Setup a continuous deployment pipeline using AWS CodePipeline

- Create a deployment Pipeline
    - Continuous deployment needs a target environment containing virtual servers, or EC2 instances, where it will deploy our code. We will set this environment before we deploy our code
    - Here, I am creating my target environment as AWS Elastic BeanStalk.
    - Elastic Beanstalk lets you easily host web applications without needing to launch, configure, or operate virtual servers on your own. It automatically provisions and operates the infrastructure (e.g. virtual servers, load balancers, etc.) and provides the application stack (e.g. OS, language and framework, web and application server, etc.) for you.
    - Go to the AWS Elastic BeanStalk console and click on create new environment, give a simple name for your environment, select a language and click on create environment.
    - Elastic BeanStalk will begin creating your target environment where you will deploy your code
- Get a copy of your sample code
    - Upload your code to GitHub repository or AWS S3 bucket or AWS codecommit
    - We can choose any of these repositories.
- Create your Pipeline
    - Go to the AWS code Pipeline console, click on get started
    - Click on create pipeline, give a name to pipeline, click on next
    - Here, we need to add our code repository. If you want to connect to a GitHub account, select GitHub. It will prompt a page to sign to a github account.
    - After sign in, you can able to see all your repositories present in your github account, select your sample code one, select branch
    - In the next step, no need to select any build.
    - Next step is to select where to deploy your code, select AWS BeanStalk from the options, it will show you the applications and environments present.
    - Select your environment and proceed. Review your pipeline settings.
- Activate your code Pipeline to deploy code
    - After reviewing your selections, click create pipeline.
    - It will deploy your code to the target environment, you can see success status after successful deployment.
- Check the changes deployed to your AWS BeanStalk environment, check with your URI
- Commit a change and then update you app

    - Make a small changes in your code and commit to the repository
    - AWS code Pipeline will automatically trigger the commit and deploy the new changes to the target environment.
- Clean up your resources
    - Clean up all the services we launched in AWS to avoid charges