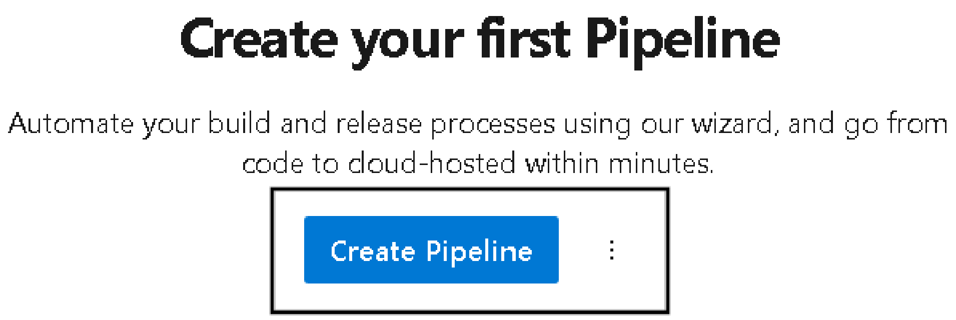
#How to Trigger Pipeline Automatically

What is Azure Pipeline Trigger?

Azure Pipeline triggers are automated ways to start a pipeline at a specific time or in response to an event. They are the building blocks of continuous integration (CI) and continuous delivery (CD).

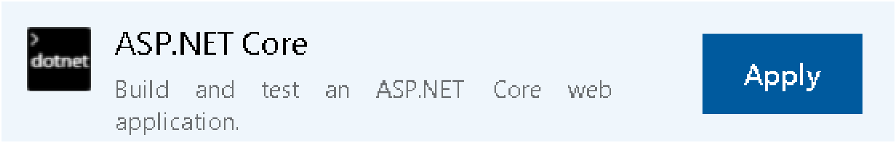
Steps—

Push .Net Application code on repository → enable agent → create pipeline → create pipeline → click on use the classic editor →select azure git → provide repo name and all → continue → search ASP .Net → Click on ASP .Net core → Apply → click inside agent pool → select default → click on triggers → right side under Webapplications → click on enable continuous integration → click on save and queue button → save and run → click on agent job1 → there should executing pipelines → its building → so we have to deploy project so we have to create a web app → duplicate and create a web app in azure.



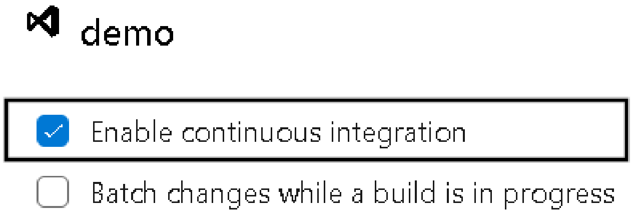
Create your first Pipeline

Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes.



ASP.NET Core

Build and test an ASP.NET Core web application.



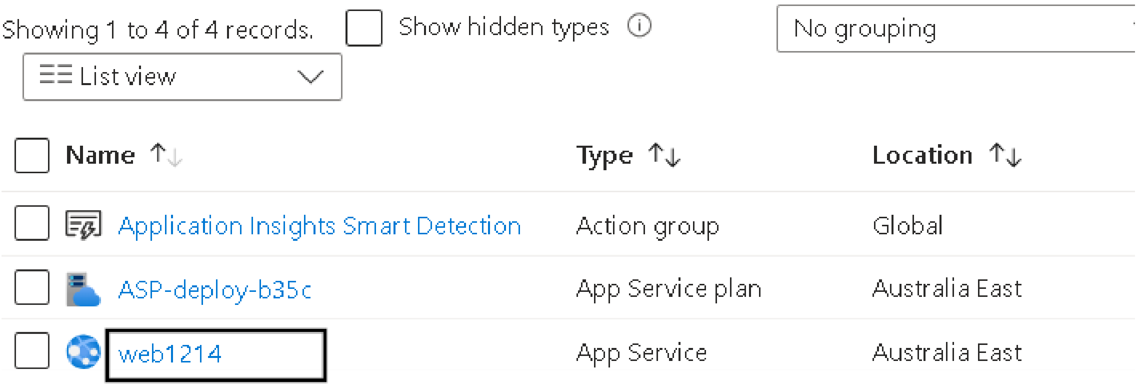
demo

☑ Enable continuous integration

□ Batch changes while a build is in progress

Steps—

After creating web app → go to pipelines tab → click on releases under pipeline → new pipeline → select azure app service deployment → click → click on three dots → save as template → provide name → there should be name 'New Release pipeline' → we have to do mapping of pipeline → click on + Add button in front of artifact → select pipeline from dropdown → add →

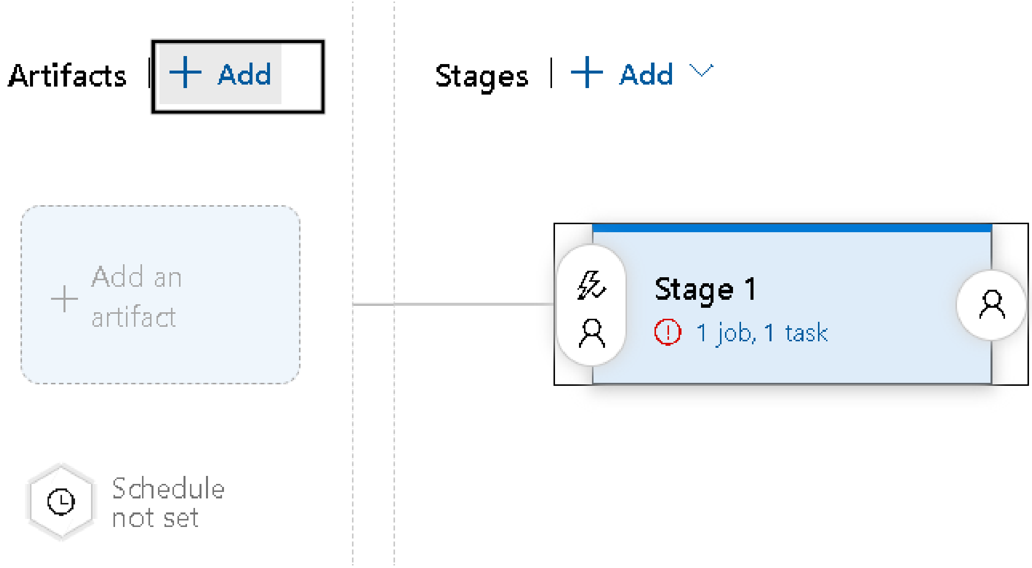


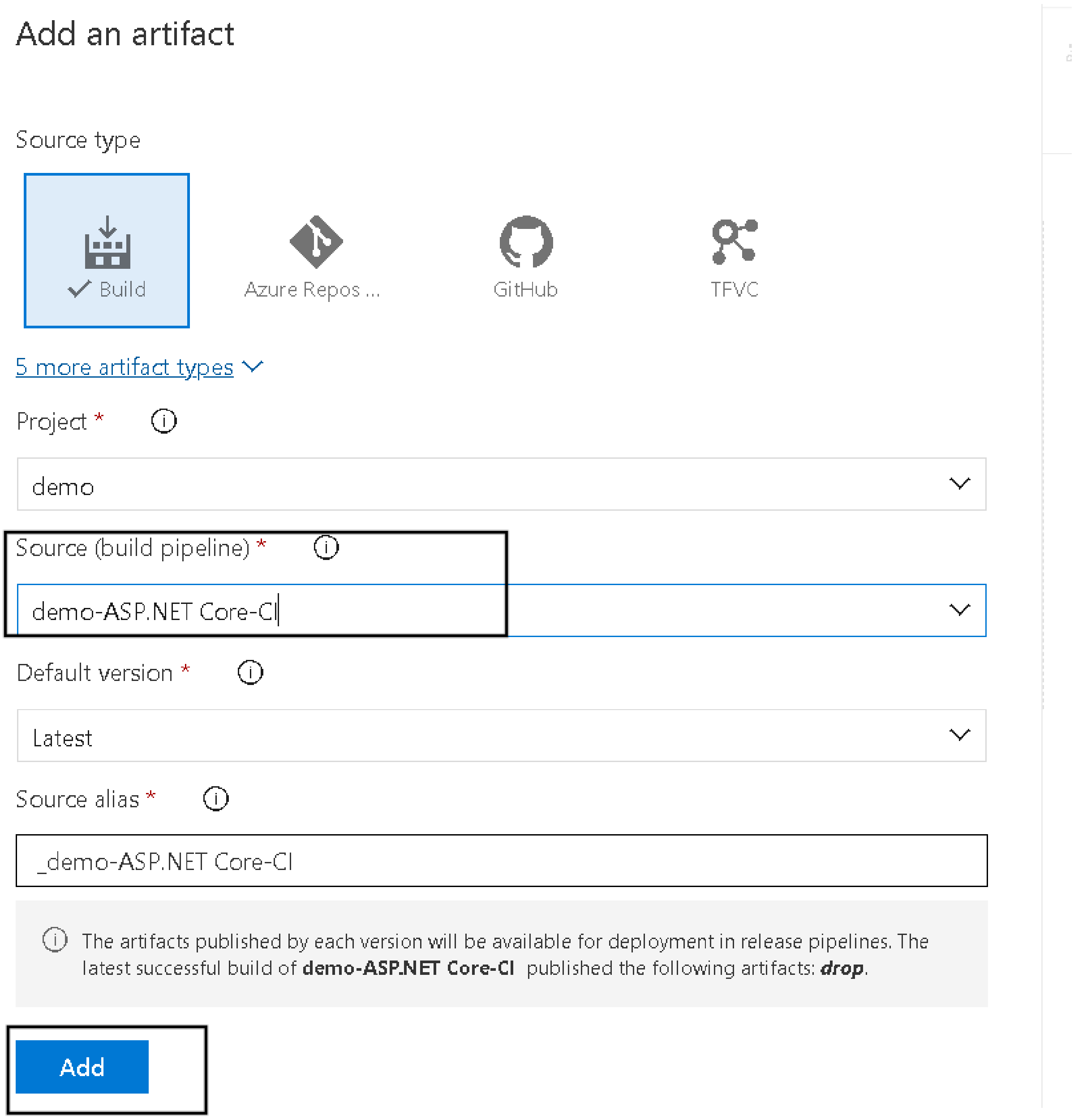
Showing 1 to 4 of 4 records. □ Show hidden types ⓘ No grouping

☰ List view

□	Name ↑↓	Type ↑↓	Location ↑↓
□	Application Insights Smart Detection	Action group	Global
□	ASP-deploy-b35c	App Service plan	Australia East
□	web1214	App Service	Australia East

Here we have created web app. Now duplicate the window of azure.





Add an artifact

Source type

Project * demo

Source (build pipeline) * demo-ASP.NET Core-CI

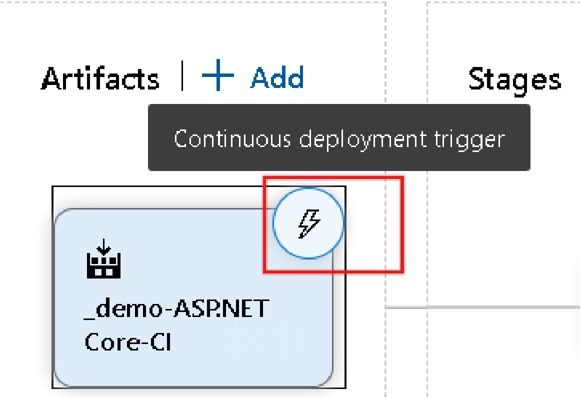
Default version * Latest

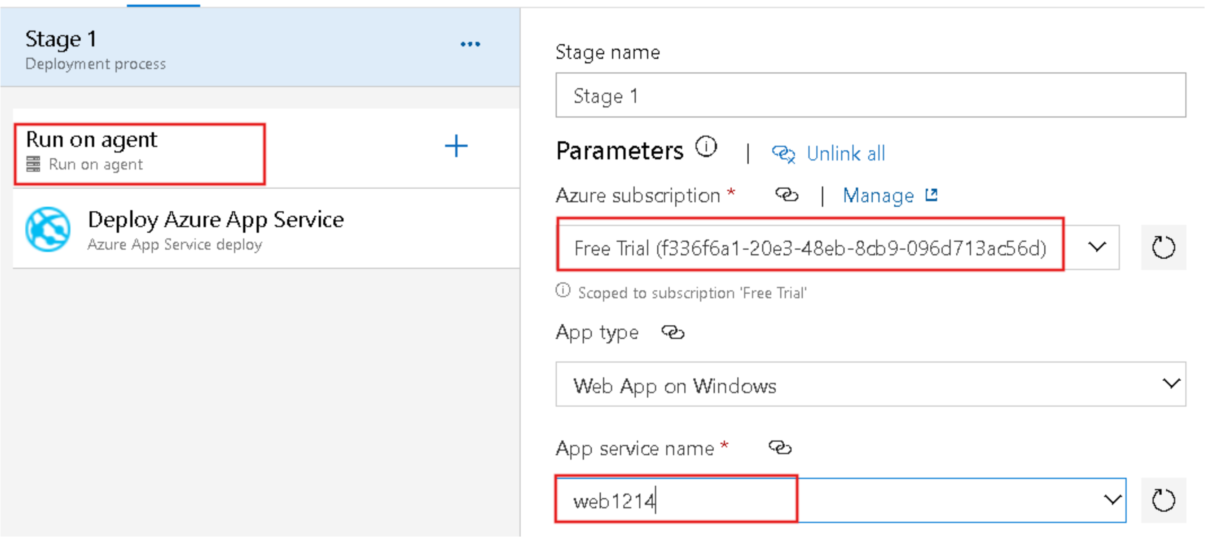
Source alias * _demo-ASP.NET Core-CI

The artifacts published by each version will be available for deployment in release pipelines. The latest successful build of **demo-ASP.NET Core-CI** published the following artifacts: ***drop***.

Steps—

Click on the thunder icon → enable continuous deployment trigger → Save → click on stages one job one task → click on azure subscription → click on free tier → authorized → click web app from azure service app → save → ok → click on run agent → select default agent → save → click on create release → create → click on release 1 from yellow bar → after execute → click on succeed → go to web app → copy domain name and paste it in new tab → there should be running project





Stage name: Stage 1

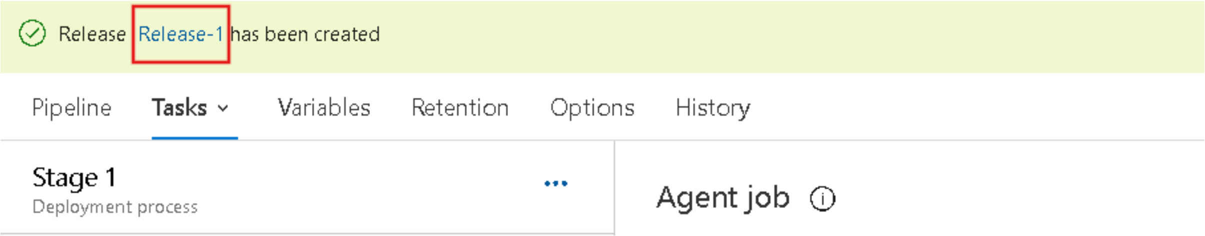
Azure subscription *: Free Trial (f336f6a1-20e3-48eb-8cb9-096d713ac56d)

Scoped to subscription 'Free Trial'

App type: Web App on Windows

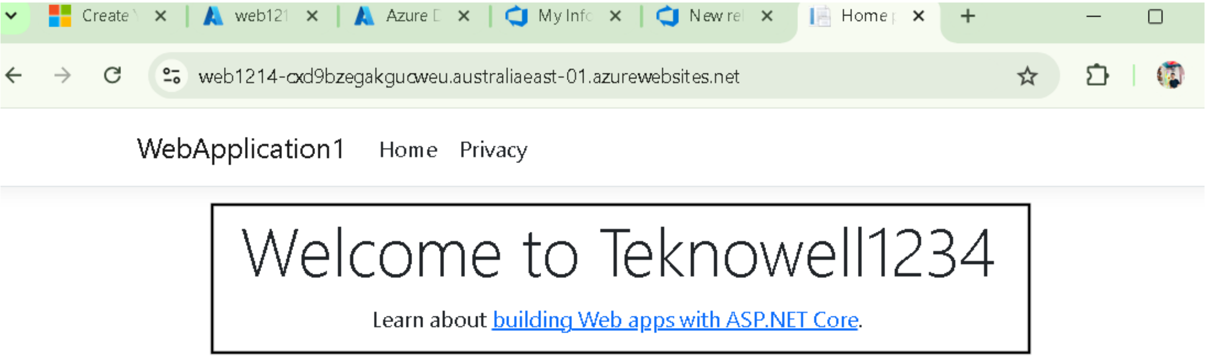
App service name *: web1214

We have created CICD pipeline. Now we have to run this.



Release Release-1 has been created

So the deployment has started. Now copy the domain name and paste it in new tab.



web1214-cxd9bzegakguoweu.australiaeast-01.azurewebsites.net

Welcome to Teknowell1234

Learn about building Web apps with ASP.NET Core.

We have successfully trigger pipeline automatically. Change something in the view pages. Check the CI and CD Pipeline and check the tab again after refresh.