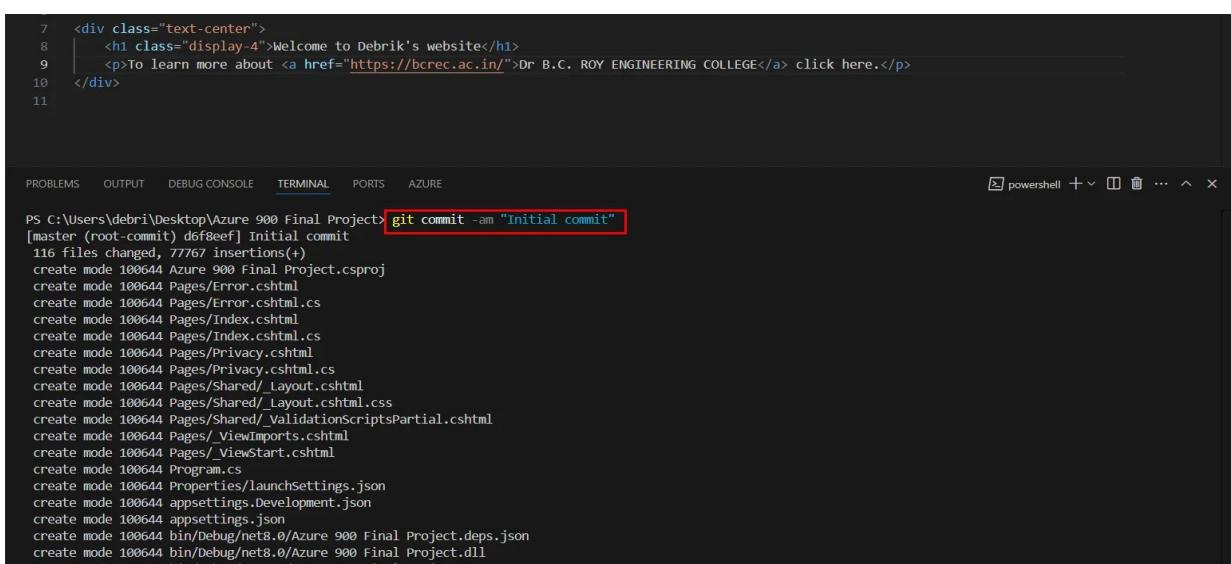
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE

```
PS C:\Users\debri\Desktop\Azure 900 Final Project> git init
Initialized empty Git repository in C:/users/debri/Desktop/Azure 900 Final Project/.git/
PS C:\Users\debri\Desktop\Azure 900 Final Project> git add .
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-grid.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-grid.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-grid rtl.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-grid rtl.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot rtl.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot rtl.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-utilities.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-utilities.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot rtl.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot rtl.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-utilities.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-utilities.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-grid.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-grid.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-grid rtl.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-grid rtl.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot rtl.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot rtl.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-utilities.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-utilities.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot rtl.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-reboot rtl.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-utilities.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/css/bootstrap-utilities.min.css', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/js/bootstrap.bundle.js', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'wwwroot/lib/bootstrap/dist/js/bootstrap.esm.js', LF will be replaced by CRLF the next time Git touches it
```

Solution

powerShell

3. Commit the changes:



```
7 <div class="text-center">
8     <h1 class="display-4">Welcome to Debrick's website</h1>
9     <p>To learn more about <a href="https://bcrec.ac.in/">Dr B.C. ROY ENGINEERING COLLEGE</a> click here.</p>
10 </div>
```

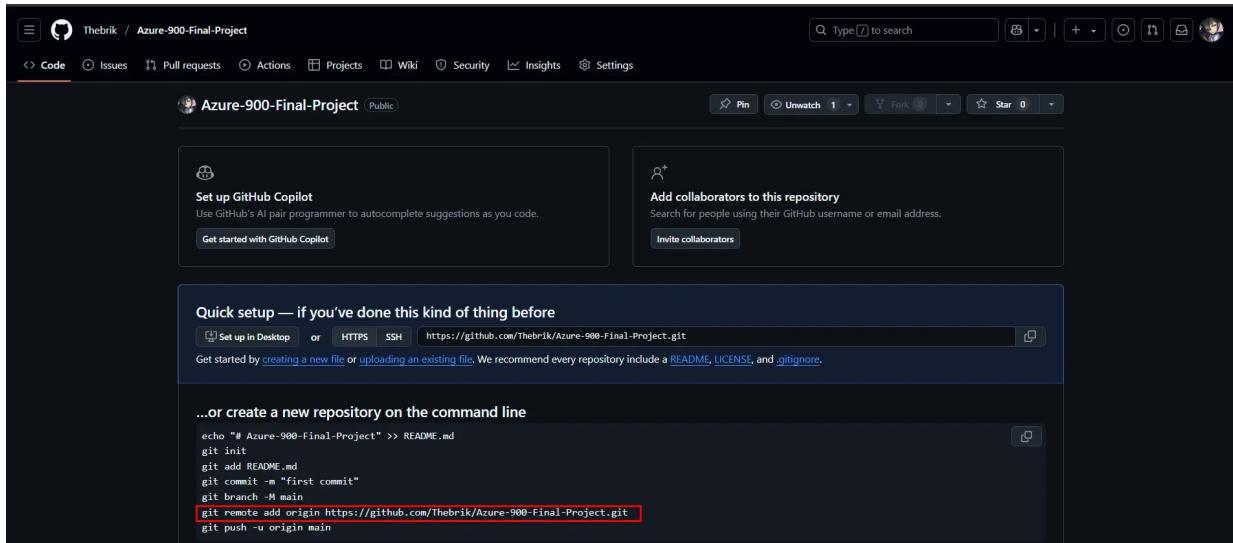
TERMINAL

```
PS C:\Users\debri\Desktop\Azure 900 Final Project> git commit -am "Initial commit"
[master (root-commit) d6f8eeef] Initial commit
 116 files changed, 77767 insertions(+)
 create mode 100644 Azure 900 Final Project.csproj
 create mode 100644 Pages/Error.cshtml
 create mode 100644 Pages/Error.cshtml.cs
 create mode 100644 Pages/Index.cshtml
 create mode 100644 Pages/Index.cshtml.cs
 create mode 100644 Pages/Privacy.cshtml
 create mode 100644 Pages/Privacy.cshtml.cs
 create mode 100644 Pages/Shared/_Layout.cshtml
 create mode 100644 Pages/Shared/_Layout.cshtml.css
 create mode 100644 Pages/Shared/_ValidationScriptsPartial.cshtml
 create mode 100644 Pages/_ViewImports.cshtml
 create mode 100644 Pages/_ViewStart.cshtml
 create mode 100644 Program.cs
 create mode 100644 Properties/launchSettings.json
 create mode 100644 appsettings.Development.json
 create mode 100644 appsettings.json
 create mode 100644 bin/Debug/net8.0/Azure 900 Final Project.deps.json
 create mode 100644 bin/Debug/net8.0/Azure 900 Final Project.dll
 create mode 100644 bin/Debug/net8.0/Azure 900 Final Project.exe
```

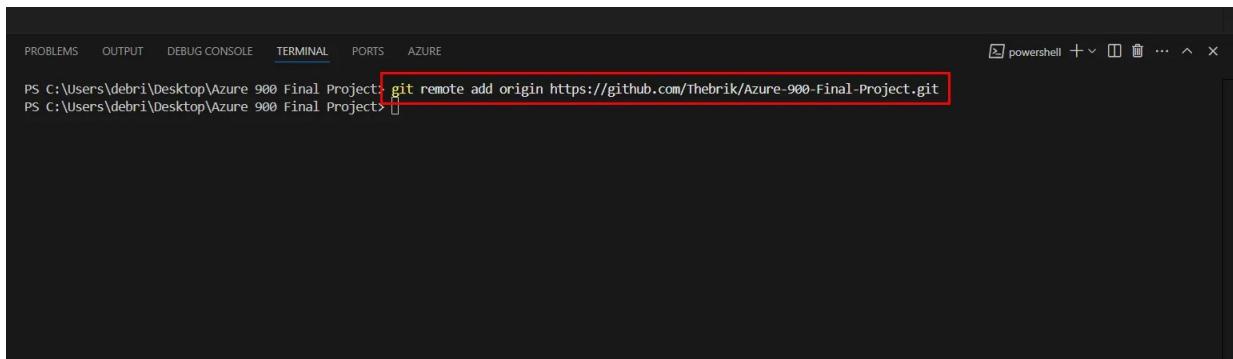
powerShell

3. Pushing the Repository to GitHub

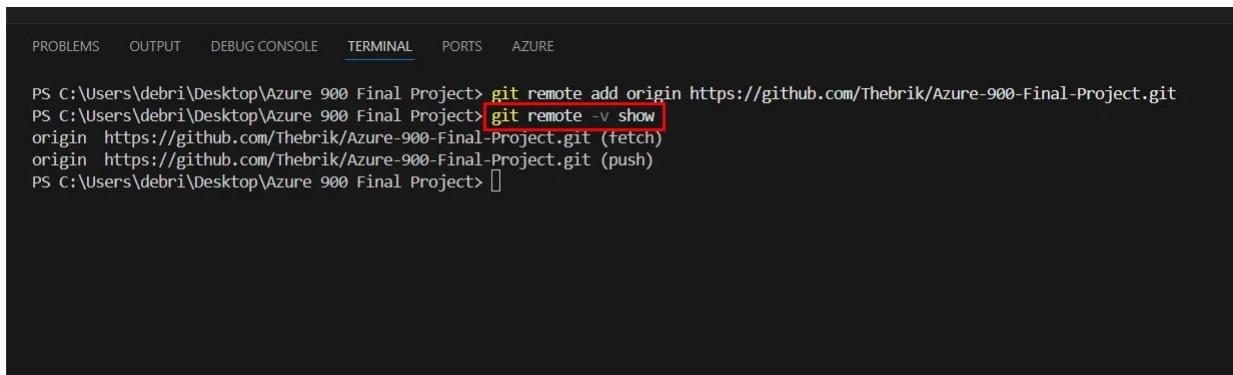
1. Create a new GitHub repository.



2. Copy the GitHub repository URL and add it as the remote origin:



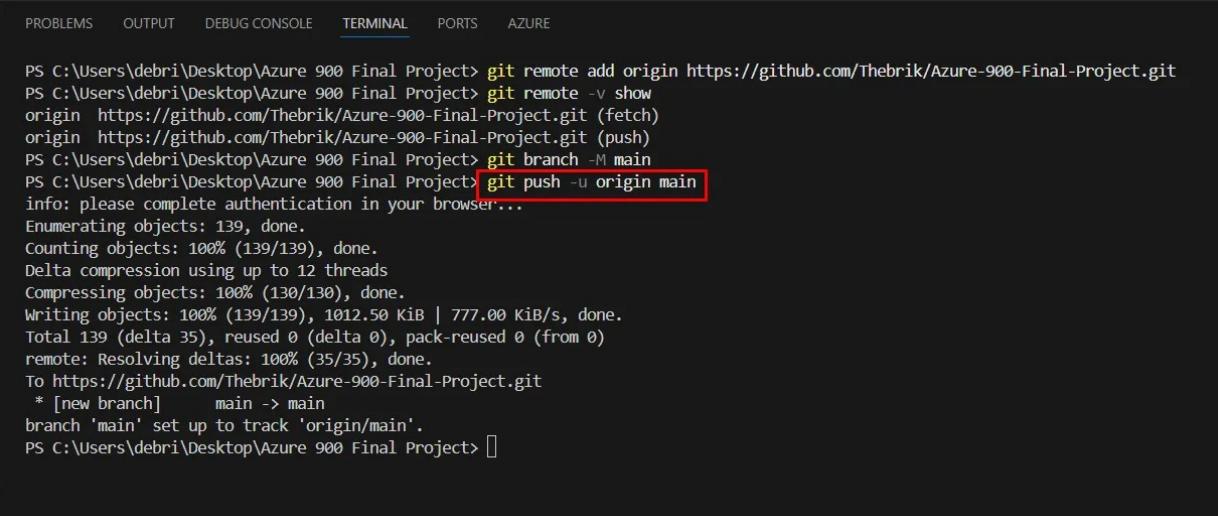
3. Check the configured repositories:



4. Create a new branch:

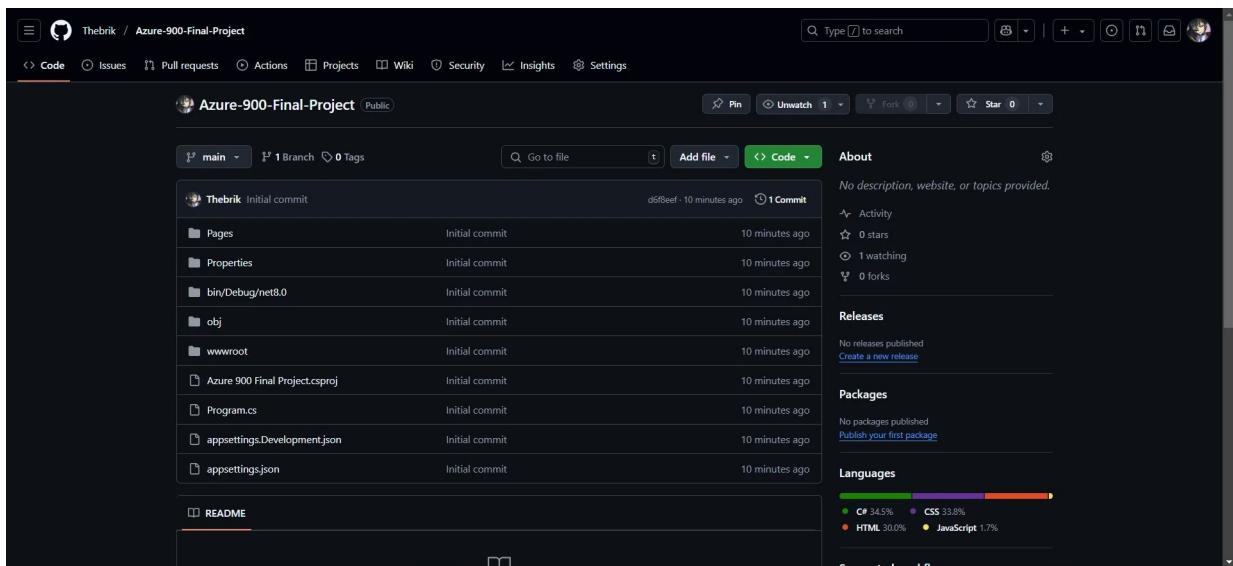
```
PS C:\Users\debri\Desktop\Azure 900 Final Project> git remote add origin https://github.com/Thebrik/Azure-900-Final-Project.git
PS C:\Users\debri\Desktop\Azure 900 Final Project> git remote -v show
origin https://github.com/Thebrik/Azure-900-Final-Project.git (fetch)
origin https://github.com/Thebrik/Azure-900-Final-Project.git (push)
PS C:\Users\debri\Desktop\Azure 900 Final Project> git branch -M main
PS C:\Users\debri\Desktop\Azure 900 Final Project>
```

5. Push the project to GitHub (authentication may be required):



```
PS C:\Users\debri\Desktop\Azure 900 Final Project> git remote add origin https://github.com/Thebrik/Azure-900-Final-Project.git
PS C:\Users\debri\Desktop\Azure 900 Final Project> git remote -v show
origin https://github.com/Thebrik/Azure-900-Final-Project.git (fetch)
origin https://github.com/Thebrik/Azure-900-Final-Project.git (push)
PS C:\Users\debri\Desktop\Azure 900 Final Project> git branch -M main
PS C:\Users\debri\Desktop\Azure 900 Final Project> git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 139, done.
Counting objects: 100% (139/139), done.
Delta compression using up to 12 threads
Compressing objects: 100% (130/130), done.
Writing objects: 100% (139/139), 1012.50 KiB | 777.00 KiB/s, done.
Total 139 (delta 35), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (35/35), done.
To https://github.com/Thebrik/Azure-900-Final-Project.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\debri\Desktop\Azure 900 Final Project>
```

6. Verify that the files have been successfully uploaded to the GitHub repository.



4. Integrating GitHub Repository with Azure App Service

1. Create an App Service Plan:

- Open Azure sandbox
- Open Azure Portal.

Unit 4 of 9 ▾ Ask Learn

Task 4: Use the Azure portal

You'll also have the option of using the Azure portal during sandbox exercises. You need to use the link provided in the exercise to access the Azure portal. Using the provided link, instead of opening the portal yourself, ensures the correct subscription is used and the exercise remains free for you to complete.

Sign in to the [Azure portal](#) to check out the Azure web interface. Once in the portal, you can see all the services Azure has to offer as well as look around at resource groups and so on.

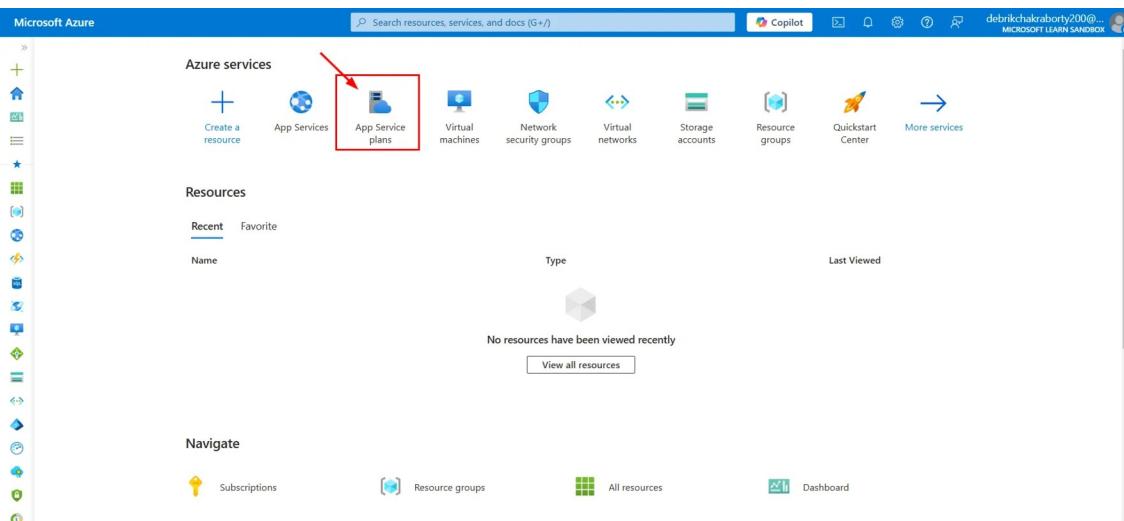
Continue

You're all set for now. We'll come back to this sandbox later in this module and actually create an Azure resource!

Next unit: Describe Azure physical infrastructure

< Previous Next >

- Navigate to App Service Plans.



- Click on Create.

The screenshot shows the Microsoft Azure portal interface. In the top left, there's a sidebar with various icons. The main area is titled 'App Service plans'. At the top, there's a search bar and several filter options like 'Subscription equals all', 'Resource group equals all', and 'Location equals all'. A prominent red arrow points to the 'Create' button, which is highlighted in blue. Below the button, the text 'No app service plans to display' is centered. There's also a small icon of a server with two sliders. At the bottom right, there's a 'Give feedback' link.

- Enter required details and select a plan.

This screenshot shows the 'Create App Service Plan' wizard. It's on the 'Project Details' step. The 'Subscription' dropdown is set to 'Concierge Subscription'. The 'Resource Group' dropdown is highlighted with a red box and contains the value 'learn-17eab1a-c54d-4d13-887a-1d54a212a9e9'. Below that is a 'Create new' button. The 'App Service Plan details' section includes fields for 'Name' (set to 'debrasp01'), 'Operating System' (set to 'Windows'), and 'Region' (set to 'Central India'). The 'Pricing Tier' section shows 'Standard S1 (100 total ACU, 1.75 GB memory, 1 vCPU)' selected. At the bottom, there are 'Review + create' and 'Next : Tags >' buttons. A red arrow points to the 'Review + create' button.

- Click Review + Create, then Create.
- App service plan is created.

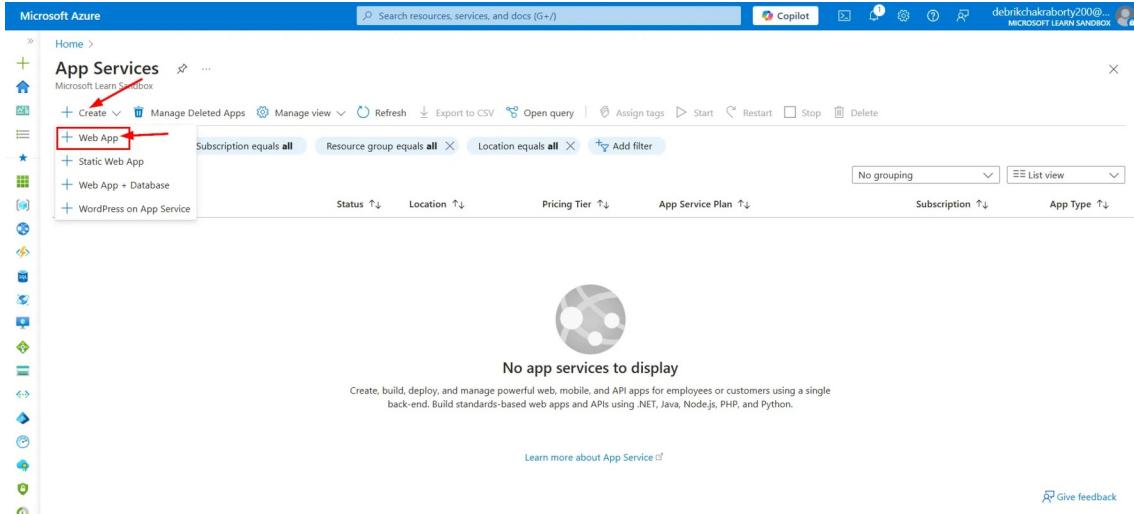
This screenshot shows the 'Microsoft.Web-ASP-Portal-40f25ee2-86b3 | Overview' page. On the left, there's a sidebar with icons for Home, App Service plans, and other services. The main area has tabs for 'Overview', 'Inputs', 'Outputs', and 'Template'. Under 'Overview', it says 'Your deployment is complete' with a green checkmark. It lists deployment details: Deployment name: Microsoft.Web-ASP-Portal-40f25ee2-86b3, Subscription: Concierge Subscription, Resource group: learn-17eab1a-c54d-4d13-887a-1d54a212a9e9. It also shows 'Deployment details' and 'Next steps' sections. At the bottom, there are 'Give feedback' and 'Tell us about your experience with deployment' links. On the right, there are sections for 'Cost management', 'Microsoft Defender for Cloud', 'Free Microsoft tutorials', 'Work with an expert', and 'Find an Azure expert >'. A red arrow points to the 'Go to resource' button in the 'Your deployment is complete' section.

2. Create a Web App Service:

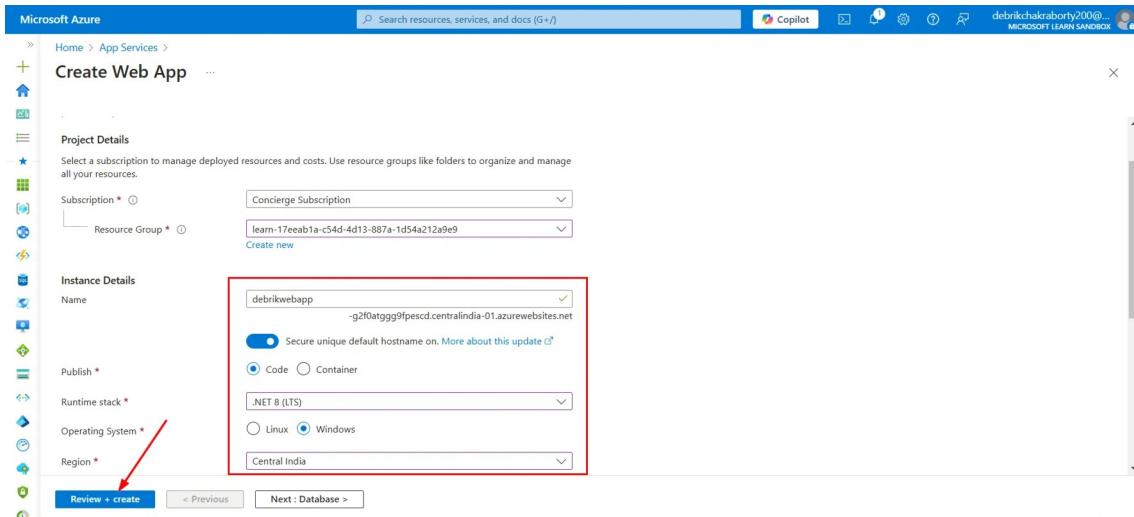
- Open App Services.



- Click on Create and select Web App.



- Fill out the required details.



- Click Review + Create, then Create.

- Deployment is complete.

The screenshot shows the Microsoft Azure Overview page for a web app named "Microsoft.Web-WebApp-Portal-3712073f-9347". The main message is "Your deployment is complete". It provides deployment details: name, subscription, resource group, start time, and correlation ID. Below this, there are sections for "Deployment details" and "Next steps". On the right side, there are promotional cards for "Cost Management", "Microsoft Defender for Cloud", "Free Microsoft tutorials", and "Work with an expert".

3. Verify that the platform is running:

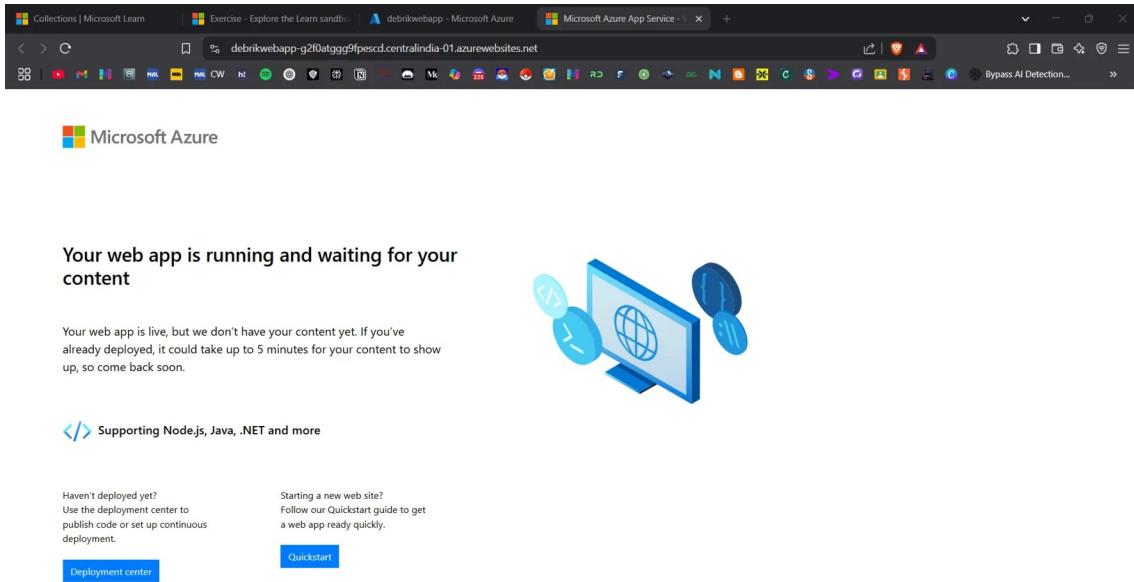
- Navigate to App Services.
- Select the created Web App.

The screenshot shows the Microsoft Azure App Services blade. It lists a single web app named "debrikwebapp". A red arrow points to the "Browse" button next to the app's name. The blade also includes filters, sorting options, and a search bar.

- Click Browse to check if the platform is operational.

The screenshot shows the Microsoft Azure App Services blade for the "debrikwebapp" web app. The "Overview" tab is selected. A red arrow points to the "Browse" button in the top navigation bar. The blade displays various details about the app, including its name, status, location, and subscription information.

- If the platform is operational, it should display the following interface:



4. Configure Deployment Settings:

- In App Services, go to **Settings > Configuration**.

The screenshot shows the 'Configuration' page for the 'debrikwebapp' web app in the Azure portal. On the left, there's a sidebar with 'App Services' selected. The main pane shows various configuration options under 'Settings'. A red arrow points to the 'Configuration' section. Another red box highlights the 'SCM Basic Auth Publishing Credentials' section, which contains two radio buttons: 'On' (selected) and 'Off'. Below this, there are sections for 'FTP Basic Auth Publishing Credentials' and 'FTP state'. A note at the bottom states: 'When selecting HTTP version 2.0, incoming client certificates must be ignored.'

- Enable necessary options to allow deployment.

The screenshot shows the 'Deployment Center' page for the 'debrikwebapp' web app. On the left, there's a sidebar with 'Deployment' selected. The main pane shows deployment settings. A red arrow points to the 'Source' dropdown, which is set to 'GitHub'. Another red arrow points to the 'Authorize' button in the 'GitHub' section. A note at the top of the main pane says: 'You're now in the production slot, which is not recommended for setting up CI/CD. Learn more'.

- Navigate to **Deployment > Deployment Center**.
- Select **GitHub** as the source and authorize GitHub.

The screenshot shows the Azure Deployment Center interface for a Web App named "debrikwebapp". The left sidebar has a "Deployment" section expanded, with "Deployment Center" selected. The main pane shows the "Settings" tab with "Source" set to "GitHub". A red box highlights the "GitHub" dropdown, and a red arrow points to the "Authorize" button below it. The status bar at the bottom of the main pane says "You're now in the production slot, which is not recommended for setting up CI/CD. Learn more".

5. Provide Repository Details:

- Fill in the repository details.

The screenshot shows the Azure Deployment Center interface for the same "debrikwebapp". The "Deployment Center" section is selected in the sidebar. The main pane shows the "GitHub" configuration with "Organization" set to "Thebrik", "Repository" set to "Azure-900-Final-Project", and "Branch" set to "main". A red box highlights the "Organization", "Repository", and "Branch" fields.

- Select **Basic Authentication** and save the settings.

The screenshot shows the Azure Deployment Center interface for the "debrikwebapp". The "Deployment Center" section is selected in the sidebar. The main pane shows the "Build" configuration with "Authentication settings" selected. Under "Authentication type*", the "Basic authentication" option is selected, indicated by a red box and a red arrow.

6. Deploy the Web Application:

- Wait for the deployment process to complete.

The screenshot shows the Azure Deployment Center interface for a web application named "debrikwebapp". The left sidebar lists various monitoring tools like Microsoft Defender for Cloud, Events (preview), Recommended services (preview), Log stream, Resource visualizer, Deployment, Deployment slots, and the Deployment Center itself. The main area is titled "Logs" and shows a deployment log for March 28, 2025. The log details a build and deployment step initiated by "DEBRIK CHAKRA..." at 03:28:08. The status is "In Progress...". A message indicates that the user needs to "Add or update the Azure App build and deployment work config".

- Click **Browse** to verify that the web application is successfully deployed.

This screenshot is similar to the previous one but focuses on the "Browse" button in the top navigation bar of the Deployment Center. A red arrow points to the "Browse" button, which is highlighted in blue. The deployment log table shows two entries: one for "App Logs" at 08:10 and another for "Build/Deploy Lo..." at 08:08, both marked as "Success (Active)".

- Web app have been deployed.

A screenshot of a web browser displaying the deployed web application. The address bar shows the URL "debrikwebapp-g2f0atggg9lpescd.centralindia-01.azurewebsites.net". The page content is the "Welcome to Debrik's website" from the application source code.

A screenshot of the deployed web application's home page. The title is "Welcome to Debrik's website". Below it, a footer note says "To learn more about Dr.B.C. ROY ENGINEERING COLLEGE click here." and includes a link. The footer also contains copyright information: "© 2025 - Azure_900_Final_Project - Privacy".

7. Testing Continuous Deployment:

- Make a small change to the code (e.g., edit the text in index.cshtml) and save it.

```
        ViewData["Title"] = "Welcome page";
    }

<div class="text-center">
    <h1 class="display-4">Welcome to Debrik Chakraborty's website</h1>
    <p>To learn more about <a href="https://bcrec.ac.in/">Dr B.C. ROY ENGINEERING COLLEGE</a> click here.</p>
</div>
```

- Commit and push the change to the GitHub repository.

```
Pages > Index.cshtml
1 @page
2 @model IndexModel
3 @{
4     ViewData["Title"] = "Welcome page";
5 }
6
7 <div class="text-center">
8     <h1 class="display-4">Welcome to Debrik Chakraborty's website</h1>
9     <p>To learn more about <a href="https://bcrec.ac.in/">Dr B.C. ROY ENGINEERING COLLEGE</a> click here.</p>
10 </div>
11
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE

```
ps C:\Users\debri\Desktop\Azure 900 Final Project> git add .
ps C:\Users\debri\Desktop\Azure 900 Final Project> git commit -m "Test deployment via CI/CD pipeline"
[main 03da5d0] Test deployment via CI/CD pipeline
 1 file changed, 1 insertion(+), 1 deletion(-)
ps C:\Users\debri\Desktop\Azure 900 Final Project> git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 394 bytes | 394.00 KiB/s, done.
Total 4 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/Thebrik/Azure-900-Final-Project.git
 2a3ef9..03da5d0 main -> main
ps C:\Users\debri\Desktop\Azure 900 Final Project> [ ]
```

- Go to the Azure portal → App Services → Your Web App → “Deployment Center” and monitor the deployment logs.

debrikwebapp | Deployment Center

Search resources, services, and docs (G+)

Copilot

debrikchakraborty200@... MICROSOFT LEARN SANDBOX

Web App

Events (preview)

Recommended services (preview)

Log stream

Resource visualizer

Deployment

Deployment slots

Deployment Center

Settings

Logs

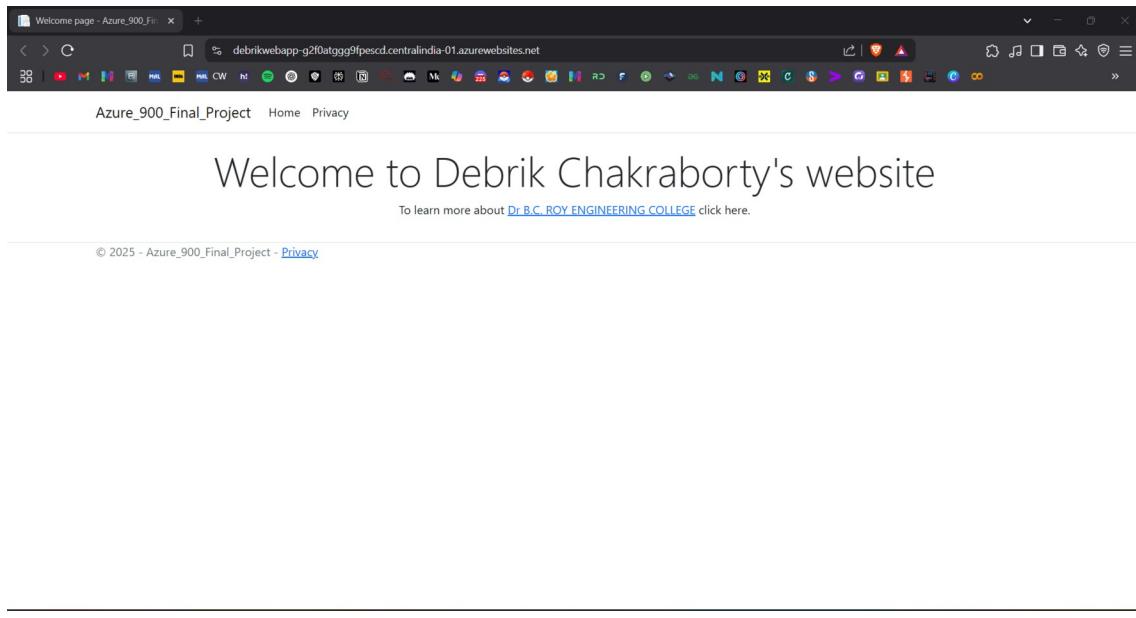
FTP credentials

Refresh

Delete

Time	Commit	Logs	Commit Author	Status	Message
Sunday, April 20, 2025 (8)	04/20/2025, 2:09:... 6e24d36	App Logs	N/A	Success (Active)	OneDeploy
	04/20/2025, 2:05:... 03da5d0	Build/Deploy Lo...	Debrik Chakrabo...	Success	Test deployment via CI/C

- Once deployment is complete, click **Browse** again to verify the update is reflected live.



- This verifies that every push to the repository triggers an automatic deployment, a key aspect of continuous deployment.

Conclusion

The objective was to establish a fully automated web deployment pipeline using GitHub and Azure App Services. This was expected to reduce manual errors, improve efficiency, and facilitate faster updates through CI/CD.

The GitHub-Azure integration allowed for real-time, consistent deployments. By simply pushing changes to the repository, new versions were automatically deployed without requiring manual intervention. The system proved to be stable and efficient for basic deployment workflows.

This project provided hands-on experience with key DevOps practices. I gained insights into setting up CI/CD pipelines, using cloud platforms for hosting, and managing version control with Git. The implementation highlighted the importance of automation in modern development practices and its critical role in maintaining scalable applications.

Reference

<https://learn.microsoft.com/en-us/azure/app-service/deploy-continuous-deployment?tabs=github%2Cgithubactions>