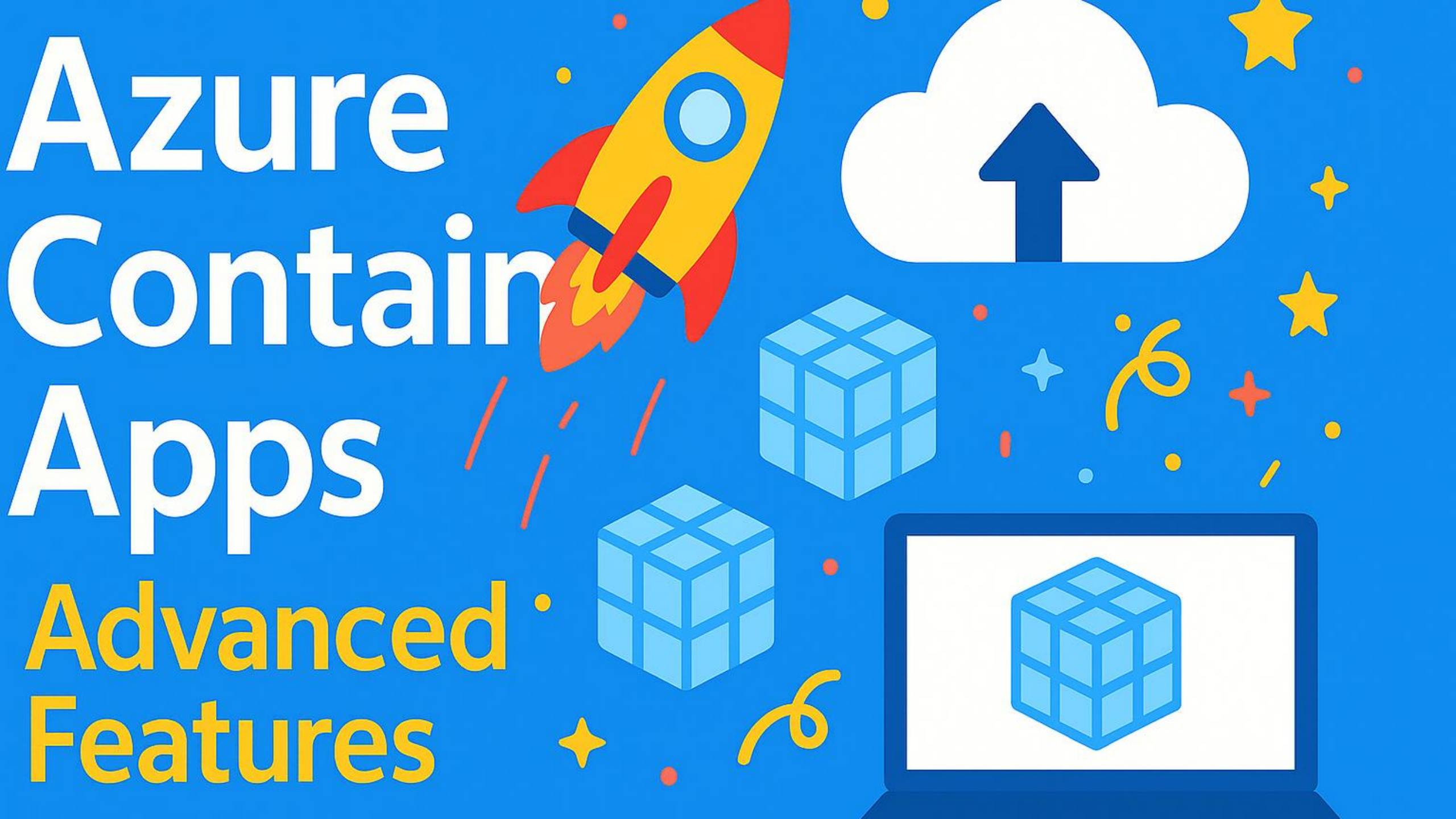
Azure Container Apps

Advanced Features



Agenda

- Azure Container Apps
- Deployment via Azure CLI
- Custom DNS & SSL
- Scaling/Replicas
- Revisions
- Traffic Splitting

Introduction to Azure Container Apps

- Fully managed serverless container service
- Ideal for microservices, APIs, background proc
- Built on Kubernetes and KEDA
  - (KEDA) Kubernetes-based Event Driven Autoscaler

Benefits

- Serverless scalability
- No need to manage Kubernetes
- Integrated with Azure ecosystem
- Fast deployment and revisions

Deploy Azure Container App

Demo1

Demo1 – Prerequisites

- Create git repository
  git init Demo1
- Create Application (index.html)
  cd Demo1
  echo '<h1>Demo Website</h1><h2>Version 1</h2>' > index.html
  cat index.html
- Create Dockerfile
  echo FROM nginx:alpine > Dockerfile
  echo COPY index.html /usr/share/nginx/html/index.html >> Dockerfile
  cat Dockerfile
- Check files in
  git status
  git add .
  git commit -m "Initial commit: Demo1 v1 on nginx"

Demo1 –
Setup Resource Group & Azure Container Registry

- Create Resource Group

  az group create -n **demo-rg** --location **southcentralus**

- Create Azure Container Registry/Deploy Container Image

  az acr create -n **cloudacr** --resource-group **demo-rg** --sku **Basic**

  az acr build -r **cloudacr** -t **demo1:v1 .**

- Register namespaces and add containerapp extensions
  - az provider register --namespace **Microsoft.App**
  - az provider register --namespace **Microsoft.OperationalInsights**
  - az extension add --name **containerapp** --upgrade --allow-preview **true**

Demo1 –Azure Container App

- Create Azure Container App Environment

  az containerapp env create -n **demoenv** -g **demo-rg** -l **southcentralus**

- Create Azure Container App

  pw='**********’

  az containerapp create -n **demoaca** -g **demo-rg** --environment **demoenv** --image **cloudacr.azurecr.io/demo1:v1** --ingress **external** --target-port 80 --registry-server **cloudacr.azurecr.io** --registry-username **cloudacr** --registry-password **$pw**

Custom Domain & SSL

Demo2

Demo2 – Prerequisites

- Domain Name
- DNS

Demo2 - Custom Domain & SSL

- Setup CNAME & txt Records
- Validate

Scaling/Replicas
Console
Revisions
Traffic Splitting

Demo3

Demo3 – Scaling/Replicas

- Modify http-scaler rule from 10 concurrent requests to 1
- Modify Min replicas from 0 to 4

Demo3 - Console

Demo3 – Revision

- Update Application (index.html)

  # Make sure you are in the Demo1 repository
  echo '<h1>Demo Website</h1><h2>Version 2</h2>' > index.html
  cat index.html
  git status
  git add .
  git commit -m "Demo3 v2 on nginx“

- Build New Container Image

  az acr build -r **cloudacr** -t **demo1:v2 .**

- Update Azure Container App

  az containerapp update -n **demoaca** -g **demo-rg** --image **cloudacr.azurecr.io/demo1:v2**

Demo3 – Traffic Splitting

This Demo was done through the Azure Portal

- Make last revision Active. This will revert us back to v1
- Change Revision Mode from single to multi-version mode
- Activate latest revision with 0% traffic
- Modify traffic to 25% for v2 and 75% to v1

Resources

- Azure Container Apps documentation
  https://learn.microsoft.com/en-us/azure/container-apps/
- Quickstart: Deploy your first container app using the Azure portal
  https://learn.microsoft.com/en-us/azure/container-apps/quickstart-portal