

Build a CI/CD Pipeline

Step 1. Create Azure DevOps Project

<https://portal.azure.com>

The screenshot displays the Azure portal interface. On the left, the navigation pane includes a 'Create a resource' button (highlighted with a red box) and a list of services. The main area shows the 'New' page with a search bar containing 'DevOps Project' (highlighted with a red box). Below the search bar, the 'Azure Marketplace' section lists various services. The 'Popular' section highlights several options, including '.NET' (highlighted with a red box), which is described as 'New Web App using ASP.NET or ASP.NET Core, or a new IoT app'. Other options include Node.js, PHP, Java, Static Website, Python, Ruby, Go, C, and 'Bring your own code'. At the bottom right, the 'Next' button is highlighted with a red box.

Home > New

Create a resource

Home

Dashboard

All services

FAVORITES

All resources

Resource groups

App Services

Function Apps

SQL servers

SQL databases

Azure Cosmos DB

Virtual machines

Load balancers

New

DevOps Project

Azure Marketplace See all Popular

Get started

Recently created

Compute

Networking

Storage

Web

Mobile

Containers

Databases

Analytics

Windows Server 2016 Datacenter Quickstart tutorial

Ubuntu Server 18.04 LTS Learn more

Web App Quickstart tutorial

SQL Database Quickstart tutorial

Function App

Start fresh with a new application

.NET New Web App using ASP.NET or ASP.NET Core, or a new IoT app

Node.js New Web app using Node.js, Express.js or Sails.js, or a new IoT app

PHP New Web app using Laravel or Symfony

Java New Web App using Spring or JSF, or a new IoT app

Static Website New static website using HTML, CSS, and JavaScript

Python New Web App using Bottle, Django, or Flask, or a new IoT app

Ruby New Web App using Ruby on Rails

Go New Web App using Go

C New IoT app using C

or start with your application

Bring your own code Deploy your existing application to Azure

Previous Next

Step 2. Application Framework & Host

The screenshot shows the 'Choose an application framework' step in the Azure portal. At the top, a progress bar indicates four steps: Runtime (checked), Framework (active), Service, and Create. Below the progress bar, the title 'Choose an application framework' is displayed. Three framework options are shown in cards: 'ASP.NET' (highlighted with a red box), 'ASP.NET Core', and 'Simple IoT (Preview)'. The 'ASP.NET' card describes it as an 'Open source web framework for building modern web apps and services'. Below the framework cards, there is a section titled 'Add a database' with a toggle switch. Under this section, a 'SQL Database' option is shown, described as 'A relational database-as-a service using the Microsoft SQL Server Engine'. At the bottom of the screen, there are 'Previous' and 'Next' buttons, with the 'Next' button highlighted by a red box.

The screenshot shows the 'Select an Azure service to deploy the application' step in the Azure portal. At the top, a progress bar indicates four steps: Runtime (checked), Framework (checked), Service (active), and Create. Below the progress bar, the title 'Select an Azure service to deploy the application' is displayed. Three service options are shown in cards: 'Windows Web App' (highlighted with a red box), 'Function App', and 'Virtual machine'. The 'Windows Web App' card describes it as a 'Fully managed compute platform on Windows for web applications and websites.' The 'Function App' card describes it as 'A serverless compute service to run code on-demand without managing infrastructure.' The 'Virtual machine' card describes it as a 'Windows virtual machine to run your app'. At the bottom of the screen, there is a message: 'Don't see a service you're looking for? We're continuously adding support for more Azure services and app frameworks. [Learn more](#)'. Below this message, there are 'Previous' and 'Next' buttons, with the 'Next' button highlighted by a red box.

Step 3. Complete Project Settings

✓

✓

✓

4

RuntimeFrameworkServiceCreate

Almost there!

Ready to deploy ASP.NET app to Azure Windows Web App.

* Project name

Hello World ✓

* Azure DevOps Organization

wlliamwang-azbootcamp ✓
https://dev.azure.com/wlliamwang-azbootcamp

* Subscription ⓘ

Pay-As-You-Go ▼

Web app name ⓘ

helloworldbootcamp ✓
.azurewebsites.net

Location ⓘ

Australia East ▼

Pricing tier: S1 Standard (1 Core, 1.75 GB RAM)

Additional settings

By continuing, you agree to the [Terms of Service](#) and the [Privacy Statement](#). The new app code is published under the MIT license.

Previous

Done

Additional settings

DevOps project

* Create new Azure DevOps organization

YesNo

Your project will be hosted in selected DevOps Services organization: william-wang

Web App on Windows

Resource group ⓘ

aklbootcamphelloworld-william-rg ✓

Pricing tier ⓘ

S1 Standard (1 Core, 1.75 GB RAM) ▼

Application Insights Location ⓘ

Australia East ▼

Step 4. Deployment Complete

✓ Your deployment is complete

Check the status of your deployment, manage resources, or troubleshoot deployment issues. Pin this page to your dashboard to easily find it next time.



Deployment name: Deploy_DevOps_Project_HelloWorld
Subscription: [Visual Studio Enterprise – MPN](#)
Resource group: VstsRG-william-wang-52b3

DEPLOYMENT DETAILS [\(Download\)](#)

Start time: 19/04/2019, 5:45:11 pm

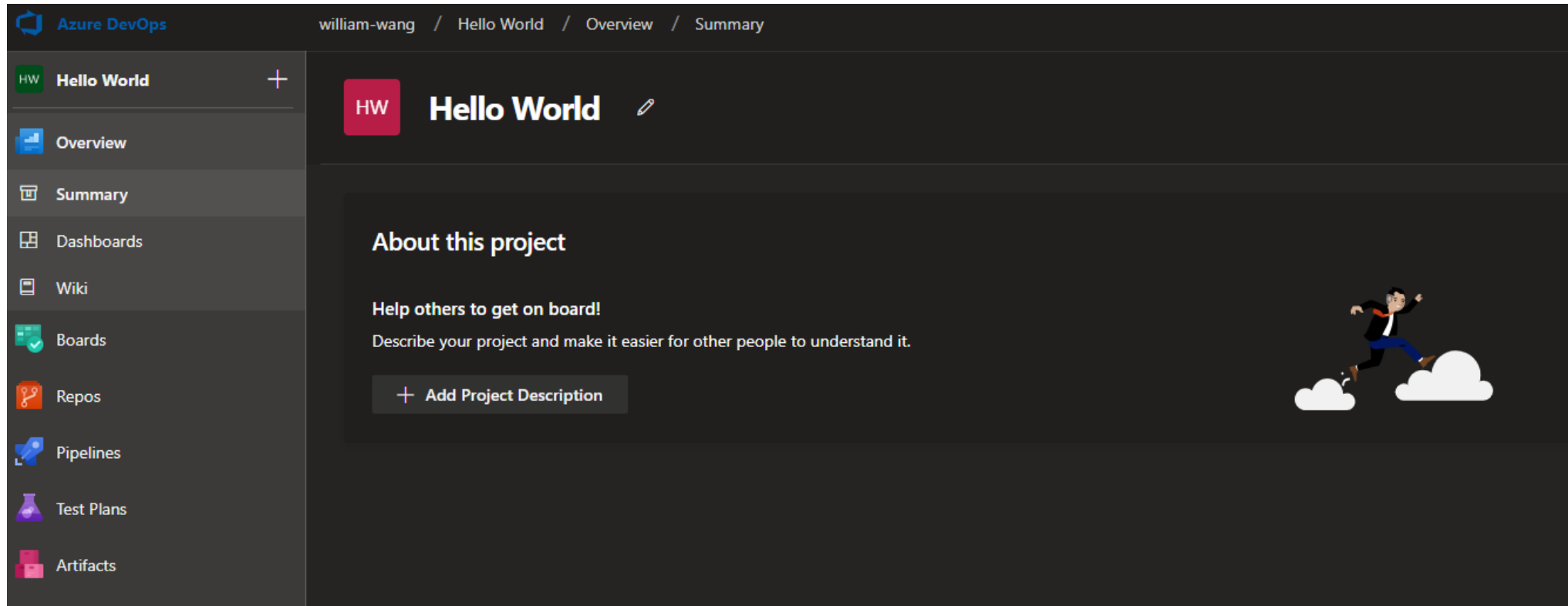
Duration: 1 minute 3 seconds

Correlation ID: 38ff8183-1731-4ae3-b2b3-365fe6a8ad3f

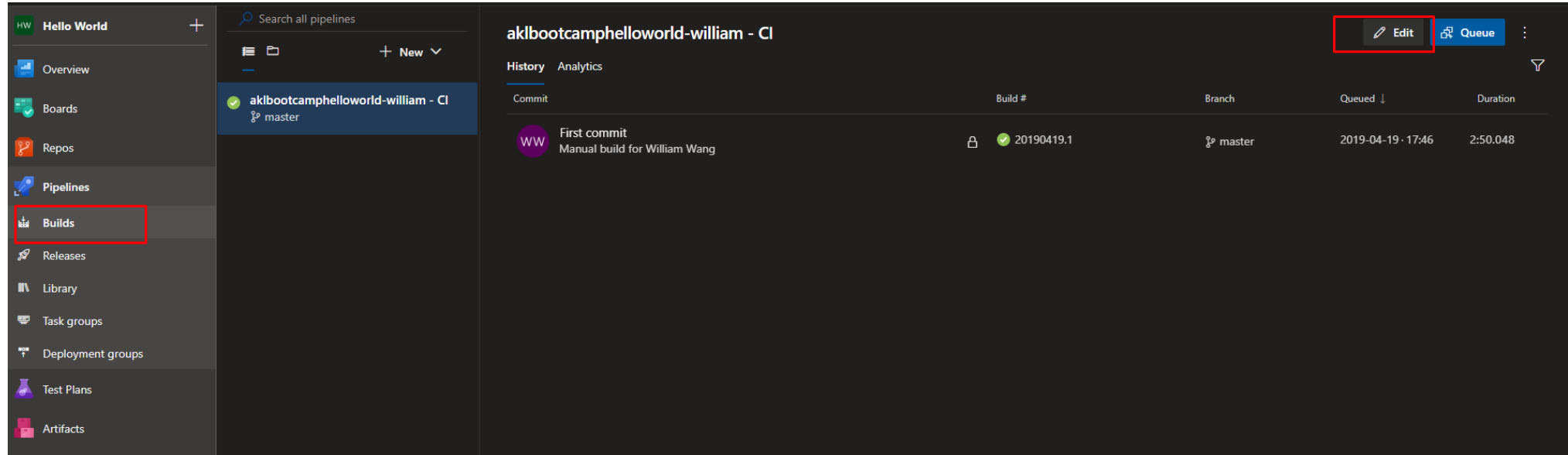
RESOURCE	TYPE	STATUS	OPERATION DETAILS
✓ william-wang/Hello World	microsoft.visualstudio/account/project	OK	Operation details
✓ william-wang	microsoft.visualstudio/account	OK	Operation details

Step 5. Azure DevOps Portal

<https://dev.azure.com/william-wang>

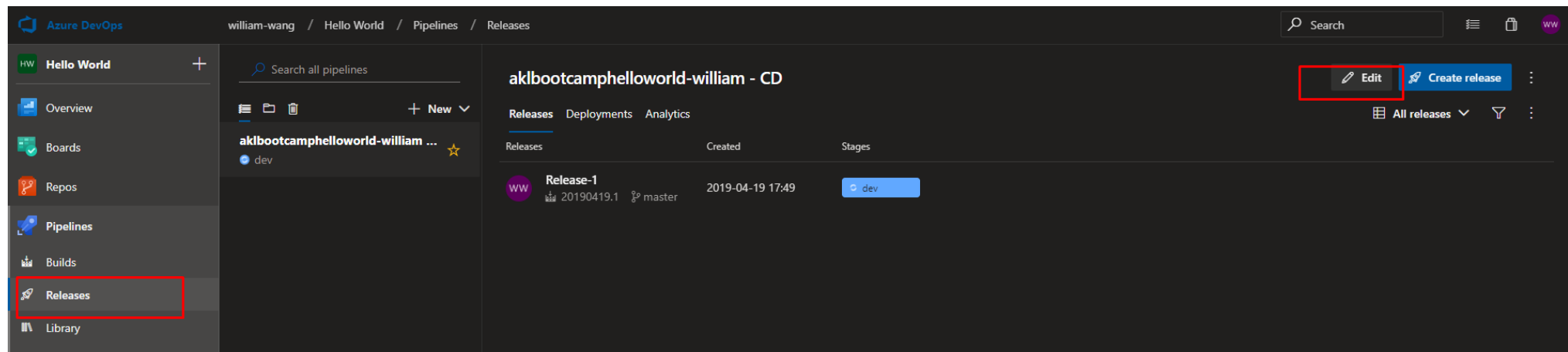


Step 6. CI/CD Pipeline



This screenshot shows the 'Builds' view for a CI pipeline named 'aklbootcamphellworld-william - CI'. The left sidebar has 'Builds' highlighted. The main panel shows a table of build history with one entry: 'First commit' (Manual build for William Wang) with build number 20190419.1, branch 'master', and a duration of 2:50.048. The 'Edit' button is highlighted with a red box.

Commit	Build #	Branch	Queued ↓	Duration
WW First commit Manual build for William Wang	20190419.1	master	2019-04-19 17:46	2:50.048



This screenshot shows the 'Releases' view for a CD pipeline named 'aklbootcamphellworld-william - CD'. The left sidebar has 'Releases' highlighted. The main panel shows a table of release history with one entry: 'Release-1' (20190419.1) on the 'master' branch, created on 2019-04-19 17:49, with a 'dev' environment. The 'Edit' button is highlighted with a red box.

Releases	Created	Stages
WW Release-1 20190419.1 master	2019-04-19 17:49	dev

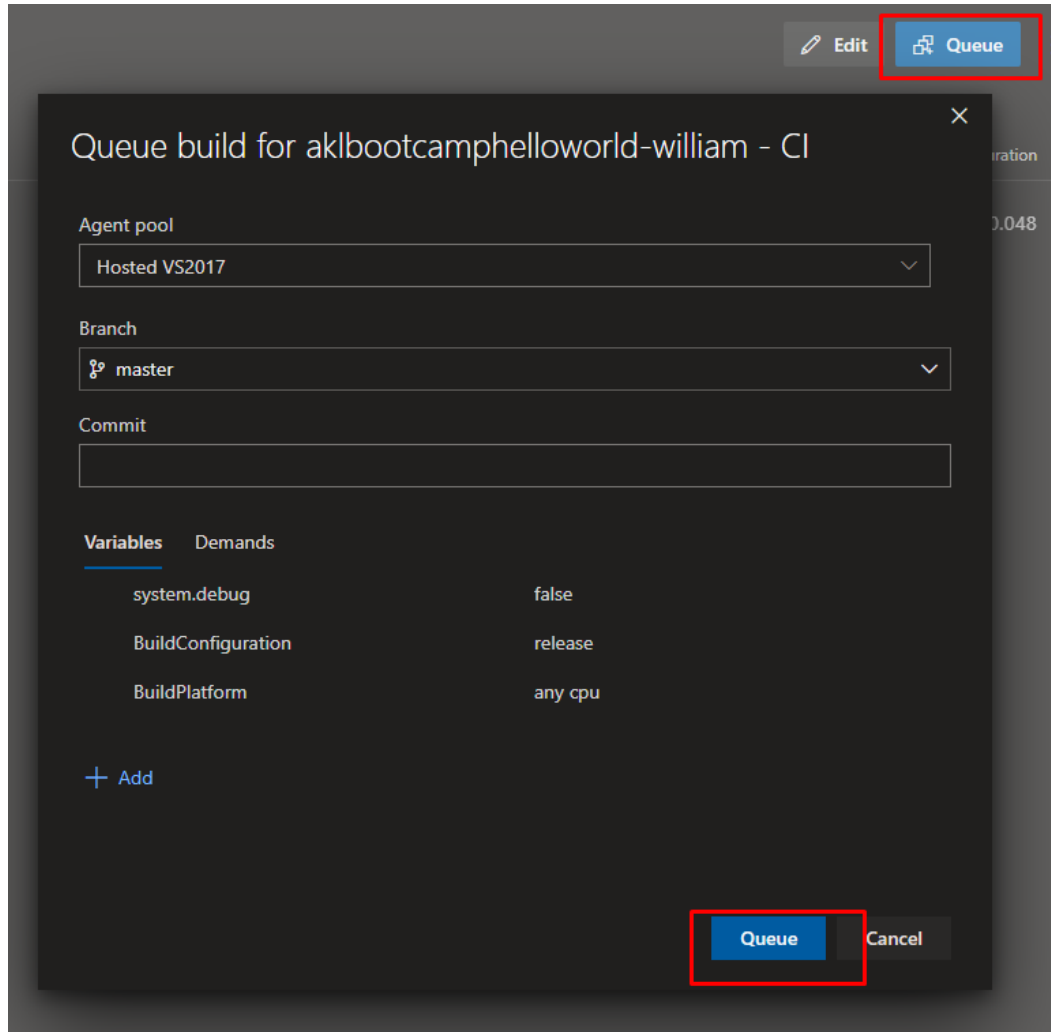
Step 7. Build & Release

The screenshot shows the 'Queue build' dialog box in Azure DevOps. The dialog is titled 'Queue build for aklbootcamphellworld-william - CI'. It has a dark theme. At the top right, there are buttons for 'Edit' and 'Queue', with the 'Queue' button highlighted by a red rectangle. The dialog contains the following fields and sections:

- Agent pool:** A dropdown menu showing 'Hosted VS2017'.
- Branch:** A dropdown menu showing 'master'.
- Commit:** An empty text input field.
- Variables and Demands:** A table with two columns: 'Variables' and 'Demands'.

Variables	Demands
system.debug	false
BuildConfiguration	release
BuildPlatform	any cpu
- + Add:** A button to add new variables or demands.
- Buttons:** At the bottom right, there are 'Queue' and 'Cancel' buttons, with the 'Queue' button highlighted by a red rectangle.

Step 8. Build & Release



The screenshot shows the 'Queue build' dialog box in Azure DevOps. The title bar reads 'Queue build for aklbootcamphello-world-william - CI'. The 'Agent pool' is set to 'Hosted VS2017'. The 'Branch' is set to 'master'. The 'Commit' field is empty. Below these fields, there are tabs for 'Variables' and 'Demands'. Under 'Variables', three items are listed: 'system.debug' with value 'false', 'BuildConfiguration' with value 'release', and 'BuildPlatform' with value 'any cpu'. At the bottom right, there are 'Queue' and 'Cancel' buttons, both highlighted with red rectangles.

Queue build for aklbootcamphello-world-william - CI

Agent pool
Hosted VS2017

Branch
master

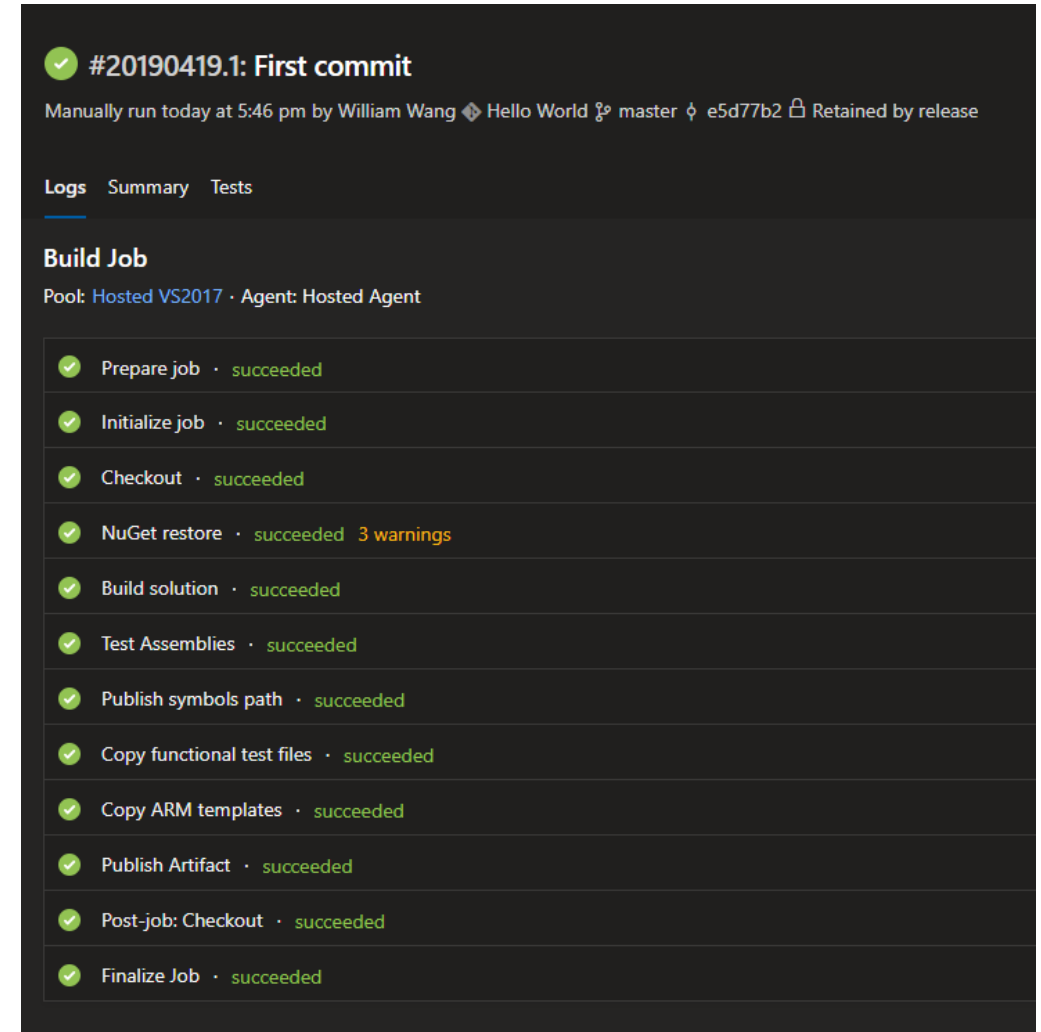
Commit

Variables Demands

system.debug	false
BuildConfiguration	release
BuildPlatform	any cpu

+ Add

Queue Cancel



The screenshot shows the build job summary page for a successful build. The job ID is '#20190419.1: First commit'. It was manually run today at 5:46 pm by William Wang. The build job details show a list of steps, all of which succeeded. The steps are: Prepare job, Initialize job, Checkout, NuGet restore (with 3 warnings), Build solution, Test Assemblies, Publish symbols path, Copy functional test files, Copy ARM templates, Publish Artifact, Post-job: Checkout, and Finalize Job.

✓ #20190419.1: First commit

Manually run today at 5:46 pm by William Wang Hello World master e5d77b2 Retained by release

Logs Summary Tests

Build Job

Pool: Hosted VS2017 Agent: Hosted Agent

- ✓ Prepare job · succeeded
- ✓ Initialize job · succeeded
- ✓ Checkout · succeeded
- ✓ NuGet restore · succeeded 3 warnings
- ✓ Build solution · succeeded
- ✓ Test Assemblies · succeeded
- ✓ Publish symbols path · succeeded
- ✓ Copy functional test files · succeeded
- ✓ Copy ARM templates · succeeded
- ✓ Publish Artifact · succeeded
- ✓ Post-job: Checkout · succeeded
- ✓ Finalize Job · succeeded