# Problem Statement:

Flipso is an upcoming e-commerce company. The company gets regular feedback from the customer to improve its application. For Flipso, it is essential to make the changes quickly in an automated and efficient manner so that customers are always happy and can buy the items in a hassle-free and smooth way. It is a crucial factor in the growth of the company. The company has decided to consider continuous integration and deployment to ship the feature fast to achieve this. The company has chosen the CI/CD pipeline to implement this. They have decided to create a CI/CD pipeline using Azure Pipeline for the solution hosted on GitHub (Third-party). You have been hired as an Azure DevOps specialist to help Flipso implement CI/CD using Azure Pipeline.

# Steps to Perform:

* Create an ASP.NET core application locally (Use the base empty application)
* Check in the code to GitHub
* Create a CI/CD pipeline using the code repository from GitHub
* The ASP.NET application should be deployed to Azure Web App
* Run the Azure app to validate the correct deployment

# Solution

1. Install the latest version of Visual Studio and Azure DevOps extension for Visual Studio on your local machine.

2. Open Visual Studio and create a new ASP.NET Core project. Choose "ASP.NET Core Web Application" and select the "Empty" template.

3. Add a basic home page to the application.

4. In Visual Studio, create a new Git repository and commit the code to the repository.

5. Create a new Azure DevOps project by logging into Azure DevOps and clicking on "Create project" button.

6. Select Git as the version control system for the project.

7. In Azure DevOps, navigate to "Pipelines" in the left-hand menu and click on "New pipeline". Select "GitHub" as the source control provider and select the appropriate repository.

8. Choose the ASP.NET Core (.NET Framework) template and select the appropriate version of .NET Framework for your project.

9. Configure the pipeline by adding the necessary tasks. For example, add a task to restore the NuGet packages, build the application, and publish the output to a folder.

10. Add a task to deploy the application to Azure Web App. You will need to create a new Azure Web App for this.

11. Add any additional tasks that are required for your project, such as running tests or creating release notes.

12. Save and run the pipeline to test it. Make any necessary changes to the pipeline and repeat the testing process until the pipeline is successful.

13. Once the pipeline is successful, commit and push any changes to the repository.

14. Navigate to the Azure Web App in the Azure Portal and verify that the application is deployed correctly by visiting the URL.

By following these steps, you should be able to create a CI/CD pipeline using Azure Pipeline to deploy your ASP.NET Core application to an Azure Web App.