Deploying a .NET 8 App to Azure using GitHub Actions and Bicep

Slides at scottsauber.com

scottsauber

What you need

- An Editor of some sort that supports .NET 8 and Bicep
  - Rider 2023 EAP 8
  - VS 17.8+
- VS Code with Bicep Extension recommended
- Fork this repo:
  - https://github.com/scottsauber/workshop-dotnet-azure-github-bicep
- Please let Scott know the following if you're participating:
  - Email you will use for Azure
  - GitHub account
- WiFi is LTGWHQ-Guest, password is learn@lt

Audience

- .NET Developers
- Anyone interested in Azure, GitHub or Bicep

Agenda

- What is the final state of what we’re building?
- What is Azure?
- What is Azure App Service? Plans?
- What is Bicep?
- What are GitHub Actions?
- Hands on all throughout

Goals

- Learn GitHub Actions, Bicep, and Azure
- We likely won’t get to everything in a few hours
  - This is going to be… a lot
- The feedback loop on this is slow
- Take home a few things back to work, whether beginner or expert

Who am I?

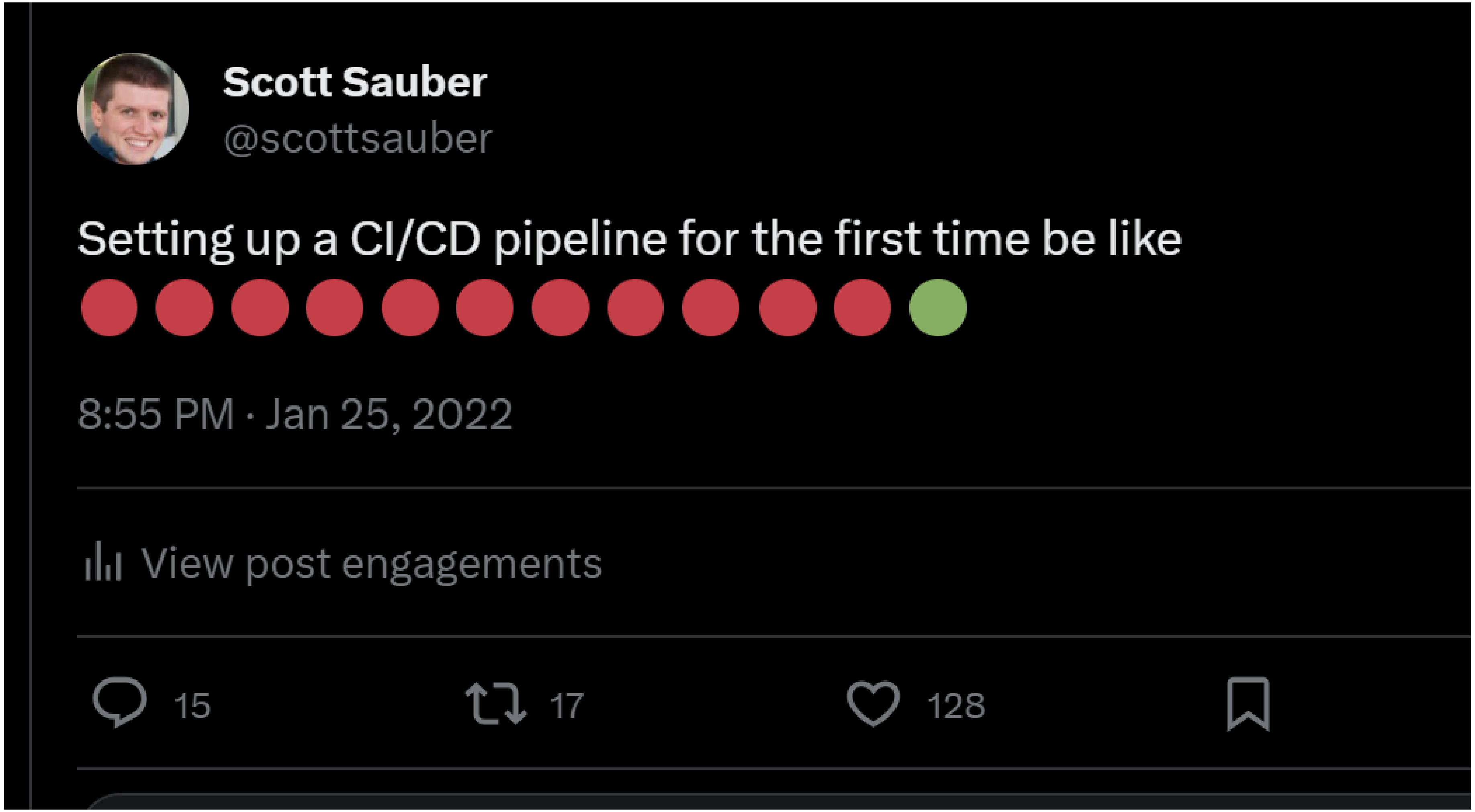
- Director of Engineering at Lean TECHniques
- Co-organizer of Iowa .NET User Group
- Microsoft MVP
- Friend of Redgate
- Blog at scottsauber.com

What are we building?

- .NET 8 API
- Running on Azure App Service
- Configured using Infrastructure as Code with Bicep
- Deployed via GitHub Actions

Features of What We’re Building

- Zero Downtime Deployment
- Infrastructure managed by code, not clicking in the portal
- Automated Build and Deploys
- Follows Azure Naming Standards for naming resources
- WhatIf on PR for infrastructure changes
- Versioning your app so you know what’s deployed
- Health Checks



Scott Sauber
@scottsauber

Setting up a CI/CD pipeline for the first time be like

🔴🔴🔴🔴🔴🔴🔴🔴🔴🔴🔴🟢

8:55 PM · Jan 25, 2022

Azure

What is Azure?

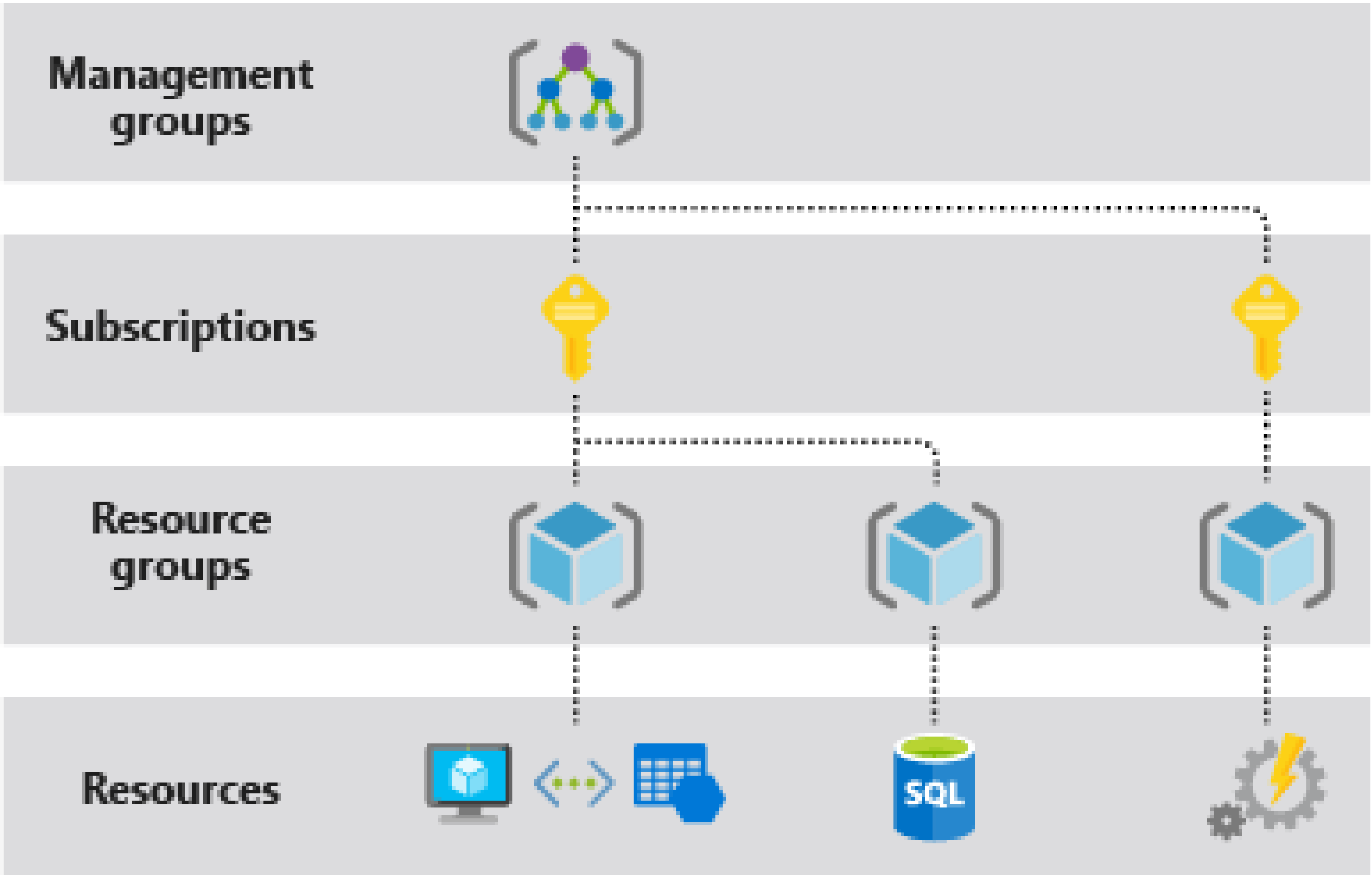
- Microsoft offering for cloud hosting
- Offers many services from hosting web apps to databases to caching to messaging to…
- You should probably be picking PaaS offerings (i.e. not VMs)

Subscriptions

- Top-ish level organization (ignoring Tenants, Management Groups for a minute)
- Recommended per team per environment
- My default naming convention: sub-<team/dept>-<environment>
  - Eg: sub-accounting-dev
- Role access separation
- Billing separation

Resource Groups

- Related groups of resources (i.e. web, DB, Key Vault, etc)
- Quickly view all resources related to that app
- Conceptually, RG = folder, and Resources = files
- Recommended per app per environment
- Default naming convention: rg-<product name>-<environment>
  - Eg: rg-fancyapp-dev
- May have many RG's in a single subscription
- Role access separation
- Billing separation



scottsauber

Azure App Service

What is Azure App Service?

- PaaS offering for hosting applications
- ☑ Handles OS patches, Framework patches
- ☑ Zero downtime deployments with slots
- ☑ SSL Certs
- ☑ Extremely simple
- ☑ Handles scaling
- ☑ Custom Domains
- ☑ And More
- ☒ Less control because PaaS

What is an Azure App Service Plan?

- Think of it like the VM for your App Service(s)
- Pick how much memory, CPU, storage you need
- You can put multiple app services on an ASP…but should you?
- Tiers for Linux
- Many apps can get away with S1 in my experience (~$70/mo for Linux)
- Need to be at least on Standard to get Deployment Slots (ZDD)
- Savings plan – commit to $ amount, 25% (1yr) or 45% savings (3yrs)
- Reservation – commit to compute, 35% (1yr) or 55% savings (3yrs)

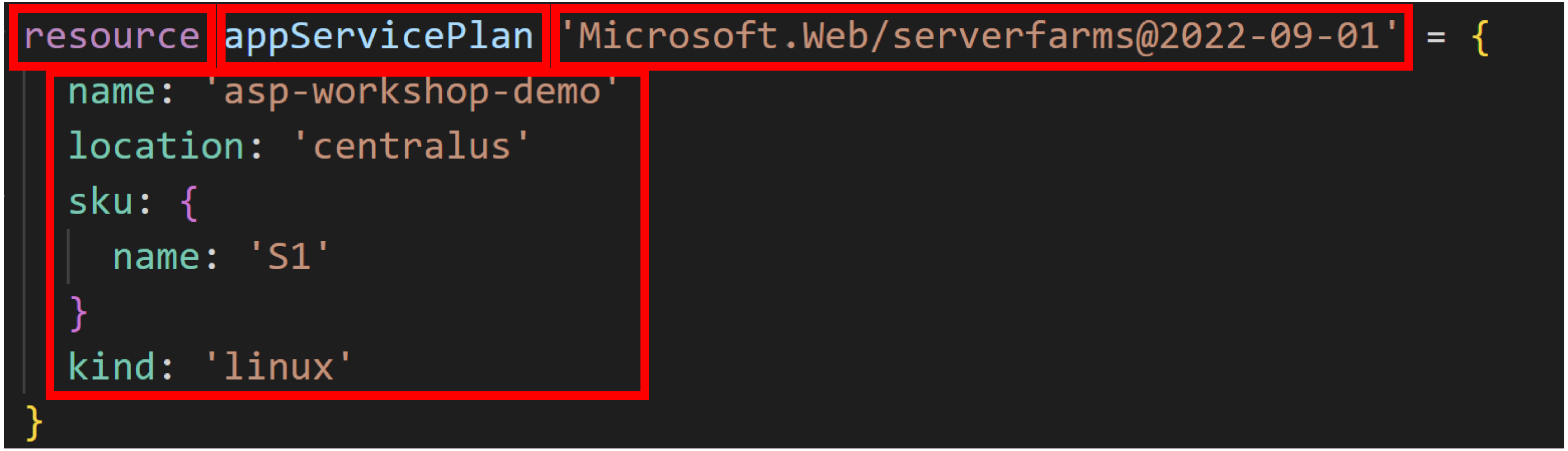
Live Demo

Bicep

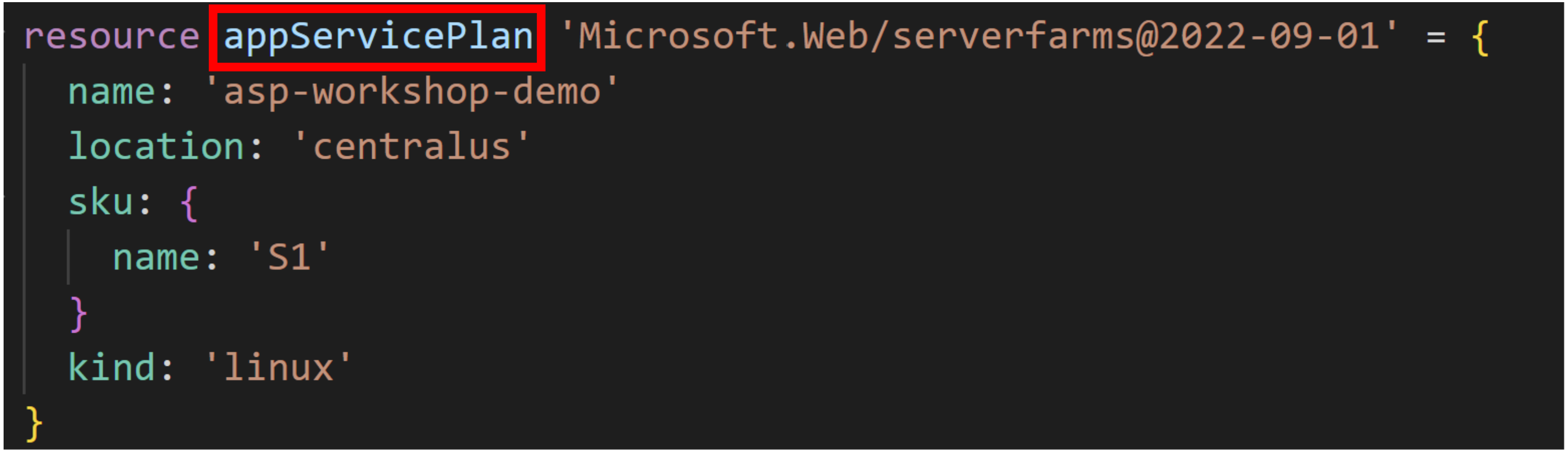
What is Bicep?

- Infrastructure as Code DSL for Azure
- Describe Azure Resources using Bicep’s language
- Stored in version control
- Repeatable
- Auditable
- Manage Configuration Drift
- Bicep runs in incremental mode by default (will not delete)
- No state file required like Terraform does

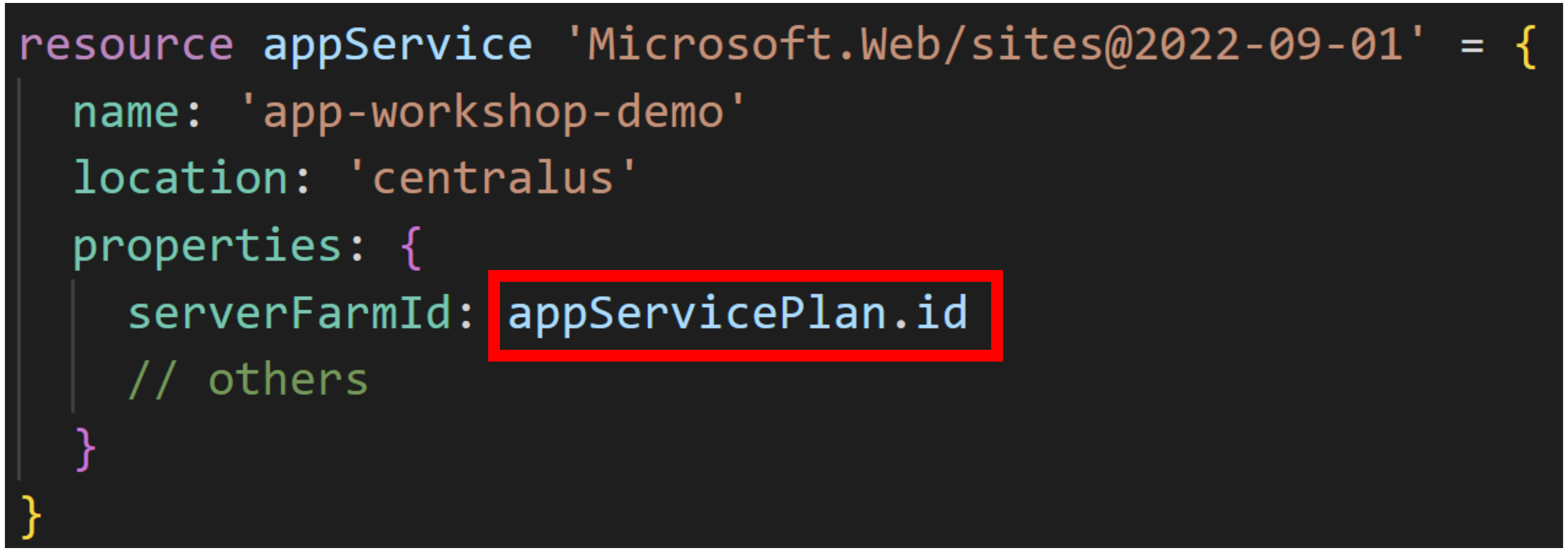
What does Bicep look like?



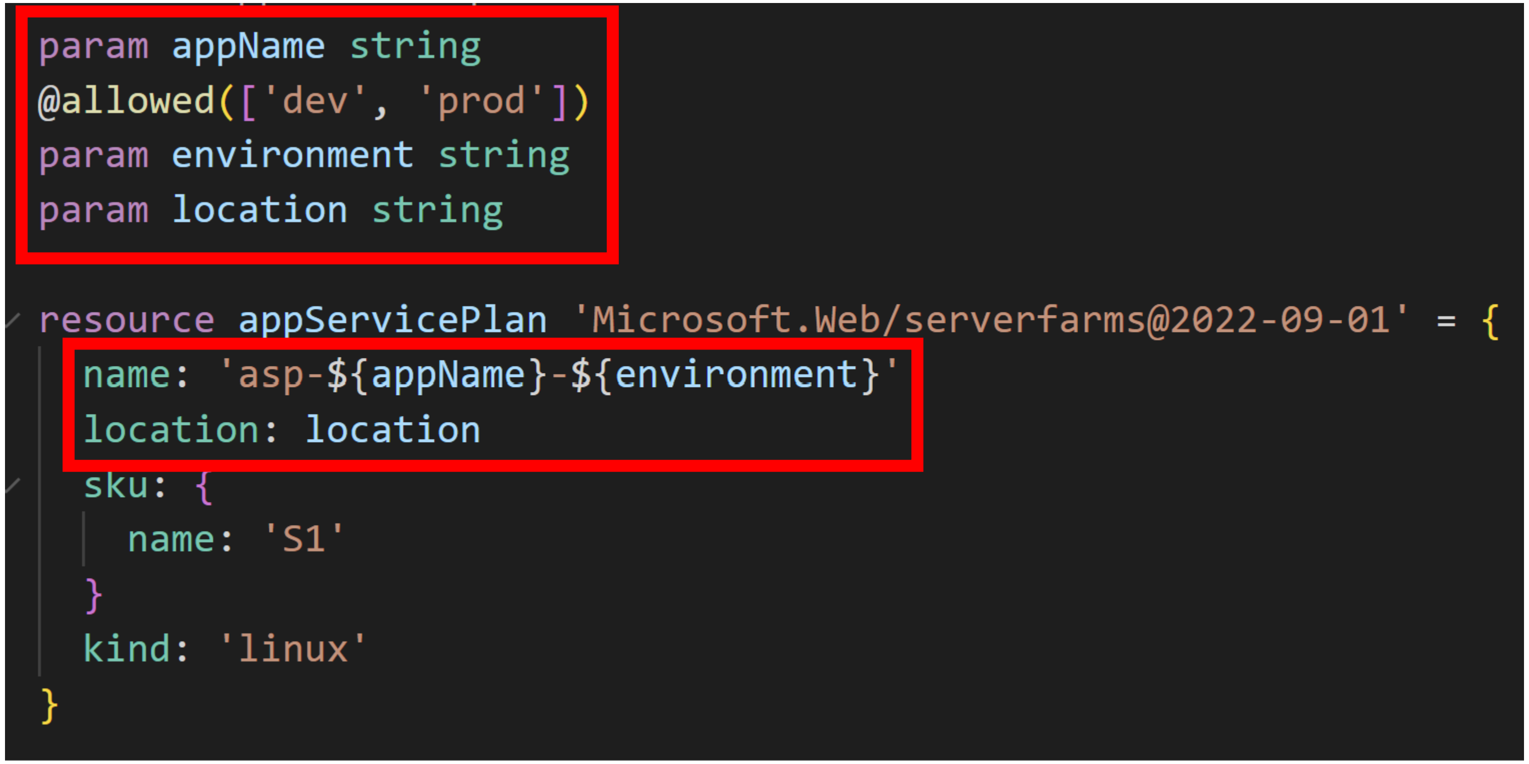
```
resource appServicePlan 'Microsoft.Web/serverfarms@2022-09-01' = {
  name: 'asp-workshop-demo'
  location: 'centralus'
  sku: {
    name: 'S1'
  }
  kind: 'linux'
}
```



```
resource appServicePlan 'Microsoft.Web/serverfarms@2022-09-01' = {
  name: 'asp-workshop-demo'
  location: 'centralus'
  sku: {
    name: 'S1'
  }
  kind: 'linux'
}
```

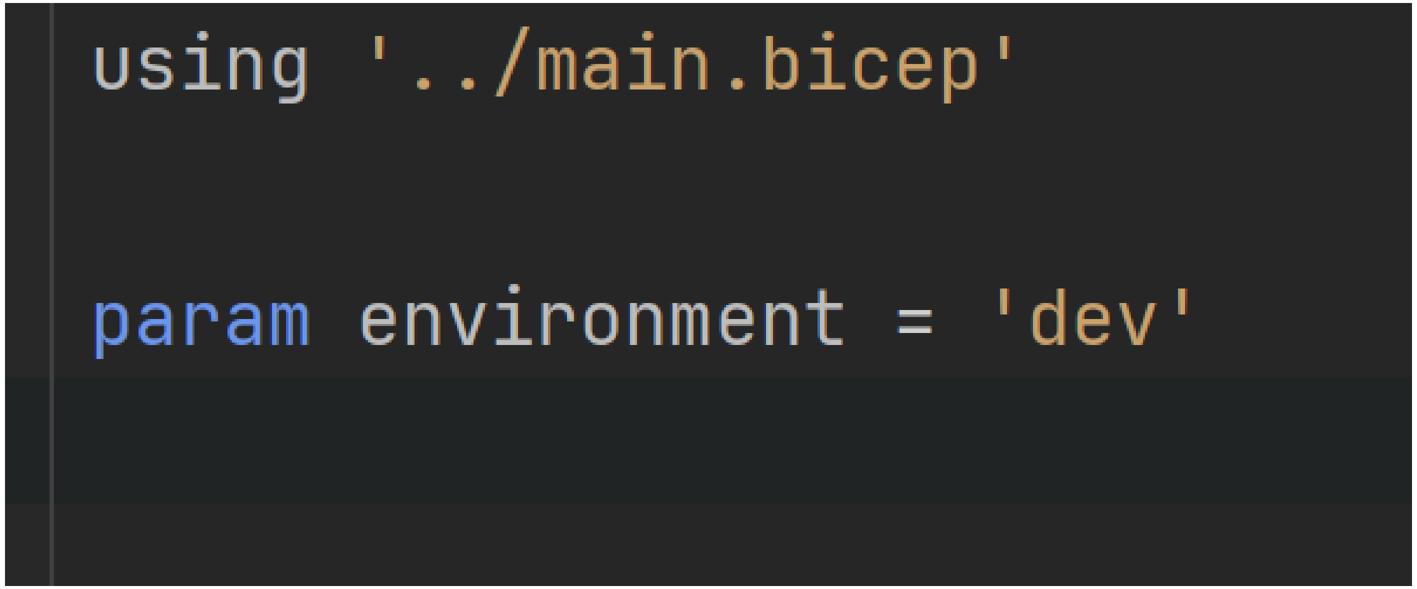


```
resource appService 'Microsoft.Web/sites@2022-09-01' = {
  name: 'app-workshop-demo'
  location: 'centralus'
  properties: {
    serverFarmId: appServicePlan.id
    // others
  }
}
```



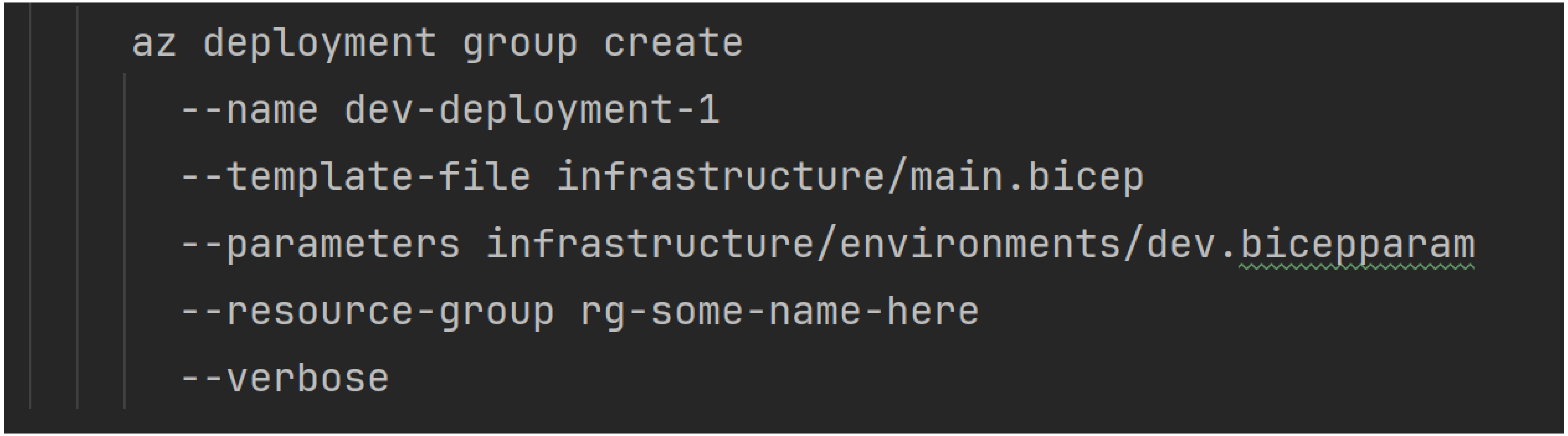
```
param appName string
@allowed(['dev', 'prod'])
param environment string
param location string

resource appServicePlan 'Microsoft.Web/serverfarms@2022-09-01' = {
  name: 'asp-${appName}-${environment}'
  location: location
  sku: {
    name: 'S1'
  }
  kind: 'linux'
}
```

dev.biccepparam file



```
using '../main.bicep'


param environment = 'dev'
```

But how do I deploy it?



```
az deployment group create
  --name dev-deployment-1
  --template-file infrastructure/main.bicep
  --parameters infrastructure/environments/dev.bicepparam
  --resource-group rg-some-name-here
  --verbose
```

Key Concepts

- Resources
- Modules
- Parameters
- .bicepparam
- Outputs
- --whatif

Benefits

- No manual work of configuring in the portal (and repeating for each env)
- Eliminate configuration drift
- Traceability of who, did what, and when
- Give Contributor access to the pipeline – not to individuals

Additional Resources

- Documentation for various Bicep resources:
  - https://learn.microsoft.com/en-us/azure/templates/microsoft.web/sites?pivots=deployment-language-bicep

scottsauber

Live Demo

Hands On

GitHub Actions

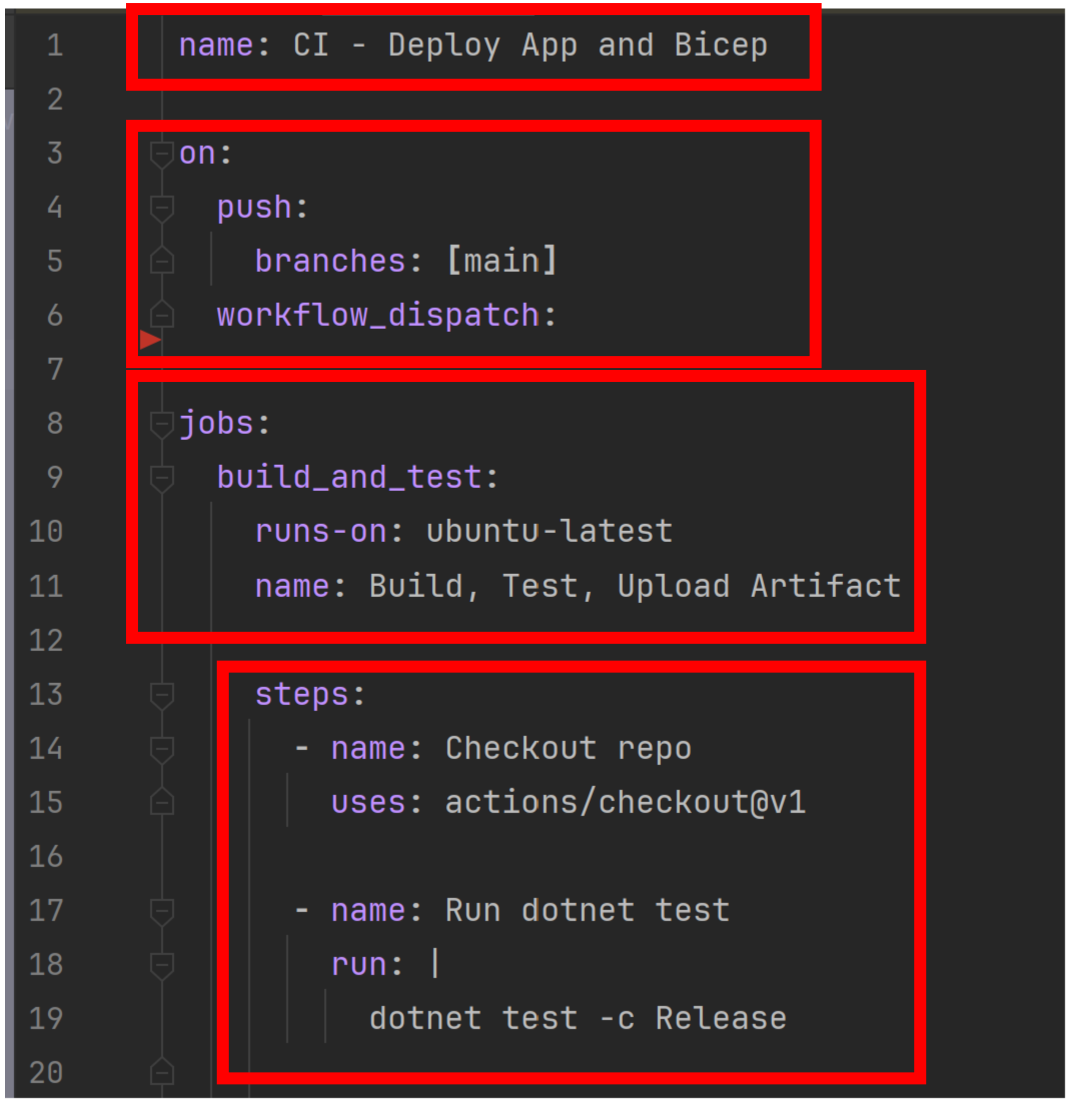
What is GitHub Actions?

- Thing doer on a trigger
- Trigger could be PR, push to main branch, open an issue, etc
- Automatically build and deploys your application
- Including the infrastructure (i.e. Bicep)

Concepts

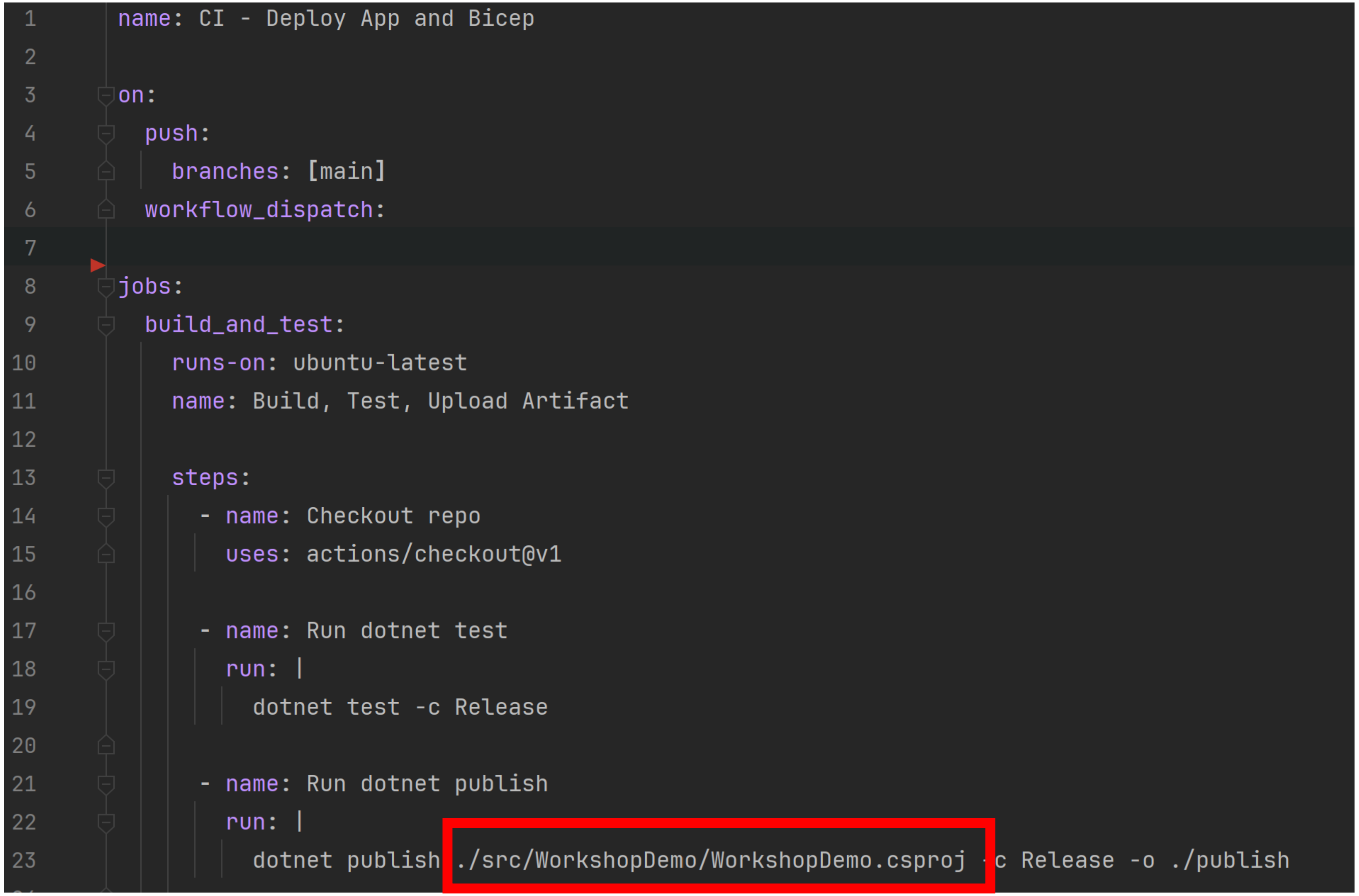
- Workflows
- Triggers
- Jobs
- Steps
- Secrets

What does GitHub Actions Look Like?



```yaml
name: CI - Deploy App and Bicep

on:
  push:
    branches: [main]
  workflow_dispatch:

jobs:
  build_and_test:
    runs-on: ubuntu-latest
    name: Build, Test, Upload Artifact

    steps:
      - name: Checkout repo
        uses: actions/checkout@v1

      - name: Run dotnet test
        run: |
          dotnet test -c Release

```

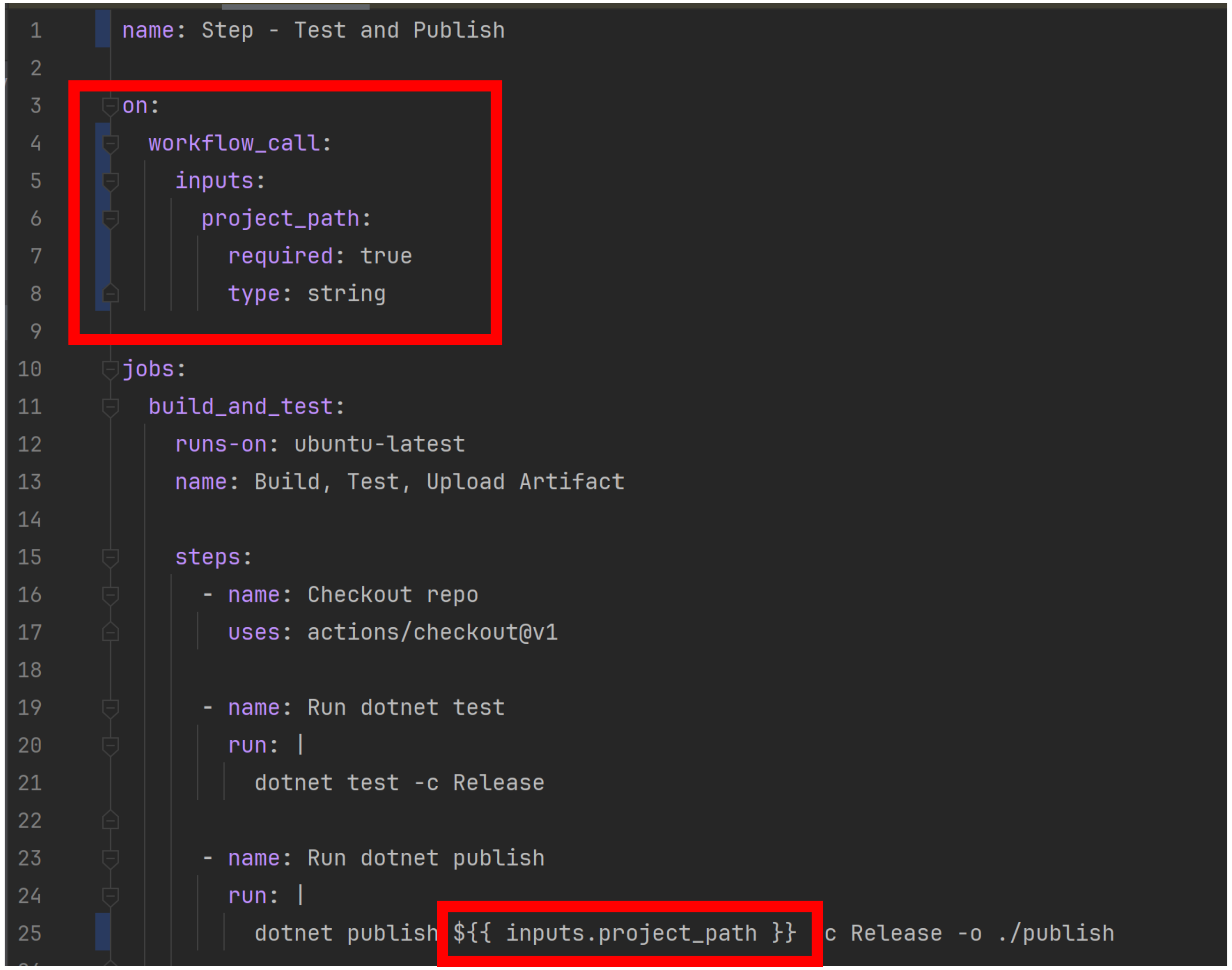
How do I reuse workflows?



```yaml
name: CI - Deploy App and Bicep

on:
  push:
    branches: [main]
  workflow_dispatch:

jobs:
  build_and_test:
    runs-on: ubuntu-latest
    name: Build, Test, Upload Artifact

    steps:
      - name: Checkout repo
        uses: actions/checkout@v1

      - name: Run dotnet test
        run: |
          dotnet test -c Release

      - name: Run dotnet publish
        run: |
          dotnet publish ./src/WorkshopDemo/WorkshopDemo.csproj -c Release -o ./publish
```

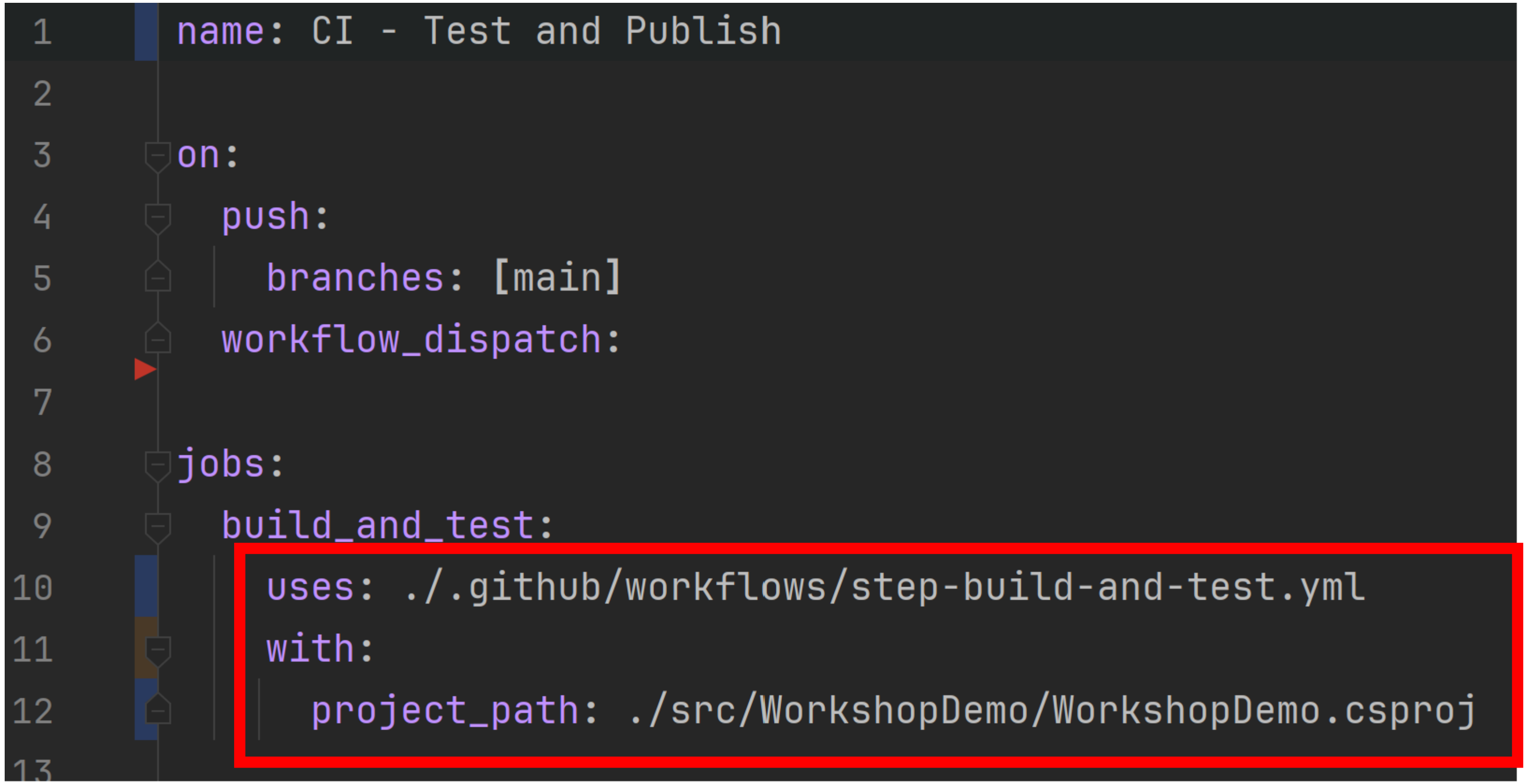
scottsauber

How do I reuse workflows?



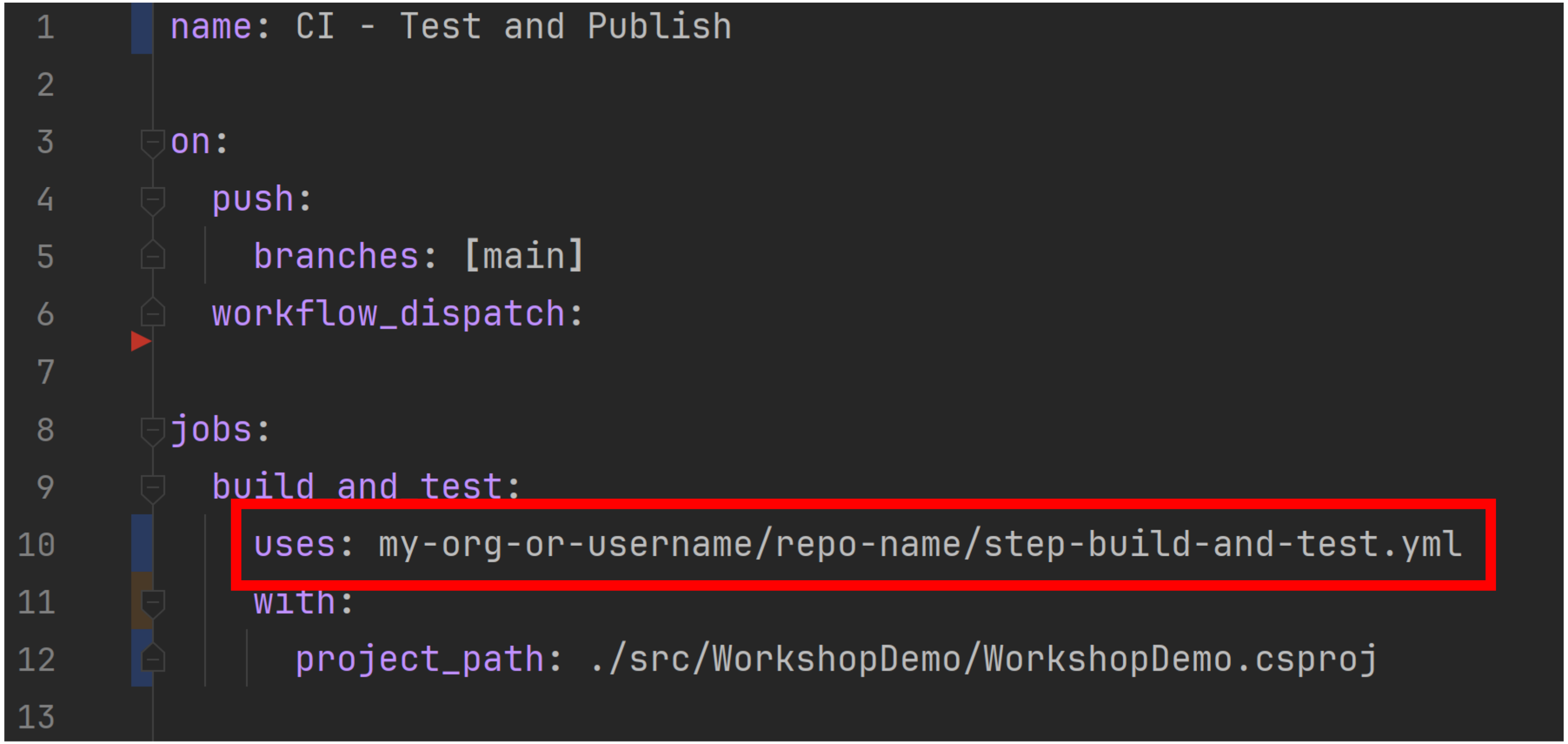
```yaml
name: Step - Test and Publish

on:
  workflow_call:
    inputs:
      project_path:
        required: true
        type: string

jobs:
  build_and_test:
    runs-on: ubuntu-latest
    name: Build, Test, Upload Artifact

    steps:
      - name: Checkout repo
        uses: actions/checkout@v1

      - name: Run dotnet test
        run: |
          dotnet test -c Release

      - name: Run dotnet publish
        run: |
          dotnet publish ${{ inputs.project_path }}  c Release -o ./publish
```

How do I consume this reusable workflow?



```yaml
name: CI - Test and Publish

on:
  push:
    branches: [main]
  workflow_dispatch:

jobs:
  build_and_test:
    uses: ./.github/workflows/step-build-and-test.yml
    with:
      project_path: ./src/WorkshopDemo/WorkshopDemo.csproj
```

How do I consume this from another repo?



```yaml
name: CI - Test and Publish

on:
  push:
    branches: [main]
  workflow_dispatch:

jobs:
  build_and_test:
    uses: my-org-or-username/repo-name/step-build-and-test.yml
    with:
      project_path: ./src/WorkshopDemo/WorkshopDemo.csproj

```

scottsauber

Live Demo

GitHub

Hands On

GitHub

Takeaways

- How to leverage Azure
- Why Infrastructure as Code is useful and how Bicep works
- How GitHub Actions fits into the big picture
- Some takeaway tips even if you had experience with this stuff

Resources

- Slides at scottsauber.com
- https://github.com/scottsauber/workshop-dotnet-azure-github-bicep
    - The “final” branch has the final state of things

Questions?

Contact: ssauber@leantechniques.com

Thanks!