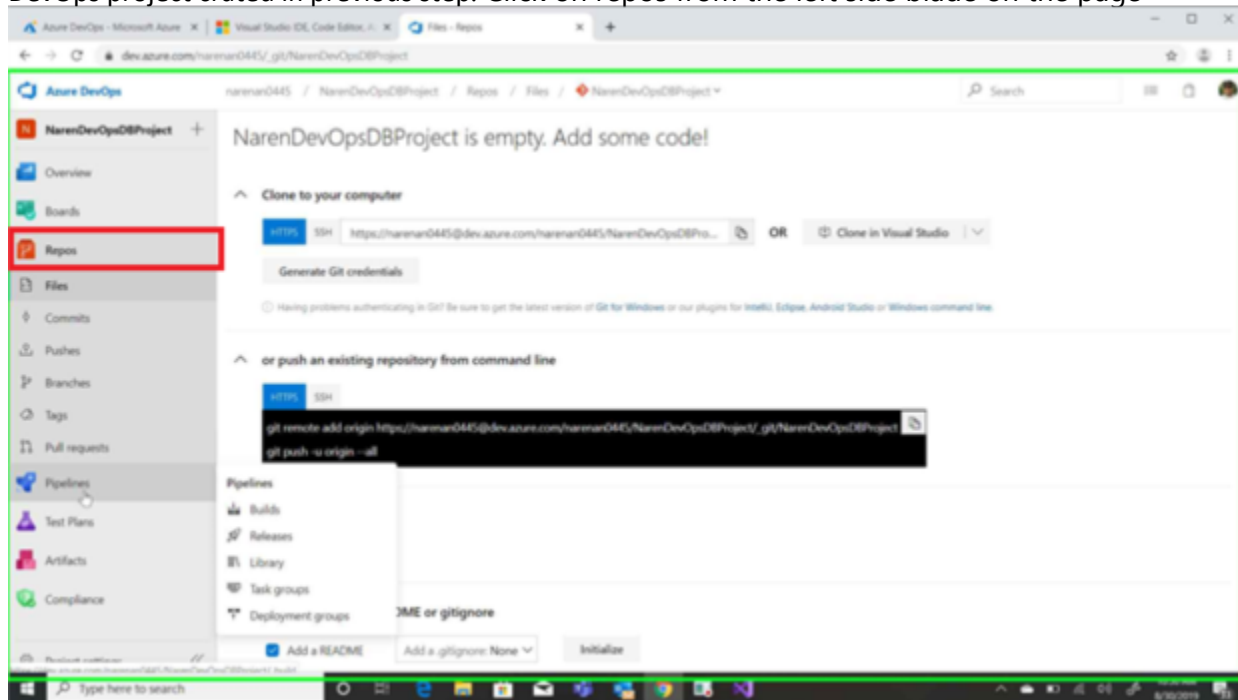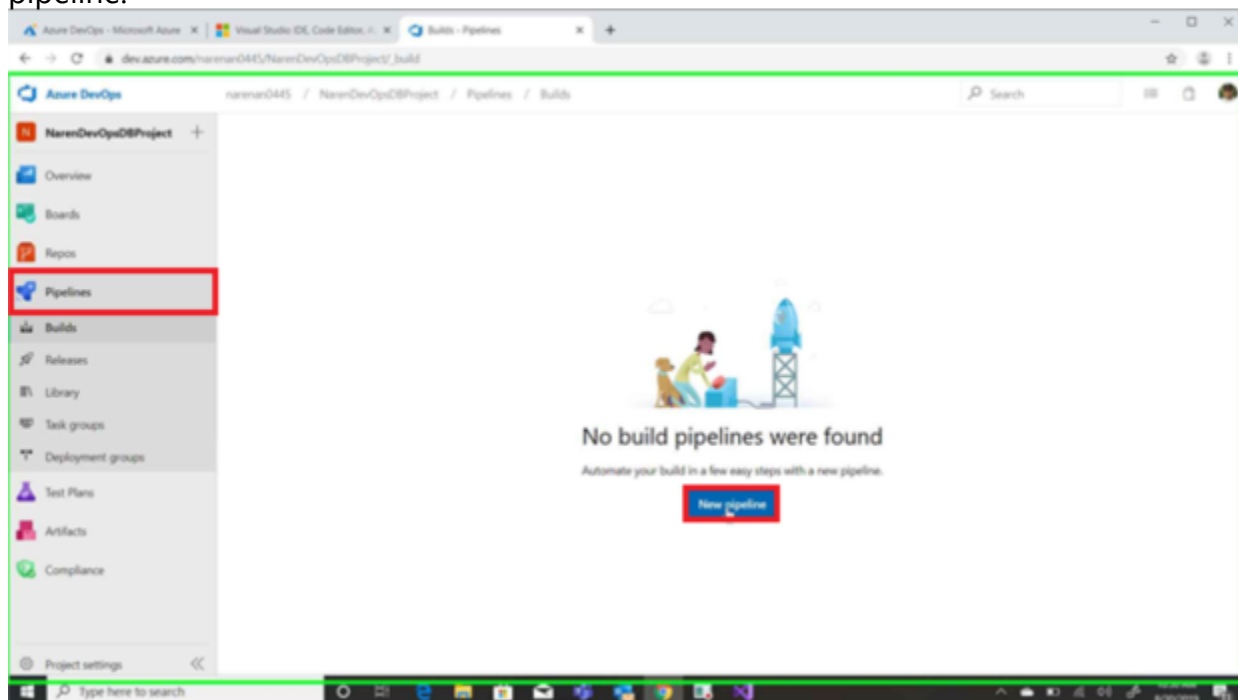# Part 3: Build Pipeline in Azure DevOps

Next step is to create & deploy build pipeline. Let's start creating the build pipeline step by step:

Step 1: If Azure DevOps already open then go to the Azure DevOps page in the browser else Open any browser and enter https://app.vssps.visualstudio.com/ logged in with the proper credentials and select the DevOps project crated in previous step. Click on repos from the left side blade on the page
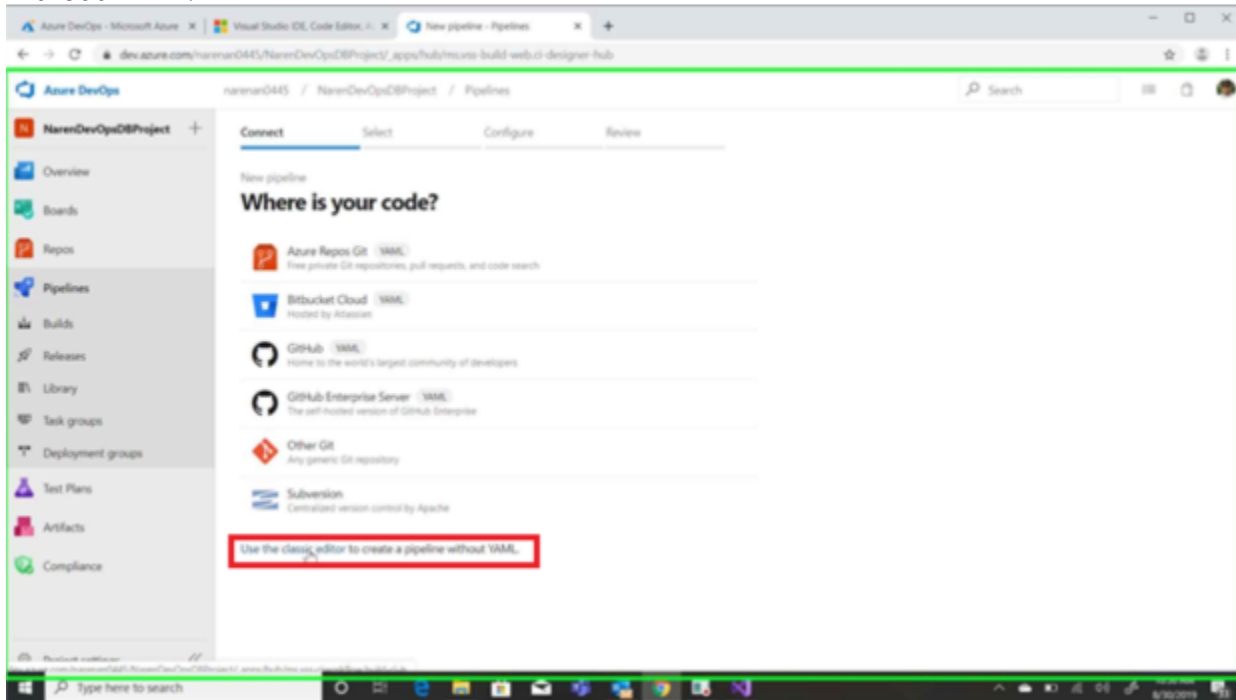

Step 2: On DevOps page click on Pipeline and click on New Pipeline Button to crate new build pipeline.
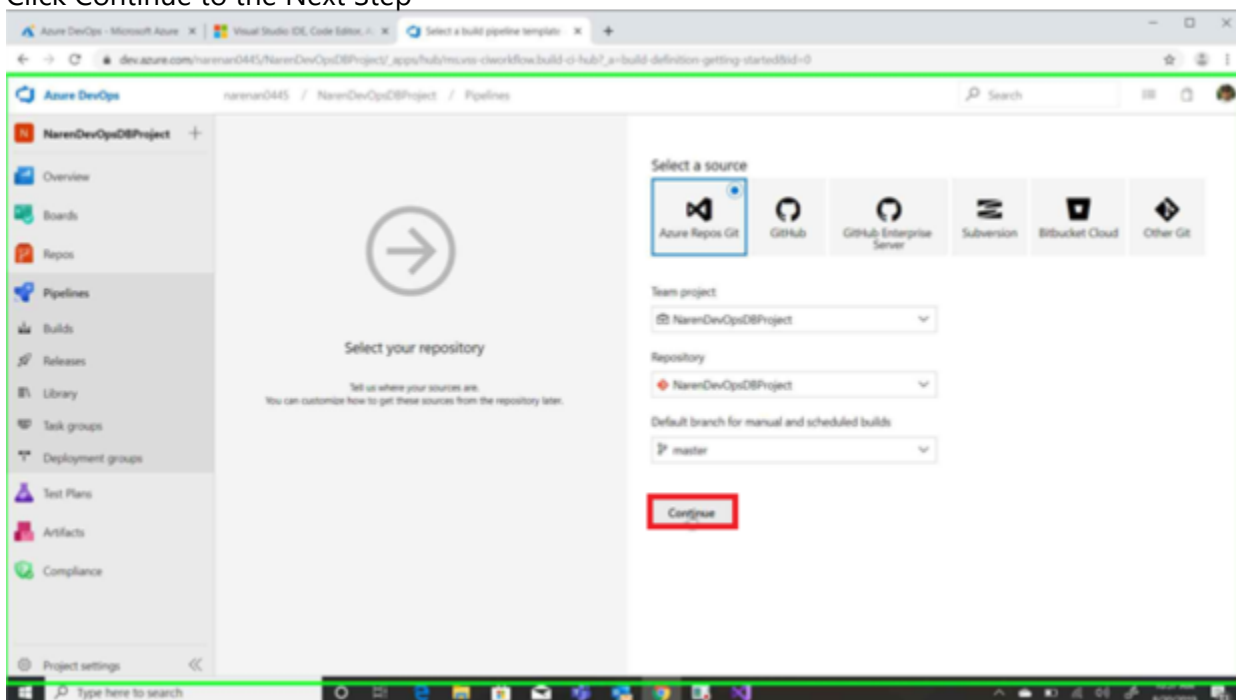

Step 3: From the bottom of the new pipeline page use the Classic editor hyperlink to create a pipeline
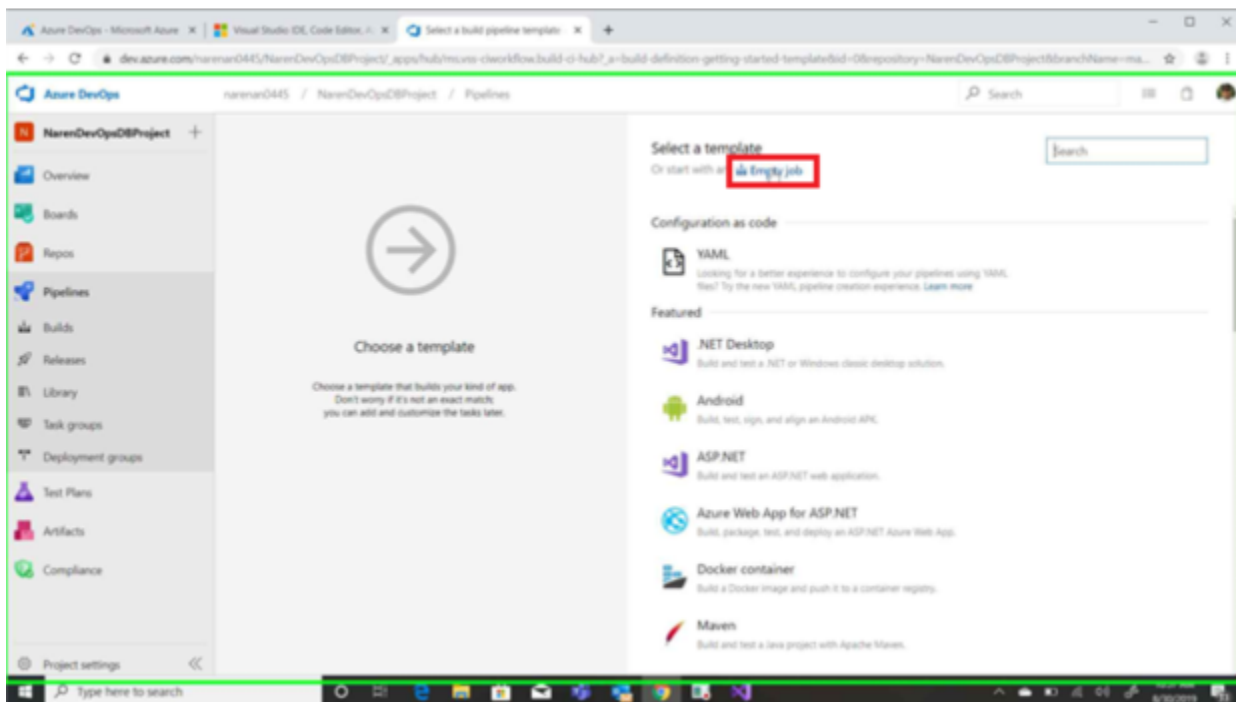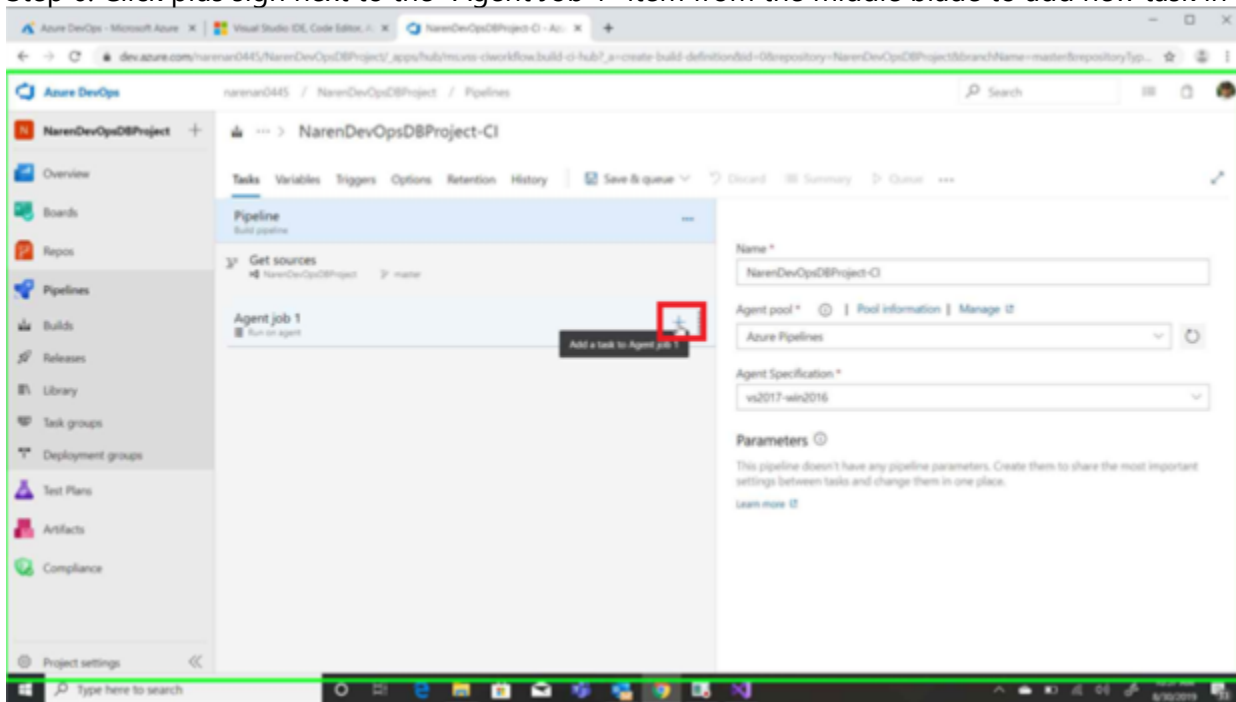
without YAML.

Step 4: It will open "Select a source" blade to the right side of the page where keep the default setting as it. Make sure that Azure Repos Git is selected. Team Project where you will select the project you created in previous part. Select repository as it is and make sure master is selected as a default branch. Click Continue to the Next Step
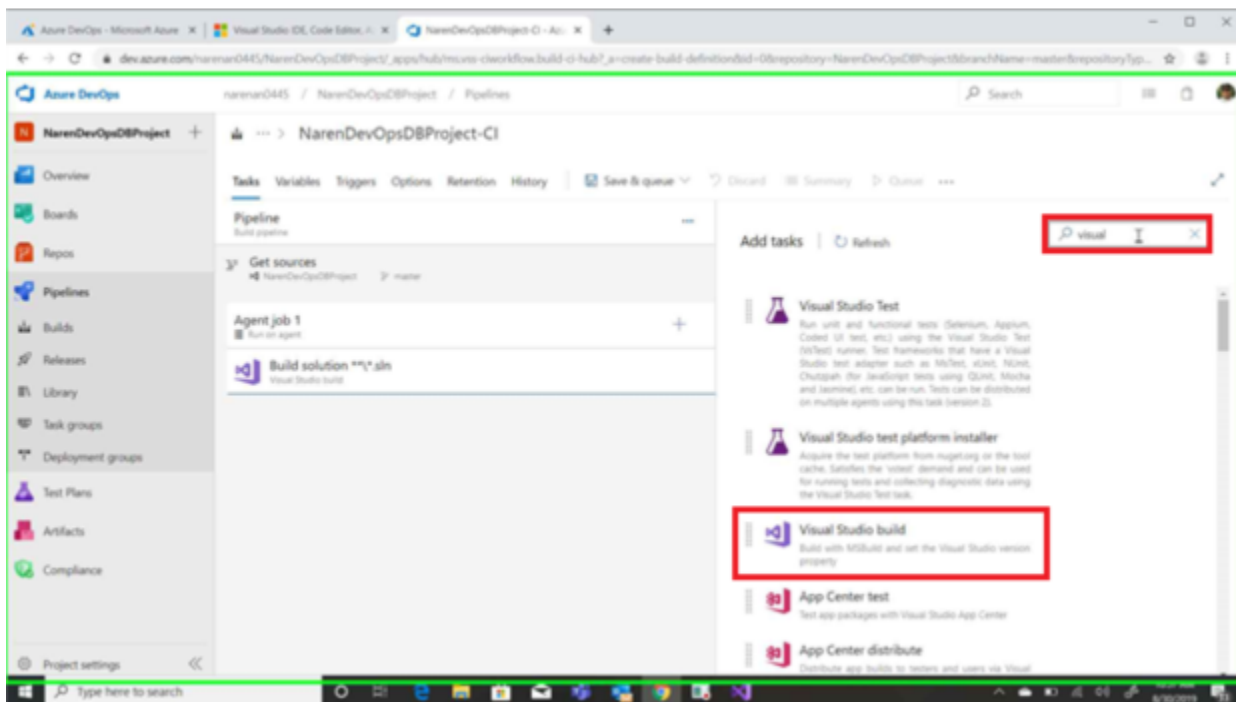

Step 5: Next you will that, "Select a Template" blade is open, click empty Job hyperlink to create new job.
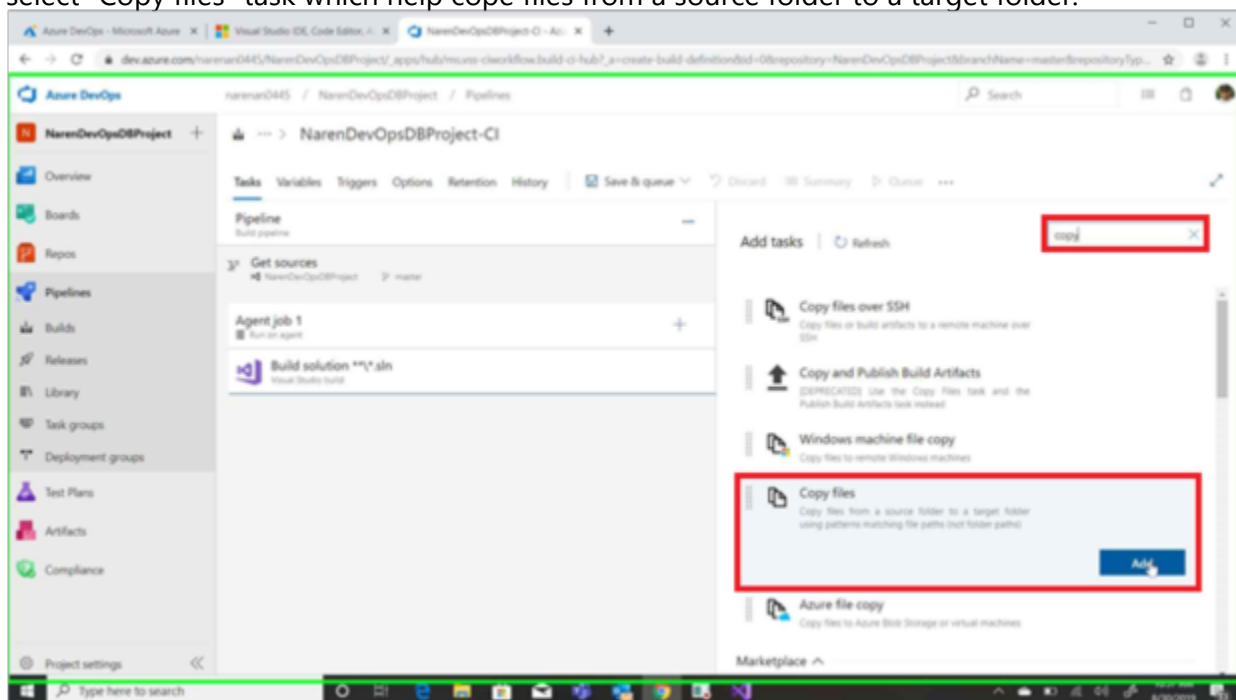
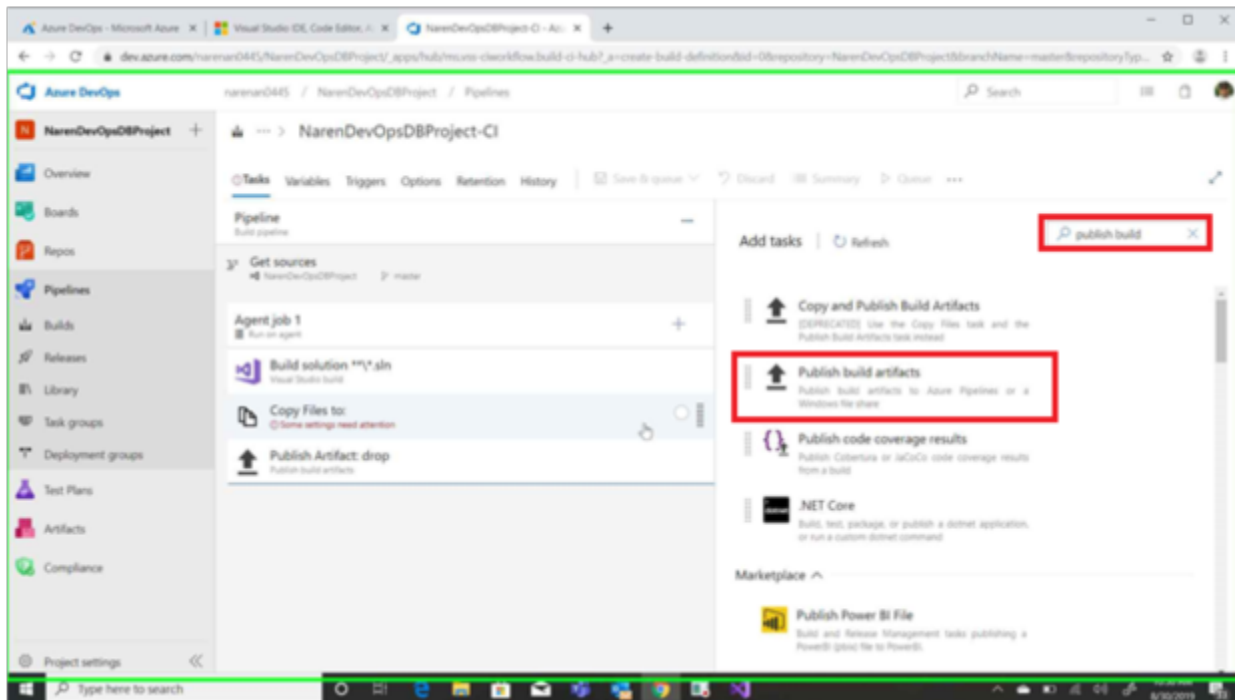Step 6: Click plus sign next to the "Agent Job 1" item from the middle blade to add new task in the job.

Step 7: In the Add Task blade click on Search text box "Visual Studio build" and select "Visual Studio build" task from the "Add Tasks" blade to build with MSBuild and set the visual studio version property.

Step 8: After adding "Visual Studio Build" task, go to the search text box again and type "Copy files" to select "Copy files" task which help cope files from a source folder to a target folder.
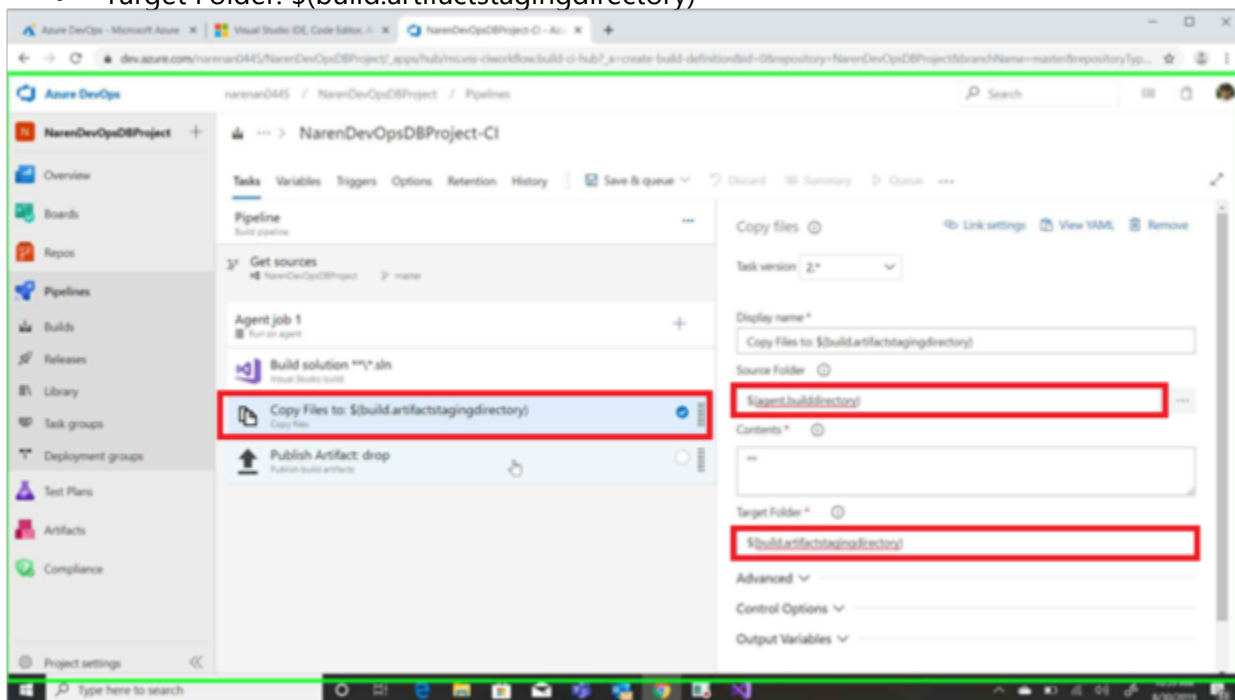

Step 9: After adding copy file task, go back to the search text box again and type "Publish build artifacts" and select "Publish build artifacts" task to publish build artifacts to azure pipeline.
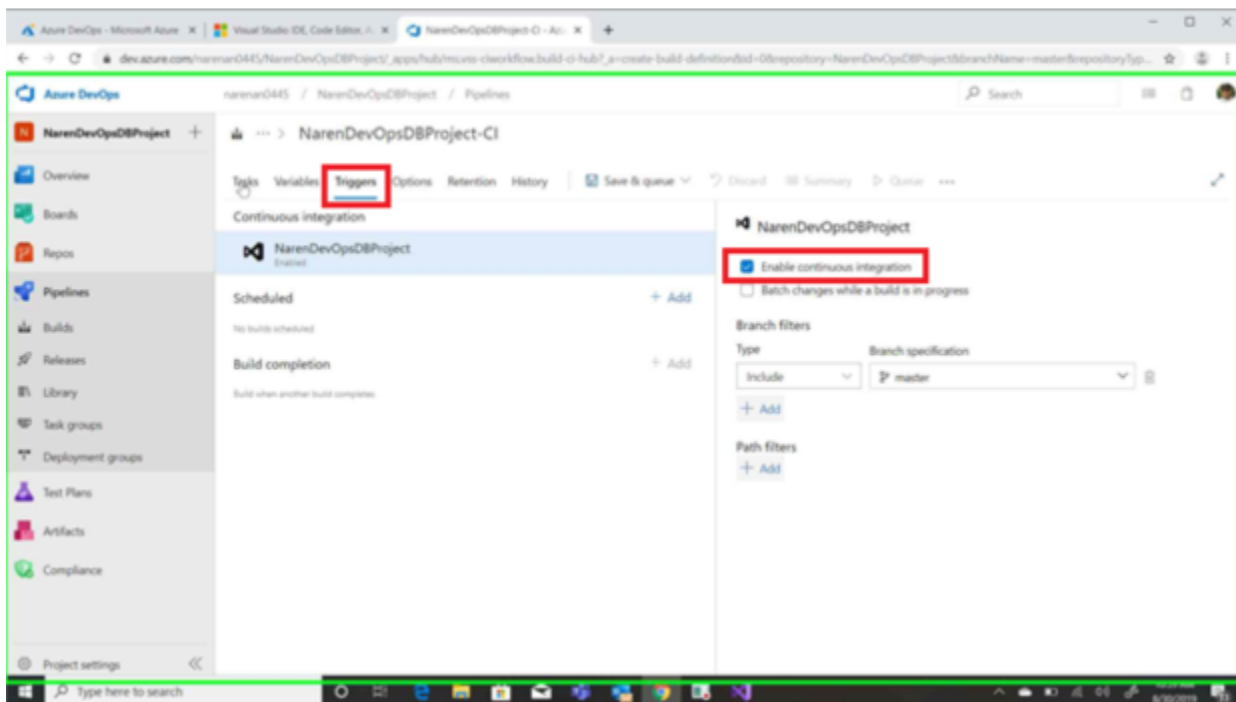
Step 10: Select "Copy files" task from the middle blade to set the property related to the "Copy files" task. Set the properties as follows and make sure that all warnings/errors are gone:
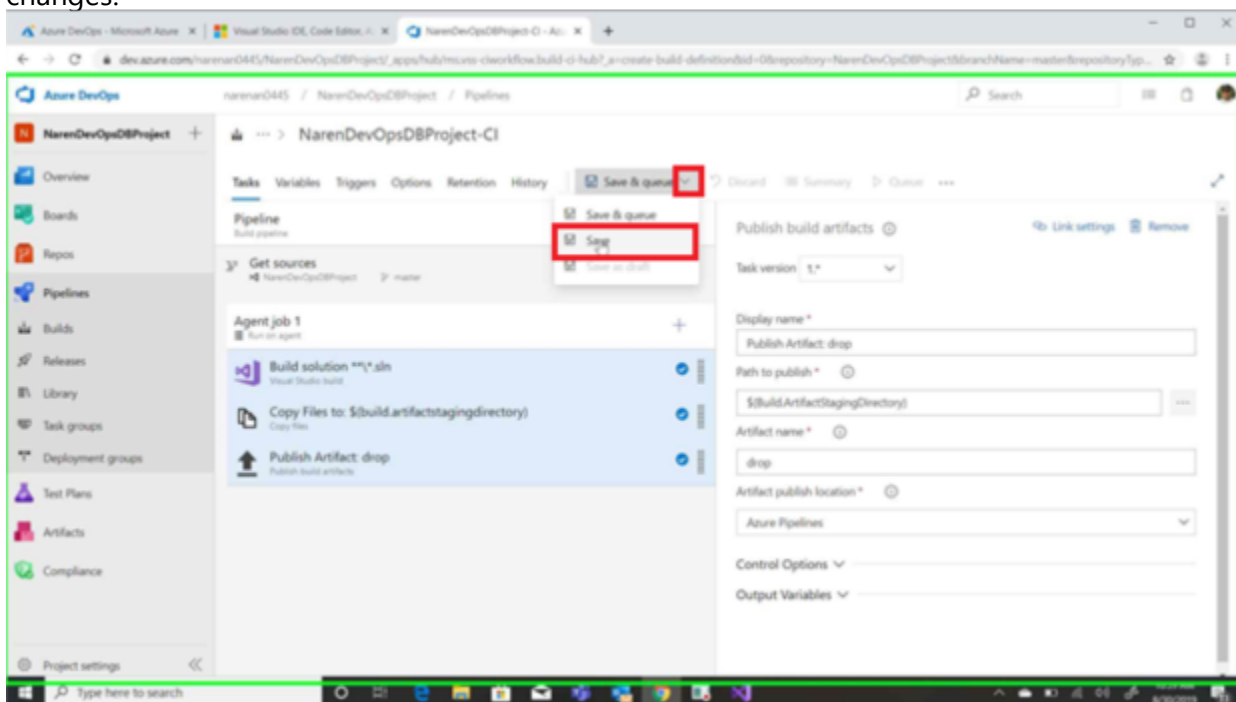- Source Folder: $(agent.builddirectory)
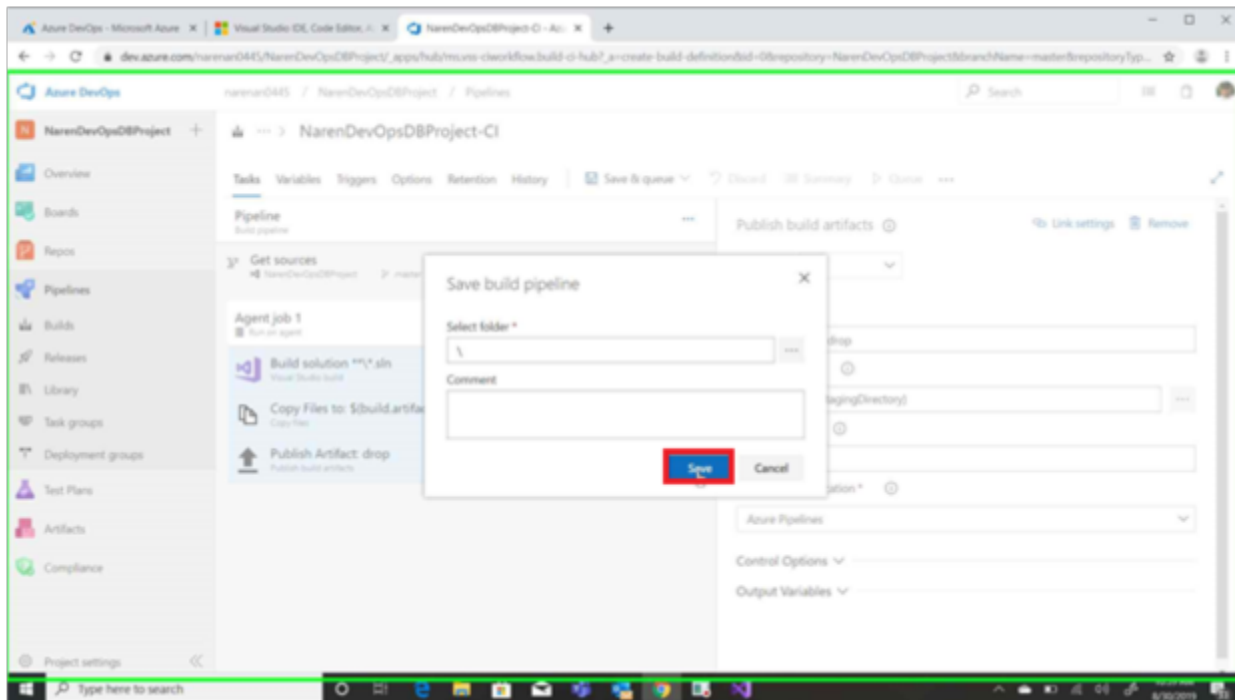- Target Folder: $(build.artifactstagingdirectory)

Step 11: click on Trigger tab from the middle blade to enable continues integration
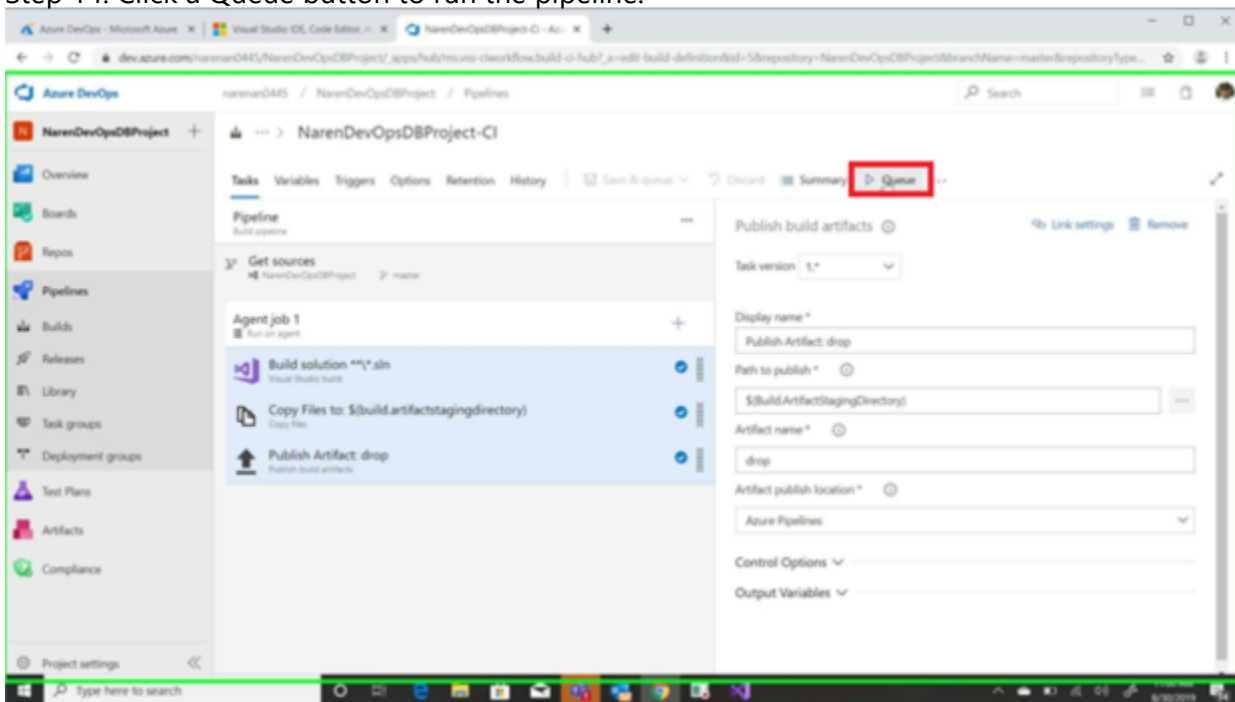
Step 12: Next step is click drop down button next to save & queue and select save option to save the changes.
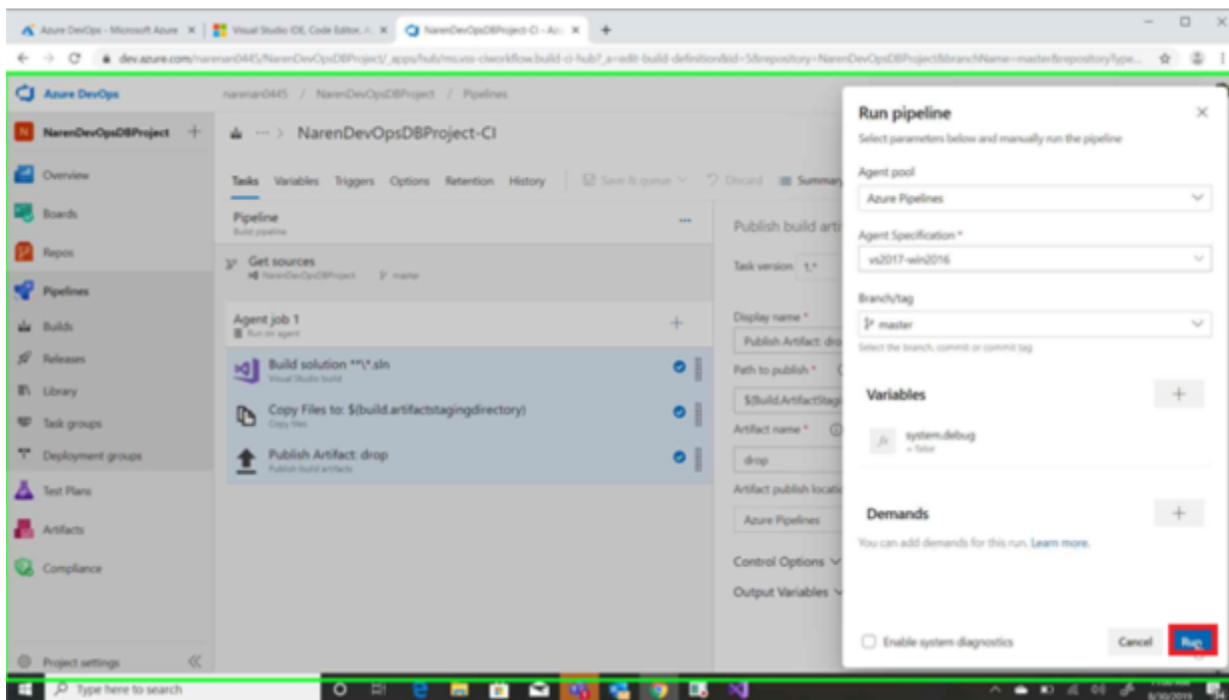

Step 13: It will open save build pipeline window, if you want you can select different folder to save the changes and enter the comments. I am not changing any default setting. click a Save button to save the changes.
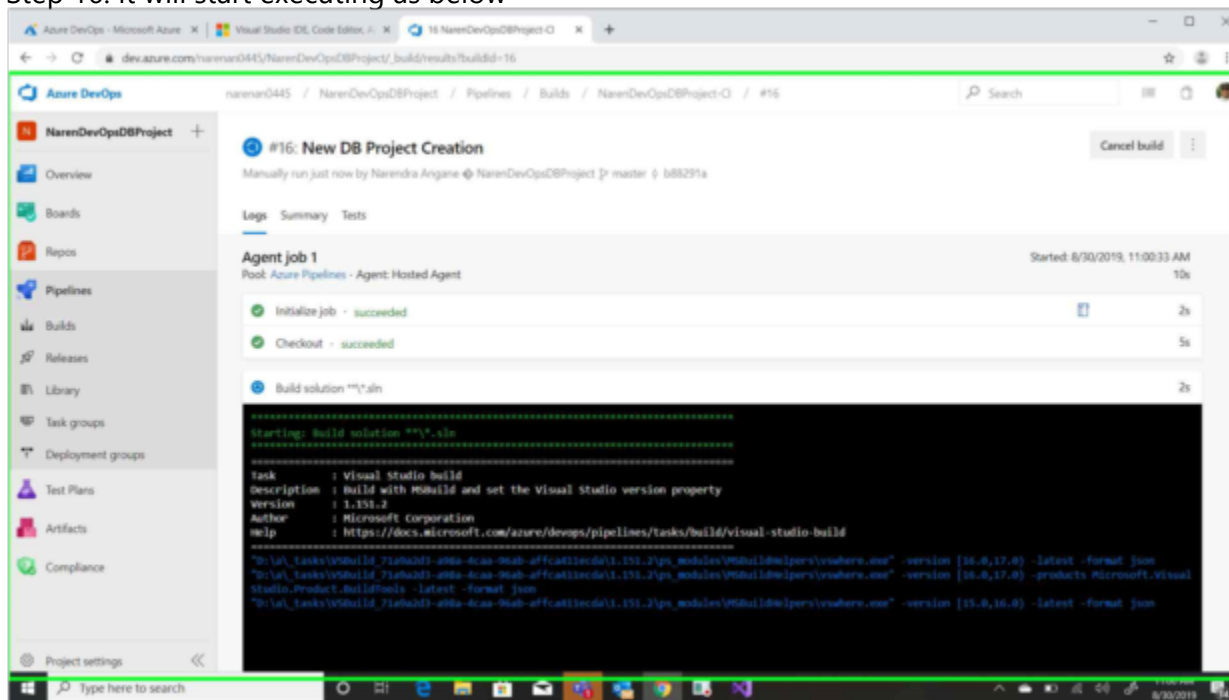
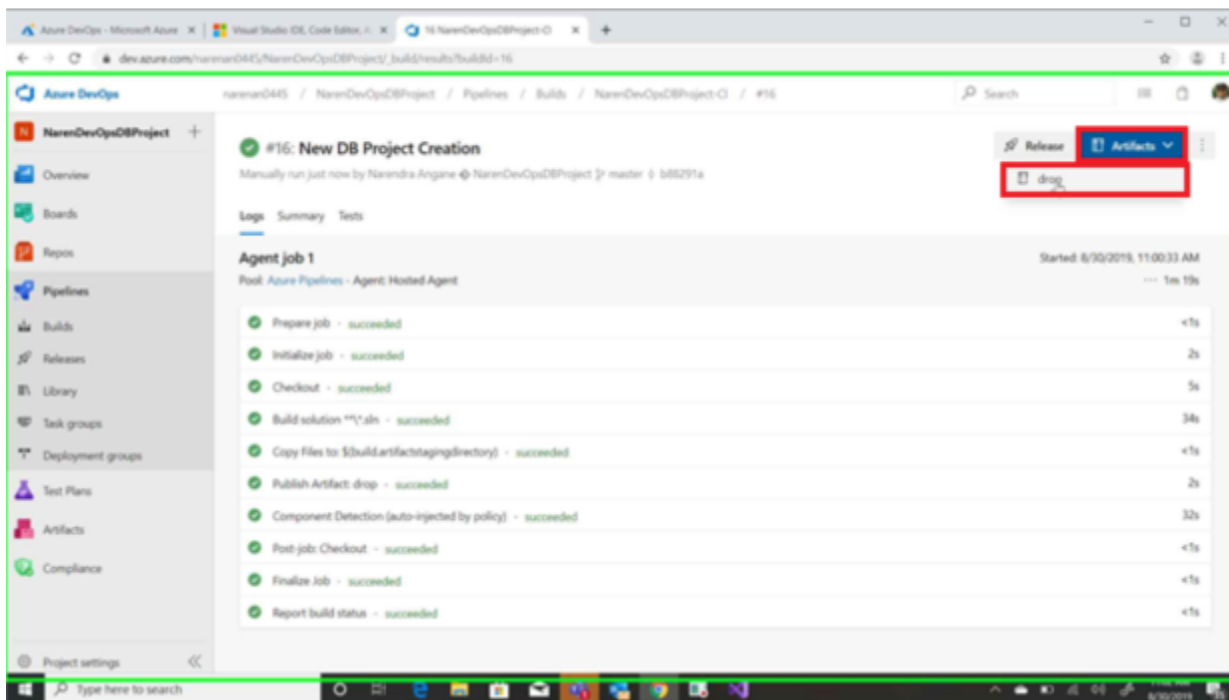Step 14: Click a Queue button to run the pipeline.

Step 15: This will open the Run Pipeline blade. You can set the various parameters related to the run of the pipeline. Will keep all the setting as default and click on Run button to execute the pipeline.
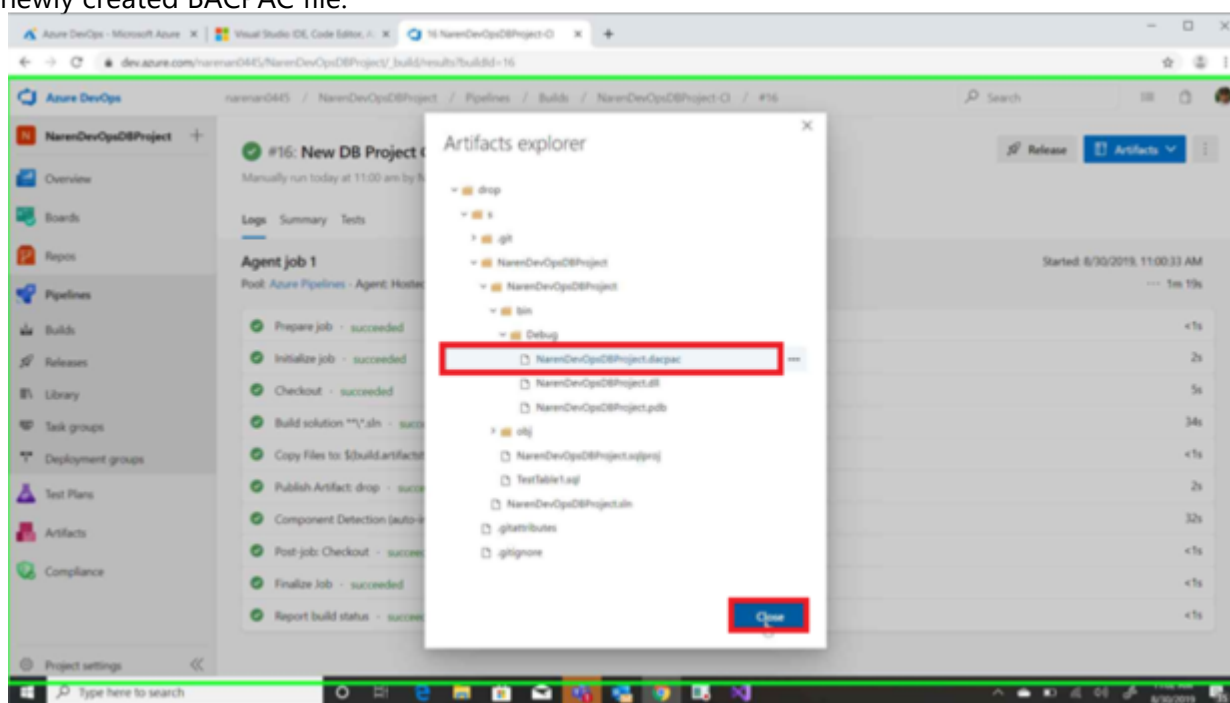
Step 16: It will start executing as below

Step 17: After successful execution you will see the similar screen as below. Next click on the Artifacts dropdown button from the top right of the window and click on drop option.

Step 18: Next from the artifacts explorer you can go to the debug folder to check the availability of newly created BACPAC file.

Congratulations! You have successfully created and executed the build Pipeline.