# Problem Statement:

You have been hired by Sator co. as a DevOps administrator. You and the other users want to monitor the success or failure of deployment remotely without connecting to Azure DevOps. For the distributed teams, it is not easy for them to be online in Azure DevOps. They want to know the DevOps status of the Azure pipeline. For this, the company has chosen to monitor the DevOps through Microsoft Team. All users connected to the Microsoft team can get the status of the DevOps pipeline. As a task, you have to integrate the Azure DevOps pipeline with Microsoft Team to visualize the runtime parameters through teams.

# Steps to Perform:

* Create an ASP.Net application locally
* Check in the code to Azure Repos
* Create an Azure pipeline to deploy the code to the Azure web app
* Integrate the pipeline with Microsoft teams
* Users can visualize the Azure pipeline operation/outcome from Microsoft teams

# Solution

1. Create an ASP.NET application:

- Open Visual Studio and create a new ASP.NET web application.

- Choose the .NET framework and the web application template that suits the requirements.

- Add some functionality to the web application to test the pipeline and the integration with Microsoft Teams.

2. Check in the code to Azure Repos:

- Create a new project in Azure DevOps.

- Create a new repository in the project.

- Push the code from the local machine to the Azure Repos repository.

3. Create an Azure pipeline:

- Open the Azure DevOps project.

- Click on the Pipelines tab and then click on the New pipeline button.

- Select the Azure Repos Git as the source.

- Choose the repository and branch from which the code should be deployed.

- Select the ASP.NET web application template from the list of templates.

- Configure the pipeline by adding the necessary tasks, such as build, test, and deployment.

4. Integrate the pipeline with Microsoft Teams:

- Open the Microsoft Teams application.

- Go to the Teams channel where the pipeline status needs to be displayed.

- Click on the + button and then click on the More apps button.

- Search for the Azure Pipelines app and click on the Add button.

- Follow the prompts to connect to Azure DevOps and choose the project and pipeline that should be displayed in the Teams channel.

- Select the appropriate options to customize the pipeline status display in the Teams channel, such as success/failure status, duration, etc.

5. Test the integration:

- Trigger a new pipeline run by committing and pushing a change to the Azure Repos repository.

- Check the Microsoft Teams channel to ensure that the pipeline status is displayed correctly.

- Verify that the pipeline status is updated in real-time, reflecting any changes in the pipeline run status.

That's it! You have successfully integrated the Azure DevOps pipeline with Microsoft Teams, enabling distributed teams to visualize the runtime parameters of the pipeline in real-time.