Deloitte.

Tech Treks

Building a DevOps Pipeline

Welcome!

- ROI leads the industry in designing and delivering customized technology and management training solutions
- Meet your instructor
  - Name
  - Background
  - Contact info
- Let’s get started!

Course Objectives

In this course, you will:

- Create a complete DevOps pipeline
- Manage application code and versions using Git and GitHub
- Execute CI/CD pipelines using GitHub Actions
- Package application code and dependencies using Docker Images
- Deploy containers using Serverless cloud environments

Agenda

This course is taught over three 2-hour sessions.

Session 1:

Managing Software Development with Git

Session 2:

Packaging Code with Docker

Session 3:

Automating Deployment to Serverless Compute

Agenda

This is session three.

Session 1:

Managing Software Development with Git

Session 2:

Packaging Code with Docker

Session 3:

Automating Deployment to Serverless Compute

Session 3: Automating Deployment to Serverless Compute

Session Objectives

In this session, you will:

- Learn benefits of and use cases for serverless computing
- Deploy serverless applications using a GitHub Workflows and Actions

Session Concepts

Serverless Computing

Serverless Deployment with GitHub Actions

Lab

What Is Serverless Computing?

- Serverless computing allows developers to build and run applications without managing infrastructure
- Cloud provider automatically provisions, scales, and manages the servers
- Key characteristics
  - No server management
  - Automatically scales with demand, handling large or small traffic seamlessly
  - Pay-as-You-Go

Serverless Applications are Event Driven

- Code is executed in response to some external event

Benefits of Serverless Architecture

- Cloud provider handles all infrastructure
- Scales automatically handle traffic spikes and dips
- Built for high availability
- No need to pay for idle or underutilized resources
- Faster time to market

Popular Serverless Platforms

AWS
Lambda

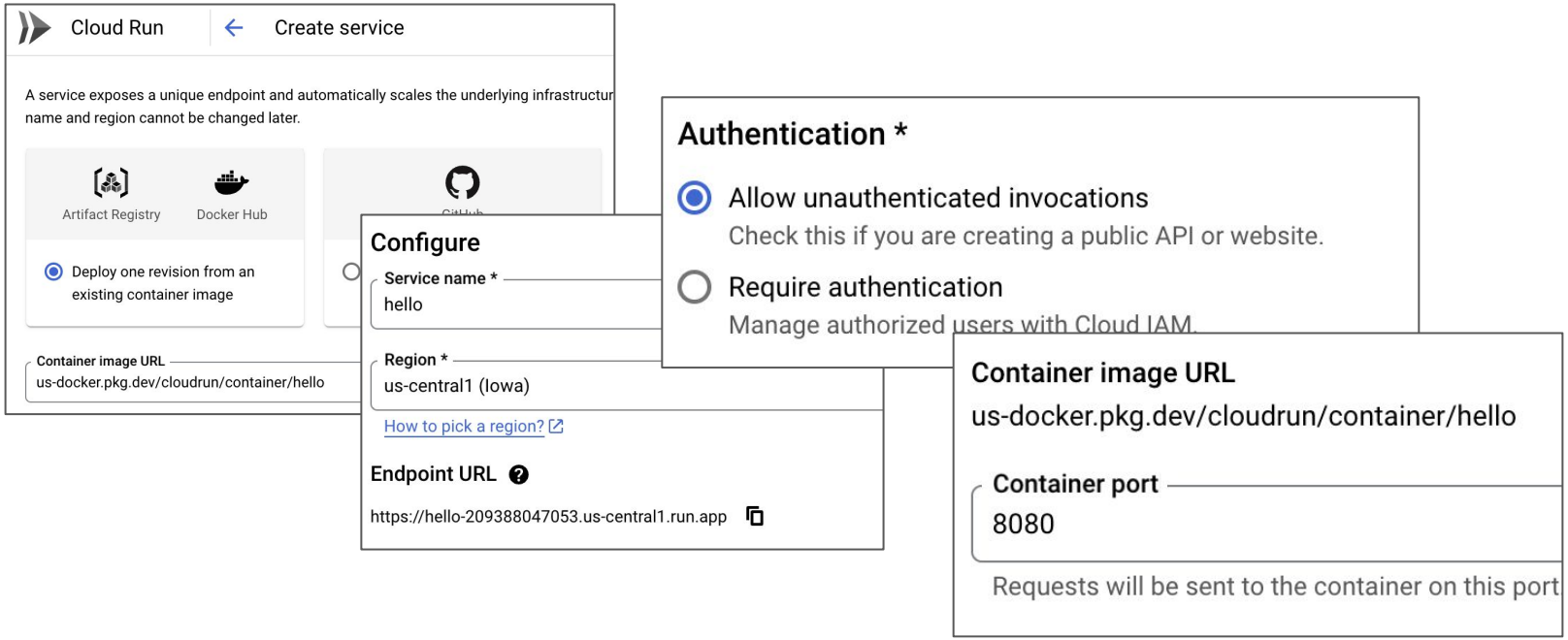
Google
Cloud Run

Azure
Functions

Use Cases for Serverless

- Microservice Architectures
- Real-Time Data Processing
- Scheduled Tasks and Automation
- Chatbots and Voice Assistants
- APIs and Webhooks
- Web Applications

Demo: Deploy a Docker Image to Cloud Run



Session Concepts

Serverless Computing

Serverless Deployment with GitHub Actions

Lab

Deploy to Serverless Compute with GitHub Workflows

- Requires cloud credentials
  - Store in GitHub secrets
  - AWS Access/Secret key pairs or service account keys
- Use the GitHub action for whichever serverless environment you are using
  - Can alternatively use cloud provider CLI commands to deploy your application

Using GitHub Actions to Deploy to AWS Lambda

```
name: Deploy Python App to AWS Lambda
<< Code Omitted to Save Space>>
jobs:
  deploy:
    runs-on: ubuntu-latest

    - name: Configure AWS credentials
      uses: aws-actions/configure-aws-credentials@v2
      with:
        aws-access-key-id: ${{ secrets.AWS_ACCESS_KEY_ID }}
        aws-secret-access-key: ${{ secrets.AWS_SECRET_ACCESS_KEY }}
        aws-region: us-east-1

    - name: Deploy to AWS Lambda
      uses: aws-actions/aws-lambda-deploy@v1
      with:
        function-name: my-lambda-function
        zip-file: lambda.zip
```

Requires AWS access/secret keys for authentication

Use AWS Lambda action to deploy the code

Using GitHub Actions to Deploy to Azure Functions

```
name: Deploy Python App to AWS Lambda
<< Code Omitted to Save Space>>
jobs:
  deploy:
    runs-on: ubuntu-latest

    - name: 'Login via Azure CLI'
      uses: azure/login@v1
      with:
        creds: ${{ secrets.AZURE_CREDENTIALS }}

    - name: 'Deploy to Azure Functions'
      uses: azure/functions-action@v1
      with:
        app-name: ${{ secrets.AZURE_FUNCTIONAPP_NAME }}  # Replace with Function name
        package: '.'  # Path to the directory or zip file to deploy
```

Log in with Azure credentials

Use Azure Function action to deploy the code

Using GitHub Actions to Deploy to Cloud Run

```
<< Code Omitted to Save Space>>
jobs:
  deploy:
    runs-on: ubuntu-latest
    steps:
      - name: Authenticate to Google Cloud
        uses: google-github-actions/auth@v1
        with:
          credentials_json: ${{ secrets.GCP_SERVICE_ACCOUNT_KEY }}

    - name: Deploy to Cloud Run
        uses: google-github-actions/deploy-cloudrun@v1
        with:
          service: ${{ github.actor }}-tech-trek
          image: docker.io/${{ secrets.DOCKER_HUB_USERNAME }}/tech-trek:${{ github.sha }}
          region: us-central1
          flags: "--allow-unauthenticated"
```

Requires Google Service Account Key for authentication

Use the Cloud Run GitHub Action to deploy a Docker Image

Session Concepts

Serverless Computing

Serverless Deployment with GitHub Actions

Lab

Hands-On Exercise

- Do the following exercise:
  - Session 3 Lab: Deploy to Cloud Run with GitHub Actions

Session Summary

In this session, you have:

- Learned benefits of and use cases for serverless computing
- Deployed serverless applications using a GitHub Workflows and Actions

Discussion: Recap

Course Summary

In this course, you have:

- Created a complete DevOps pipeline
- Managed application code and versions using Git and GitHub
- Executed CI/CD pipelines using GitHub Actions
- Packaged application code and dependencies using Docker Images
- Deployed containers using Serverless cloud environments