# 在 Azure Kubernetes Service (AKS) 中調整應用程式

如果您一直都依照教學課程操作，就會在 AKS 中有一個正常運作的 Kubernetes 叢集，並已部署「Azure 投票」應用程式。 在本教學課程中 (七個章節的第五部分)，您會將應用程式中的 Pod 相應放大，然後嘗試進行 Pod 自動調整。 您也會了解如何調整 Azure VM 節點的數目，以變更叢集裝載工作負載的容量。 您會了解如何：

* 調整 Kubernetes 節點
* 手動調整執行應用程式的 Kubernetes Pod
* 設定執行應用程式前端的自動調整 Pod

在後續的教學課程中，Azure 投票應用程式會更新為新版本。

**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、將此映像上傳至 Azure Container Registry，並已建立 Kubernetes 叢集。 該應用程式接著便在 Kubernetes 叢集上執行。 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)。

在本教學課程中，您必須執行 Azure CLI 2.0.38 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)。

**手動調整 Pod**

在先前的教學課程中部署 Azure Vote 前端與 Redis 執行個體時，已建立單一複本。 若要查看叢集中的 Pod 數目和狀態，請使用 [kubectl get](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands" \l "get) 命令，如下所示：

Console

kubectl get pods

下列範例輸出會顯示一個前端 Pod 和一個後端 Pod：

複製

NAME READY STATUS RESTARTS AGE

azure-vote-back-2549686872-4d2r5 1/1 Running 0 31m

azure-vote-front-848767080-tf34m 1/1 Running 0 31m

若要手動變更 *azure-vote-front* 部署中的 Pod 數目，請使用 [kubectl scale](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands" \l "scale) 命令。 下列範例會將前端 Pod 的數目增加為 *5*：

Console

kubectl scale --replicas=5 deployment/azure-vote-front

再次執行 [kubectl get pods](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands" \l "get)，以確認 Kubernetes 是否建立了其他 Pod。 大約一分鐘後，其他 Pod 就會成為叢集中的可用項目：

Console

$ kubectl get pods

READY STATUS RESTARTS AGE

azure-vote-back-2606967446-nmpcf 1/1 Running 0 15m

azure-vote-front-3309479140-2hfh0 1/1 Running 0 3m

azure-vote-front-3309479140-bzt05 1/1 Running 0 3m

azure-vote-front-3309479140-fvcvm 1/1 Running 0 3m

azure-vote-front-3309479140-hrbf2 1/1 Running 0 15m

azure-vote-front-3309479140-qphz8 1/1 Running 0 3m

**自動調整 Pod**

Kubernetes 支援[水平 Pod 自動調整](https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/)，可根據 CPU 使用率或其他選取的計量來調整部署中的 Pod 數目。 [計量伺服器](https://kubernetes.io/docs/tasks/debug-application-cluster/core-metrics-pipeline/)可用來將資源使用率提供給 Kubernetes。 若要安裝計量伺服器，請複製 metrics-server GitHub 存放庫並安裝範例資源定義。 若要檢視這些 YAML 定義的內容，請參閱[Kuberenetes 1.8+ 的計量伺服器](https://github.com/kubernetes-incubator/metrics-server/tree/master/deploy/1.8%2B)。

Console

git clone https://github.com/kubernetes-incubator/metrics-server.git

kubectl create -f metrics-server/deploy/1.8+/

若要使用自動調整程式，必須為您的 Pod 定義 CPU 要求和限制。 在 azure-vote-front 部署中，前端容器會要求 0.25 個 CPU，限制為 0.5 個 CPU。 設定看起來會像這樣：

Yaml

resources:

requests:

cpu: 250m

limits:

cpu: 500m

下列範例會使用 [kubectl autoscale](https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands" \l "autoscale) 命令自動調整 *azure-vote-front* 部署中的 Pod 數目。 如果 CPU 使用率超過 50%，自動調整程式就會增加 Pod，最多可達 10 個執行個體：

Console

kubectl autoscale deployment azure-vote-front --cpu-percent=50 --min=3 --max=10

若要查看自動調整程式的狀態，請使用 kubectl get hpa 命令，如下所示：

$ kubectl get hpa

NAME REFERENCE TARGETS MINPODS MAXPODS REPLICAS AGE

azure-vote-front Deployment/azure-vote-front 0% / 50% 3 10 3 2m

幾分鐘之後，如果 Azure Vote 應用程式上的負載降到最低，Pod 複本數目就會自動降低成三個。 您可以再次使用 kubectl get pods，以查看移除的非必要 Pod。

**手動調整 AKS 節點**

如果您在上一個教學課程中使用命令建立了 Kubernetes 叢集，它會有一個節點。 如果您打算在叢集上增加或減少容器工作負載，可以手動調整節點的數目。

下列範例會在名為 myAKSCluster 的 Kubernetes 叢集中，將節點的數目增加到三個。 此命令需要幾分鐘的時間來完成。

Azure CLI

az aks scale --resource-group=myResourceGroup --name=myAKSCluster --node-count 3

輸出如下：

複製

"agentPoolProfiles": [

{

"count": 3,

"dnsPrefix": null,

"fqdn": null,

"name": "myAKSCluster",

"osDiskSizeGb": null,

"osType": "Linux",

"ports": null,

"storageProfile": "ManagedDisks",

"vmSize": "Standard\_D2\_v2",

"vnetSubnetId": null

}

**後續步驟**

在本教學課程中，您已在 Kubernetes 叢集中使用不同的調整功能。 您已了解如何︰

* 調整 Kubernetes 節點
* 手動調整執行應用程式的 Kubernetes Pod
* 設定執行應用程式前端的自動調整 Pod