# 更新 Azure Kubernetes Service (AKS) 中的應用程式

在 Kubernetes 中部署應用程式之後，您可以藉由指定新的容器映像或映像版本來進行更新。 當您更新應用程式時，更新會分段進行，所以只有一部分的部署會同時更新。 此分段更新方式可讓應用程式在更新期間保持運作， 此外也能當作部署失敗時的復原機制。

在本教學課程 (6/7 部分) 中，已更新範例 Azure Vote 應用程式。 您會了解如何：

* 更新前端應用程式程式碼
* 建立已更新的容器映像
* 將容器映像推送至 Azure Container Registry
* 部署已更新的容器映像

**開始之前**

在先前的教學課程中，已將應用程式封裝成容器映像、將這些映像上傳至 Azure Container Registry (ACR)，並已建立 Kubernetes 叢集。 該應用程式接著便在 Kubernetes 叢集上執行。

應用程式存放庫也會一併複製，其中包括應用程式原始程式碼，以及本教學課程使用的預先建立 Docker Compose 檔案。 請確認您已建立存放庫的複製品，而且已將目錄變更為複製的目錄。 如果您尚未完成這些步驟，而想要跟著做，請回到[教學課程 1 – 建立容器映像](https://docs.microsoft.com/zh-tw/azure/aks/tutorial-kubernetes-prepare-app)。

在進行本教學課程時，您必須執行 Azure CLI 2.0.44 版或更新版本。 執行 az --version 以尋找版本。 如果您需要安裝或升級，請參閱[安裝 Azure CLI](https://docs.microsoft.com/cli/azure/install-azure-cli)。

**更新應用程式**

我們將變更範例應用程式，然後更新已部署至 AKS 叢集的版本。 您可以在 *azure-vote* 目錄中找到範例應用程式的原始程式碼。 使用編輯器 (例如 vi) 開啟 *config\_file.cfg* 檔案：

Console

vi azure-vote/azure-vote/config\_file.cfg

將 *VOTE1VALUE* 和 *VOTE2VALUE* 的值變更為不同的色彩。 下列範例顯示已更新的色彩值：

# UI Configurations

TITLE = 'Azure Voting App'

VOTE1VALUE = 'Blue'

VOTE2VALUE = 'Purple'

SHOWHOST = 'false'

儲存並關閉檔案。

**更新容器映像**

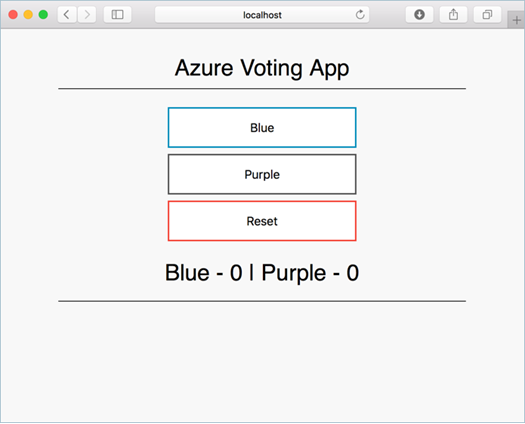
若要重新建立前端映像並測試已更新的應用程式，請使用 [docker-compose](https://docs.docker.com/compose/)。 --build 引數可用來指示 Docker Compose 重新建立應用程式映像：

Console

docker-compose up --build -d

**在本機測試應用程式**

若要確認更新的容器映像已顯示您的變更，請開啟本機網頁瀏覽器並進入 http://localhost:8080。



*config\_file.cfg* 檔案中提供的更新色彩值顯示於執行中的應用程式上。

**標記並推送映像**

若要正確使用更新的映像，請為 *azure-vote-front* 映像標記您 ACR 登錄的登入伺服器名稱。 使用 [az acr list](https://docs.microsoft.com/cli/azure/acr" \l "az_acr_list)命令取得登入伺服器名稱：

Azure CLI

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

使用 [docker tag](https://docs.docker.com/engine/reference/commandline/tag/) 來標記映像。 將 <acrLoginServer> 取代為您的 ACR 登入伺服器名稱或公用登錄主機名稱，並將映像版本更新為 *:v2*，如下所示：

Console

docker tag azure-vote-front <acrLoginServer>/azure-vote-front:v2

現在，使用 [docker push](https://docs.docker.com/engine/reference/commandline/push/) 將映像上傳至您的登錄。 將 <acrLoginServer> 取代為您的 ACR 登入伺服器名稱。 如果您在推送到 ACR 登錄時發生問題，請確定已執行 [az acr login](https://docs.microsoft.com/cli/azure/acr" \l "az_acr_login) 命令。

Console

docker push <acrLoginServer>/azure-vote-front:v2

**部署已更新的應用程式**

若要確保最大執行時間，則應用程式 pod 必須有多個執行個體正在執行中。 使用 [kubectl get pods](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands" \l "get) 命令確認執行中的前端執行個體數目：

$ kubectl get pods

NAME READY STATUS RESTARTS AGE

azure-vote-back-217588096-5w632 1/1 Running 0 10m

azure-vote-front-233282510-b5pkz 1/1 Running 0 10m

azure-vote-front-233282510-dhrtr 1/1 Running 0 10m

azure-vote-front-233282510-pqbfk 1/1 Running 0 10m

如果您沒有多個前端 Pod，請調整 *azure-vote-front* 部署，如下所示：

Console

kubectl scale --replicas=3 deployment/azure-vote-front

若要更新應用程式，請使用 [kubectl set](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands" \l "set) 命令。 以容器登錄的登入伺服器或主機名稱來更新 <acrLoginServer>，並指定 *v2* 應用程式版本：

Console

kubectl set image deployment azure-vote-front azure-vote-front=<acrLoginServer>/azure-vote-front:v2

若要監視部署，請使用 [kubectl get pod](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands" \l "get) 命令。 部署已更新的應用程式後，您的 pod 會終止並以新的容器映像重建。

Console

kubectl get pods

下列範例輸出顯示在部署進行期間正在終止的 Pod 和執行中的新執行個體：

$ kubectl get pods

NAME READY STATUS RESTARTS AGE

azure-vote-back-2978095810-gq9g0 1/1 Running 0 5m

azure-vote-front-1297194256-tpjlg 1/1 Running 0 1m

azure-vote-front-1297194256-tptnx 1/1 Running 0 5m

azure-vote-front-1297194256-zktw9 1/1 Terminating 0 1m

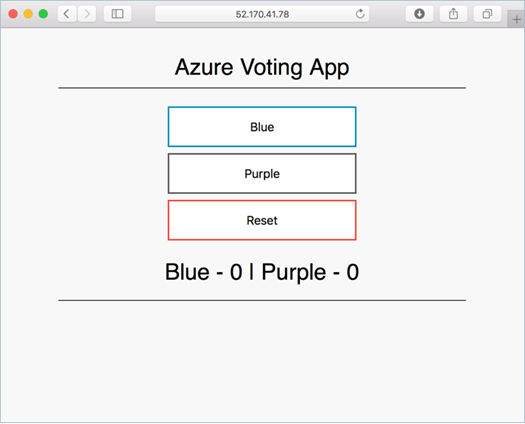
**測試已更新的應用程式**

若要檢視更新應用程式，請先取得 azure-vote-front 服務的外部 IP 位址：

Console

kubectl get service azure-vote-front

現在，開啟本機網頁瀏覽器並前往該 IP 位址。



**後續步驟**

在本教學課程中，您已更新應用程式並將此更新推出至 Kubernetes 叢集。 您已了解如何︰

* 更新前端應用程式程式碼
* 建立已更新的容器映像
* 將容器映像推送至 Azure Container Registry
* 部署已更新的容器映像

繼續進行下一個教學課程，以了解如何將 AKS 叢集升級為新版的 Kubernetes。