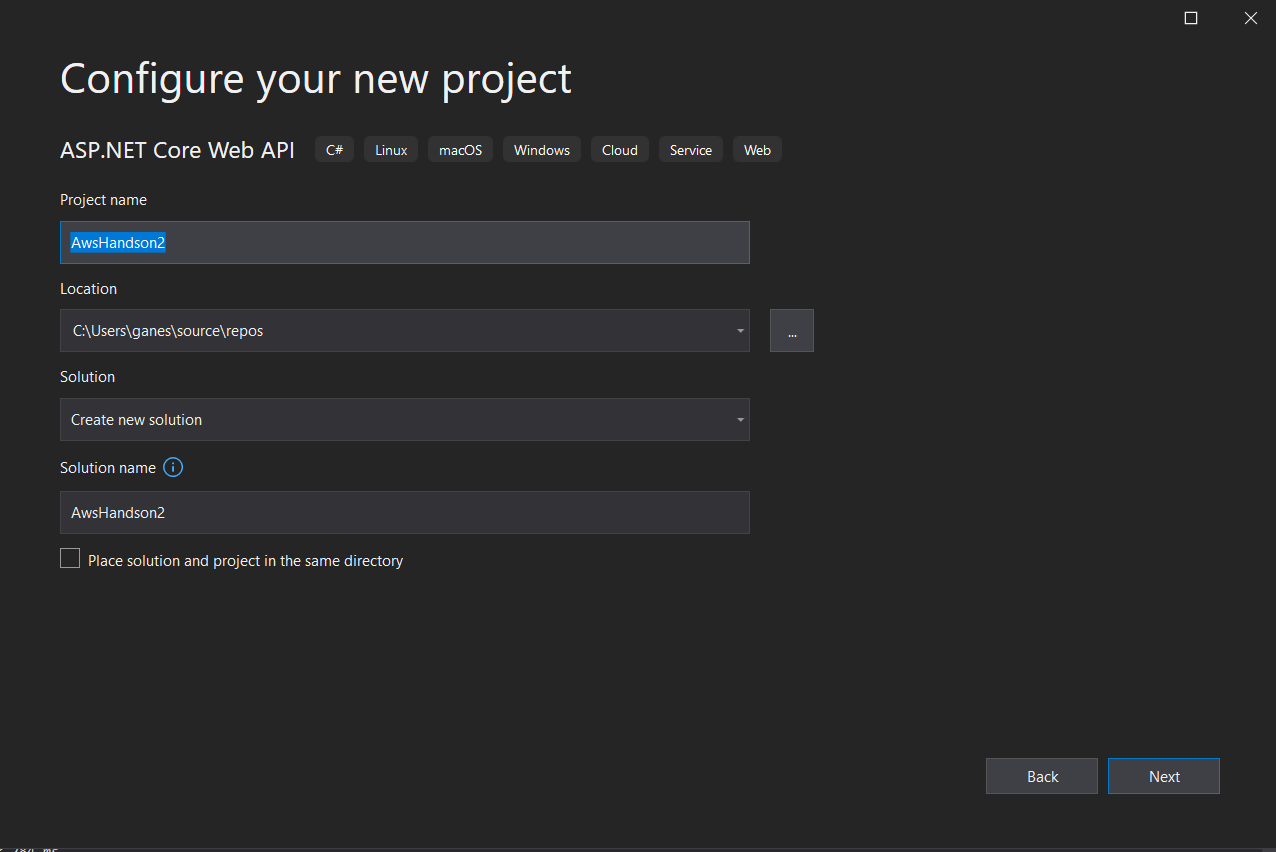
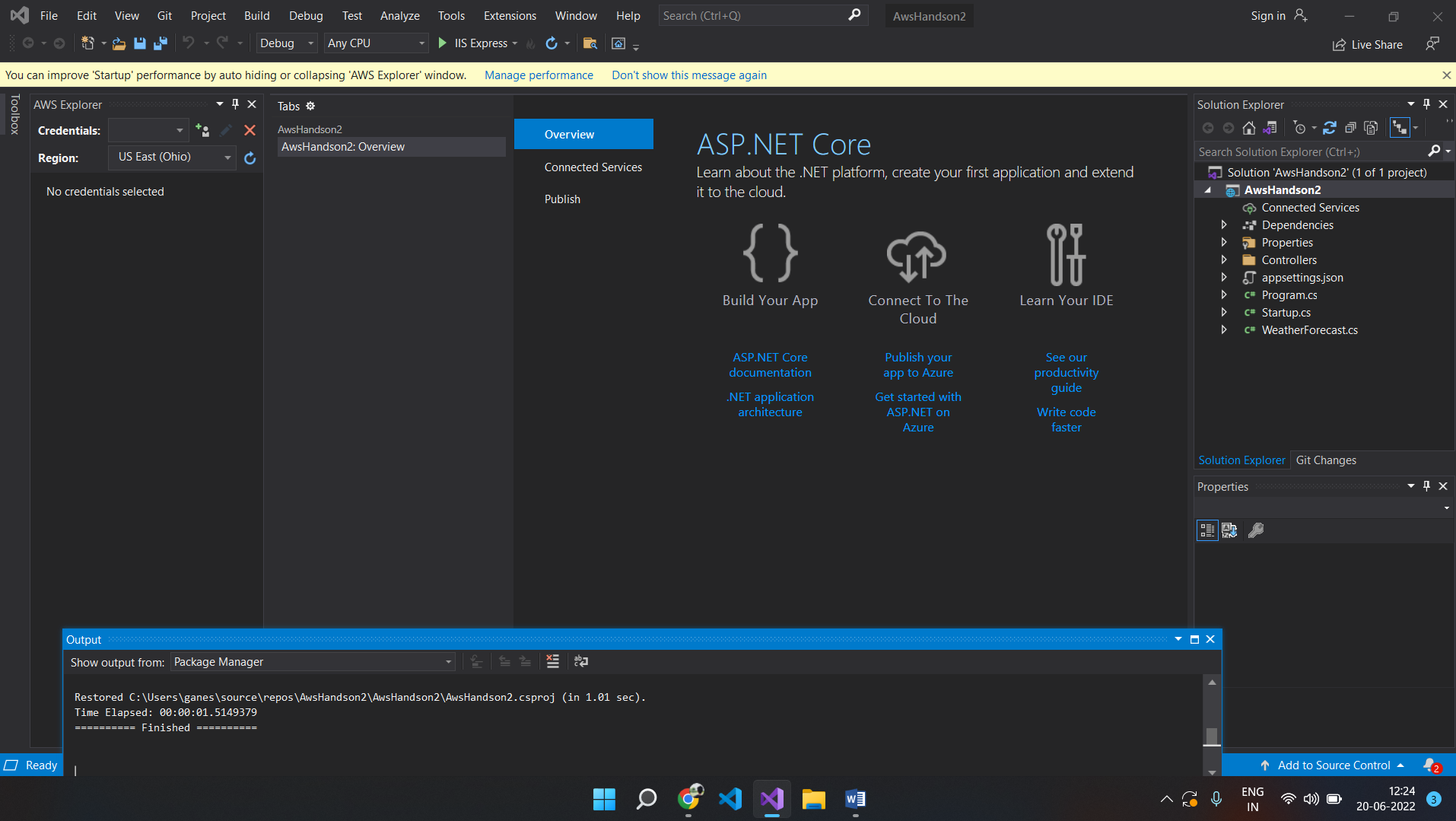
Hands On 11: **Solution**

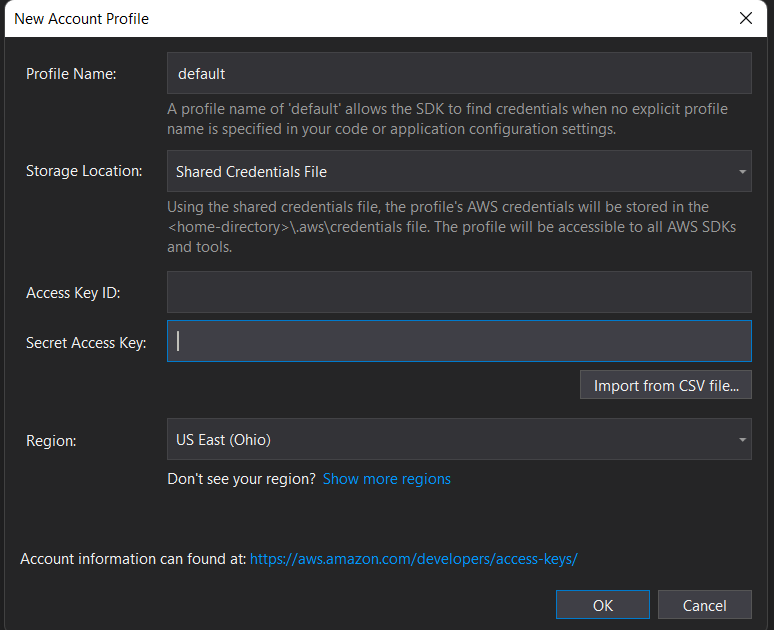
Publishing a .NET Core Microservice as an Elastic Beanstalk using AWS

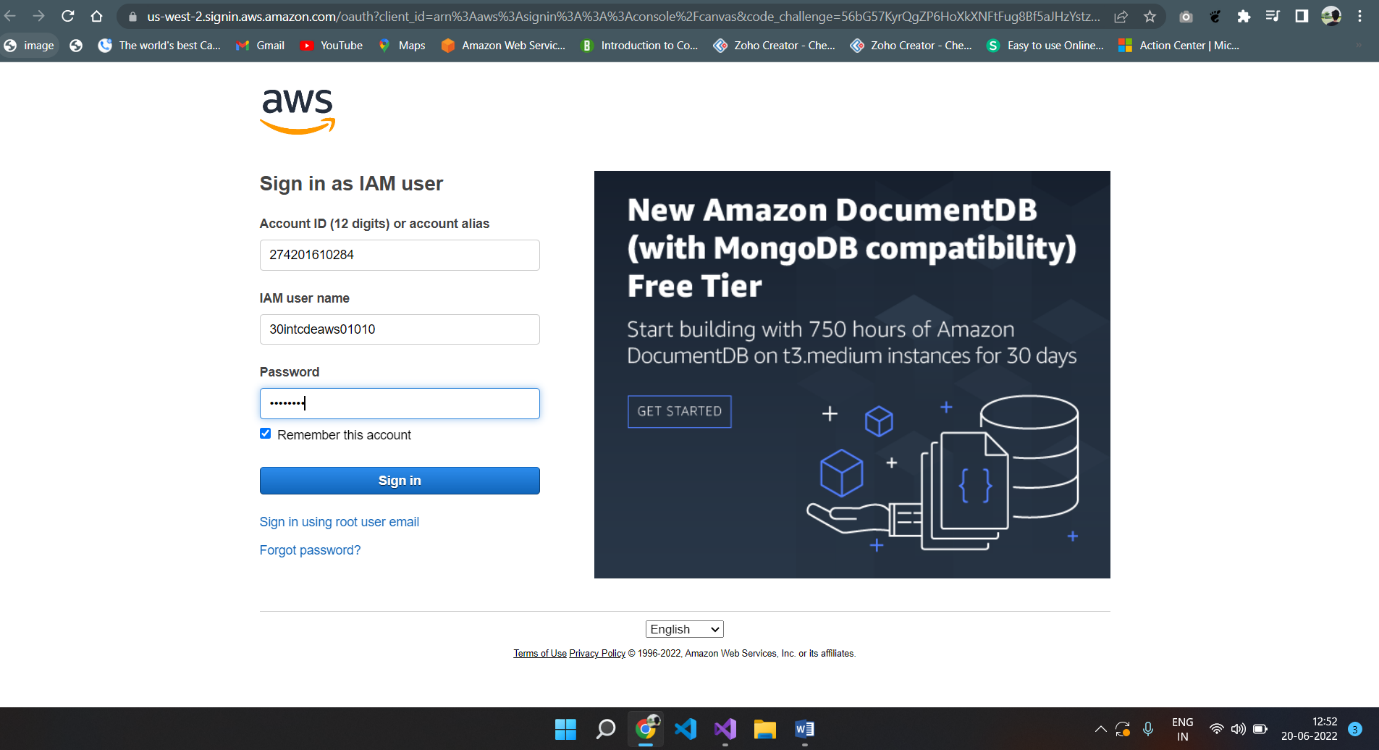
1. Create a ASP.NET Core Web API Project in VS 2019

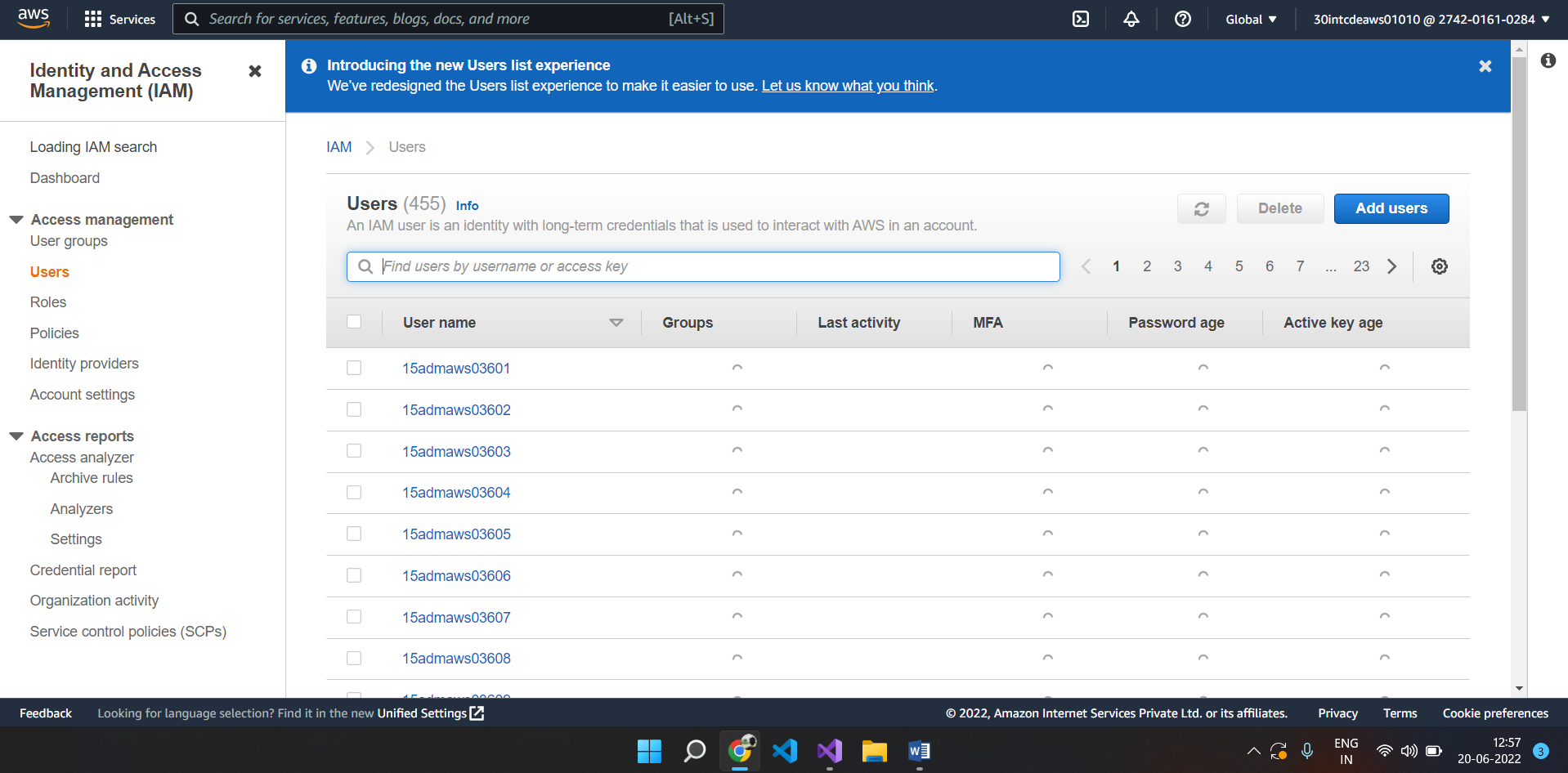




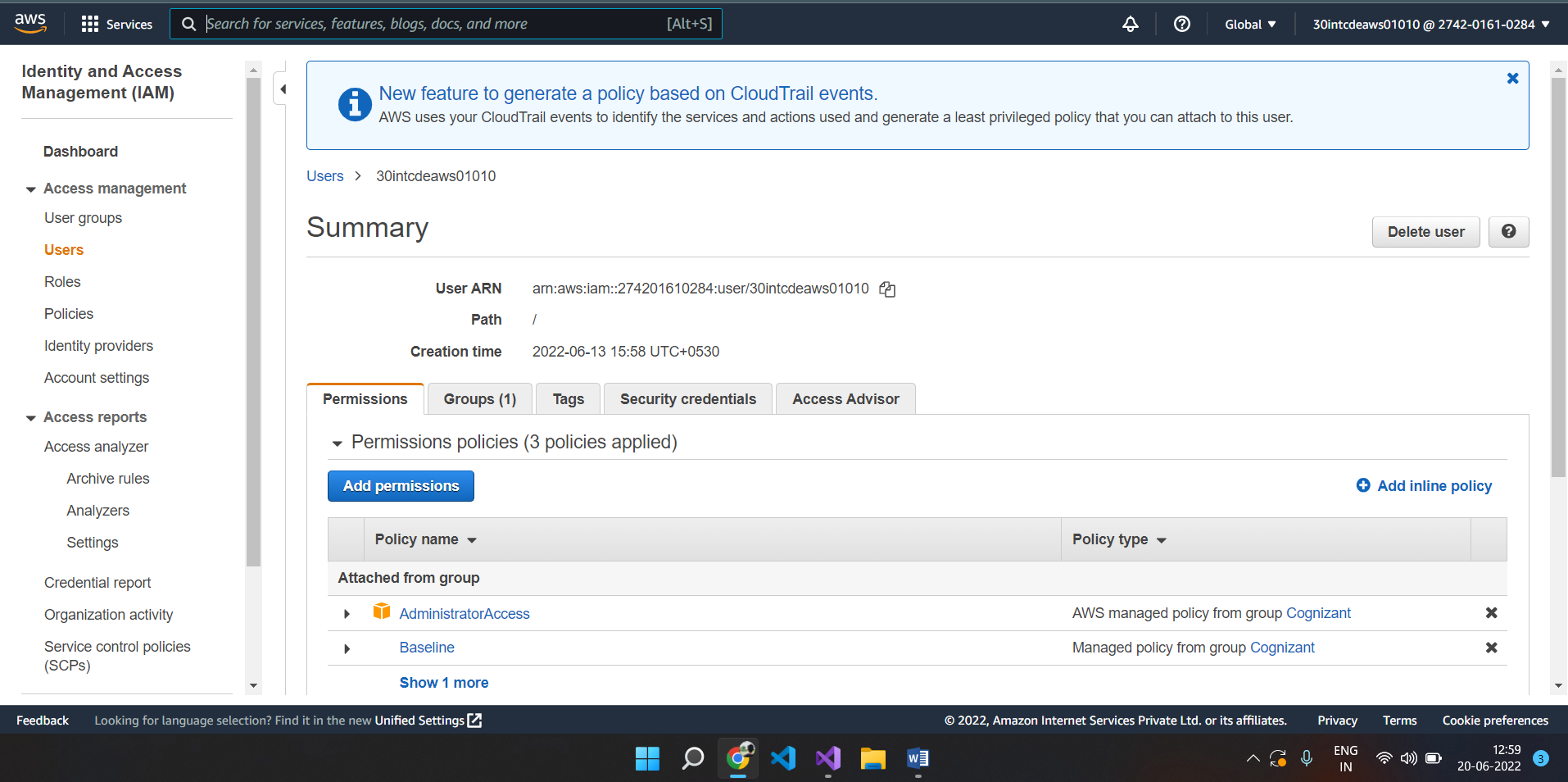
1. Once the project is created, the following screen will appear where the AWS toolkit has to be configured.



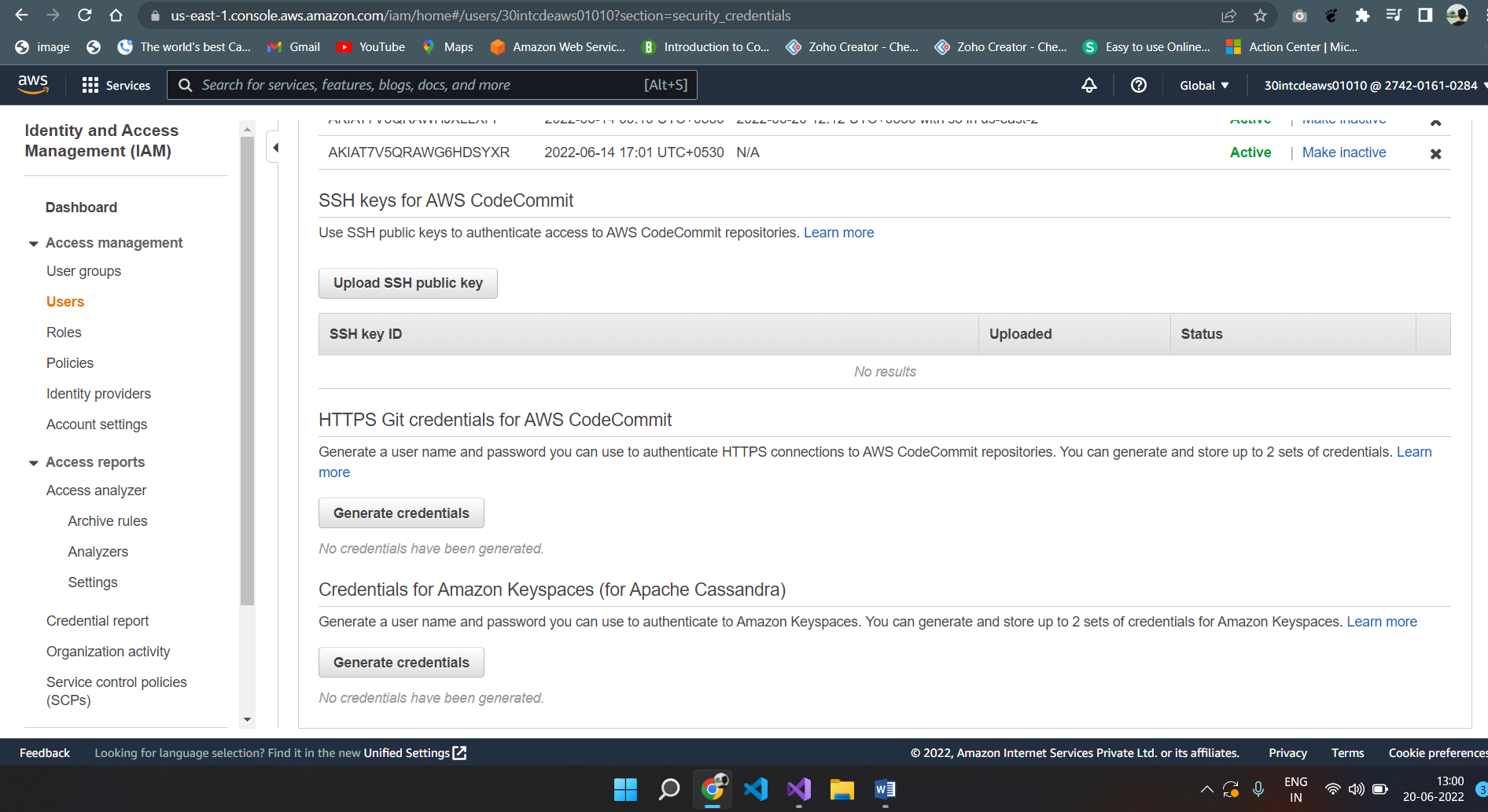
1. Go to AWS Portal and login to your account
2. Go to IAM à Users in the left side navigation panel à Choose your username from the list



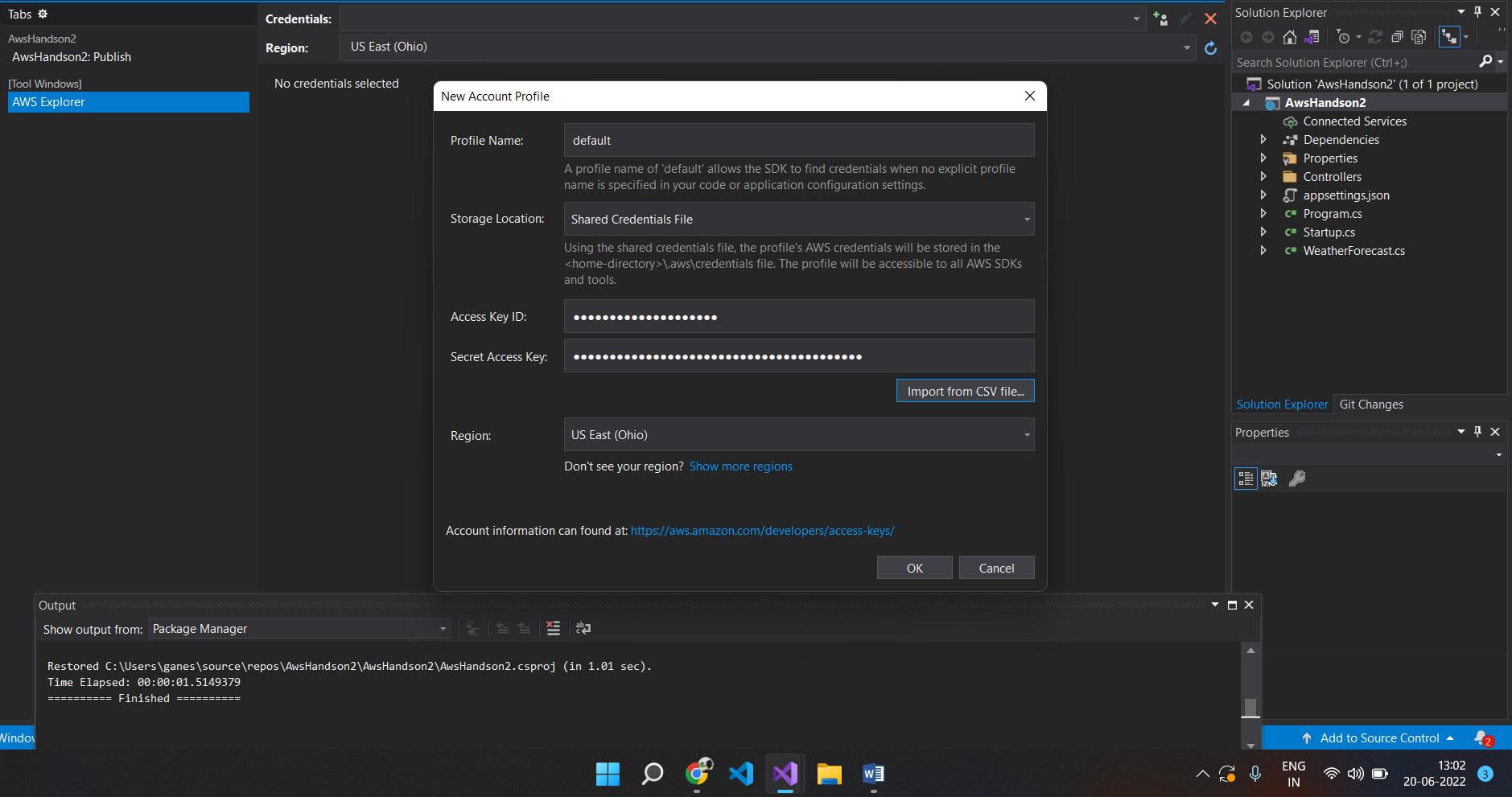
1. The following screen will appear



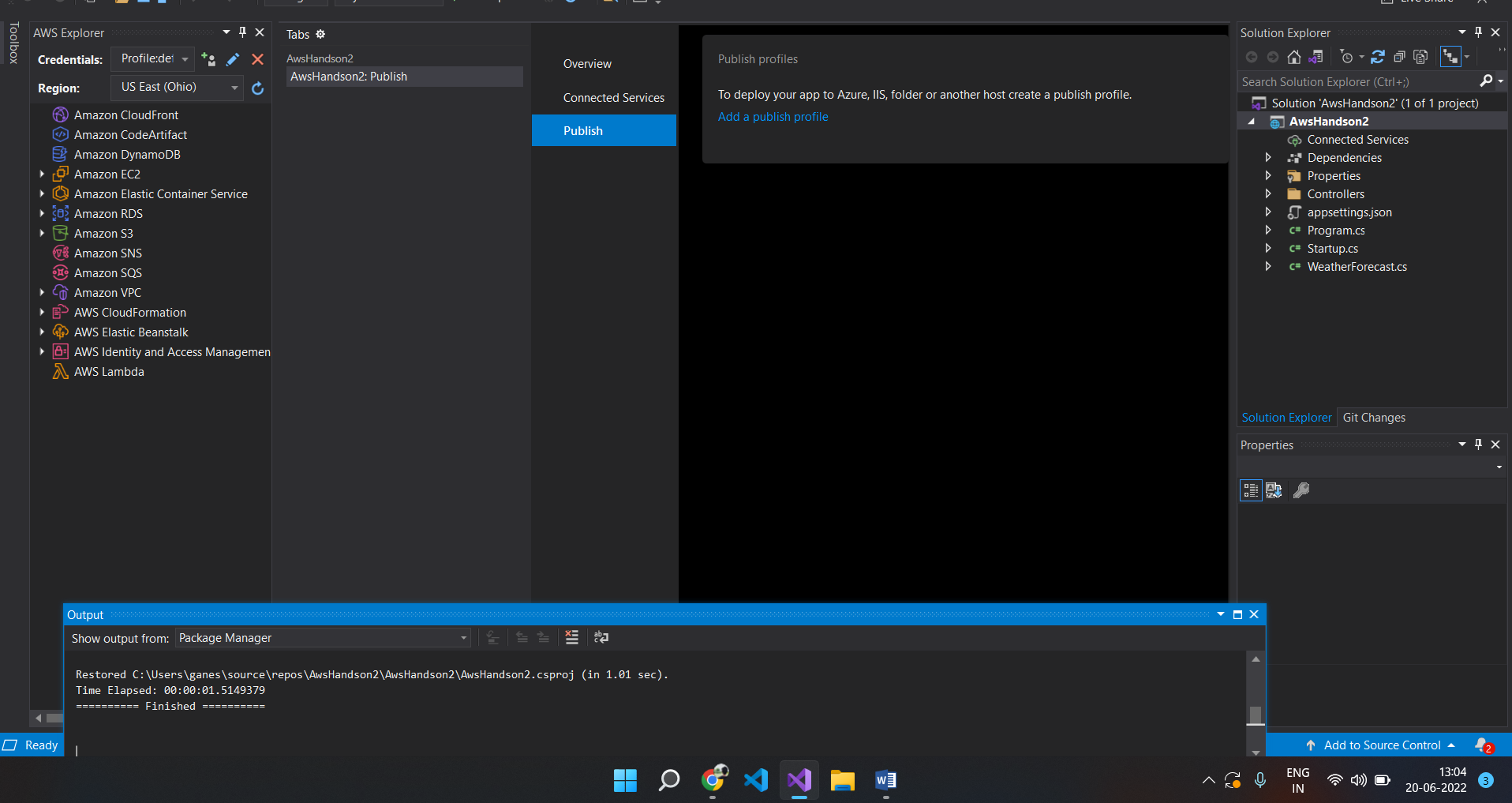
1. Click on Security Credentials and there will be an option to download csv file which has Access key and security credentials.



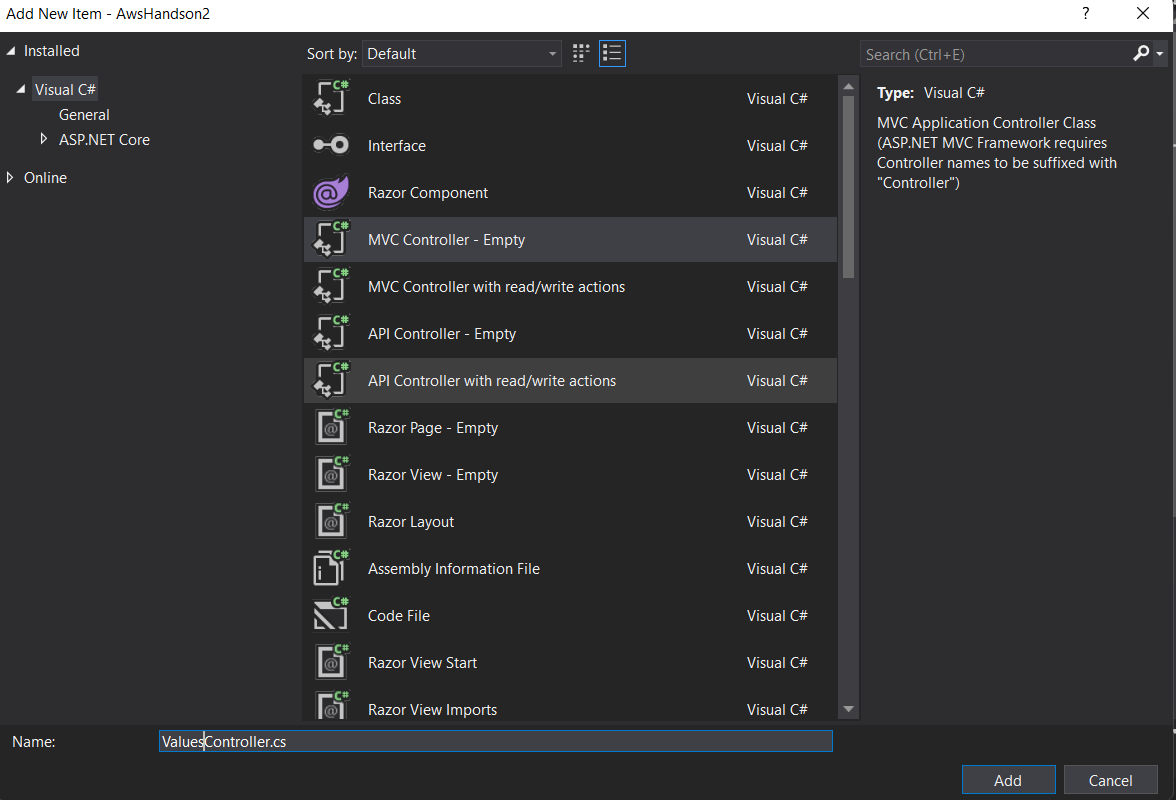
1. Download the csv file and in VS 2019, there will be an option to import csv file in “AWS getting started” page.
2. Once you upload the csv the access key and secret key will get filled as per below screenshot



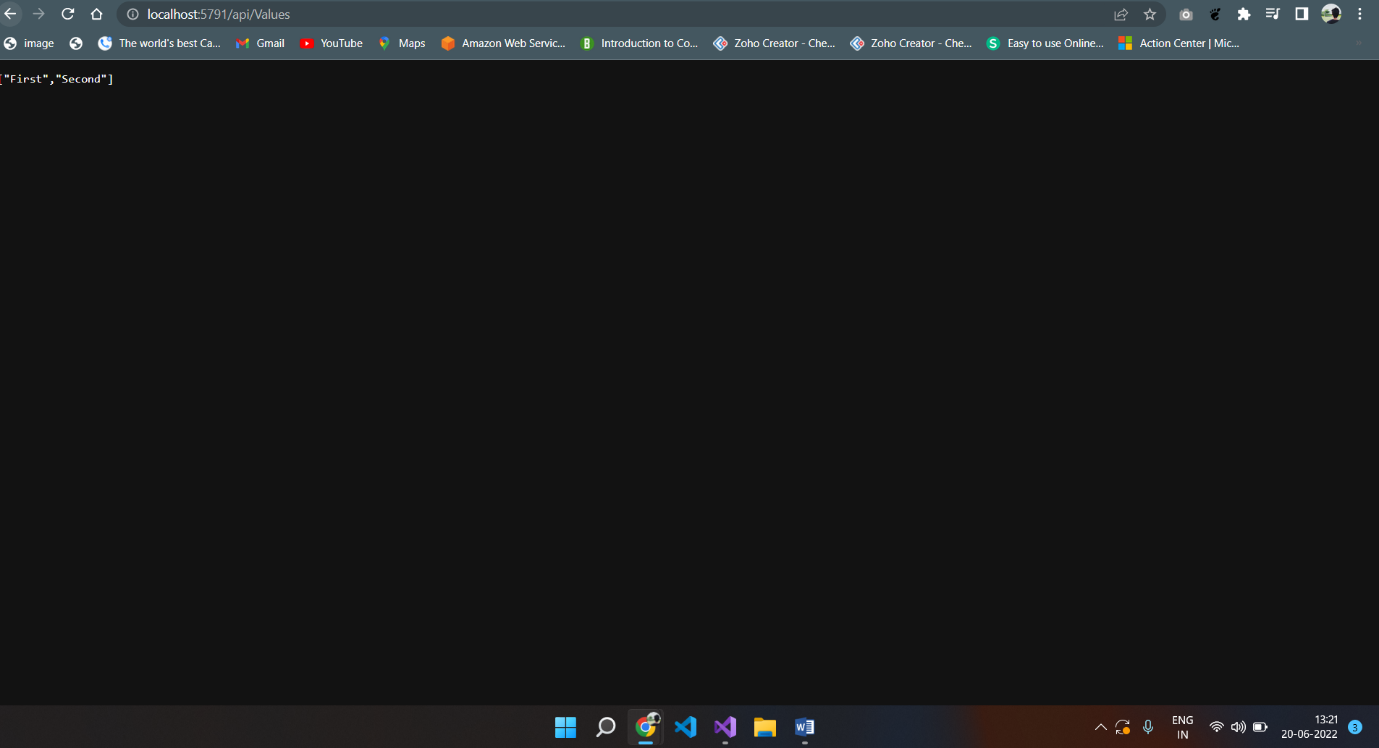
1. Click on Save and Close
2. AWS Explorer will get opened as per the following screenshot. The default profile will be loaded



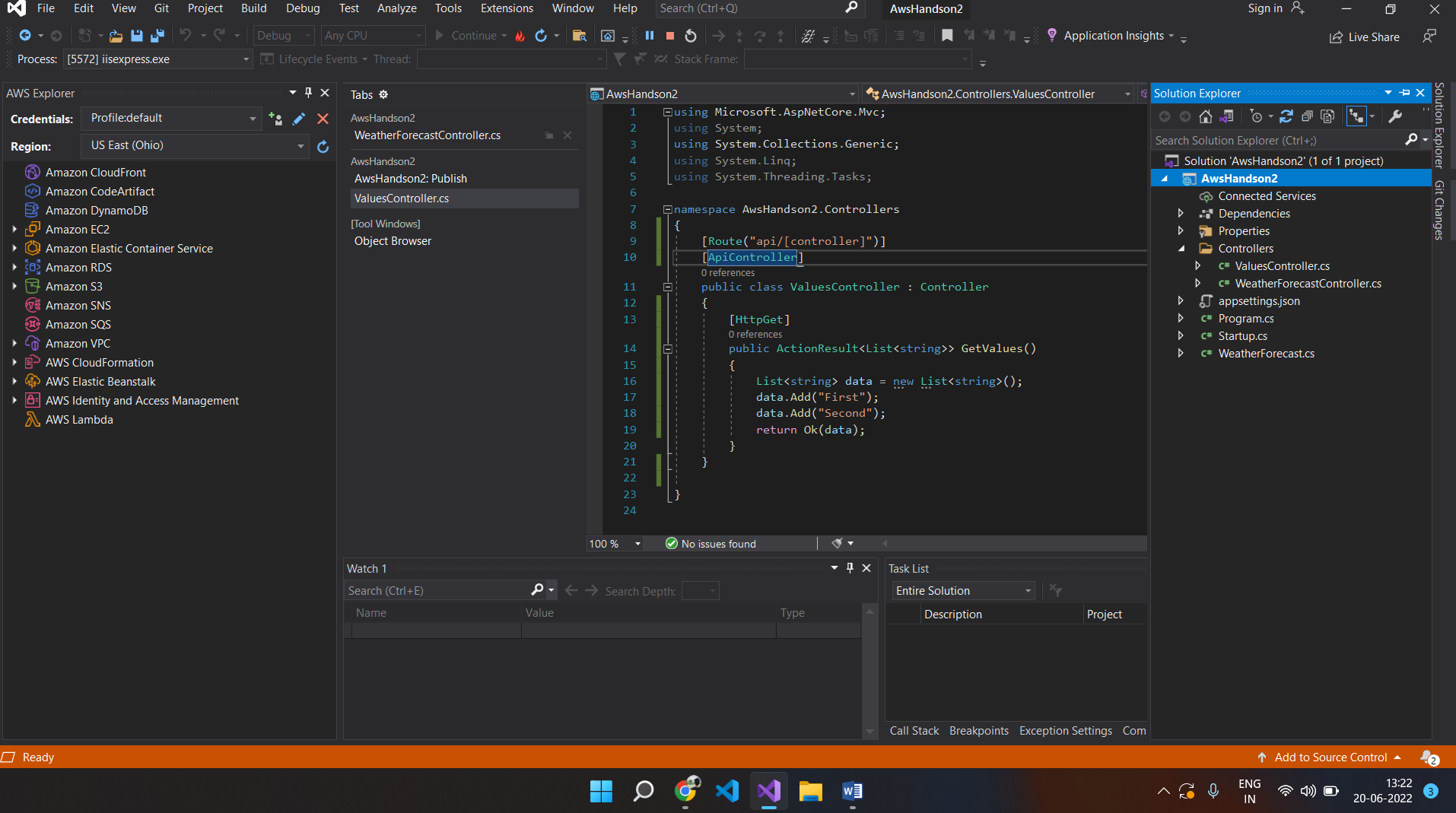
1. Add a new controller in the solution explorer named “ValuesController” and create a HTTPGET method which will return list of string.



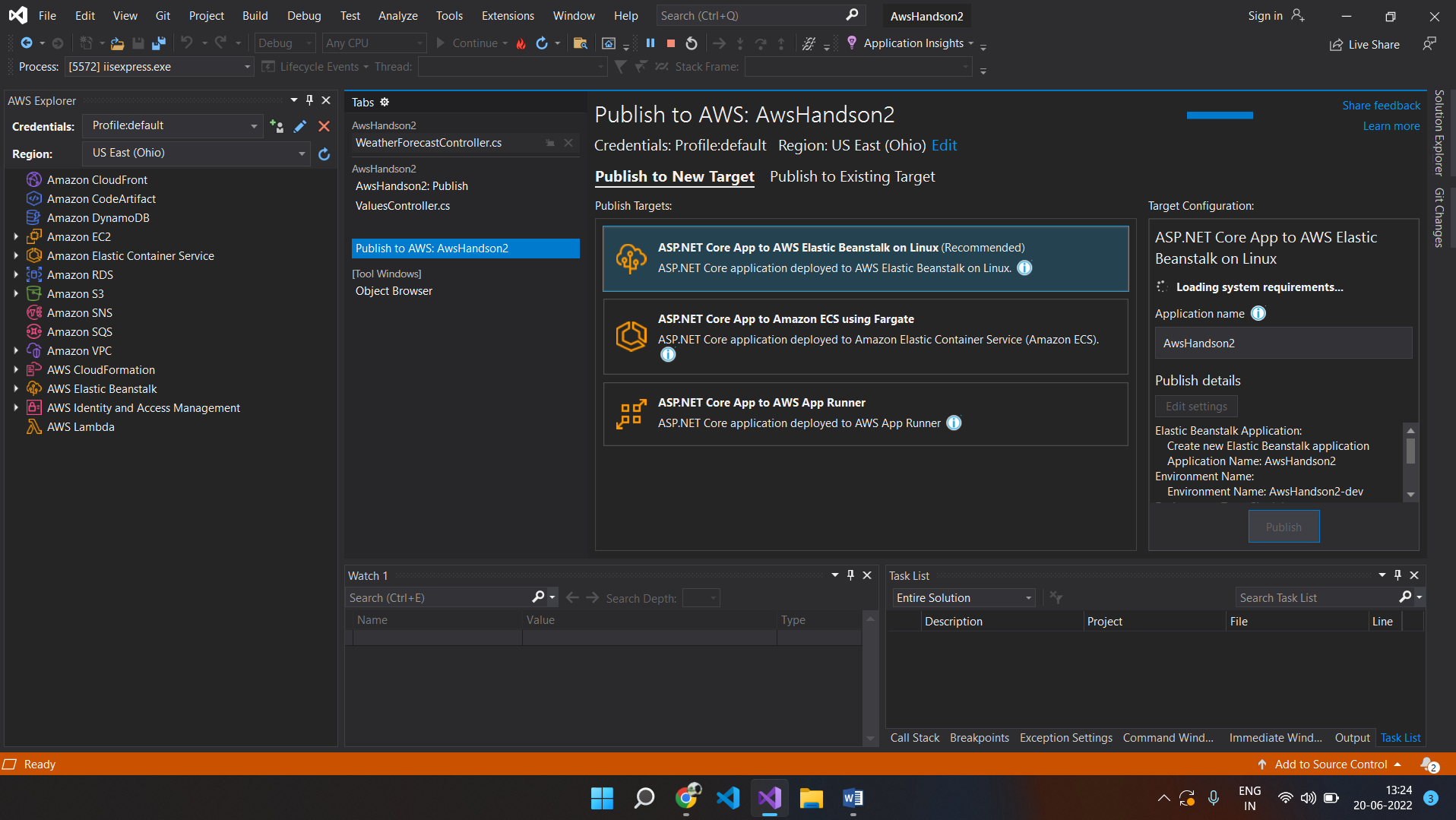
1. When Executed the following output should be generated



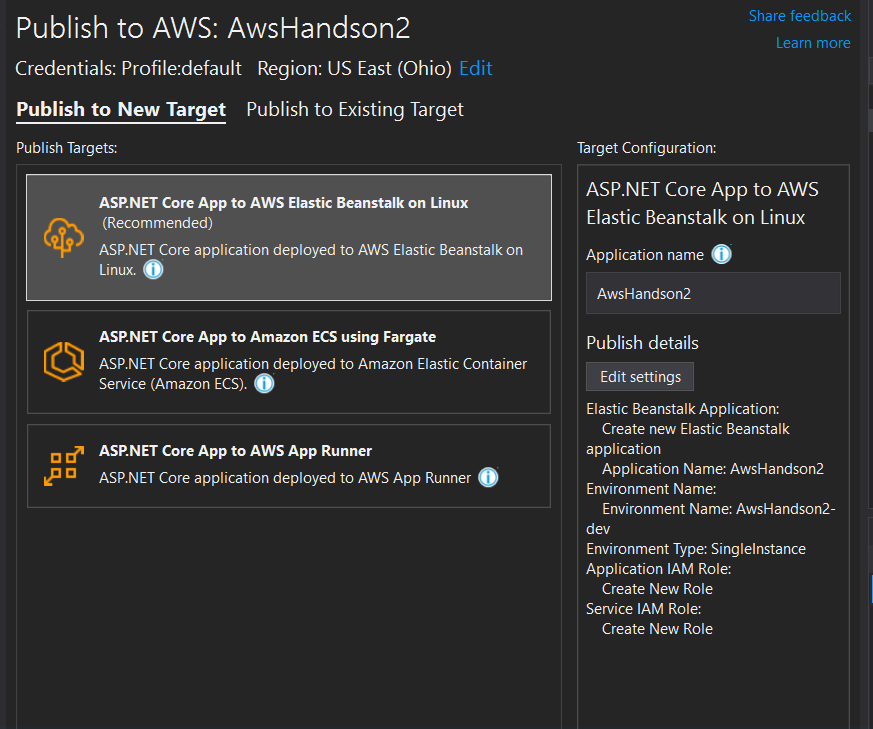
1. Right click on the project in the Solution explorer and there will be an option to publish the microservice on AWS



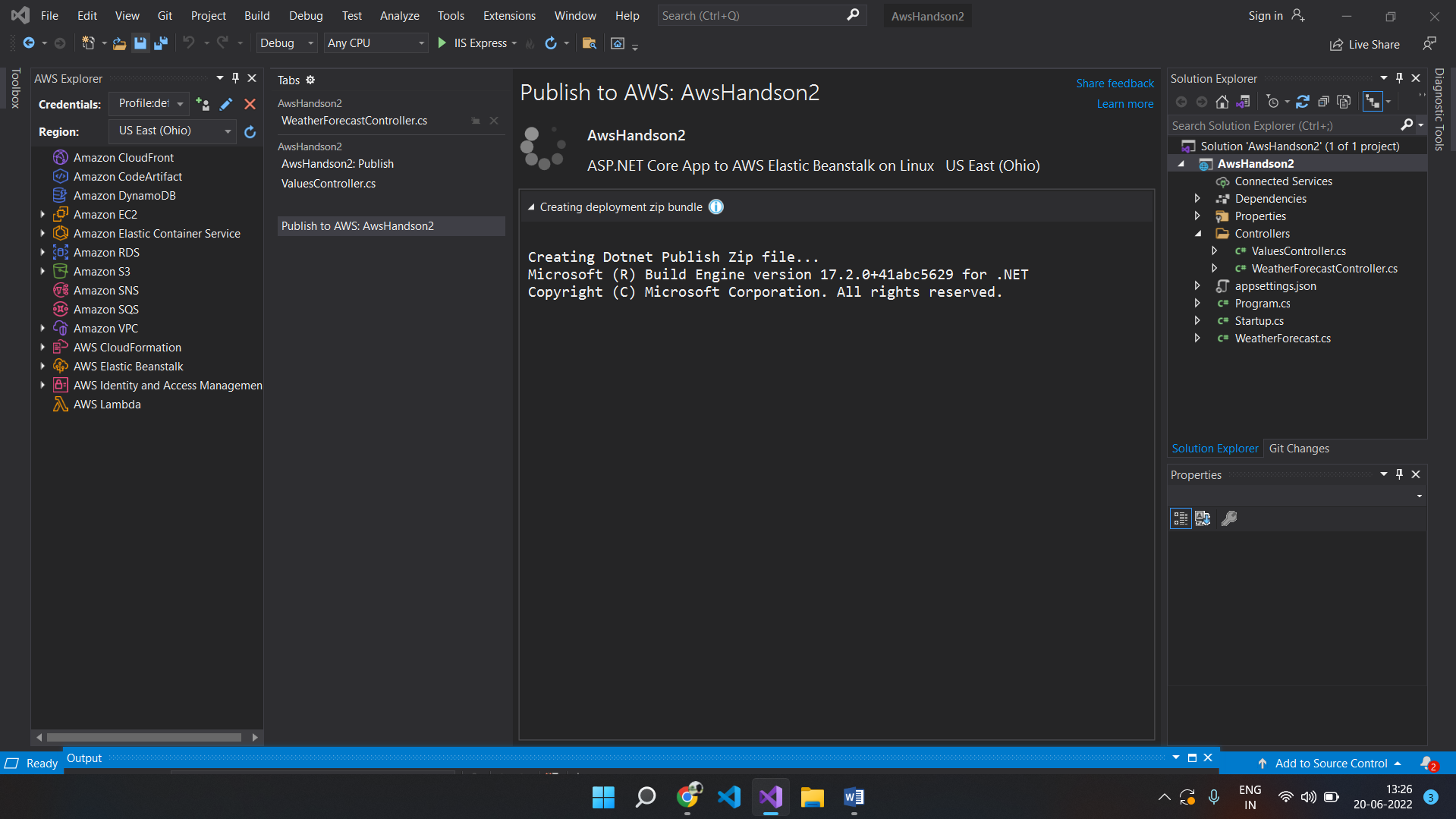
1. The following screen will appear

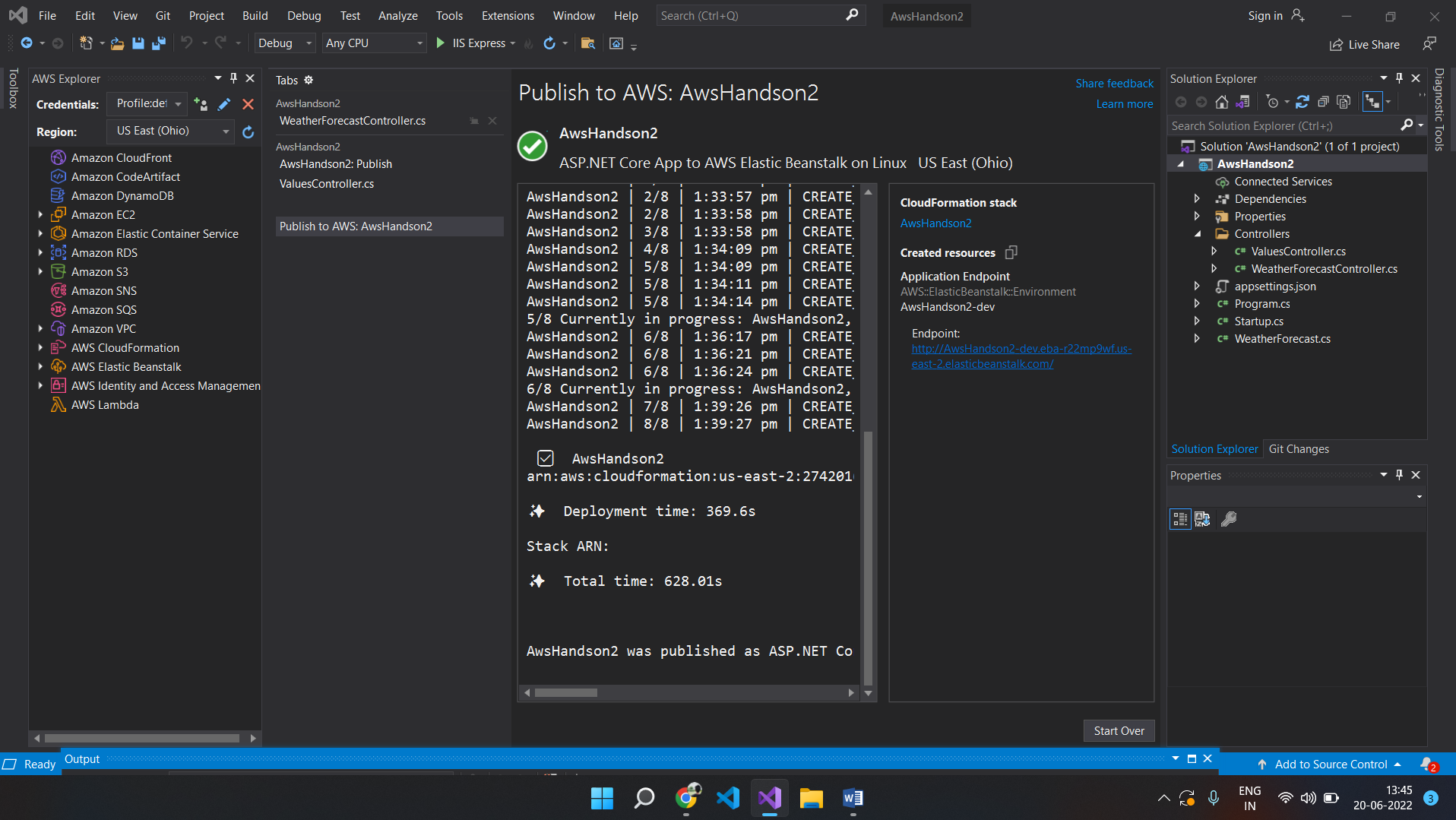


1. Publish to “New Target” and click on “ASP.NET core App to AWS Elastic BeanStalk on Linux” option and click on Publish.

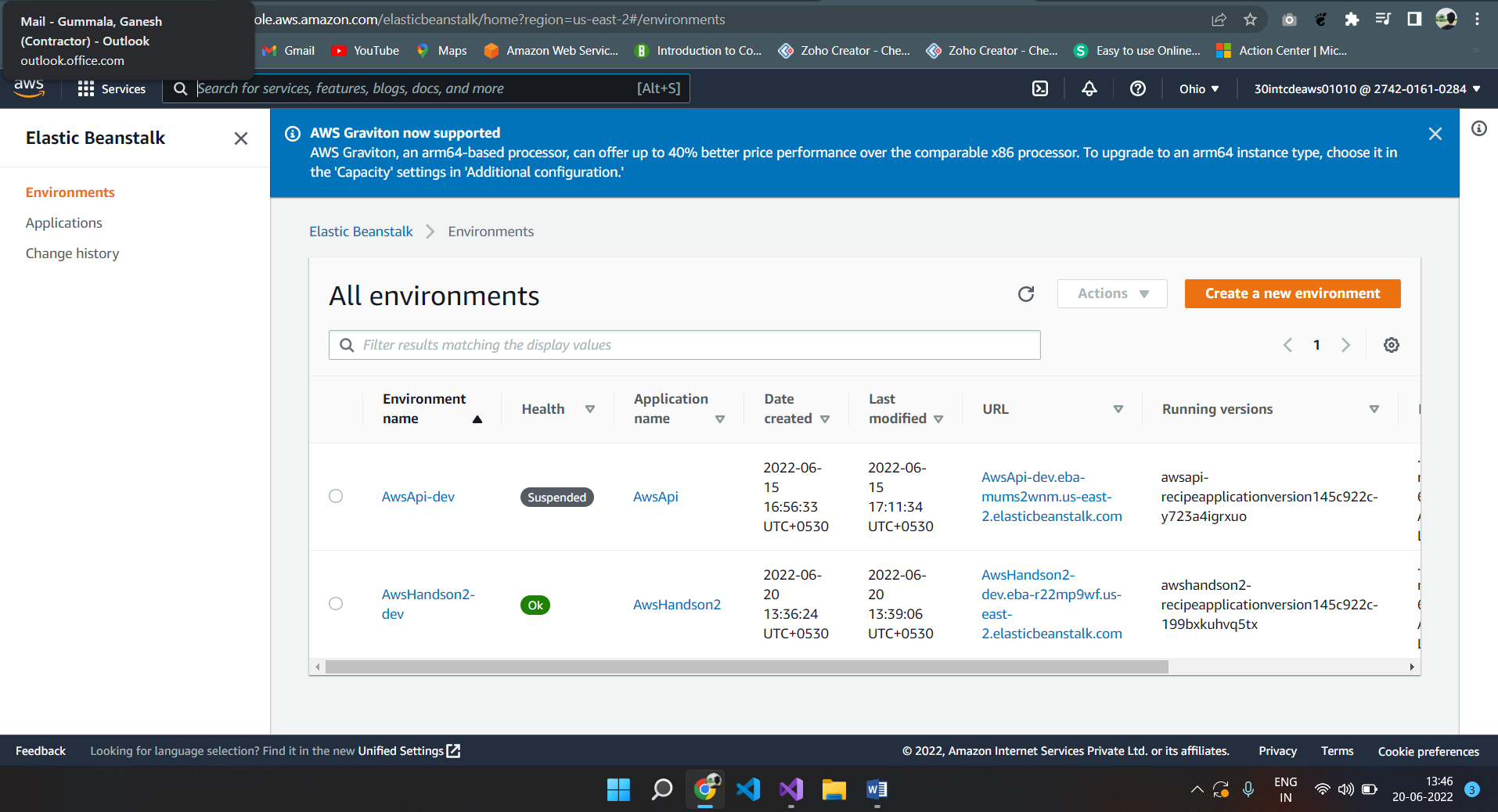


1. Following screen will appear

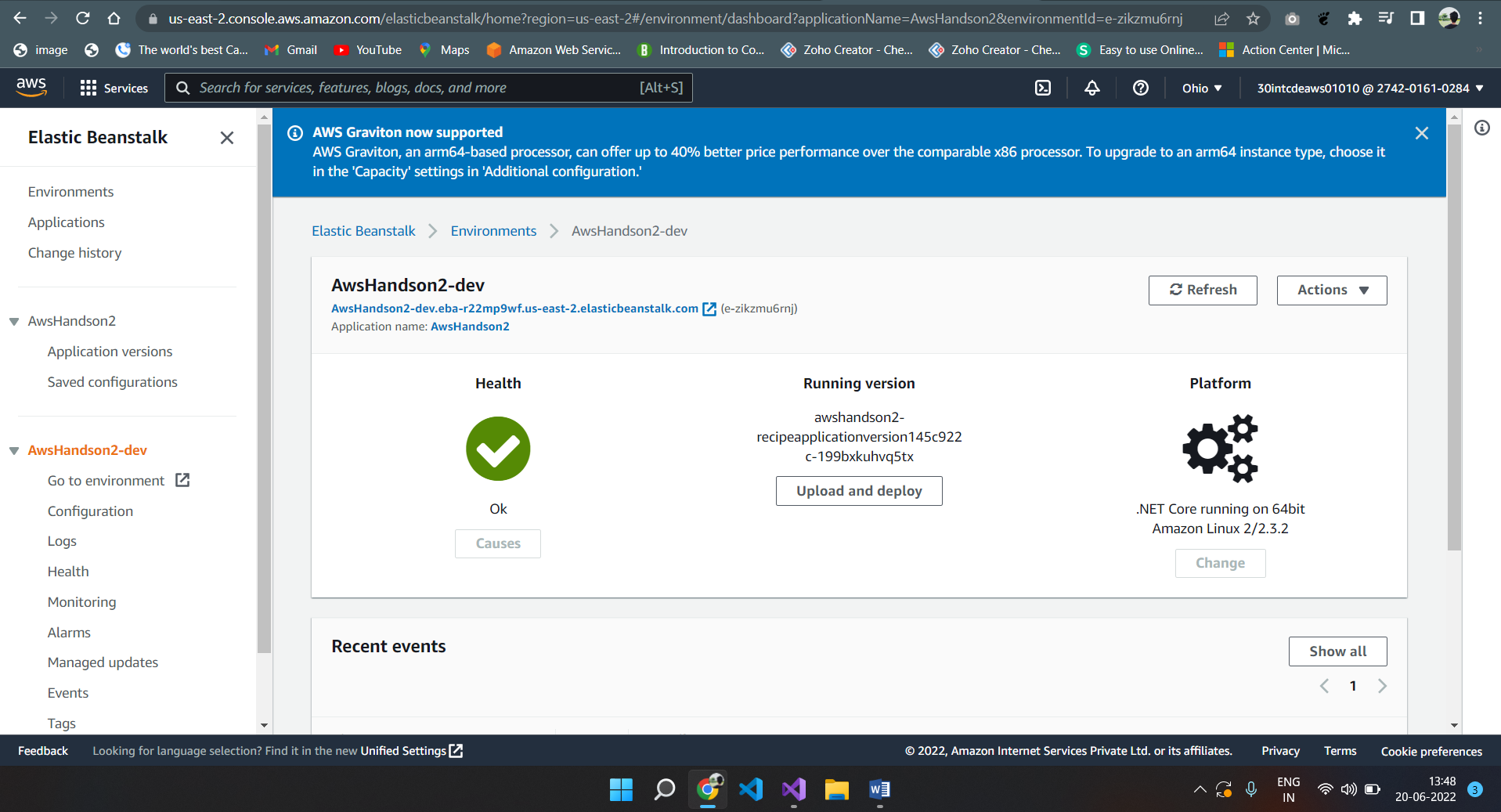




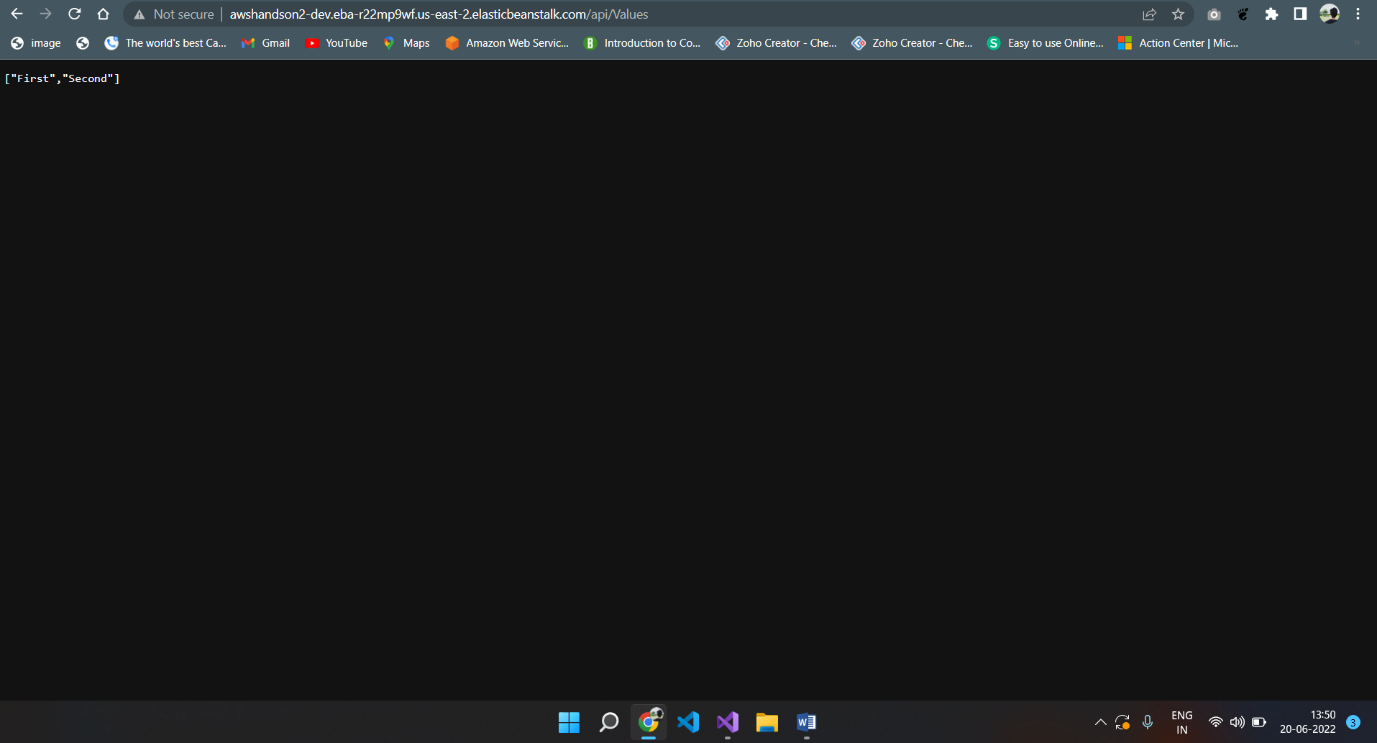
1. Go to Portal and search for Elastic Bean Stalk in the services and find the Environment and Application that we published from VS 2019.



1. Click on the Environment (AWSHandson2-dev) and the following screen will appear



1. Click on the URL and browse for the end point (/api/Values) and the following screen will appear



20. We have successfully deployed the microservice on AWS as Elastic Beanstalk.