





# Azure DevOps 를 사용해서 아주 쉽고 재미나게

## 가상 머신에 코드를 배포해보기!





# 주민규

Azure MVP, MCT

knighp@gmail.com





**CLOUDMATE**  
Azure Expert Group

**CLOUDZEN**



**Microsoft**

Global Azure Virtual 2020 는 클라우드메이트, 클라우드Zen, 마이크로소프트의 후원을 받아 진행되고 있습니다.

#gav2020kr

// Resource Group 생성

```
$ az group create -n TestRG -l eastus
```

// VM 생성

```
$ az vm create -g TestRG -n vm01 --admin-username mingyu --image UbuntuLTS --ssh-key-value ~/.ssh/abc_rsa.pub
```

// VM 의 22/80/443 포트 열기

```
$ az vm open-port -g TestRG -n vm01 --port 22 --priority 1010
$ az vm open-port -g TestRG -n vm01 --port 80 --priority 1020
$ az vm open-port -g TestRG -n vm01 --port 443 --priority 1030
```

// VM 접속

```
$ ssh -i ~/.ssh/abc_rsa myID@123.123.123.123
```

## Nginx 및 Node.js 설치

```
$ sudo apt update  
$ sudo apt install nginx  
$ sudo apt update  
$ sudo curl -sL https://deb.nodesource.com/setup_12.x | sudo -E bash -  
$ sudo apt install -y nodejs  
$ sudo apt update  
$ sudo npm install -g --save express-generator e
```

// Nginx 를 Reverse Proxy로 설정

\$ sudo nano /etc/nginx/sites-available/default

// try\_files \$uri \$uri/ =404; 주석 처리 후, 아래 입력  
proxy\_pass http://127.0.0.1:3000/;

\$ sudo service nginx restart



**Pipelines**

- Pipelines
- Environments
- Releases
- Library
- Task groups
- Deployment groups

### Deployment groups > GAV2020Group

Details Targets Save Share Security Help

Deployment group name: GAV2020Group

Type of target to register: Linux

Description:

Deployment pool: GAV2020-GAV2020Group

Registration script:

```
mkdir azagent;cd azagent;curl -fksL -o vstsagent.tar.gz https://vstsagentpackage.azureedge.net/agent/2.166.3/vsts-agent-linux-x64-2.166.3.tar.gz;tar -zxf vstsagent.tar.gz; if [ -x "$(command -v systemctl)" ]; then ./config.sh --deploymentgroup --deploymentgroupname "GAV2020Group" --acceptteeeula --agent $HOSTNAME --url https://dev.azure.com/codingstar/ --work _work --projectname 'GAV2020' --runasservice; sudo ./svc.sh install; sudo ./svc.sh start; else ./config.sh --deploymentgroup --deploymentgroupname "GAV2020Group" --acceptteeeula --agent $HOSTNAME --url https://dev.azure.com/codingstar/ --work _work --projectname 'GAV2020'; ./run.sh; fi
```

Use a personal access token in the script for authentication

**Copy script to the clipboard**

Run from an administrator command prompt

### Deployment groups > GAV2020Group

Details Targets Save Share Security Help

Healthy (1)

vm01

Summary

No deployments yet

Tags

web X



codingstar / MyProj / Repos / Files / MyProj

New Repos pull request experience: Try out the new modern, fast, and mobile-friendly pull request experience.

MyProj is empty. Add some code!

Clone to your computer

HTTPS

SSH

[https://codingstar@dev.azure.com/codingstar/MyProj/\\_git/MyPrc](https://codingstar@dev.azure.com/codingstar/MyProj/_git/MyPrc)

Generate Git Credentials

Having problems authenticating in Git? Be sure to generate Git credentials.

Push an existing repository from command line

HTTPS

SSH

```
git remote add origin  
https://codingstar@dev.azure.com/codings...
```

New Repos pull request experience: Try out the new modern, fast, and mobile-friendly pull request experience within Repos. [Learn more](#) [Not now](#) [Try it!](#)

GAV2020

.gitignore  
azure-pipelines.yml  
index.js  
package-lock.json  
package.json  
vm01.json

Git 을 사용하여 Local 에서 Commit 하고 Push 한다!

master / Type to find a file or folder...

Files

Contents History

Name ↑	Last change	Commits
.gitignore	3월 17일	c0515136 first commit MinGy...
azure-pipelines.yml	Yesterday	36a863cf Third commit MinG...
index.js	27m ago	0b637647 Updated index.js ...
package-lock.json	3월 17일	c0515136 first commit MinGy...
package.json	3월 19일	81887b39 test to start MinGy...
vm01.json	3월 17일	c0515136 first commit MinGy...



codingstar / GAV2020 / Pipelines

Connect

Select

Configure

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.



codingstar / GAV2020 / Pipelines

✓ Connect

Select

Configure

Review

New pipeline

## Select a repository

Filter by keywords

GAV2020



New pipeline

## Review your pipeline YAML

◆ GAV2020 / azure-pipelines-1.yml \*

```
1 # Node.js
2 # Build a general Node.js project with npm.
3 # Add steps that analyze code, save build artifacts, deploy, and more:
4 # https://docs.microsoft.com/azure/devops/pipelines/languages/javascript
5
6 trigger:
7 - master
8
9 pool:
10   vmImage: 'ubuntu-latest'
11
12 steps:
```

# Build Pipeline

<https://github.com/microsoft/azure-pipelines-tasks/tree/master/Tasks>



주민규

관리자 · 4월 12일 오후 7:09

#Azure #DevOps 에서 #CI #CD 에 필요한 작업들이 정리되는 가장 중요한 yaml 파일 작성이 웬지 쉽지 않은데요. ^^". 생략되는 것들이 있곤해서, 여러 문서를 비교해봐도 구조가 눈에 안 들어오기도...

azure-pipelines.yml 파일 작성에 대한 공부를 처음 하실 때 살짝 참고로 하시면 좋을 것 같습니다. 이 글에서는 3개의 단계에 대해서만 아주 간단히 보겠습니다. 구조적으로, **Stages > Jobs > Steps** 3개의 단계로 나누어 집니다. steps 안에는 일련의 실제 작업들이 정의 됩니다. 그리고 Azure DevOps 에서는 스텝별로 하나의 프로세서가 기동되어 실행됩니다. 그렇기 때문에 프로세스 환경변수는 스텝간에 공유되지 않습니다. steps 내에는 task, script, bash, pwsh or PowerShell, upload 등의 작업들이 들어갈 수 있습니다.

Stages 와 Jobs 는 잘 생략되기도 합니다. 예를 들면,

**steps:**

- script: |

```
npm install
```

```
npm run build --if-present
```

위와 같이 Stages 와 Jobs 가 생략될 수도 있는 거죠. 정석대로 써보자면, 아래와 같이 쓰시면 됩니다.

**stages:**

- stage: Build

**jobs:**

- job: Build

**steps:**

- script: |

```
npm install
```

```
npm run build --if-present
```



주민규님이 링크를 공유했습니다.

관리자 · 4월 12일 오후 8:40

이번에는 #Azure #DevOps 에서 #stages 의 실행 순서에 대해서 알아보겠습니다. 아래 내용은 참고 링크에 설명이 더 자세히 나옵니다.

**stages:**

- **stage: stageA**

**jobs:**

- **job:**

- **stage: stageB**

**jobs:**

- **job:**

위와 같이 작성하면, stageA 가 실행된 이후에 stageB 가 실행되게 됩니다.

**stages:**

- **stage: Test**

- **stage: stageA**

**dependsOn: Test**

- **stage: stageB**

**dependsOn: Test**

- **stage: stageC**

**dependsOn:**

- **stageA**

- **stageB**

...  
...



위와 같이 작성하면, stageA 와 stageB 는 Test stage 가 실행된 후에 병렬로 동시에 실행되며, stageC 는 stageA 과 stageB 가 모두 실행된 이후에 실행되게 됩니다.

**stages:**

- **stage: stageA**

**jobs:**

- **job:**

- **stage: stageB**

**dependsOn: []**

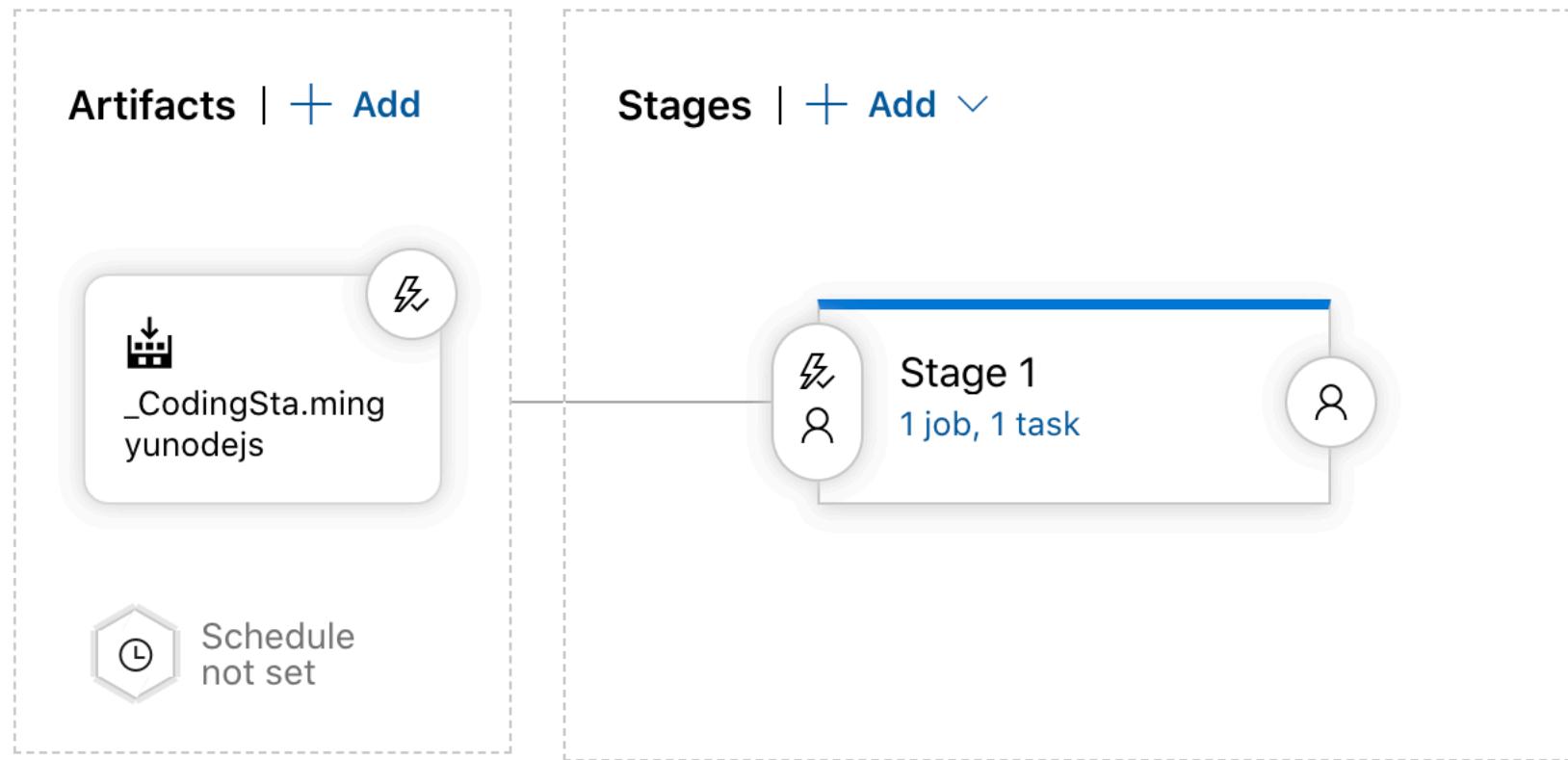
**jobs:**

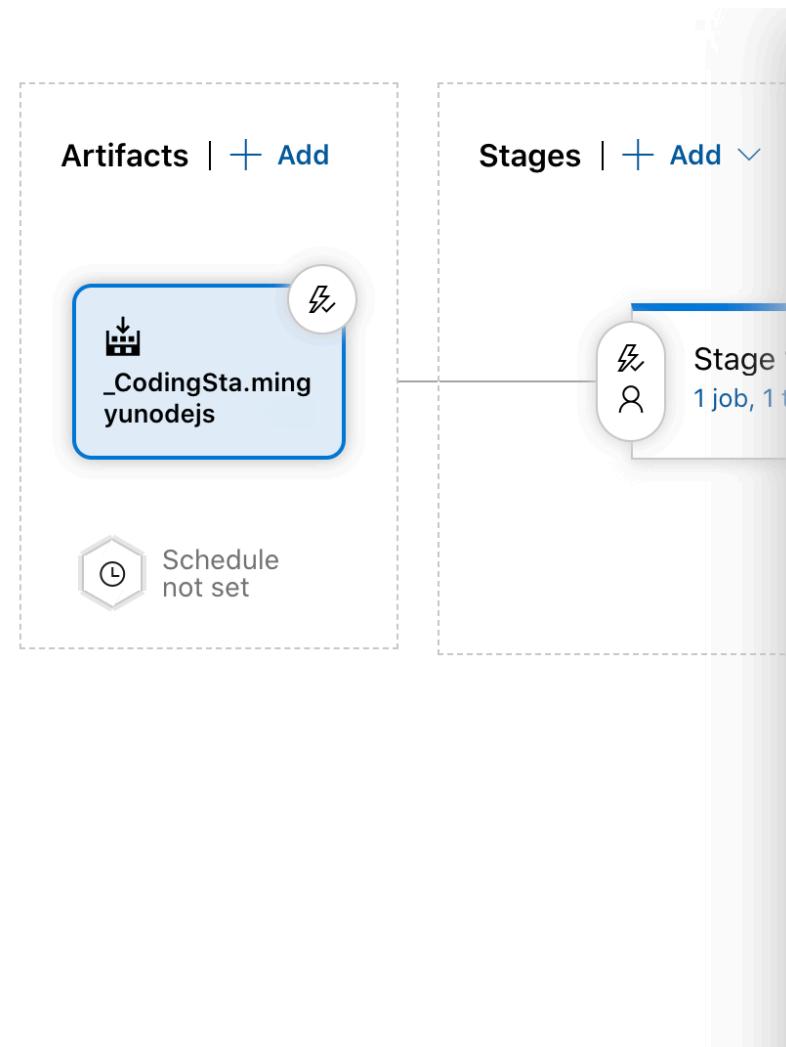
- **job:**

위와 같이 dependsOn: [] 를 넣어서 작성하면, 두 개의 stage들이 병렬로 동시에 실행되게 됩니다.

참고링크 : <https://docs.microsoft.com/en-us/azure/devops/pipelines/process/stages?view=azure-devops&tabs=yaml>

Azure DevOps 를 사용해서 아주 쉽고 재미나게 가상 머신에 코드를 배포해보기!





### Artifact

Build - \_CodingSta.mingyunodejs

Delete ...

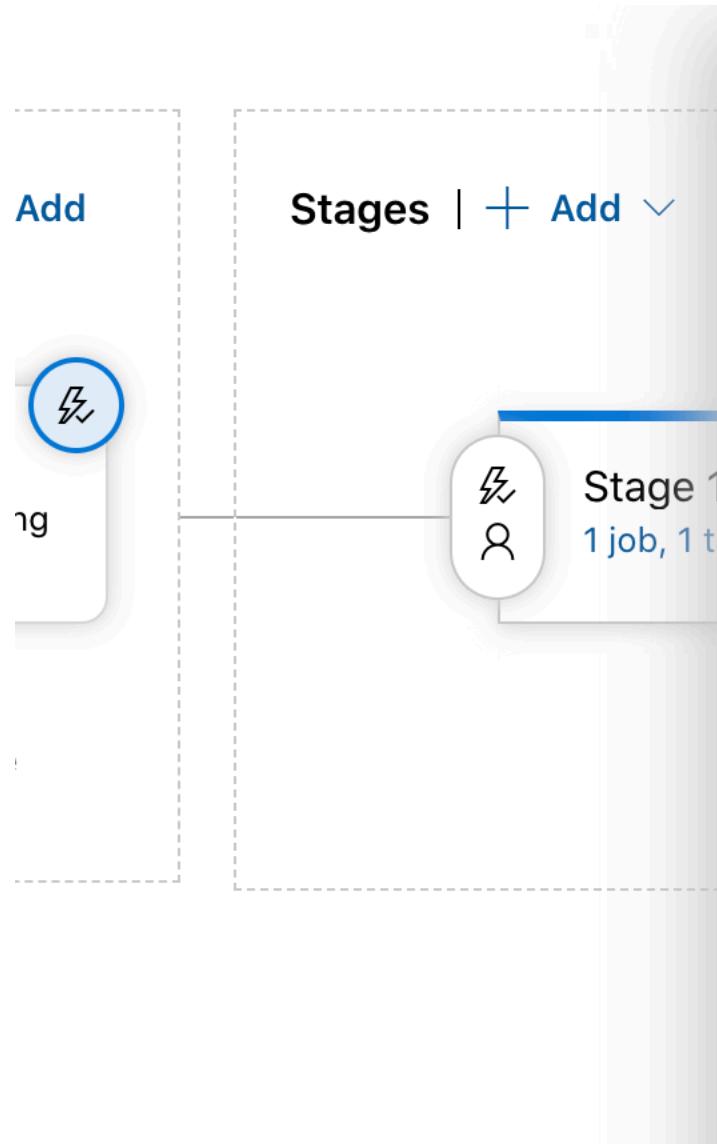
Project \* ⓘ  
NodePrj

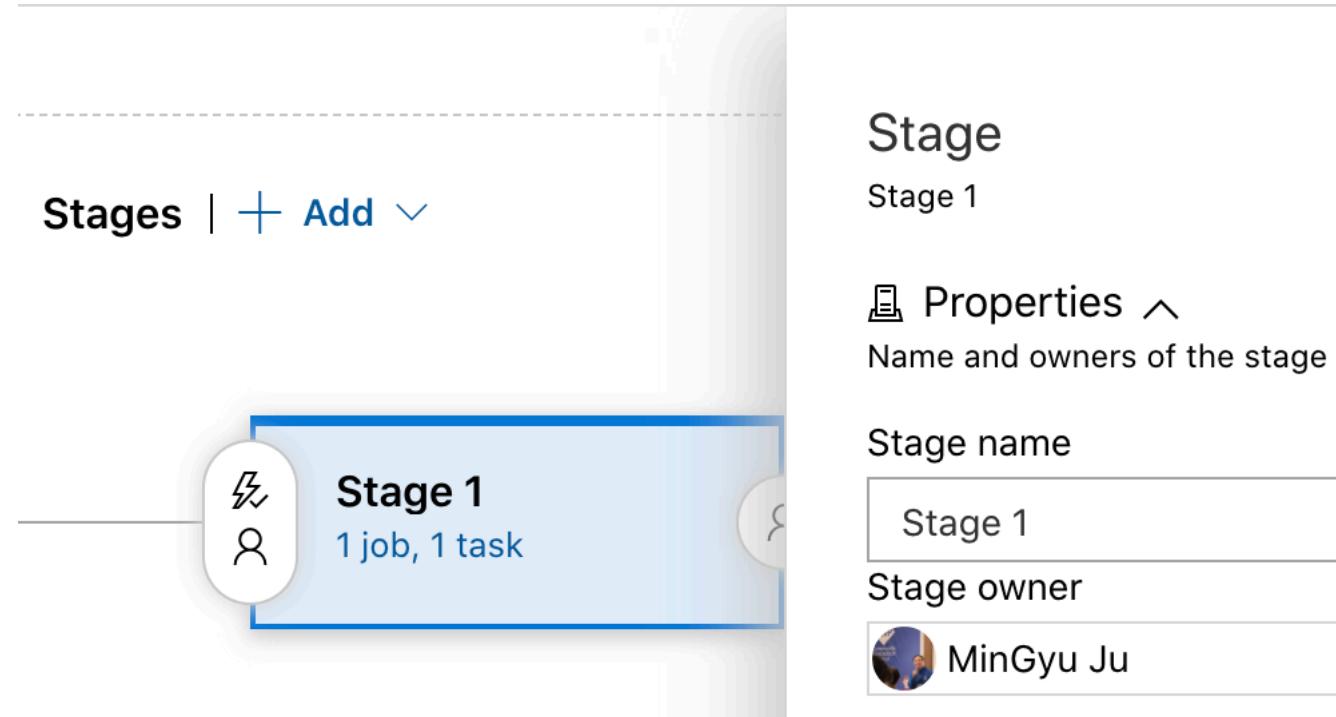
Source (build pipeline) \* ⓘ  
CodingSta.mingyunodejs

Default version \* ⓘ  
Latest

Source alias \* ⓘ  
\_CodingSta.mingyunodejs

ⓘ The artifacts published by each version will be available for deployment in release pipelines. The latest successful build of **CodingSta.mingyunodejs** published the following artifacts: **drop**.





The screenshot shows the Azure DevOps Release pipeline configuration. On the left, there's a list of stages with "Stage 1" selected. The stage details pane on the right shows the following information:

- Stage**: Stage 1
- Properties**: Name and owners of the stage
- Stage name**: Stage 1
- Stage owner**: MinGyu Ju (with a profile picture)

[All pipelines](#) > [↑ New release pipeline](#)[Pipeline](#) [Tasks](#) [Variables](#) [Retention](#) [Options](#) [History](#)**Stage 1**

Deployment process

...

Stage name

Stage 1

**Deployment group job** Run on deployment group

+

**>\_ Command Line Script**

Command line



## All pipelines &gt; New release pipeline

Save Create release ...

Pipeline Tasks Variables Retention Options History

## Stage 1

Deployment process

...

## Deployment group job

Run on deployment group

+



## Command Line Script

Command line

## Deployment group job

Remove

Display name \*

Deployment group job

## Deployment targets

Deployment group \*

GR01



Required tags

web



1 matching targets in GR01 deployment group

Targets to deploy to in parallel

 Multiple    One target at a time

Maximum number of targets in parallel

 100% targets (1)

Timeout \*

0

Job cancel timeout \*

1



## Artifact download ^



\_CodingSta.mingyunodejs Latest

▼ Selected all artifacts

## Additional options ^

Allow scripts to access the OAuth token ⓘ

Run this job ⓘ

Only when all previous jobs have succeeded





## All pipelines &gt; New release pipeline

Save Create release ...

Pipeline Tasks Variables Retention Options History

## Stage 1

Deployment process

...

## Deployment group job

Run on deployment group

+

## Command Line Script

Command line

✓

⋮

## Command line

View YAML

Remove

Task version 2.\*

Display name \*

Command Line Script

Script \*

```
cd /home/mingyu/azagent/_work/r1/a/_CodingSta.mingyunodejs/drop  
sudo nohup node index.js &
```

Advanced

Control Options

Environment Variables

Output Variables

# 당신의 목소리를 들려주세요!

Global Azure Virtual 2020는 여러분의 목소리를 기다립니다.  
가감 없는 목소리가 발표자 분에게 매우 큰 힘이 됩니다.  
앞으로 더 좋은 행사가 될 수 있도록 목소리를 내주세요.  
감사합니다!

세션에 대한 목소리:

<https://sv.kazure.com/>(세션명)\_(순서)

파트너사 행사:

<https://bit.ly/2RvJQzR>

