

LABORATORIO 10

Cree un contenedor en Azure y expóngalo en Internet con un nombre de dominio completo (FQDN).

Creación de un contenedor

1. Inicie sesión en [Azure Portal](#) con la suscripción de Azure.
2. Abra Azure Cloud Shell desde Azure Portal seleccionando el icono de Cloud Shell. Asegúrese de seleccionar **Bash** como entorno interactivo de Cloud Shell.



3. Cree un grupo de recursos llamado **learn-deploy-aci-rg-iniciales** para que sea más fácil limpiar estos recursos cuando haya terminado con el módulo. Si elige otro nombre de grupo de recursos, recuérdelo para usarlo en los demás ejercicios de este módulo. También debe elegir una región en la que quiera crear el grupo de recursos, por ejemplo, **Este de EE. UU.**

```
az group create --name learn-deploy-aci-rg-iniciales --location eastus
```

Cree un contenedor, para lo que debe especificar un nombre, una imagen de Docker y un grupo de recursos de Azure en el comando `az container create`. Este contenedor se puede exponer opcionalmente a Internet si se especifica una etiqueta de nombre DNS. En este ejemplo, se implementa un contenedor que hospeda una aplicación web pequeña. También puede seleccionar la ubicación donde se va a colocar la imagen; aquí se usa la región **Este de EE. UU.**, pero puede cambiarla por una ubicación cercana a la suya.

Especifique un nombre DNS para exponer el contenedor en Internet. El nombre DNS debe ser único. Con fines de aprendizaje, ejecute este comando desde Cloud Shell para crear una variable de Bash que contenga un nombre único:

```
DNS_NAME_LABEL=aci-demo-iniciales
```

Ejecute el siguiente comando `az container create` para iniciar una instancia de contenedor:

```
az container create \  
  --resource-group learn-deploy-aci-rg-iniciales \  
  --name mycontainer \  
  --image mcr.microsoft.com/azuredocs/aci-helloworld \  
  --ports 80 \  
  --dns-name-label $DNS_NAME_LABEL \  
  --location eastus
```

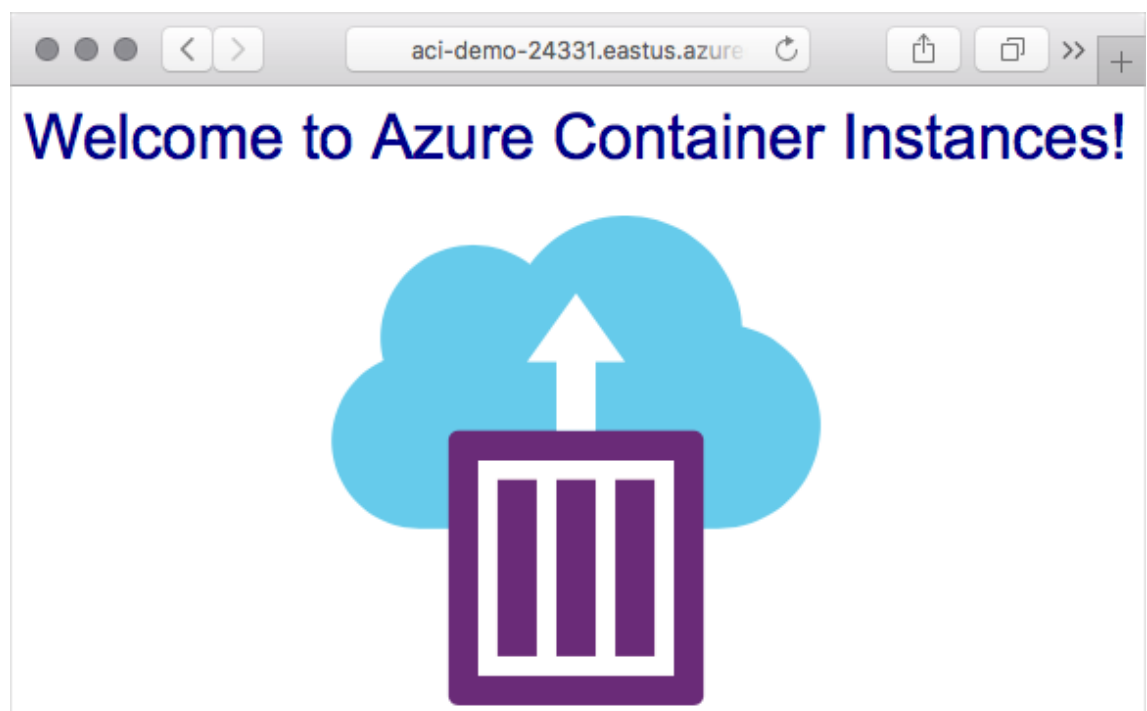
\$DNS_NAME_LABEL especifica el nombre DNS. El nombre de la imagen, **azuredocs/aci-helloworld**, hace referencia a una imagen de contenedor hospedada en Microsoft Container Registry que ejecuta una aplicación web Node.js básica.

Cuando finalice el comando `az container create`, ejecute `az container show` para comprobar su estado:

```
az container show \
--resource-group learn-deploy-aci-rg-iniciales \
--name mycontainer \
--query "{FQDN:ipAddress.fqdn,ProvisioningState:provisioningState}" \
--output table
```

Obtendrá el nombre de dominio completo (FQDN) del contenedor y su estado de aprovisionamiento. Este es un ejemplo:

Desde un explorador, vaya al FQDN del contenedor para verlo en ejecución. Asegúrese de escribir el prefijo `http://` delante de la cadena `aci-demo....`. Debería obtener esta página de bienvenida:



Aquí, ha creado una instancia de contenedor de Azure para ejecutar un servidor y una aplicación web. También ha accedido a esta aplicación con el FQDN de la instancia de contenedor.

Implementación de Azure Cosmos DB

1. Al implementar Azure Cosmos DB, debe proporcionar un nombre de base de datos único. Con fines de aprendizaje, ejecute este comando desde

Cloud Shell para crear una variable de Bash que contenga un nombre único:

```
COSMOS_DB_NAME=aci-cosmos-db-iniciales
```

Ejecute este comando `az cosmosdb create` para crear la instancia de Azure Cosmos DB:

```
COSMOS_DB_ENDPOINT=$(az cosmosdb create \
--resource-group learn-deploy-aci-rg-iniciales \
--name $COSMOS_DB_NAME \
--query documentEndpoint \
--output tsv)
```

Este comando puede tardar unos minutos en completarse.

`$COSMOS_DB_NAME` especifica el nombre de la base de datos único. El comando imprime la dirección del punto de conexión para la base de datos. Aquí, el comando guarda la dirección en la variable Bash `COSMOS_DB_ENDPOINT`.

Ejecute `az cosmosdb keys list` para obtener la clave de conexión de Azure Cosmos DB y almacénela en una variable Bash llamada `COSMOS_DB_MASTERKEY`:

```
COSMOS_DB_MASTERKEY=$(az cosmosdb keys list \
--resource-group learn-deploy-aci-rg-iniciales \
--name $COSMOS_DB_NAME \
--query primaryMasterKey \
--output tsv)
```

Implementación de un contenedor que funcione con la base de datos

Cree una instancia de contenedor de Azure que pueda leer y escribir registros en la instancia de Azure Cosmos DB.

Las variables de dos entornos que ha creado en la última parte, `COSMOS_DB_ENDPOINT` y `COSMOS_DB_MASTERKEY`, conservan los valores que necesita para conectarse a la instancia de Azure Cosmos DB.

1. Ejecute el siguiente comando `az container create` para crear el contenedor:

```
az container create \
--resource-group learn-deploy-aci-rg-iniciales \
--name aci-demo \
--image mcr.microsoft.com/azuredocs/azure-vote-front:cosmosdb \
--ip-address Public \
--location eastus \
--environment-variables \
COSMOS_DB_ENDPOINT=$COSMOS_DB_ENDPOINT \
COSMOS_DB_MASTERKEY=$COSMOS_DB_MASTERKEY
```

azuredocs/azure-vote-front:cosmosdb hace referencia a una imagen de contenedor que ejecuta una aplicación ficticia para votar.

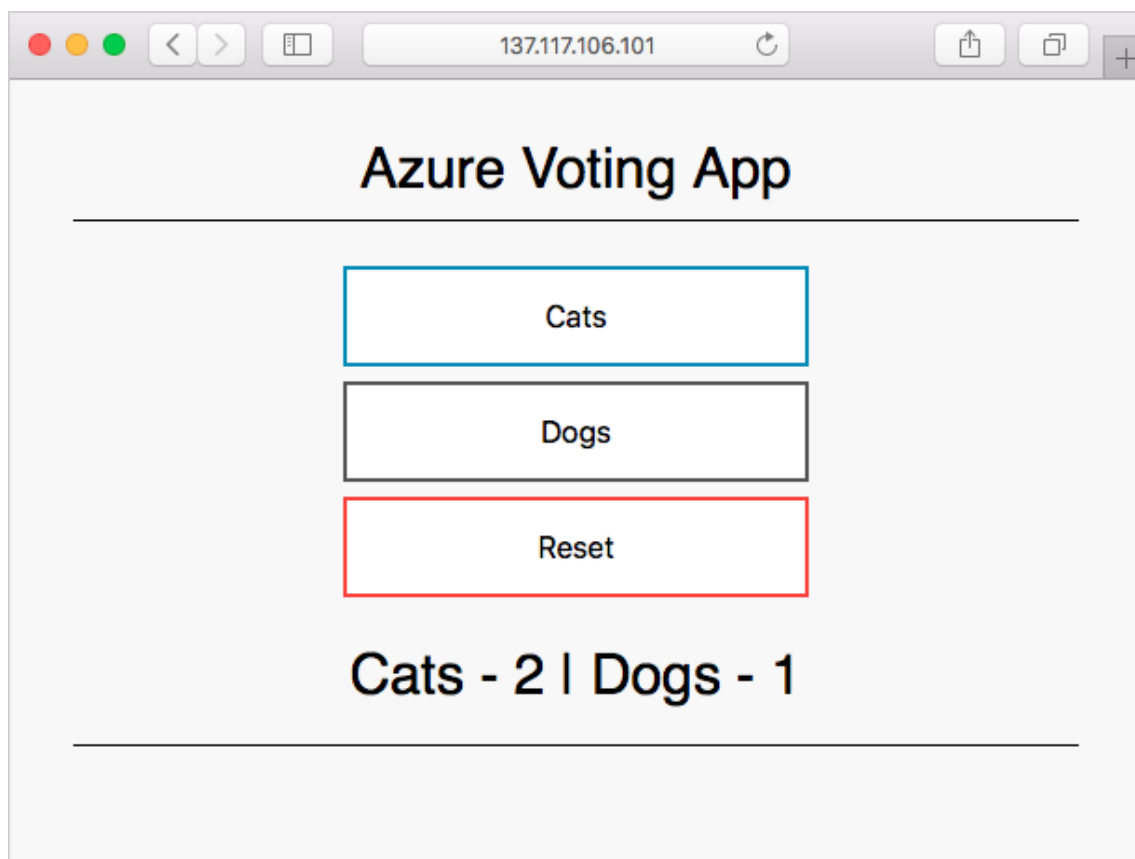
Tenga en cuenta el argumento `--environment-variables`. Este argumento especifica variables de entorno que se pasan al contenedor cuando este se inicia. La imagen de contenedor está configurada para buscar estas variables de entorno. Use el nombre del punto de conexión de Azure Cosmos DB y su clave de conexión.

Ejecute el comando `az container show` para obtener la dirección IP pública del contenedor:

```
az container show \
--resource-group learn-deploy-aci-rg-iniciales \
--name aci-demo \
--query ipAddress.ip \
--output tsv
```

En un explorador, vaya a la dirección IP del contenedor.

Una vez que la aplicación está disponible, se ve esta página:



Intente votar "Gatos" o "Perros". Cada voto se almacena en la instancia de Azure Cosmos DB.

Creación de un recurso compartido de archivos de Azure

Cree una cuenta de almacenamiento y un recurso compartido de archivos. Posteriormente, se puede hacer que una instancia de contenedor de Azure acceda a este recurso.

1. La cuenta de almacenamiento necesita un nombre único. Con fines de aprendizaje, ejecute este comando para almacenar un nombre único en una variable de Bash:

```
STORAGE_ACCOUNT_NAME=mystorageaccountnombre
```

Ejecute el siguiente comando `az storage account create` para crear la cuenta de almacenamiento:

```
az storage account create \  
  --resource-group learn-deploy-aci-rg-iniciales \  
  --name $STORAGE_ACCOUNT_NAME \  
  --sku Standard_LRS \  
  --location eastus
```

Ejecute el siguiente comando para colocar la cadena de conexión de la cuenta de almacenamiento en una variable de entorno denominada `AZURE_STORAGE_CONNECTION_STRING`:

```
export AZURE_STORAGE_CONNECTION_STRING=$(az storage account show-connection-  
string \  
  --resource-group learn-deploy-aci-rg-iniciales \  
  --name $STORAGE_ACCOUNT_NAME \  
  --output tsv)
```

`AZURE_STORAGE_CONNECTION_STRING` es una variable de entorno especial que la CLI de Azure comprende. El elemento `export` permite que otros comandos CLI que se ejecutan después accedan a la variable.

Ejecute este comando para crear un recurso compartido de archivos denominado **aci-share-demo** en la cuenta de almacenamiento:

```
az storage share create --name aci-share-demo
```

Obtención de credenciales de almacenamiento

Para montar un recurso compartido de archivos de Azure como un volumen en Azure Container Instances, necesita estos tres valores:

- Nombre de la cuenta de almacenamiento
- Nombre del recurso compartido
- Clave de acceso de cuenta de almacenamiento

Ya tiene los dos primeros valores. El nombre de cuenta de almacenamiento se almacena en la variable de Bash `STORAGE_ACCOUNT_NAME`. Se especificó **aci-share-demo** como el nombre del recurso compartido en el paso anterior. Aquí obtendrá el valor que falta: la clave de acceso de la cuenta de almacenamiento.

1. Ejecute el siguiente comando para obtener la clave de la cuenta de almacenamiento:

```
STORAGE_KEY=$(az storage account keys list \
--resource-group learn-deploy-aci-rg-iniciales \
--account-name $STORAGE_ACCOUNT_NAME \
--query "[0].value" \
--output tsv)
```

El resultado se almacena en una variable de Bash llamada `STORAGE_KEY`.

Como paso opcional, imprima el nombre de la clave de almacenamiento en la consola.

```
echo $STORAGE_KEY
```

Implementación de un contenedor y montaje del recurso compartido de archivos

Para montar un recurso compartido de archivos de Azure como volumen en un contenedor, especifique el recurso compartido y el punto de montaje del volumen al crear el contenedor.

1. Ejecute este comando `az container create` para crear un contenedor que monte `/aci/logs/` en el recurso compartido de archivos:

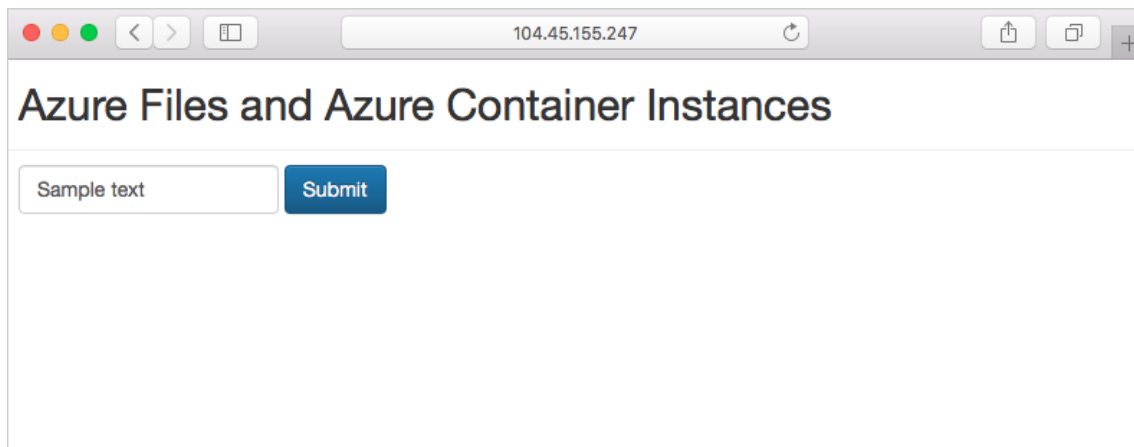
```
az container create \
--resource-group learn-deploy-aci-rg-iniciales \
--name aci-demo-files \
--image mcr.microsoft.com/azuredocs/aci-hellofiles \
--location eastus \
--ports 80 \
--ip-address Public \
--azure-file-volume-account-name $STORAGE_ACCOUNT_NAME \
--azure-file-volume-account-key $STORAGE_KEY \
```

```
--azure-file-volume-share-name aci-share-demo \  
--azure-file-volume-mount-path /aci/logs/
```

Ejecute `az container show` para obtener la dirección IP pública de su contenedor:

```
az container show \  
--resource-group learn-deploy-aci-rg-iniciales \  
--name aci-demo-files \  
--query ipAddress.ip \  
--output tsv
```

Desde un explorador, vaya a la dirección IP de su contenedor. Aparece esta página:



Escriba cualquier texto en el formulario y seleccione **Enviar**. Esta acción crea un archivo que contiene el texto que escribió en el recurso compartido de archivos de Azure.

Ejecute este comando `az storage file list` para mostrar los archivos que se encuentran en el recurso compartido de archivos:

```
az storage file list -s aci-share-demo -o table
```

Ejecute `az storage file download` para descargar un archivo en la sesión de Cloud Shell. Reemplace **<filename>** por uno de los archivos que aparecían en el paso anterior.

```
az storage file download -s aci-share-demo -p <filename>
```

Ejecute el comando `cat` para imprimir el contenido del archivo.

```
cat <filename>
```

Recuerde que los datos se conservan cuando se cierra el contenedor. Puede montar el recurso compartido de archivos en otras instancias de contenedor para hacer que los datos estén disponibles.