

# Microsoft Cloud

# Workshop

DevOps on Azure AKS  
Hands-On Lab

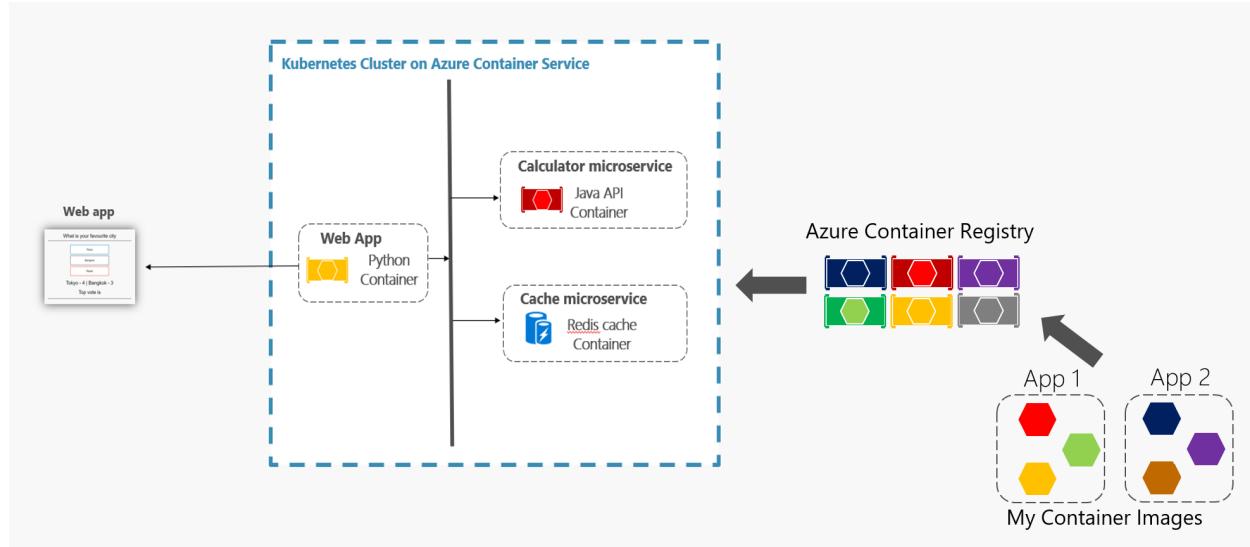
版本	日期	修改历史	作者
1. 0	2018. 12. 17	创建文档	马平

# DevOps on Azure AKS Hands-On Lab

## 多容器应用运行在 Azure AKS 上

### 摘要

这是一个在线投票应用，部署架构如下：



应用包含如下组件：

- 一个前端 Web 应用，使用 Python 语言开发
- 一个计算微服务，使用 Java 语言开发
- 一个缓存微服务，使用 Redis 实现

本教程包括以下内容：

## 目录

DevOps on Azure AKS Hands-On Lab.....	2
实验前提条件 .....	5
任务 1：本地环境准备 .....	5
任务 2：部署 AKS .....	5
任务 3：部署 Azure Container Registry.....	5
基础练习 1：创建 Docker Image .....	5
任务 1：获取应用程序代码 .....	5
任务 2：构建 calculator-api 服务 .....	5
任务 3：构建 Docker Image .....	5
基础练习 2：推送 Docker Image 到 Azure Container Registry .....	5
任务 1：登录国际版 Azure .....	5
任务 2：创建资源组 .....	6
任务 3：创建 Azure Container Registry.....	6
任务 4：登录 Azure Container Registry.....	6
任务 5：将 Docker Image 推送到 Azure Container Registry .....	7
任务 6：列出 Azure Container Registry 中的 Docker Image .....	7
基础练习 3：部署应用程序到 AKS .....	7
任务 1：修改部署文件 .....	7
任务 2：部署应用到 AKS .....	7
基础练习 4：更新应用程序到 AKS .....	8
任务 1：修改应用程序代码 .....	8
任务 2：更新 Docker Image .....	9
任务 3：标记和推送镜像到 Azure Container Registry.....	9
任务 4：更新 azure-vote-front 镜像.....	9
DevOps 练习 1：使用 Jenkins 自动化部署应用到 AKS 群集 .....	10
任务 1：安装 Jenkins VM.....	10
任务 2：设置 Jenkins 环境变量 .....	10
任务 3：设置 Jenkins 凭据 .....	10

任务 4：创建 /var/lib/jenkins/.kube/ 目录 .....	11
任务 5：复制本机的~/.kube/config 到 /var/lib/jenkins/.kube/ 目录 .....	12
任务 6：创建新任务并执行构建 .....	12
DevOps 练习 2：使用 Azure DevOps 自动化部署应用到 AKS 群集.....	14
任务 1：登录到 Azure DevOps.....	14
任务 2：创建 Project .....	14
任务 3：设置认证 .....	15
任务 4：克隆项目到本地 .....	16
任务 5：创建 Build Process.....	17
点击 +，增加一个 Maven task .....	20
点击 +，增加一个 Docker Compose task .....	21
删除 Build an Image 步骤.....	22
点击 +，增加一个 Docker task，给 Image 打 Tag.....	22
Push images 到 Azure Container Registry .....	23
部署 Kubernetes 配置文件 .....	24
测试 Build Process.....	25
任务 6：创建 Release Process .....	26
修改 Kubernetes deployment 和 service.....	29
修改 Kubernetes 中的镜像 .....	32
测试 Release Process .....	33
任务 7：使用 CI/CD 触发器 .....	35
Enable build process trigger.....	35
Enable release process trigger .....	35
测试 CI/CD pipeline .....	37
清除实验资源 .....	39
任务 1：删除资源组 .....	39
任务 2：删除 Service Principle.....	39
任务 3：删除本地容器和镜像 .....	39

# 实验前提条件

## 任务 1：本地环境准备

请提前安装并配置好如下软件：

- JDK 8
- Maven
- Git
- Docker
- kubectl
- Azure CLI

## 任务 2：部署 AKS

## 任务 3：部署 Azure Container Registry

## 基础练习 1：创建 Docker Image

### 任务 1：获取应用程序代码

```
git clone https://github.com/maping/python-voting-web-app.git
```

### 任务 2：构建 calculator-api 服务

```
cd python-voting-web-app
```

```
cd calculator-api
```

```
mvn clean package
```

### 任务 3：构建 Docker Image

```
cd python-voting-web-app
```

```
docker-compose build
```

```
docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
azure-vote-front	latest	491d99820c3c	46 seconds ago	949MB
azure-calculator-api	latest	1d12164ff685	2 hours ago	481MB
tomcat	latest	48dd385504b1	3 days ago	475MB
redis	latest	5958914cc558	12 days ago	94.9MB
tiangolo/uwsgi-nginx-flask	python3.6	1947008cce7	2 weeks ago	945MB

## 基础练习 2：推送 Docker Image 到 Azure Container Registry

### 任务 1：登录国际版 Azure

```
az cloud set -n AzureCloud
```

```
az login -u <Your-Account-Name> -p <Your-Account-Password>
```

访问 <http://portal.azure.com>，用你的帐户登陆。

## 任务 2：创建资源组

### 创建 AKS 群集资源组

```
az group create --name qianhongAksRG --location westeurope
```

如果之前已经创建 AKS 群集资源组，此步跳过。

### 创建 Azure Container Registry 资源组

```
az group create --name qianhongAcrRG --location westeurope
```

注意：资源组名称全球所有 Region 共享名称，所以不能重名。

说明：目前支持 AKS 的区域请查看[这里](https://docs.microsoft.com/zh-cn/azure/aks/container-service-quotas) <https://docs.microsoft.com/zh-cn/azure/aks/container-service-quotas>。

## 任务 3：创建 Azure Container Registry

```
az acr create --resource-group qianhongAcrRG --name qianhongwesteurope --sku Basic --admin-enabled true
```

如果之前已经创建 ACR，此步跳过。

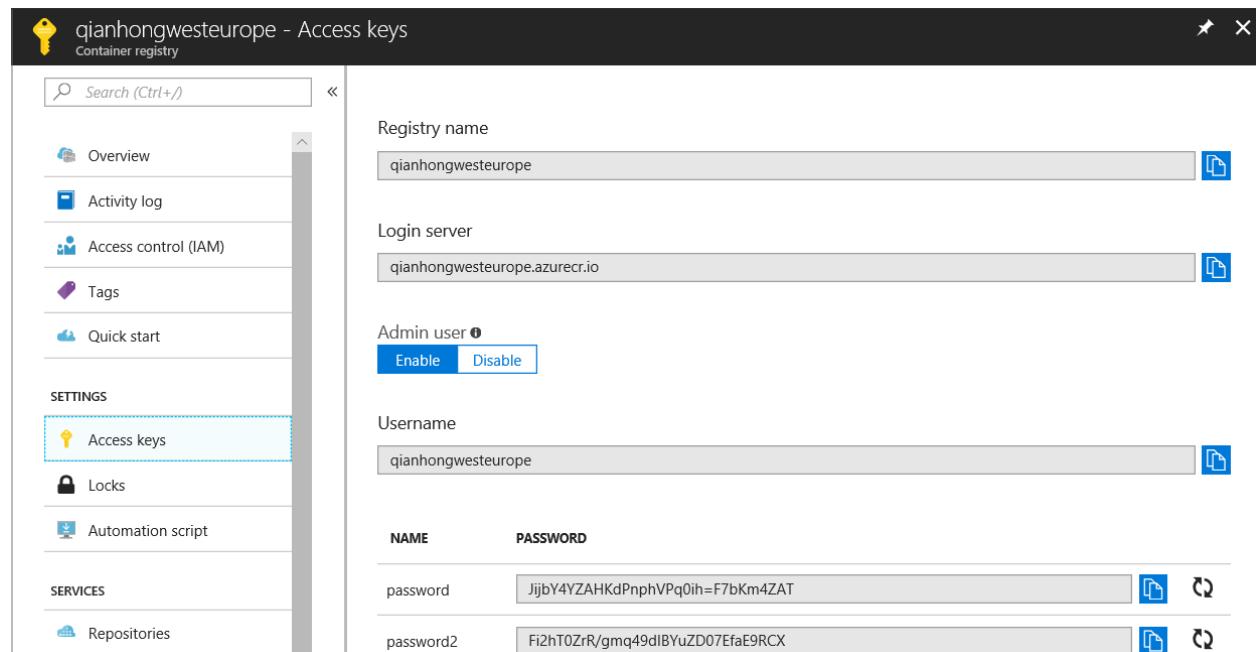
说明：ACR name 必须全球唯一，比如 qianhongwesteurope。

## 任务 4：登录 Azure Container Registry

```
az acr login --name qianhongwesteurope
```

说明：如果用 az acr login 有问题，可以用 docker login 试试。

```
docker login <Your-ACR-Login-Server> -u <Your-ACR-Username> -p <Your-ACR-Password>
```



列出 ACR Login Server 名称

```
az acr list --resource-group qianhongAcrRG --query "[].{acrLoginServer:loginServer}" --output table
```

输出: qianhongwesteurope.azurecr.io

## 任务 5：将 Docker Image 推送到 Azure Container Registry

```
docker tag azure-vote-front qianhongwesteurope.azurecr.io/azure-vote-front:v2.0
```

```
docker push qianhongwesteurope.azurecr.io/azure-vote-front:v2.0
```

说明: 此步在中国大陆网络环境下非常慢, 需要配置容器镜像加速器。

## 任务 6：列出 Azure Container Registry 中的 Docker Image

```
az acr repository list --name qianhongwesteurope --output table
```

输出: azure-vote-front

```
az acr repository show-tags --name qianhongwesteurope --repository azure-vote-front --output table
```

输出: v2.0

访问 Azure 门户, 点击 Container registries > qianhongwesteurope - Repositories

## 基础练习 3：部署应用程序到 AKS

### 任务 1：修改部署文件

修改 python-voting-web-app\azure-vote-all-in-one-redis.yaml

```
image: qianhongwesteurope.azurecr.io/azure-vote-front:v2.0
```

.....

```
image: qianhongwesteurope.azurecr.io/azure-calculator-api
```

### 任务 2：部署应用到 AKS

```
cd python-voting-web-app
```

```
kubectl create -f azure-vote-all-in-one-redis.yaml
```

如果部署失败, 删除 deployment 和 service, 再重新部署。

删除 deployment 和 service 的命令如下 :

```
kubectl delete deployment azure-vote-front azure-calculator-api azure-vote-back
```

```
kubectl delete service azure-vote-front azure-calculator-api azure-vote-back
```

查看应用

```
kubectl get pod
```

```
mapingdeMBP:python-voting-web-app maping$ kubectl get pod
NAME                               READY   STATUS    RESTARTS   AGE
azure-calculator-api-58c698f4bb-jkpp6   1/1     Running   0          62s
azure-vote-back-655476c7f7-ghp68       1/1     Running   0          63s
azure-vote-front-7957855d8d-vnfzv      1/1     Running   0          63s
```

kubectl get service

```
[mapingdeMBP:python-voting-web-app maping$ kubectl get service
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
azure-calculator-api   LoadBalancer  10.0.74.201  40.68.21.81  8080:30632/TCP  5m4s
azure-vote-back        ClusterIP    10.0.189.50  <none>        6379/TCP   5m5s
azure-vote-front       LoadBalancer  10.0.217.110  40.68.21.68  80:31816/TCP  5m4s
kubernetes           ClusterIP   10.0.0.1     <none>        443/TCP   68d
```

访问 <http://40.68.21.68>



## 基础练习 4：更新应用程序到 AKS

### 任务 1：修改应用程序代码

修改 python-voting-web-app\azure-vote\azure-vote\config\_file.cfg 文件，修改后如下：

```
# UI Configurations
TITLE = 'Azure Voting App'
VOTE1VALUE = 'Beijing'
VOTE2VALUE = 'Shanghai'
```

```
SHOWHOST = 'false'
```

保存。

## 任务 2：更新 Docker Image

```
cd python-voting-web-app
```

```
docker-compose build
```

## 任务 3：标记和推送镜像到 Azure Container Registry

```
docker tag azure-vote-front qianhongwesteurope.azurecr.io/azure-vote-front:v2.1
```

```
az acr login --name qianhongwesteurope
```

```
docker push qianhongwesteurope.azurecr.io/azure-vote-front:v2.1
```

注意: docker tag 和 docker push 的 ACR 要一致, 而且大小写敏感。

访问 Azure 门户, 点击 Container registries > qianhongwesteurope - Repositories

## 任务 4：更新 azure-vote-front 镜像

设置 azure-vote-front 镜像为 azure-vote-front:v2.1

```
kubectl set image deployment azure-vote-front azure-vote-front=qianhongwesteurope.azurecr.io/azure-vote-front:v2.1
```

注意: <Your-ACR-Login-Server> 必须全部小写。

监控部署

```
kubectl get pod --watch
```

可以看到 azure-vote-front Pod 是先创建新的 Pod, 然后再把旧的 Pod 逐个停止。

直接刷新原来的页面, 会显示和在本地运行一样的画面。因为修改镜像, 不会导致 Service External IP 发生改变。

# Azure Voting App

Beijing

Shanghai

Reset

Beijing - 6 | Shanghai - 7

Top vote is Shanghai

DevOps 练习 1：使用 Jenkins 自动化部署应用到 AKS 群集

任务 1：安装 Jenkins VM

任务 2：设置 Jenkins 环境变量

Jenkins 管理门户，点击系统管理，点击系统设置，找到全局属性，勾上环境变量，添加 ACR\_LOGIN\_SERVER 环境变量。

全局属性

Tool Locations

环境变量

键值对列表

键 ACR\_LOGIN\_SERVER

值 qianhongwesteurope.azurecr.io

删除

增加

点击保存。

任务 3：设置 Jenkins 凭据

出于安全考虑，现在将 ACR 凭据存储在 Jenkins 凭据对象中。

## Jenkins 管理门户

The screenshot shows the Jenkins 'Credentials' management page. At the top, there are tabs for 'T' (Text), 'P' (Properties), 'Store ↓', 'Domain', 'ID', and 'Name'. Below this is a toolbar with icons for '图标: S M L' and a search bar. A section titled 'Stores scoped to Jenkins' lists two entries: 'Jenkins' and '(global)'. The 'Jenkins' entry has a house icon and the '(global)' entry has a globe icon.

, 点击 Credentials

点击 Jenkins, 点击 Global credentials (unrestricted) 旁边的下拉三角, 点击 Add credentials

The screenshot shows the Jenkins 'System' configuration page under the 'Global credentials' section. It displays a table with one row. The row contains a house icon, the text 'Global credentials (unrestricted)', and a description: 'Credentials that should be available irrespective of domain specification to requirements matching.' Below the table is a toolbar with icons for '图标: S M' and a button for 'Add credentials'.

Kind 选择 Username with password (默认)

Scope 选择 Global (Jenkins, nodes, items, and all child items, etc) (默认)

Username 输入 <ACR\_Username>

Password 输入 <ACR\_Password>

Description 输入 ACR qianhongwesteurope

The screenshot shows the 'Add credentials' form for 'Username with password' type. The form fields are as follows:

Kind	Username with password
Scope	Global (Jenkins, nodes, items, all child items, etc)
Username	qianhongwesteurope
Password	.....
ID	
Description	ACR qianhongwesteurope

任务 4：创建 /var/lib/jenkins/.kube/ 目录

登录到 Jenkins VM, 执行如下命令:

切换到 jenkins 用户: sudo su - jenkins

```
mkdir -m 777 /var/lib/jenkins/.kube/
```

```
exit
```

由于不知道 jenkins 用户的口令，按照如下操作修改 jenkins 用户的口令：

切换到 root 用户: sudo -i

设置 jenkins 用户口令为 jenkins: passwd jenkins

## 任务 5：复制本机的~/.kube/config 到 /var/lib/jenkins/.kube/ 目录

打开 Git Bash，运行如下命令：

```
export JENKINS_USER=jenkins
```

```
export JENKINS_SERVER=mapingjenkins.southeastasia.cloudapp.azure.com
```

```
scp ~/.kube/config $JENKINS_USER@$JENKINS_SERVER:~/.kube/config
```

```
$ scp ~/.kube/config $JENKINS_USER@$JENKINS_SERVER:~/.kube/config  
jenkins@mapingjenkins.southeastasia.cloudapp.azure.com's password:  
config                                         100%  9558      81.3KB/s   00:00
```

在 Jenkins VM 上，切换到 jenkins 用户: sudo su - jenkins

执行 kubectl cluster-info

```
jenkins@mapingJenkinsVM:~$ kubectl cluster-info  
Kubernetes master is running at https://qianhongk8-qianhongaksrg-731332-abee9d55.hcp.westeurope.azurek8s.io:443  
Heapster is running at https://qianhongk8-qianhongaksrg-731332-abee9d55.hcp.westeurope.azurek8s.io:443/api/v1/namespaces/kube-system/services/heapster/proxy  
KubeDNS is running at https://qianhongk8-qianhongaksrg-731332-abee9d55.hcp.westeurope.azurek8s.io:443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy  
Kubernetes-dashboard is running at https://qianhongk8-qianhongaksrg-731332-abee9d55.hcp.westeurope.azurek8s.io:443/api/v1/namespaces/kube-system/services/kubernetes-dashboard/proxy  
Metrics-server is running at https://qianhongk8-qianhongaksrg-731332-abee9d55.hcp.westeurope.azurek8s.io:443/api/v1/namespaces/kube-system/services/https-metrics-server/proxy  
To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
```

说明 jenkins 用户可以连接到 Kubernetes 群集。

## 任务 6：创建新任务并执行构建

输入 MyPythonVotingWebApp-AKS 作为作业名称。选择 **自由风格项目**。

源码管理，选择 Git，Repository URL 输入 <https://github.com/maping/python-voting-web-app.git>

点击构建环境，勾选 **Use secret text(s) or file(s)**

### 构建环境

- Delete workspace before build starts
- Use secret text(s) or file(s)

会出现 **Bindings**，下拉列表选择 **Username and password (separated)**

Username Variable 输入 ACR\_USERNAME

Password Variable 输入 ACR\_PASSWORD

Credentials 选择 Specific credentials, 选择之前创建的 qianhongwesteurope

**构建环境**

Delete workspace before build starts  
 Use secret text(s) or file(s) ?

**Bindings**

**Username and password (separated)**

Username Variable: ACR\_USERNAME  
Password Variable: ACR\_PASSWORD  
Credentials:  Specific credentials  Parameter expression  
qianhongwesteurope/\*\*\*\*\*\*/ (ACR qianhongwesteurope) Add

**Add** X

Abort the build if it's stuck  
 Add timestamps to the Console Output  
 With Ant ?

点击**构建**, 增加一个 Execute shell, 内容如下:

```
WEB_IMAGE_NAME="${ACR_LOGIN_SERVER}/azure-vote-front:kube${BUILD_NUMBER}"
docker build -t $WEB_IMAGE_NAME ./azure-vote
docker login ${ACR_LOGIN_SERVER} -u ${ACR_USERNAME} -p ${ACR_PASSWORD}
docker push $WEB_IMAGE_NAME
```

**构建**

**Execute shell**

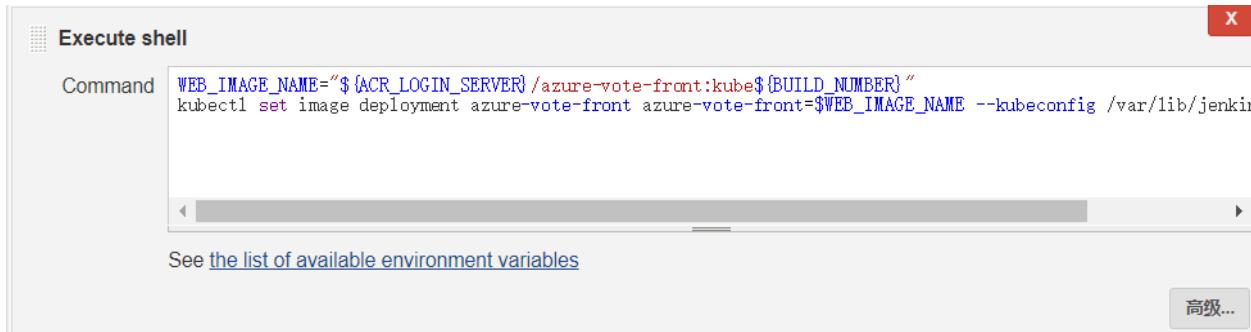
Command: `WEB_IMAGE_NAME="${ACR_LOGIN_SERVER}/azure-vote-front:kube${BUILD_NUMBER}"  
docker build -t $WEB_IMAGE_NAME ./azure-vote  
docker login ${ACR_LOGIN_SERVER} -u ${ACR_USERNAME} -p ${ACR_PASSWORD}  
docker push $WEB_IMAGE_NAME`

See [the list of available environment variables](#)

高级...

再增加一个 Execute shell, 内容如下:

```
WEB_IMAGE_NAME="${ACR_LOGIN_SERVER}/azure-vote-front:kube${BUILD_NUMBER}"
kubectl set image deployment azure-vote-front azure-vote-front=$WEB_IMAGE_NAME --kubeconfig
/var/lib/jenkins/.kube/config
```



### 点击立即构建

```
kubectl: digest: sha256:cb37d3d59a97e0349573ed466e4a0f74347d169890fc0a4d35b9eb9988106d size: 6379
[MyPythonVoting@ebApp-AKS] $ ./bin/sh -xe /tmp/jenkins976378541368268095.sh
+ WEB_IMAGE_NAME=****.azurecr.io/azure-vote-front:kube1
+ kubectl set image deployment azure-vote-front azure-vote-front=****.azurecr.io/azure-vote-front:kube1 --kubeconfig /var/lib/jenkins/.kube/config
deployment.apps "azure-vote-front" image updated
Finished: SUCCESS
```

在等待构建过程中，观察 pod 重新创建过程：kubectl get pod --watch。

可以看到 azure-vote-front Pod 是先创建新的 Pod，然后再把旧的 Pod 逐个停止。

查看 pod 定义

```
C:\Users\pinm>kubectl get pod
NAME                               READY   STATUS    RESTARTS   AGE
azure-vote-back-655476c7f7-cwfsz   1/1     Running   0          18h
azure-vote-front-877b7b7db-p8njt  1/1     Running   0          19m

kubectl describe pod azure-vote-front-877b7b7db-p8njt
Containers:
  azure-vote-front:
    Container ID:  docker://07b1592a526a3b13155dfb7fdcf118380b6308e56fb5b9756662f64850aa5613
    Image:         qianhongwesteurope.azurecr.io/azure-vote-front:kube1
    Image ID:      docker-pullable://qianhongwesteurope.azurecr.io/azure-vote-front@sha256:cb
```

直接刷新原来的页面，因为修改镜像，不会导致 Service External IP 发生改变。

## DevOps 练习 2：使用 Azure DevOps 自动化部署应用到 AKS 群集

### 任务 1：登录到 Azure DevOps

<https://dev.azure.com>

如果没有 Azure DevOps 账户，可以新建一个。

也可以直接访问你的 Azure DevOps 账户，比如：<https://dev.azure.com/maping930883/>

### 任务 2：创建 Project

点击 **Create project**

## Create new project

X

Project name \*

cicd-python-voting-web-app



Description

Visibility



Public ⓘ

Anyone on the internet  
can view the project.  
Certain features like  
TFVC are not supported.



Private

Only people you give  
access to will be able to  
view this project.



Public projects are disabled for your organization. You can turn on public visibility  
with [organization policies](#).

Advanced

Version control ⓘ

Git

Work item process ⓘ

Agile

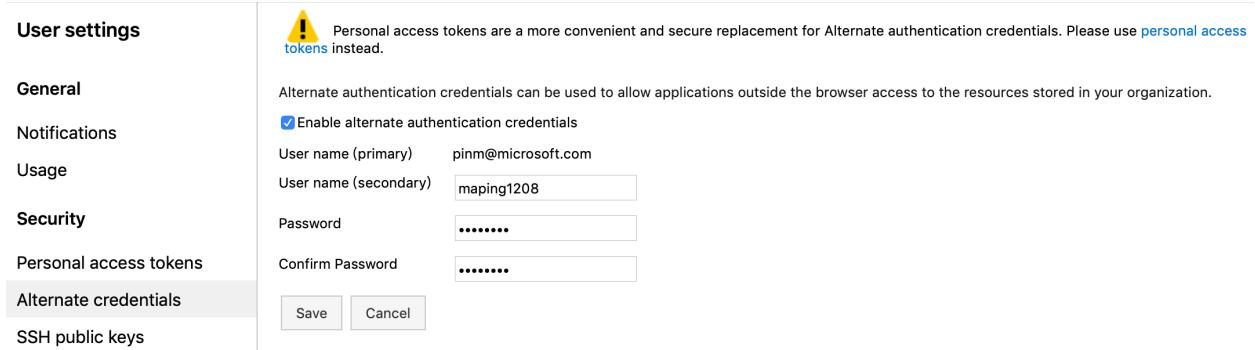
Create

Cancel

### 任务 3：设置认证

为了能够让别人克隆你的 repo，需要设置认证，这里我们选择 Alternate Authentication Credentials 方式。因为这种方式比较便捷，但生产环境建议使用 SSH 或 Git Credential Manager。

点击右上角你的账户，在下拉菜单中选择 Security，点击 Alternate Authentication Credentials，勾上 Enable alternate authentication credentials



## 任务 4：克隆项目到本地

点击 **Repos**,

cicd-python-voting-web-app is empty. Add some code!

### Clone to your computer

HTTPS    SSH   <https://maping930883.visualstudio.com/cicd-python-voting-web-a...>  OR  Clone in VS Code

Generate Git credentials

ⓘ Having problems authenticating in Git? Be sure to get the latest version of Git for Windows or our plugins for IntelliJ, Eclipse, Android Studio or macOS & Linux terminal.

### or push an existing repository from command line

HTTPS    SSH

```
git remote add origin https://maping930883.visualstudio.com/cicd-python-voting-web-app/_git/cicd-python-voting-web-app   
git push -u origin --all 
```

### or import a repository

Import

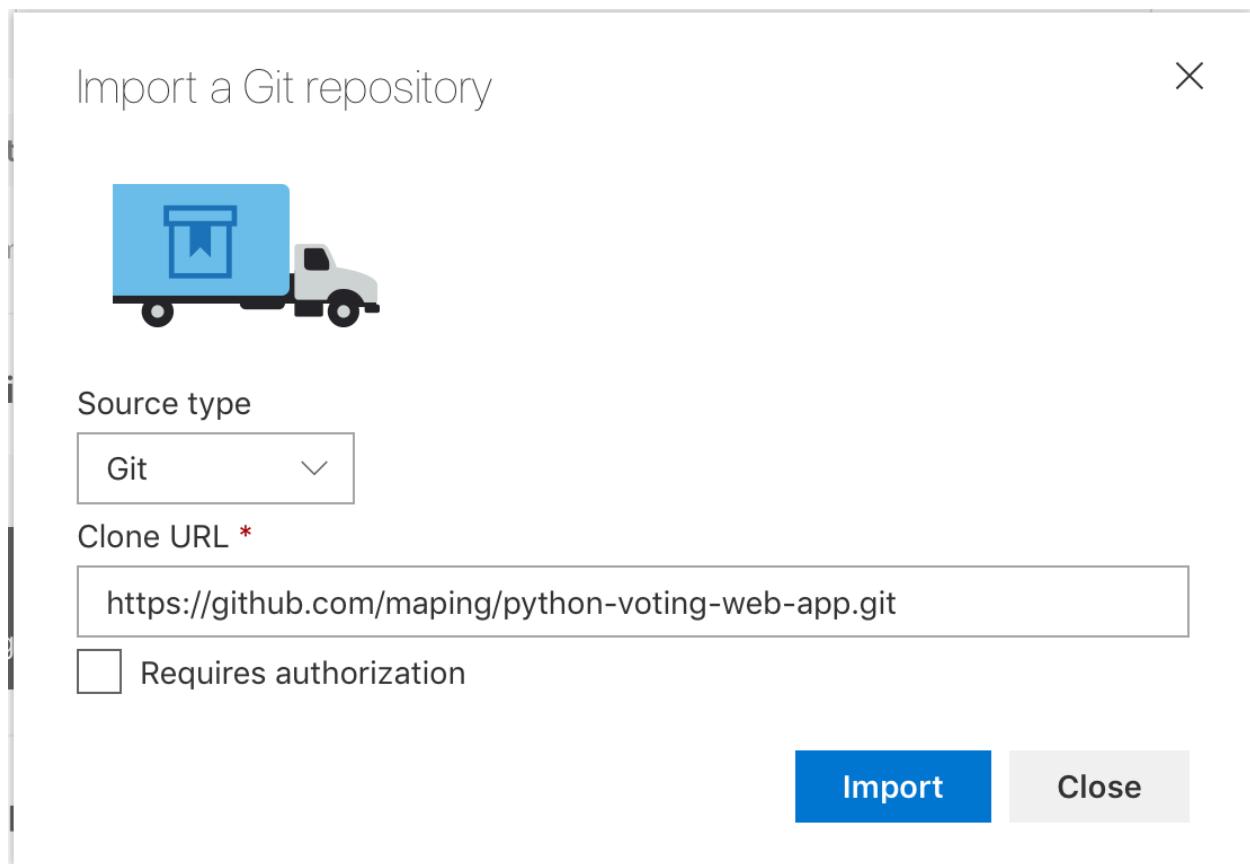
### or initialize with a README or gitignore

Add a README

Add a .gitignore: None

Initialize

点击 **or import a repository**, 输入 <https://github.com/maping/python-voting-web-app.git> , 点击 **Import**



master ✓ cicd-python-voting-web-app / Type to find a file or folder...

Set up build Fork Clone

Contents	History	README	New	Upload file(s)	Download as Zip
Name ↑	Last change	Commits			
azure-vote	30 minutes ago	bc91b645	Update config_file.cfg	Ma Ping	
calculator-api	10/4/2018	6a556958	add calculator-api,now 3 containe...		
jenkins-tutorial	10/2/2018	f2ea59e5	first commit	maping	
.gitignore	10/2/2018	f2ea59e5	first commit	maping	
azure-vote-all-in-one-redis.yaml	10/4/2018	6a556958	add calculator-api,now 3 containe...		
docker-compose.yaml	10/4/2018	6a556958	add calculator-api,now 3 containe...		
LICENSE	10/2/2018	f2ea59e5	first commit	maping	
README.md	10/2/2018	f2ea59e5	first commit	maping	

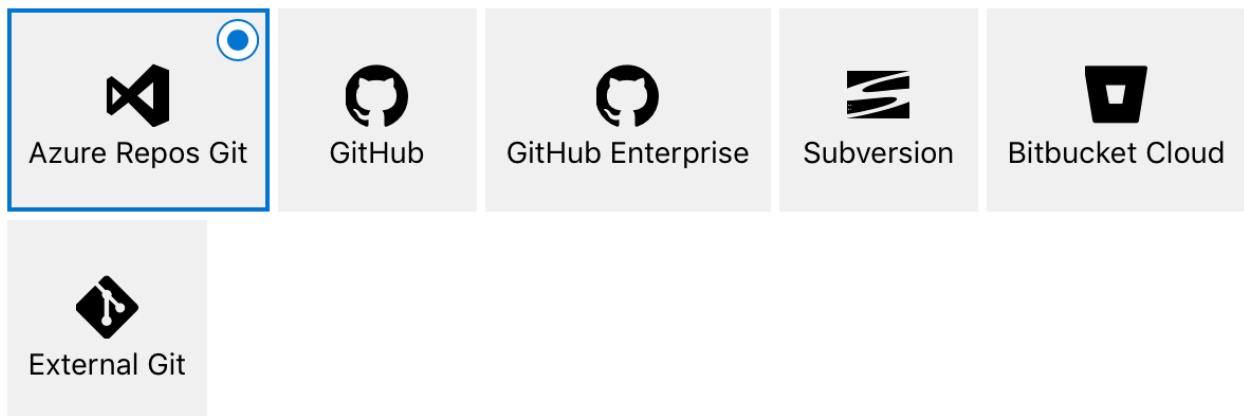
## 任务 5：创建 Build Process

准备在 Pipelines 中创建以下构建任务：

1. 编译和构建 Java Maven calculator API
2. 使用 Docker Compose 构建 Docker Image
3. 为 Docker Image 打 Tag
4. 推送 Docker Images 到 Azure Container Registry
5. 部署 Kubernetes manifest 文件，用于构建 Release Process

点击 **Builds**, 点击 **New pipeline**, 点击 **+ New**, 增加新的 Build Definition。Select your repository 选择 **Azure Repos Git**, 点击 **Continue**

### Select a source



### Team project

▼

### Repository

▼

### Default branch for manual and scheduled builds

▼

**Continue**

Choose a template, 选择 **Docker container**, 点击 **Apply**

## Select a template

Or start with an [Empty job](#)



Search



YAML

Looking for a better experience to configure your pipelines using YAML files? Try the new YAML pipeline creation experience. [Learn more](#)

### Featured



.NET Desktop

Build and test a .NET or Windows classic desktop solution.



Android

Build, test, sign, and align an Android APK.



ASP.NET

Build and test an ASP.NET web application.



Azure Web App for ASP.NET

Build, package, test, and deploy an ASP.NET Azure Web App.



Docker container

Build a Docker image and push it to a container registry.

[Apply](#)



Maven

Build and test a Java project with Apache Maven.

### Agent pool 选择 Hosted Ubuntu 1604

[... > cicd-python-voting-web-app-D...](#)

Tasks   Variables   Triggers   Options   Retention   History   |   [Save & queue](#)   [Discard](#)   [Summary](#)   [Queue](#)   ...

[View YAML](#)

Pipeline   [...](#)

Build pipeline

Get sources   [...](#)

cicd-python-voting-web-app   master

Agent job 1   [+](#)

Run on agent

Build an image   [Docker](#)

Push an image   [Docker](#)

Name \*  

Agent pool \*   [Manage](#)

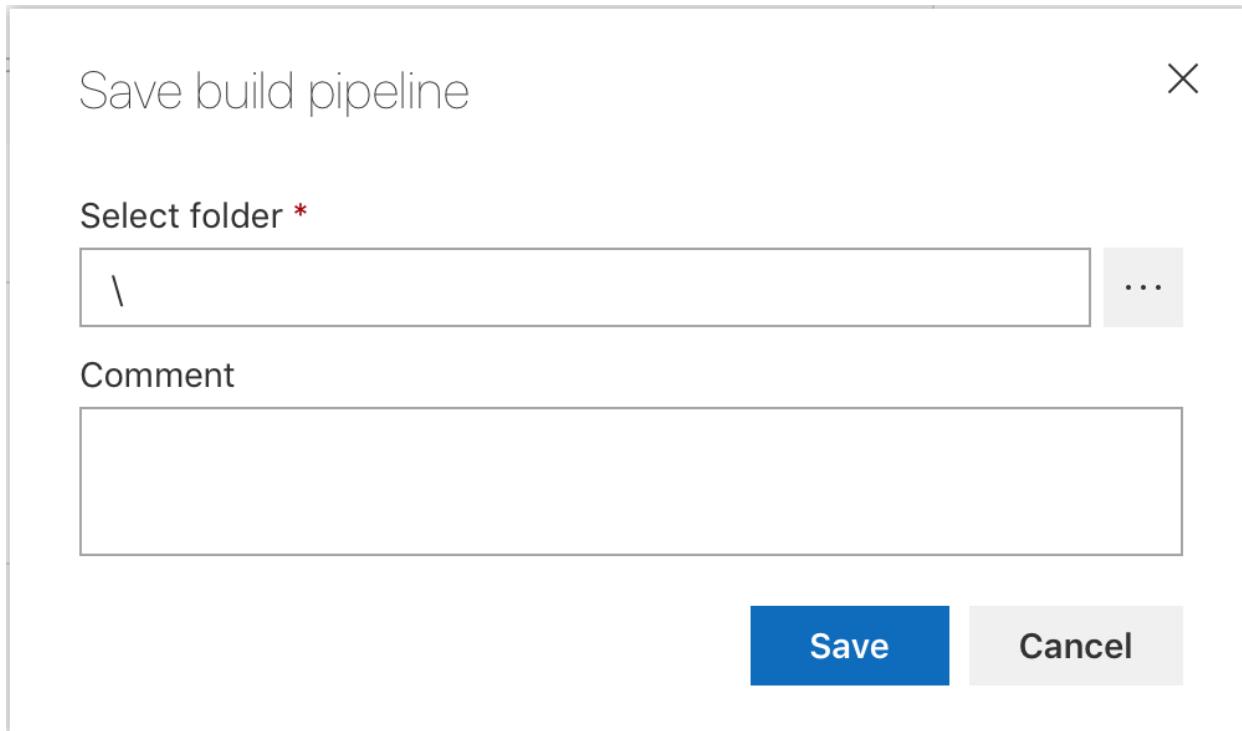
Hosted Ubuntu 1604

Parameters   [...](#)

This pipeline doesn't have any pipeline parameters. Create them to share the most important settings between tasks and change them in one place.

[Learn more](#)

点击 **Save & queue**, 选择 **Save**, 保持默认的路径。



点击 +, 增加一个 Maven task

点击 Add 增加

The screenshot shows the Azure DevOps Pipeline editor. The left sidebar lists tasks: 'Get sources', 'Agent job 1', 'Build an image', and 'Push an image'. The 'Add tasks' search bar at the top right has 'maven' typed into it. A search result for 'Maven' is shown, with an 'Add' button highlighted. Below the search results, there's a 'Marketplace' section with entries for 'Advanced JTest Automation Association' and 'Advanced Cucumber Automation Association'.

把 Maven task 移到 Agent job 1 的第 1 个位置, 并且选择 calculator-api 目录中的 pom.xml 作为 Maven POM file。

The screenshot shows the Azure Pipelines interface for a pipeline named "cicd-python-voting-web-app-D...". The pipeline consists of several steps:

- Get sources**: Fetches code from the "cicd-python-voting-web-app" repository, master branch.
- Agent job 1**:
  - Maven calculator-api/pom.xml**: Maven task, checked.
  - Build an image**: Docker task.
  - Push an image**: Docker task.

On the right, the details for the Maven task are shown:

- Maven**: Version 3.\*
- Display name**: Maven calculator-api/pom.xml
- Maven POM file**: calculator-api/pom.xml
- Goal(s)**: package
- Options**: (empty)
- JUnit Test Results**:
  - Publish to Azure Pipelines/TFS
  - Test results files**: \*\*/surefire-reports/TEST-\*.xml

点击 +, 增加一个 Docker Compose task

点击 Add 增加

... > cicd-python-voting-web-app-D...

The pipeline structure remains the same as the previous screenshot. On the right, the "Add tasks" pane is open, showing the "Docker" tab selected. A "Docker Compose" task is listed with the following details:

- Docker Compose**: Build, push or run multi-container Docker applications. Task can be used with Docker or Azure Container registry.

A large "Add" button is visible at the bottom right of the pane.

把 Docker Compose task 移到 Agent job 1 的第 2 个位置

Azure subscription 选项选择你的订阅，点击 Authorize。

Azure Container registry 选项选择你的 ACR。

Docker Compose File 选项，选择项目根目录下的 docker-compose.yml。

Command 输入项，输入 build。

The screenshot shows the Azure DevOps Pipeline editor. On the left, there's a list of tasks: 'Get sources', 'Agent job 1' (which contains 'Maven calculator-api/pom.xml'), 'Run a Docker Compose command' (which has a note 'Some settings need attention'), 'Build an image', and 'Push an image'. The 'Run a Docker Compose command' task is currently selected. On the right, there are configuration options for this task. It includes fields for 'Display name' (set to 'Run a Docker Compose command'), 'Container Registry Type' (set to 'Azure Container Registry'), 'Azure subscription' (set to 'Microsoft Azure Internal Consumption'), 'Azure Container Registry' (set to 'gianhongwesteurope'), 'Docker Compose File' (set to 'docker-compose.yaml'), and an 'Additional Docker Compose Files' field.

## 删除 Build an Image 步骤

因为使用 Docker compose task 构建镜像，因此不再需要 Build an Image 步骤。

点击 +，增加一个 Docker task，给 Image 打 Tag

… > cicd-python-voting-web-app-D...

The screenshot shows the Azure DevOps Pipeline editor. The pipeline structure is identical to the previous one, but the 'Run a Docker Compose command' task is now highlighted. A 'Docker' search result is shown in the 'Add tasks' dialog. It lists two options: 'Docker' and 'Docker Compose'. Both descriptions mention building, tagging, pushing, or running Docker images, or running a Docker command, and note that they can be used with Docker or Azure Container registry. There is a prominent 'Add' button at the bottom right of the dialog.

把 Docker task 移到 Agent job 1 的第 3 个位置。

Display name 输入项，输入 Add tag to image azure-vote-front。

Azure subscription 选项选择你的订阅，点击 Authorize。

Azure Container registry 选项选择你的 ACR。

Commands 选项，Command 选择 tag。

Arguments 输入项，输入 qianhongwesteurope.azurecr.io/azure-vote-front:cicd\$(Build.BuildId)

Image name 输入项，输入 qianhongwesteurope.azurecr.io/azure-vote-front

… > cicd-python-voting-web-app-D...

The screenshot shows the Azure DevOps Pipeline editor interface. On the left, a list of tasks is visible: 'Get sources', 'Agent job 1' (containing 'Maven calculator-api/pom.xml' and 'Run a Docker Compose command'), 'Add tag to image azure-vote-front' (selected and highlighted in blue), 'Push image azure-vote-front', and 'Publish Kubernetes Artifacts'. On the right, the 'Add tag to image azure-vote-front' task is expanded, showing its configuration. It has an 'Azure container registry' dropdown set to 'qianhongwesteurope'. Under 'Commands', 'Command' is set to 'tag' and 'Arguments' is set to 'qianhongwesteurope.azurecr.io/azure-vote-front:cicd\$(Build.BuildId)'. There are also options for 'Tag multiple images', 'Image name' (set to 'qianhongwesteurope.azurecr.io/azure-vote-front'), and 'Qualify image name' (checked).

## Push images 到 Azure Container Registry

选择默认创建的 Push image 任务，修改如下选项设置：

Display name 输入项，输入 Push image azure-vote-front。

Azure subscription 选项选择你的订阅，点击 Authorize。

Azure Container registry 选项选择你的 ACR。

Commands 选项，Command 选择 push。

Arguments 输入项，输入

Image Name 输入项，输入 azure-vote-front:cicd\$(Build.BuildId)

Azure subscription: Microsoft Azure Internal Consumption (7313321e-6d27-4b22-b...)

Azure container registry: gianhongwesteurope

Command: push

Arguments:

Image name: azure-vote-front:cicd\$(Build.BuildId)

## 部署 Kubernetes 配置文件

点击 +，增加一个 Publish Build Artifacts 任务，

Add tasks | Refresh

publish build

**Copy and Publish Build Artifacts**  
[DEPRECATED] Use the Copy Files task and the Publish Build Artifacts task instead

**Publish Build Artifacts**  
Publish build artifacts to Azure Pipelines/TFS or a file share

**Publish Code Coverage Results**  
Publish Cobertura or JaCoCo code coverage results from a build

修改如下选项设置：

Display name 输入项，输入 Publish Kubernetes Artifacts。

Path to publish 选项，选择 azure-vote-all-in-one-redis.yaml。

Artifact name 输入项，输入 yaml。

The screenshot shows the Azure Pipelines interface for a build pipeline named "cicd-python-voting-web-app-D...". The pipeline consists of a single agent job. The job includes a "Get sources" step and an "Agent job 1" step. "Agent job 1" contains the following tasks:

- Maven calculator-api/pom.xml
- Run a Docker Compose command
- Add tag to image azure-vote-front
- Push image azure-vote-front
- Publish Kubernetes Artifacts

The "Publish Kubernetes Artifacts" task is currently selected.

**Publish Build Artifacts** configuration (right side):
 

- Version: 1.\*
- Display name: Publish Kubernetes Artifacts
- Path to publish: azure-vote-all-in-one-redis.yaml
- Artifact name: yaml
- Artifact publish location: Azure Pipelines/TFS

## 测试 Build Process

点击 Save & Queue，下拉菜单选择 Save & Queue，开始构建。

点击 Build #2018XXX，查看构建详细信息

The screenshot shows the build details for build #20181211.4. The build was manually run today at 4:45 PM by Ping Ma. It used a Hosted Ubuntu 1604 pool and a Hosted Agent. The build started at 2018/12/11 4:45:32 and took 2m 47s.

**Agent job 1 Job**

Pool: Hosted Ubuntu 1604 · Agent: Hosted Agent

Step	Result	Duration
Prepare job	succeeded	<1s
Initialize job	succeeded	1s
Initialize Agent	succeeded	<1s
Checkout	succeeded	5s
Maven calculator-api/pom.xml	succeeded 1 warning	30s
Run a Docker Compose command	succeeded	1m 31s
Add tag to image azure-vote-front	succeeded	7s
Push image azure-vote-front	succeeded	25s
Publish Kubernetes Artifacts	succeeded	4s
Post-job: Checkout	succeeded	<1s
Report build status	succeeded	<1s

发现一个 Warning，点击后

```
018-12-11T08:46:09.9895204Z ##[warning]No test result files matching
/home/vsts/work/1/s/**/surefire-reports/TEST-*.xml were found, so publishing JUnit
test results is being skipped.
```

因为没有 JUnit 单元测试，可以勾掉 Publish to Azure Pipelines/TFS。

… > cicd-python-voting-web-app-D...

重新构建，这次没有警告。

构建成功后，访问 Azure 门户，点击 Container registries > [qianhongwesteurope](#) – Repositories > Tags，确认镜像已经推送到 ACR。

## 任务 6：创建 Release Process

镜像已经 Push 到 Azure Container Registry 中了，现在准备将其发布到 Kubernetes 群集中。

点击 Releases，点击 New pipeline，选择 Deploy to Kubernetes Cluster template，点击 Apply。

All pipelines > New release pipeline

Pipeline Tasks Variables Retention Options History

Artifacts | + Add

+ Add an artifact

Schedule not set

Stages | + Add ▾

Stage 1 Select a template

Select a template

Or start with an Empty job

Search

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.

Apply

Stage name 输入项，输入 cicd-python-voting-web-app。

All pipelines > New release pipeline

Pipeline Tasks Variables Retention Options History

Artifacts | + Add

+ Add an artifact

Schedule not set

Stages | + Add ▾

cicd-python-voti... 1 job, 1 task

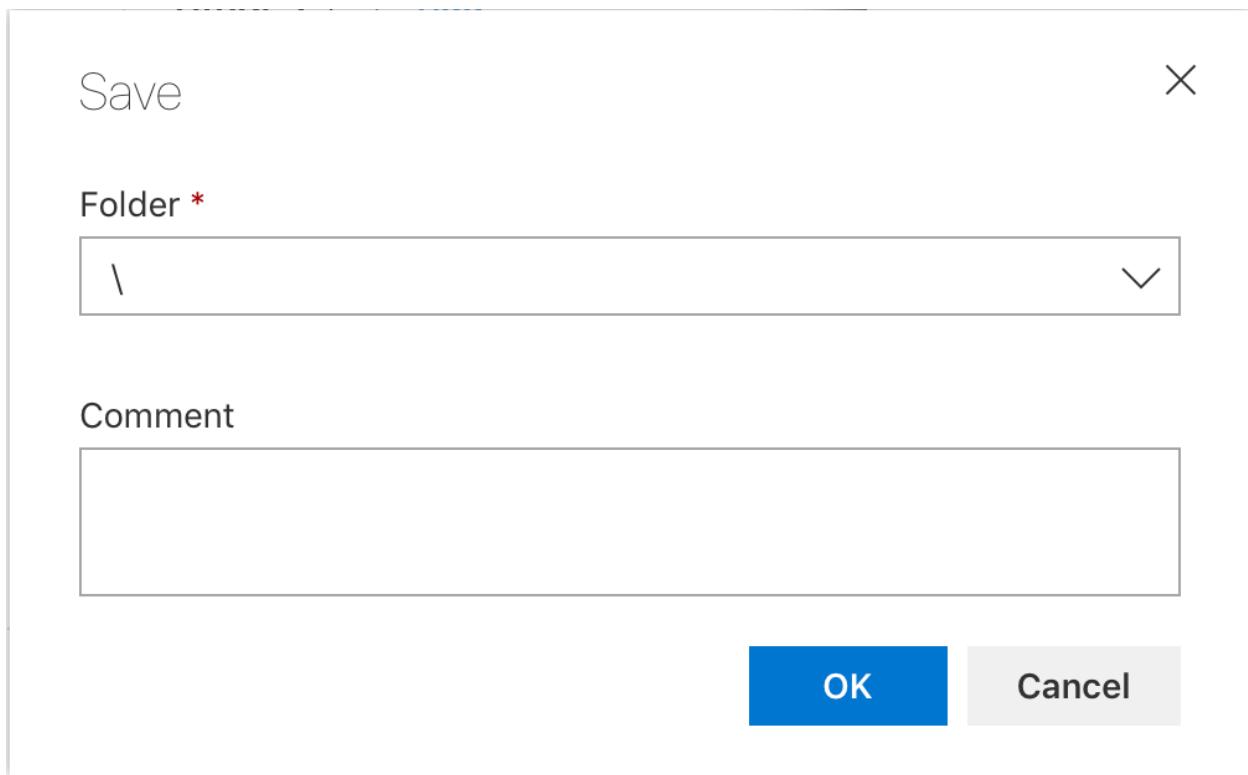
Stage cicd-python-voting-web-app

Properties Name and owners of the stage

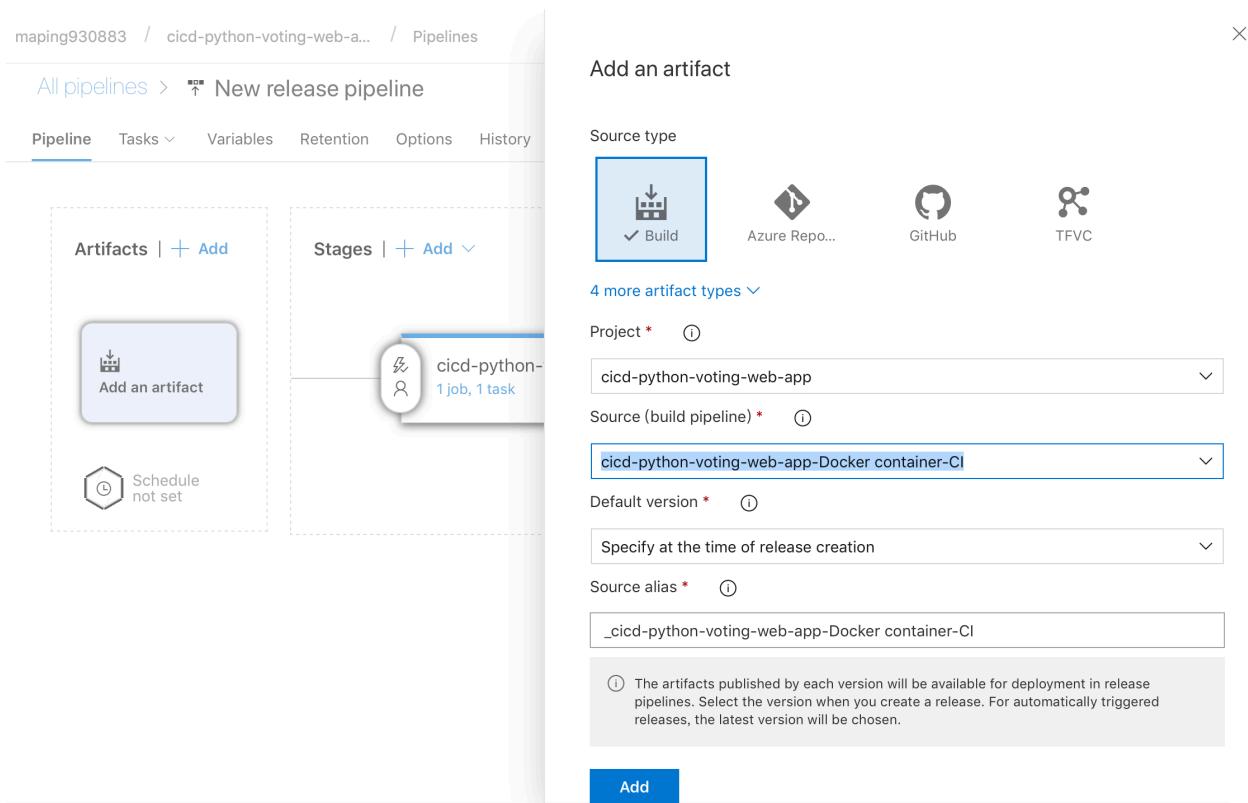
Stage name cicd-python-voting-web-app

Stage owner PM Ping Ma

点击 Save 保存。



点击 **Artifacts** 旁边的 **+ Add**, 选择之前创建的 Build process, 点击 **Add**。

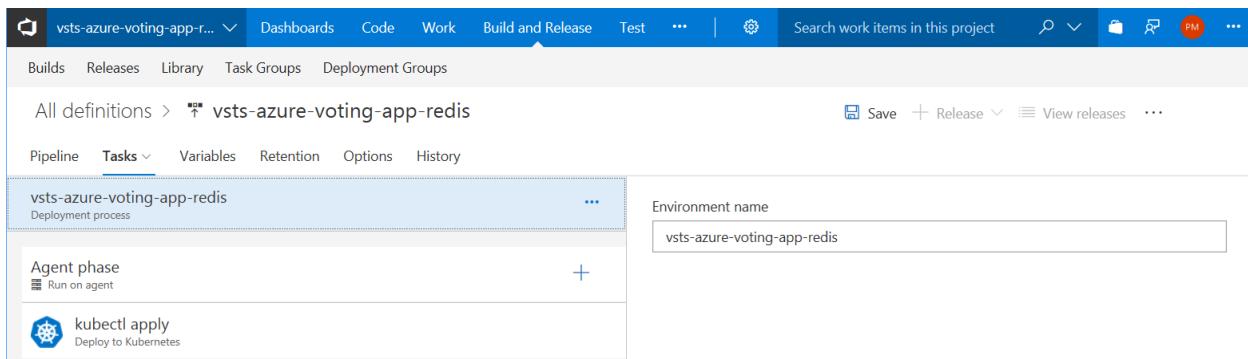
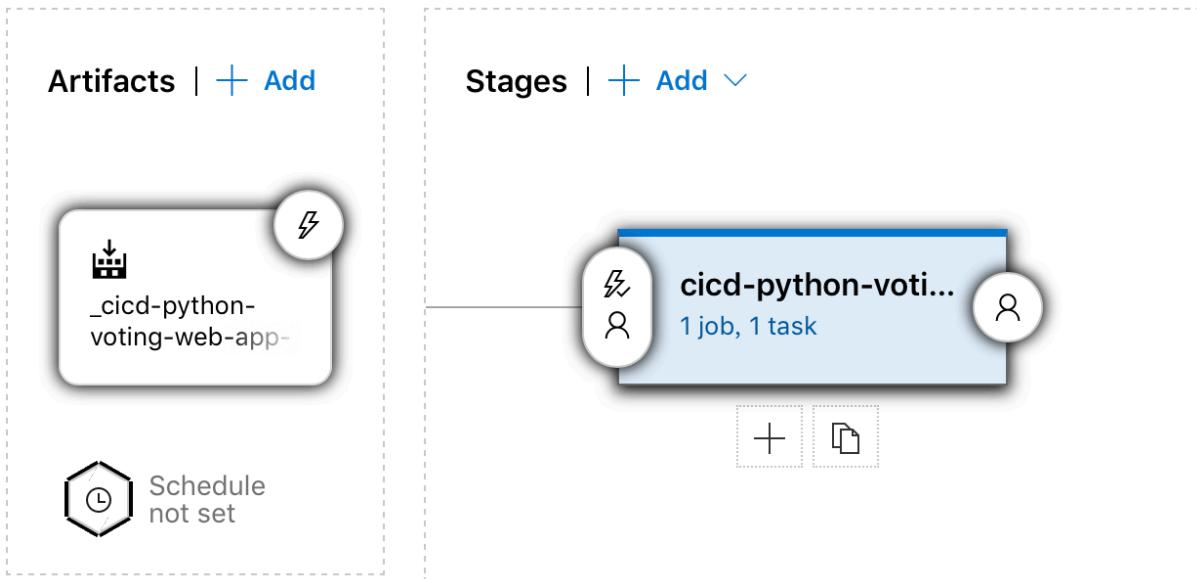


The image shows the Azure DevOps Pipelines interface for creating a new release pipeline. On the left, under 'Artifacts', there is a button '+ Add'. A tooltip 'Add an artifact' is shown above it. On the right, under 'Stages', there is a stage named 'cicd-python-voting-web-app' which contains '1 job, 1 task'. A tooltip 'Schedule not set' is shown below the stage. A modal window titled 'Add an artifact' is open on the right. It has a 'Source type' section with a selected 'Build' option (indicated by a checkmark). Below it are options for 'Azure Repo...', 'GitHub', and 'TFVC'. The 'Project' dropdown is set to 'cicd-python-voting-web-app'. The 'Source (build pipeline)' dropdown is set to 'cicd-python-voting-web-app-Docker container-CI'. The 'Default version' dropdown is set to 'Specify at the time of release creation'. The 'Source alias' input field contains '\_cicd-python-voting-web-app-Docker container-CI'. A note at the bottom states: 'The artifacts published by each version will be available for deployment in release pipelines. Select the version when you create a release. For automatically triggered releases, the latest version will be chosen.' At the bottom of the modal is a blue 'Add' button.

点击 **1 job, 1 task**, 修改 release process。

## All pipelines > New release pipeline

Pipeline Tasks  Variables Retention Options History



vsts-azure-voting-app-r... < Dashboards Code Work Build and Release Test ... | ⚙️ Search work items in this project

All definitions > vsts-azure-voting-app-redis

Save + Release View releases ...

Pipeline Tasks  Variables Retention Options History

vsts-azure-voting-app-redis Deployment process

Agent phase Run on agent

kubectl apply Deploy to Kubernetes

Environment name  
vsts-azure-voting-app-redis

修改 Kubernetes deployment 和 service

修改 kubectl apply 任务，修改如下选项设置：

Display name 输入项，输入 Update deployment and service。

Service connection type 选择 Kubernetes Service Connection

Kubernetes Service Connection 选项，点击 + New，输入如下配置信息：

## Add a Kubernetes endpoint.

Connection name

Server URL  (i)

KubeConfig  (i)

[Learn More](#)

OK Close

- Connection Name: qianhongK8sConnection
- ServerURL: <https://qianhongk8-qianhongaksrg-731332-860b40cc.hcp.westeurope.azurek8s.io>

Home > Kubernetes services > qianhongK8sCluster

**qianhongK8sCluster** Kubernetes service

» Move Delete ⟳ Refresh

Resource group (change) : qianhongAksRG	Kubernetes version : 1.9.9
Status : Succeeded	API server address : <a href="https://qianhongk8-qianhongaksrg-731332-860b40cc.hcp.westeurope.azurek8s.io">qianhongk8-qianhongaksrg-731332-860b40cc.hcp.westeurope.azurek8s.io</a>
Location : West Europe	Total cores : 6
Subscription (change) : Microsoft Azure Internal Consumption	Total memory : 21
Subscription ID : 7313321e-6d27-4b22-befb-1ff4125d35c8	HTTP application routin... : N/A
Tags (change) : Click here to add tags	

- Kubeconfig: 内容是 ~/.kube/config 文件内容

如果没有~/.kube/config 文件, 可以运行如下命令生成一个 qianhongK8sClusterConfig.txt 文件  
az aks get-credentials --resource-group=qianhongAksRG --name qianhongK8sCluster --  
file=.\qianhongK8sClusterConfig.txt

点击 Verify connection, 验证通过后, 点击 OK。

X

## Add a Kubernetes service connection.

Choose authentication

Kubeconfig  Service account

Connection name

qianhongK8sConnection

Server URL

<https://qianhongk8-qianhongaksrg-731332-860b40cc.hc>

(i)

KubeConfig

token: b55111fa00244354a3f6db4eeb09ff7e

(i)

Accept untrusted certificates

(i)

Connection:  Verified

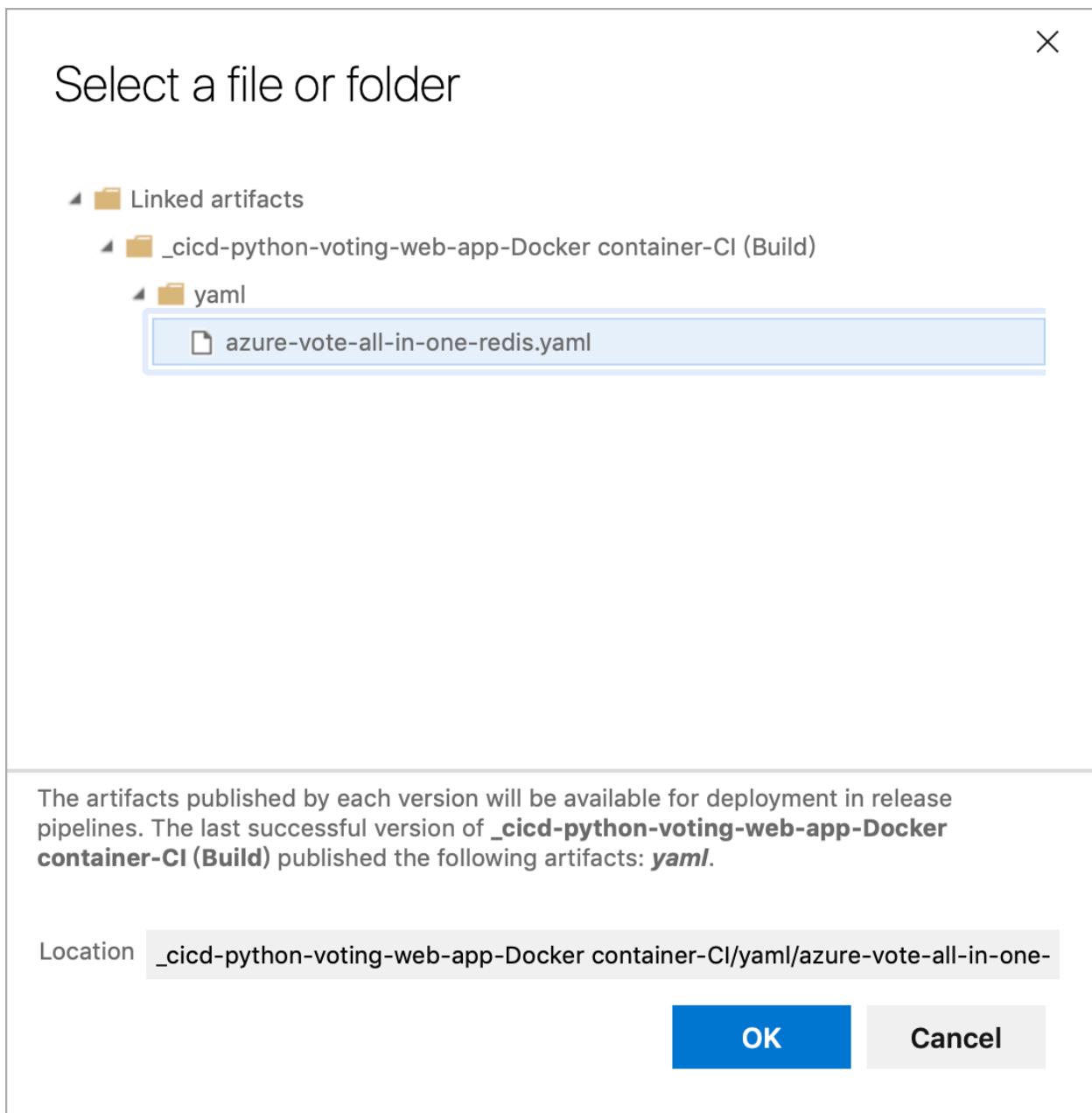
[Verify connection](#)

[Learn more about Kubernetes](#)

OK

Close

勾上 **Use Configuration File**, 选择 `azure-vote-all-in-one-redis.yaml`。



## 修改 Kubernetes 中的镜像

修改 Kubernetes 中的 azure-vote-front 镜像为最新的版本。

点击 + 增加一个 **Deploy to Kubernetes** task，修改如下选项设置：

Display name 输入项，输入 Update image azure-vote-front。

Kubernetes Service Connection 选项，选择 **qianhongK8sConnection**

Command 选项，选择 set。

Argument 输入项，输入 image deployment/azure-vote-front azure-vote-front=**qianhongwesteurope.azurecr.io/azure-vote-front:cicd\$(Build.BuildId)**

## Secret 项信息

- Secret name 输入项，输入 myregistrykey。
- Azure subscription 选项选择你的订阅。
- Azure Container registry 选项选择你的 ACR。

点击 Save。

The screenshot shows the 'New release pipeline' configuration screen. On the left, there's a list of tasks:

- cicd-python-voting-web-app (Deployment process)
- Agent job (Run on agent)
- Update deployment and service (Deploy to Kubernetes)
- Update image azure-vote-front (Deploy to Kubernetes, currently selected)

On the right, the 'Update image azure-vote-front' task is expanded. It has the following settings:

- Command: set
- Arguments: image deployment/azure-vote-front azure-vote-front=qianhongwesteurope.azurecr.io/azure-vote-front:cicd\$(Build.BuildId)
- Secrets:
  - Type of secret: dockerRegistry
  - Container registry type: Azure Container Registry
  - Azure subscription: Microsoft Azure Internal Consumption (7313321e-6d27-4b22-beff) (Scoped to subscription 'Microsoft Azure Internal Consumption')
  - Azure container registry: qianhongwesteurope

## 测试 Release Process

点击 + Release -> + Create release, 点击 Create 开始

## Create a new release

New release pipeline

### ⚡ Pipeline ▾

Click on a stage to change its trigger from automated to manual.



Stages for a trigger change from automated to manual. ⓘ

✓ cicd-python-voting-web-app



### 📄 Artifacts ▾

Select the version for the artifact sources for this release

Source alias	Version	⋮
_cicd-python-voting-web-app-D	20181211.5	▼
Docker container-Cl		

Release description

**Create**

**Cancel**

点击 Release #, 查看 Release Process。

New release pipeline > Release-1 > cicd-python-voting-web-app > Succeeded

Deployment process  
Succeeded

Agent job  
Succeeded

Started: 2018/12/11 下午5:35:33  
... 30s

Initialize Agent · succeeded 2s  
Initialize job · succeeded 3s  
Download artifact - \_cicd-python-voting-web-app-Docker cont... · succeeded 2s  
Update deployment and service · succeeded 15s  
Update image azure-vote-front · succeeded 7s

刷新访问 <http://40.118.42.67>

## 任务 7：使用 CI/CD 触发器

至此，Build Process 和 Release Process 已经创建完毕。现在可以使用 CI/CD 触发器，当代码发生改变时自动触发构建。

### Enable build process trigger

点击 Builds，选择之前定义的 Build Process，点击 Edit，点击 Triggers

勾上 Enable continuous integration，Branch specification 选择 Master

... > cicd-python-voting-web-app-D...

Tasks Variables Triggers Options Retention History | Save & queue Discard Summary Queue ...

Continuous integration  
cicd-python-voting-web-app Enabled

Scheduled + Add

No builds scheduled

Build completion + Add

Build when another build completes

cicd-python-voting-web-app

Enable continuous integration  
 Batch changes while a build is in progress

Branch filters

Type Branch specification  
Include master

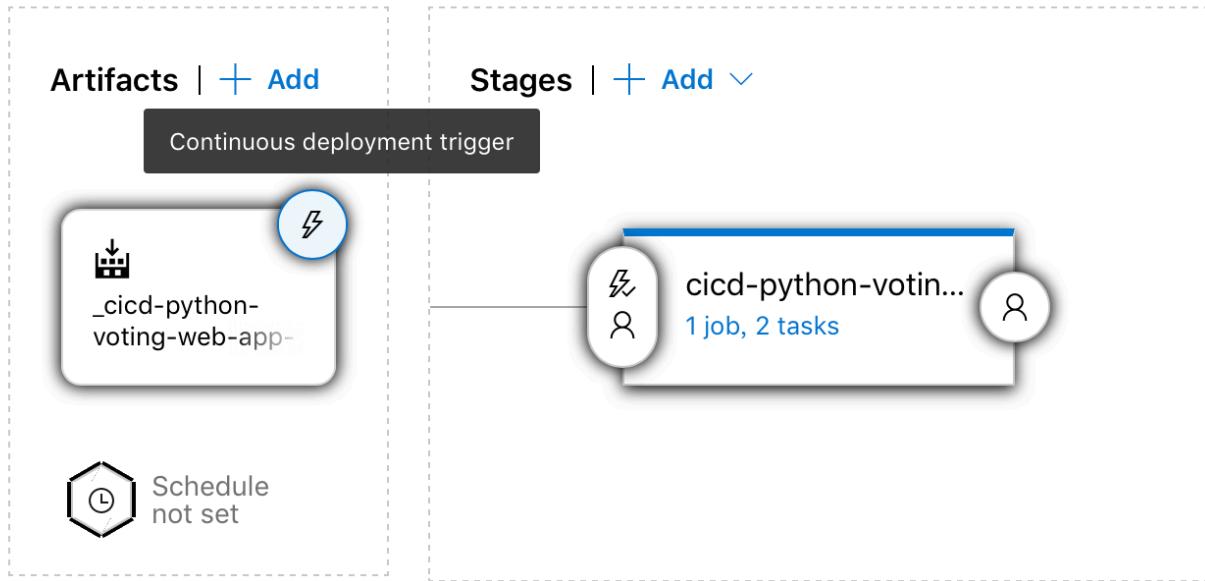
+ Add

Path filters  
+ Add

点击 Save & Queue -> Save。

### Enable release process trigger

点击 Releases，选择之前定义的 Release Process，点击 Edit，点击 Artifacts 右上角 闪电标记，勾上 Enable Continuous deployment trigger



点击 + Add, 增加 Branch, Build Branch 选择 Master

X

## Continuous deployment trigger

Build: \_cicd-python-voting-web-app-Docker container-CI



Enabled

Creates a release every time a new build is available.

### Build branch filters ⓘ

Type

Build branch

Build tags

Include ▾

master ▾



+ Add | ▾

## Pull request trigger

Build: \_cicd-python-voting-web-app-Docker container-CI



Disabled

ⓘ Enabling this will create a release every time a selected artifact is available as part of a pull request workflow

测试 CI/CD pipeline

修改 cicd-python-voting-web-app\azure-vote\azure-vote\config\_file.cfg 文件，提交代码。

观察 VSTS 检测到代码变化，自动触发了 Build and Release Process。

 #20181211.6: Updated config\_file.cfg

Triggered today at 下午9:12 for Ping Ma cicd-python-voting-web-app master 28e4192

[Cancel build](#) ...

Logs [Summary](#) Tests

### Progression

 **Deployments**  
0 No deployments were found for this build.

 **Build pipeline running**  
0 error(s) / 0 warning(s)

 **Associated changes**  
1 commit(s)  
 Updated config\_file.cfg  
Ping Ma authored 28e4192 2m ago

Queued 2m ago  
Started after 3s in queue 2m ago  
Running 2m 38s  
Builds since today at 下午4:26



↑ New release pipeline > Release-3 ▾

Pipeline Variables History | + Deploy ▾  Cancel  Refresh  Edit release

### Release

Continuous deployment...  
for  Ping Ma  
2018/12/11 下午9:15

### Artifacts

 \_cicd-python-vot...   
20181211.6  
master

### Stages

cicd-python-voting-v  
 In progress  
 4/5 tasks  
Update deployment  
 00:15

等待 Build 和 Release Process 完成。刷新页面 <http://40.118.42.67>

# 清除实验资源

## 任务 1：删除资源组

```
az group delete --name qianhongAksRG
```

说明：删除 AKS 群集所在的资源组后，名称类似  
MC\_qianhongAksRG\_qianhongK8sCluster\_westeurope 的资源组也会被自动删除。

```
az group delete --name qianhongAcrRG
```

## 任务 2：删除 Service Principle

查找 Service Principle

```
az ad app list --query "[?displayName=='qianhongK8sCluster'].{Name:displayName,Id:appId}" --output table
```

删除 Service Principle

```
az ad app delete --id <appId>
```

## 任务 3：删除本地容器和镜像

打开 Git Bash 终端，根据实际需要选择执行如下命令

停止所有容器

```
$ for id in $(docker ps -q); do docker stop $id; done
```

删除所有容器

```
$ for id in $(docker ps -aq); do docker rm $id; done
```

删除所有本地的 docker image（执行前，请确认确实要删除所有本地 Docker Image）

```
$ for id in $(docker images -aq); do docker rmi $id; done  
$ docker rmi `docker images -aq`
```

删除所有本地没有 Tag 的 docker image

```
$ docker rmi $(docker images | grep none | awk '{ print $3}')
```