



DEVLERAZUREDA - DevOps Academy

03. Create a build pipeline with Azure

Pipelines

Mehmet Kut –Technical Trainer @Microsoft



Github – DevOps Academy

<https://github.com/msdevopsacademy/>

Repo:

03-create-a-build-pipeline-with-azure-pipelines

Azure Pipelines

Cloud-hosted pipelines for Linux, Windows and macOS.



Any language, any platform, any cloud

Build, test, and deploy Node.js, Python, Java, PHP, Ruby, C/C++, .NET, Android, and iOS apps. Run in parallel on Linux, macOS, and Windows. Deploy to Azure, AWS, GCP or on-premises



Extensible

Explore and implement a wide range of community-built build, test, and deployment tasks, along with hundreds of extensions from Slack to SonarCloud. Support for YAML, reporting and more

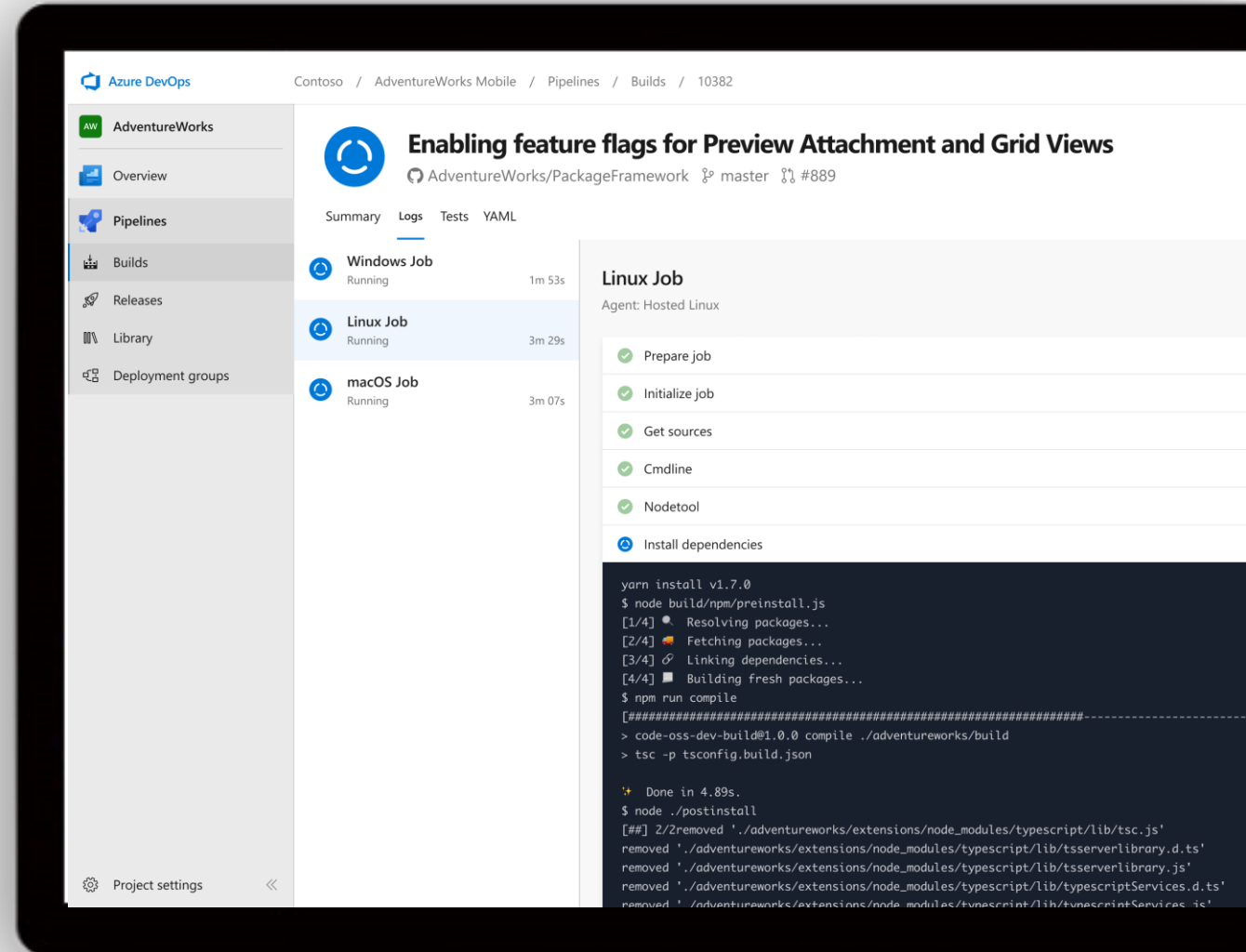


Containers and Kubernetes

Easily build and push images to container registries like Docker Hub and Azure Container Registry. Deploy containers to individual hosts or Kubernetes.

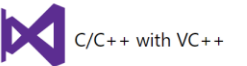
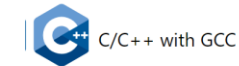
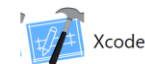
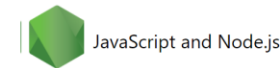


<https://azure.com/pipelines>



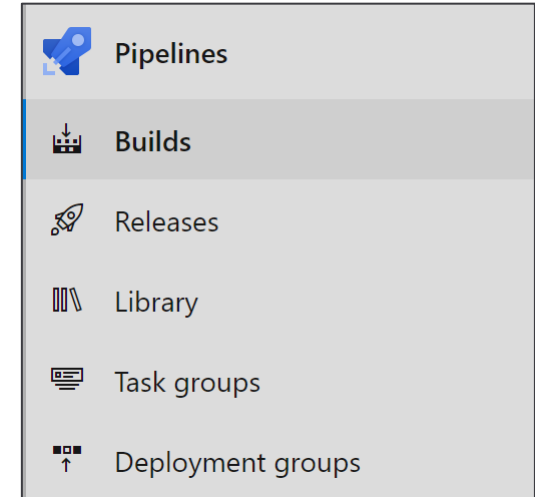
Running a Build with Azure DevOps Pipelines

- Supporting multiple languages
- **Prerequisites:**
 - A GitHub Account
 - An Azure DevOps Organization
 - Application Source Code
- Based on your source code, Azure DevOps Pipelines “recognizes” the capabilities
- The output of the Pipelines process is a “Azure-Pipelines.yml” file



Building a Pipeline for .NET Core applications

1. New Build Pipeline



Building a Pipeline for .NET Core applications

1. New Build Pipeline
2. Select Source Control environment

Connect


Select

Configure

Review


New pipeline

Where is your code?

 Azure Repos Git


YAML

Free private Git repositories, pull requests, and code search

 Bitbucket Cloud


YAML

Hosted by Atlassian

 GitHub


YAML

Home to the world's largest community of developers


 GitHub Enterprise Server

YAML

The self-hosted version of GitHub Enterprise

 Other Git

Any generic Git repository

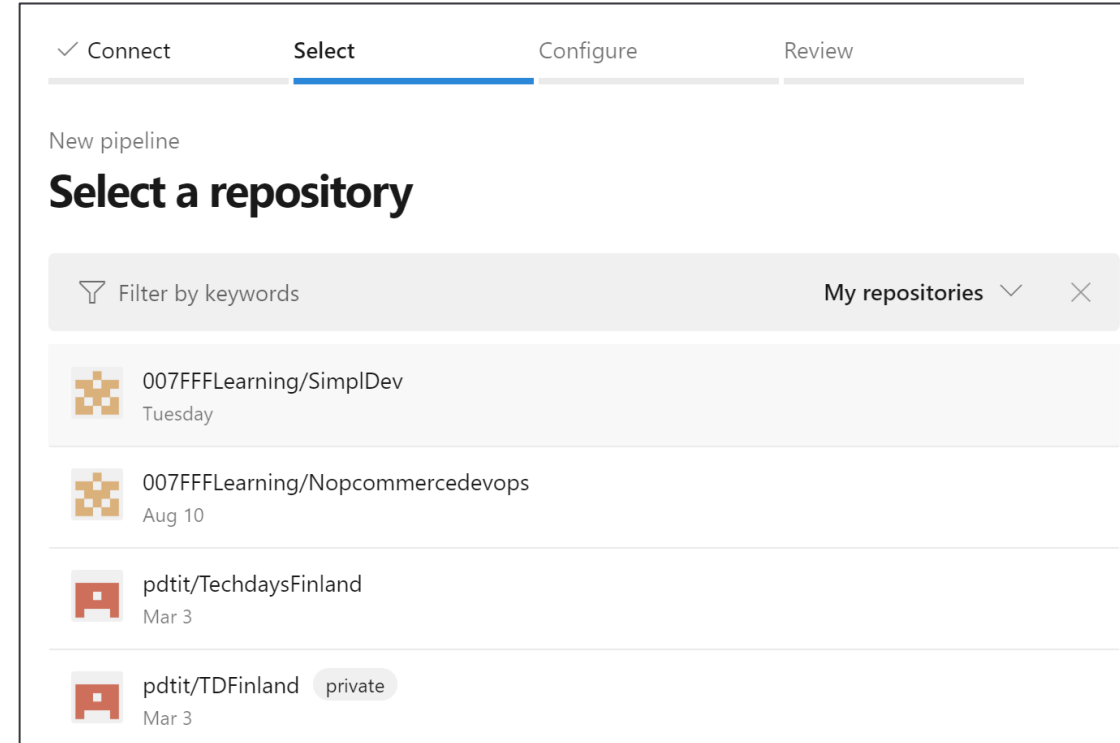
 Subversion

Centralized version control by Apache

[Use the classic editor](#) to create a pipeline without YAML.

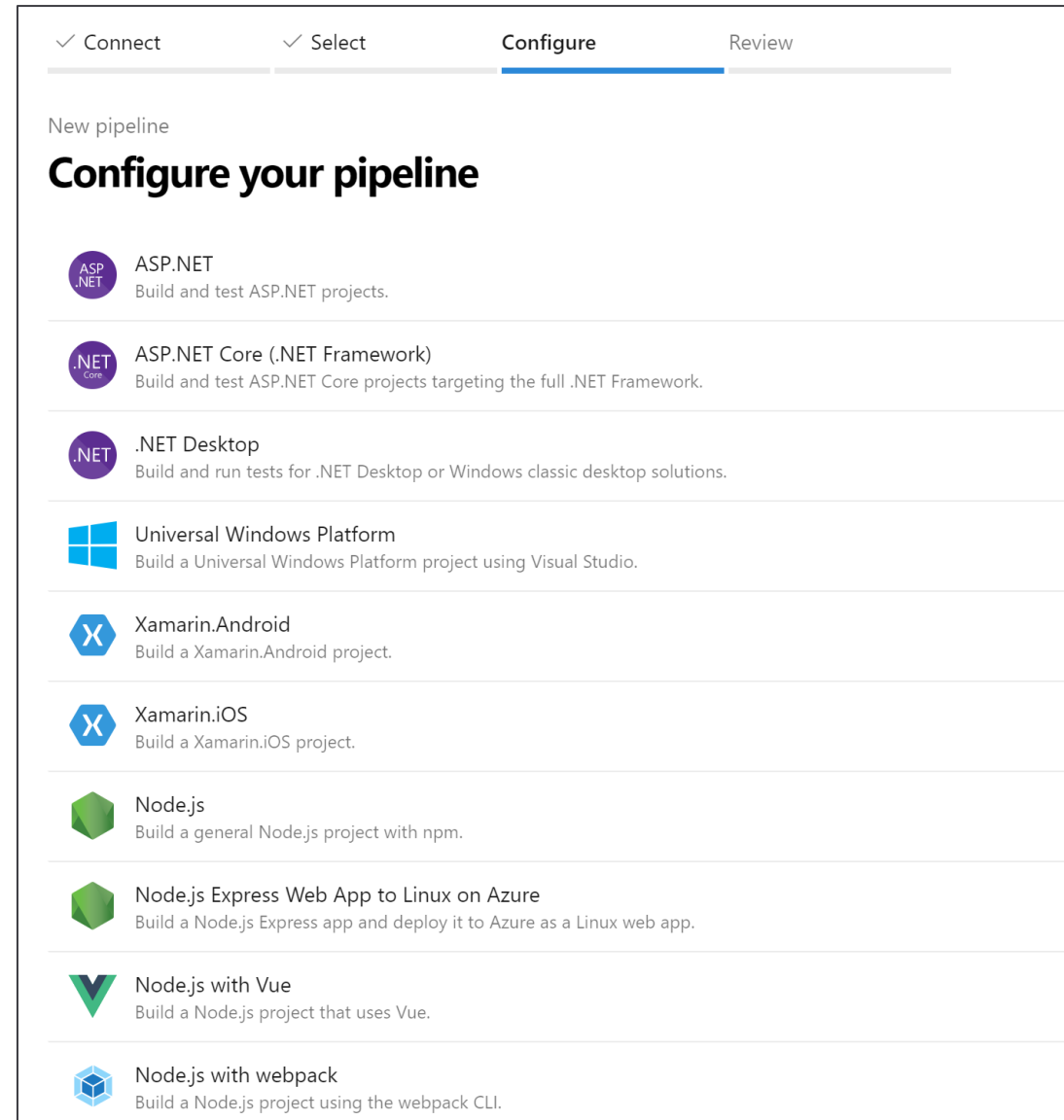
Building a Pipeline for .NET Core applications

1. New Build Pipeline
2. Select Source Control environment
3. Select Repo
- 4.



Building a Pipeline for .NET Core applications

1. New Build Pipeline
2. Select Source Control environment
3. Select Repo
4. Azure Pipelines analyzes the source code, and provides “options” for Build



Building a Pipeline for .NET Core applications

1. New Build Pipeline
2. Select Source Control environment
3. Select Repo
4. Azure Pipelines analyzes the source code, and provides “options” for Build
5. This results in a “Azure-pipelines.yml” file
6. Create and Run your Build

Linux

Pool: [Hosted Ubuntu 1604](#) · Agent: Hosted Agent

- ✓ Prepare job · succeeded
- ✓ Initialize job · succeeded
- ✓ Checkout · succeeded
- ✓ DotNetCoreInstaller · succeeded
- ✓ dotnet build · succeeded
- ✓ run tests · succeeded
- ✓ Publish Test Results **/*.trx · succeeded
- ✓ Post-job: Checkout · succeeded
- ✓ Finalize Job · succeeded

Running a Release with Azure DevOps Pipelines

- Supporting multiple languages
- Start from a template, or blank
- Based on a Pipeline build artifact, or other sources
- Single or multi-staged release scenarios

Featured



Azure App Service deployment

Deploy your application to Azure App Service. Choose from Web App on Windows, Linux, containers, Function Apps, or WebJobs.



Deploy a Java app to Azure App Service

Deploy a Java application to an Azure Web App.



Deploy a Node.js app to Azure App Service

Deploy a Node.js application to an Azure Web App.



Deploy a PHP app to Azure App Service and Azure Database for MySQL

Deploy a PHP application to an Azure Web App and database to Azure Database for MySQL.



Deploy a Python app to Azure App Service and Azure database for MySQL

Deploy a Python Django, Bottle, or Flask application to an Azure Web App and database to Azure Database for MySQL.



Deploy to a Kubernetes cluster

Deploy, configure, update your containerized applications to a Kubernetes cluster.



IIS website and SQL database deployment

Deployment Group: Deploy ASP.NET or ASP.NET Core web applications to an IIS Website and SQL database on physical or virtual machines (VM).

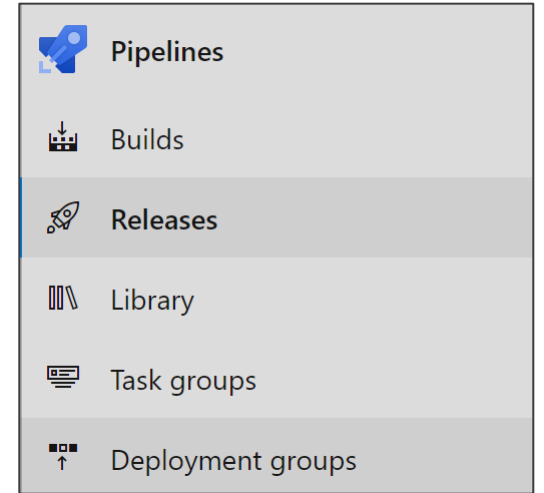
Others



Azure App Service deployment with continuous monitoring

Configuring a Release Pipeline for .NET Core applications

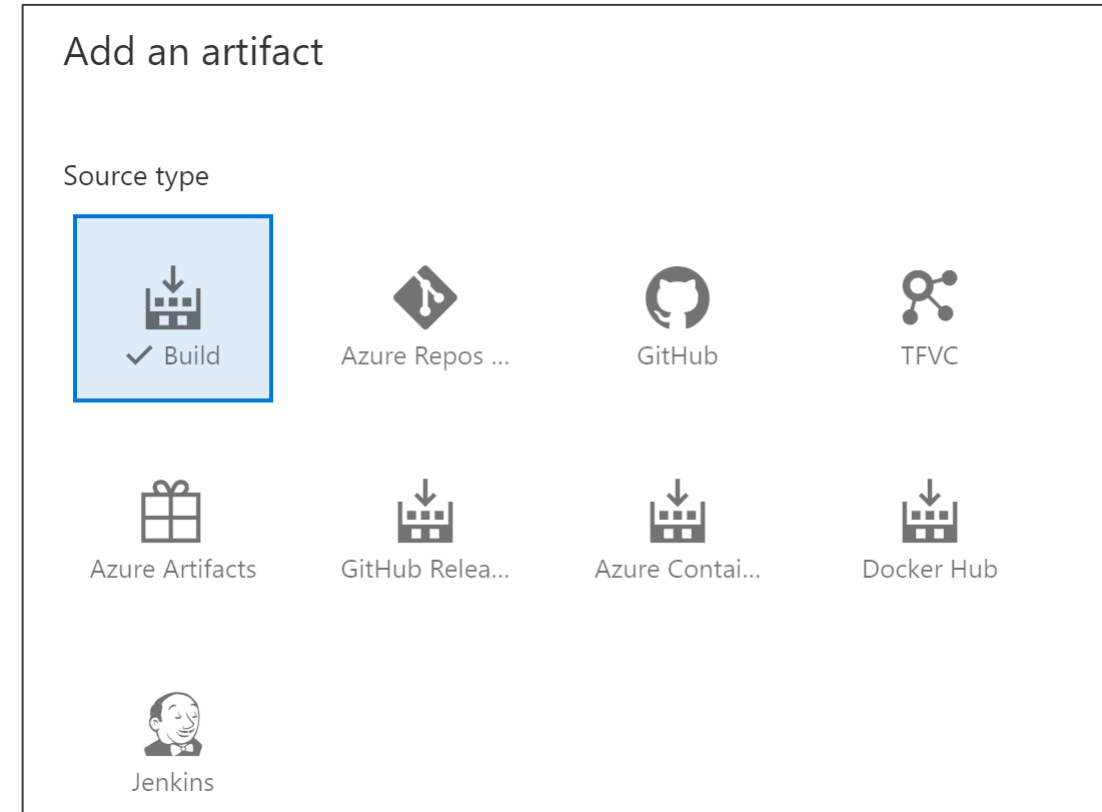
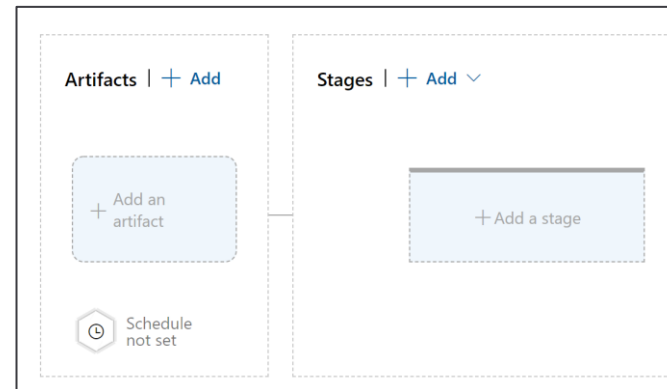
1. New Release Pipeline



Configuring a Release Pipeline for .NET Core applications

1. New Release Pipeline

2. Select Artifacts



Configuring a Release Pipeline for .NET Core applications

1. New Release Pipeline
2. Select Artifacts
3. Define a Stage Template

Featured



Azure App Service deployment

Deploy your application to Azure App Service. Choose from Web App on Windows, Linux, containers, Function Apps, or WebJobs.



Deploy a Java app to Azure App Service

Deploy a Java application to an Azure Web App.



Deploy a Node.js app to Azure App Service

Deploy a Node.js application to an Azure Web App.



Deploy a PHP app to Azure App Service and Azure Database for MySQL

Deploy a PHP application to an Azure Web App and database to Azure Database for MySQL.



Deploy a Python app to Azure App Service and Azure database for MySQL

Deploy a Python Django, Bottle, or Flask application to an Azure Web App and database to Azure Database for MySQL.



Deploy to a Kubernetes cluster

Deploy, configure, update your containerized applications to a Kubernetes cluster.



IIS website and SQL database deployment

Deployment Group: Deploy ASP.NET or ASP.NET Core web applications to an IIS Website and SQL database on physical or virtual machines (VM).

Others



Azure App Service deployment with continuous monitoring

Configuring a Release Pipeline for .NET Core applications


1. New Release Pipeline
2. Select Artifacts
3. Define a Stage Template
4. Complete Stage Template parameters

Stage name

Stage 2

Parameters ⓘ | [Unlink all](#)

Azure subscription * ⓘ | [Manage](#) [↗](#)


007FFFLearning Labs (0a407898-c077-442d-8e17-71420aa82426) ▼ 

ⓘ Scoped to subscription '007FFFLearning Labs'

App type ⓘ

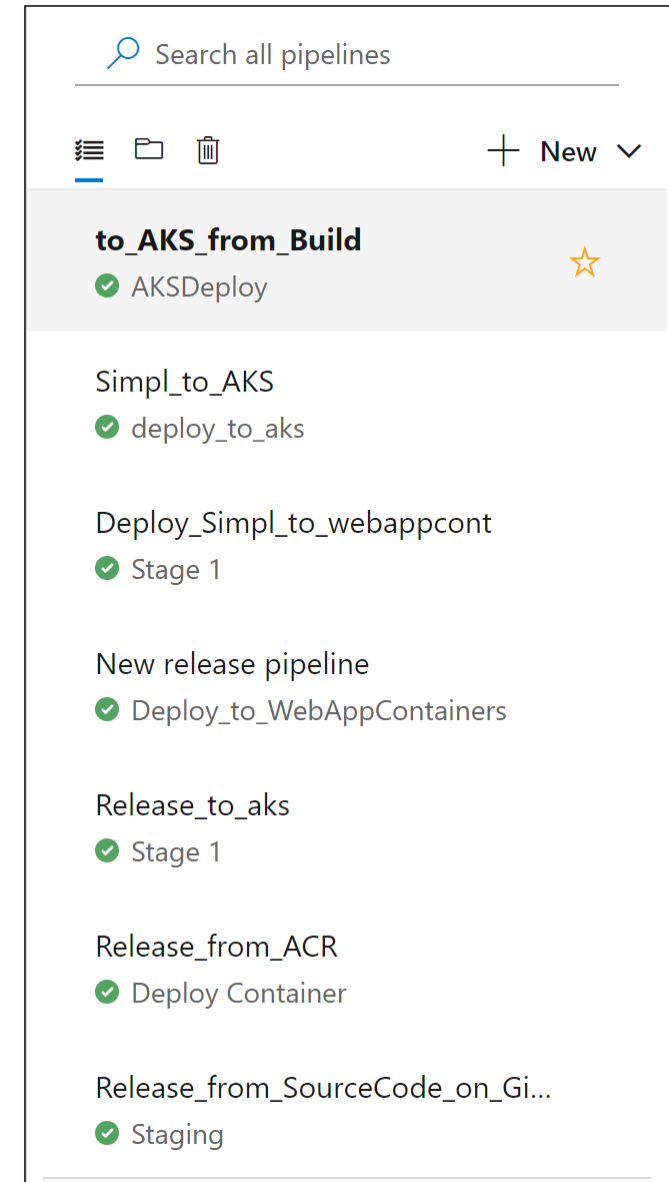
Web App for Containers (Linux) ▼

App service name * ⓘ

simplwebcontdevops ▼ 

Configuring a Release Pipeline for .NET Core applications

1. New Release Pipeline
2. Select Artifacts
3. Define a Stage Template
4. Complete Stage Template parameters
5. Save & Run



Monitoring a Release Pipeline for .NET Core applications

- Full details of each step in the Release process
- Succeeded / Failed
- Duration of the Pipeline

The screenshot displays the Azure DevOps Release Pipeline monitoring interface. At the top, the breadcrumb navigation shows 'Release_from_ACR > Release-2 > Deploy Container' with a green 'Succeeded' status badge. Below this is a horizontal menu with tabs: 'Pipeline', 'Tasks', 'Variables', 'Logs' (which is selected and underlined), 'Tests', 'Deploy', 'Cancel', 'Refresh', 'Download all logs', 'Edit', and a more options menu. The main content area is divided into two panels. The left panel, titled 'Deployment process', shows a list of steps with 'Run on agent' highlighted in light blue, indicating it is the current step. The right panel, titled 'Run on agent', provides details for this step, including the pool 'Hosted VS2017' and agent 'Hosted Agent'. It lists three sub-tasks, all of which are marked as 'succeeded': 'Initialize job', 'Azure Web App on Container Deploy: contnopwebapp', and 'Finalize Job'.

Release_from_ACR > Release-2 > Deploy Container ✓ Succeeded

← Pipeline Tasks Variables **Logs** Tests | ☁ Deploy ⛔ Cancel ↻ Refresh ⬇ Download all logs ✎ Edit ⋮

Deployment process
Succeeded

- ✓ Run on agent
Succeeded

Run on agent
Pool: Hosted VS2017 · Agent: Hosted Agent

- ✓ Initialize job · succeeded
- ✓ Azure Web App on Container Deploy: contnopwebapp · succeeded
- ✓ Finalize Job · succeeded

Monitoring a Release Pipeline for .NET Core applications

- Full details of each step in the Release process

```
1  [2019-08-11T18:16:38.0429690Z ##[section]Starting: Initialize job
2  2019-08-11
3  2019-08-11
4  2019-08-11
5  2019-08-11
6  2019-08-11
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

1  [2019-08-11T18:16:40.7798636Z ##[section]Starting: Azure Web App on Container Deploy: contnopwebapp
2  2019-08-11T18:16:41.0218887Z =====
3  2019-08-11T18:16:41.0219066Z Task      : Azure Web App for Containers
4  2019-08-11T18:16:41.0219314Z Description : Deploy containers to Azure App Service
5  2019-08-11T18:16:41.0219368Z Version    : 1.0.20
6  2019-08-11T18:16:41.0219430Z Author     : Microsoft Corporation
7  2019-08-11T18:16:41.0219492Z Help       : https://docs.microsoft.com/azure/devops/pipelines/tasks/deploy/azure-rm-web-app-containers
8  2019-08-11T18:16:41.0219597Z =====
9  2019-08-11T18:16:42.4716693Z Got service connection details for Azure App Service:'contnopwebapp'
10 2019-08-11T18:16:42.7336601Z Single-container Deployment to the webapp 'contnopwebapp' as only the image detail was sepcified.
11 2019-08-11T18:16:43.1923367Z Updating App Service Configuration settings. Data: {"appCommandLine":null,"linuxFxVersion":"DOCKER|nopacr1.azurecr.io/nopcon
12 2019-08-11T18:16:45.4827837Z Updated App Service Configuration settings
13 2019-
14 2019-
15 2019-
16 2019-
17 2019-
18 2019-
19 2019-

1  [2019-08-11T18:16:53.6649998Z ##[section]Starting: Finalize Job
2  2019-08-11T18:16:53.6694180Z Start cleaning up orphan processes.
3  2019-08-11T18:16:53.6754539Z ##[section]Finishing: Finalize Job
4
```

Demo

Github – DevOps Academy

<https://github.com/msdevopsacademy/>

Repo:

03-create-a-build-pipeline-with-azure-pipelines

