Build Your First CI/CD PIPELINE

DRUPAL CAMP ASHEVILLE

Amber Matz *Developer Advocate*

Tugboat

JULY 11, 2026

Build Your First CI/CD Pipeline

And add a quality check while you're at it!

Amber Matz *Developer Advocate*

DRUPAL CAMP ASHEVILLE

Tugboat

JULY 11, 2026

TODAY'S QUEST

- How I created a sandbox to experiment with a CI/CD pipeline
- A *process* of adding a local CLI tool to a CI/CD check
- 6 essential concepts I learned along the way

DRUPAL CAMP ASHEVILLE

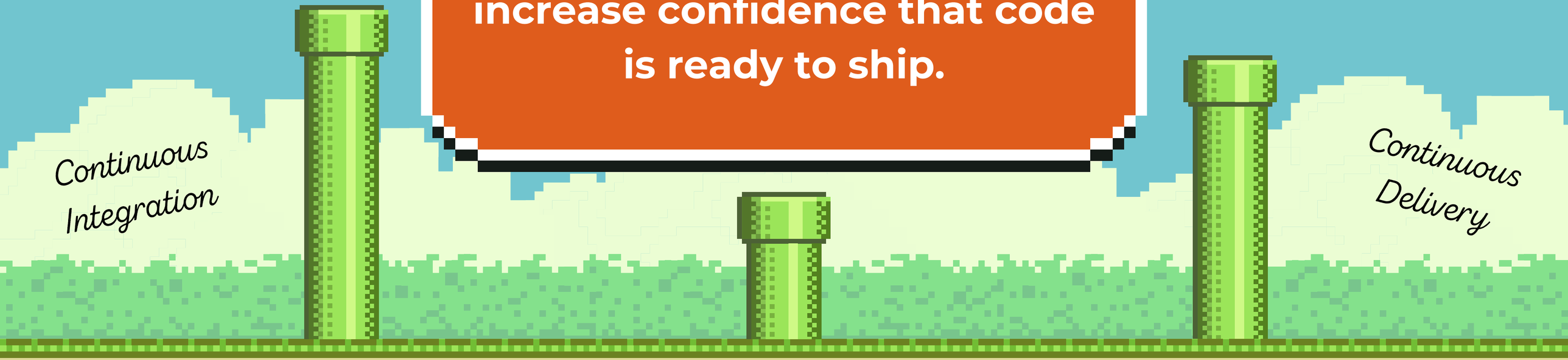
Amber Matz *Developer Advocate*

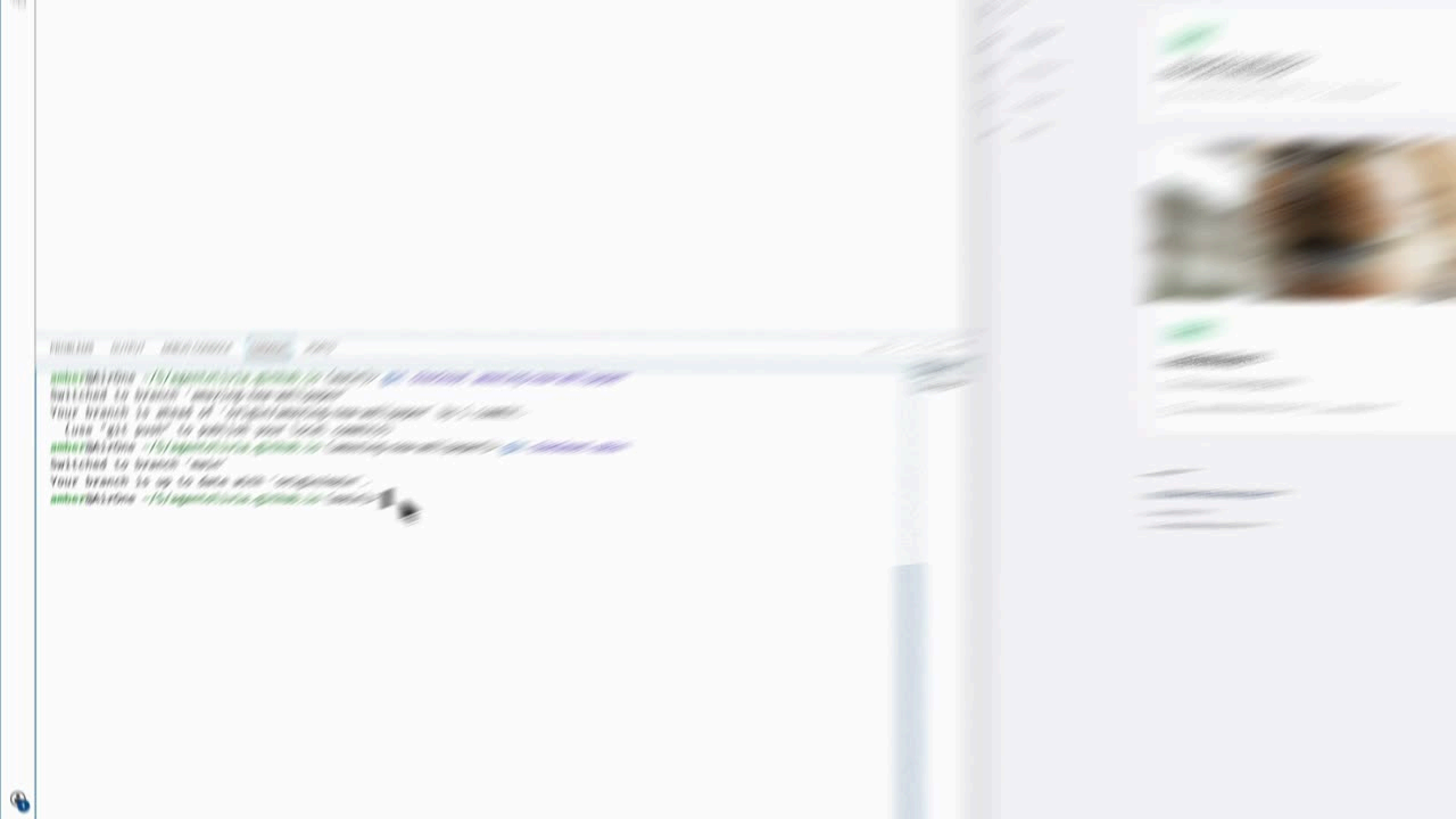
Tugboat

JULY 11, 2026

WHAT'S A CI/CD PIPELINE?

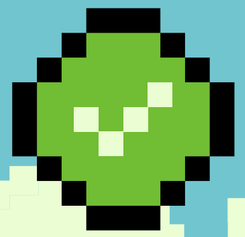
A sequence of triggered automated checks designed to increase confidence that code is ready to ship.

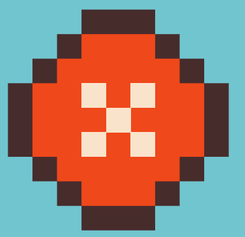
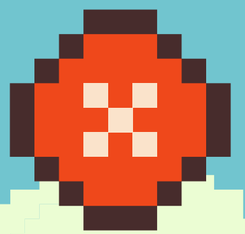




WHERE I STARTED

- Commit code changes to ‘main’ and push to repo
- Create pull request
- Trigger automatic quality checks
- Merge pull request
- Trigger automatic deployment



YOLO commits to main branch

MY CI/CD SANDBOX

Used the template:

CaiJimmy/hugo-theme-stack-starter

Named the repo:

agentolivia.github.io

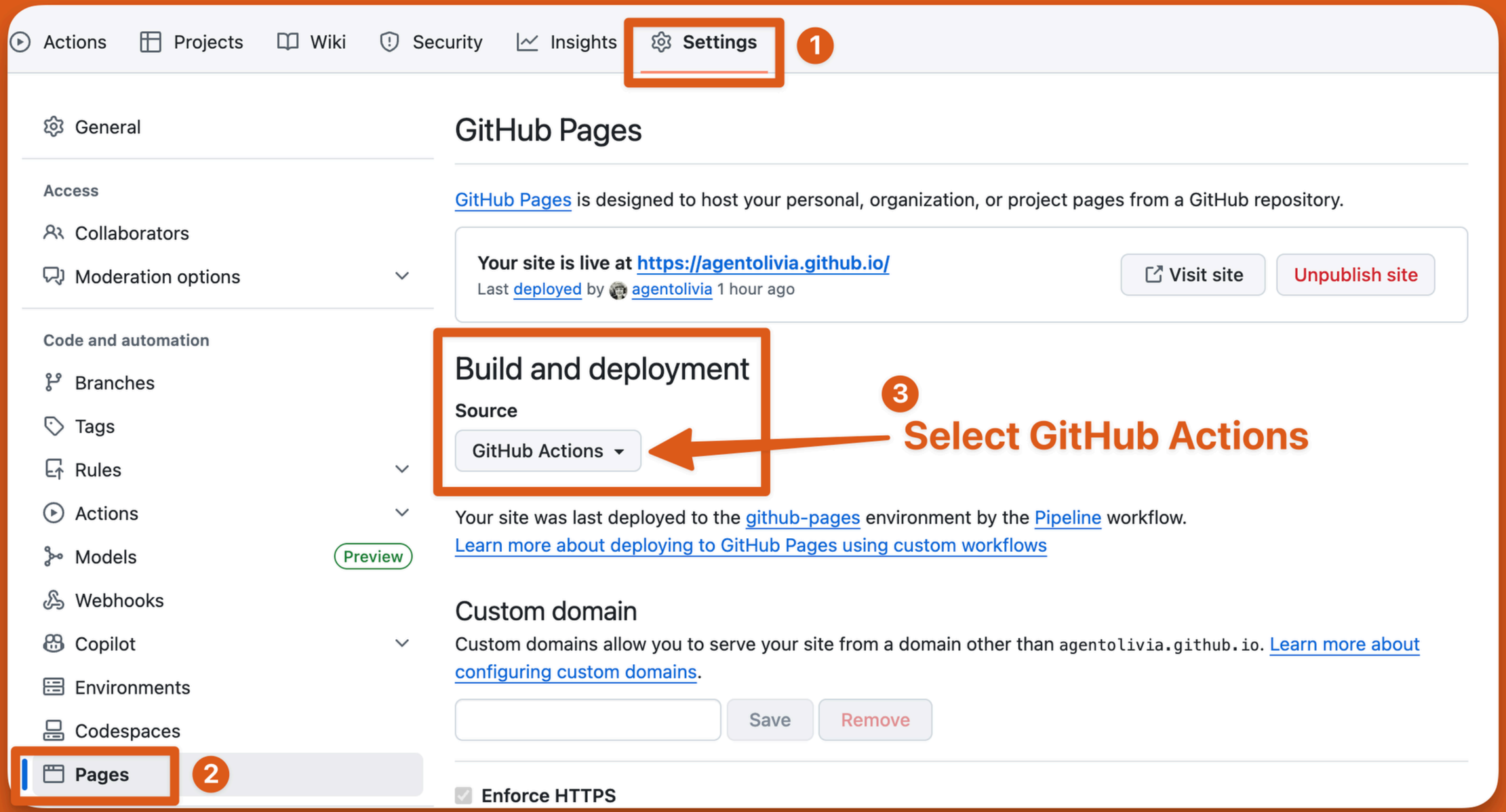
my username on GitHub

Used GitHub Actions:

Settings > Pages > Source:
GitHub Actions

I created and set up a new GitHub repo that builds and deploys a Hugo site to GitHub Pages using GitHub Actions.

Settings > Pages > Source: GitHub Actions



SET UP LOCAL

Code

```
git clone
...
```

Environment

```
brew install hugo
brew install go
```

Build and run

```
hugo serve
```

Visit localhost:1313

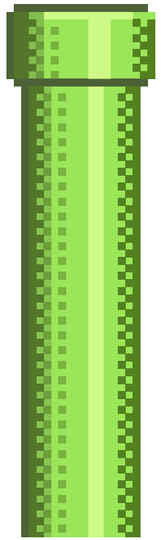
Get the code and your environment set up to build and run the site on your computer.

In VS Code

1. **git checkout step-1-first-deploy**
2. **Open .github/workflows/deploy.yml**
3. **Open Timeline to view edit and commit history.**

Filter files...

- .devcontainer
- .github/workflows
  - deploy.yml
  - update-theme.yml
- .gitignore
- .vscode
- LICENSE
- README.md
- assets
- config/_default
- content
- go.mod
- go.sum



35 files changed +831 -0 lines changed

Top

Search within code

.github/workflows/deploy.yml +105

```
@@ -0,0 +1,105 @@
+ name: Build and deploy
+ on:
+   push:
+     branches:
+       - main
+       - master
+   workflow_dispatch:
+ permissions:
+   contents: read
+   pages: write
+   id-token: write
+ concurrency:
+   group: pages
+   cancel-in-progress: false
+ defaults:
+   run:
+     shell: bash
+ jobs:
+   build:
+     runs-on: ubuntu-latest
+     env:
+       DART_SASS_VERSION: 1.97.1
+       GO_VERSION: 1.25.5
+       HUGO_VERSION: 0.154.2
+       NODE_VERSION: 24.12.0
+       TZ: Europe/Oslo
+     steps:
+       - name: Checkout
```

PROGRESS REPORT

- Commit code changes to ‘main’ and push to repo
- Create pull request
- Trigger automatic quality checks
- Merge pull request to ‘main’ branch
- Trigger automatic deployment

What I got with a new repo based on the Hugo Stack theme starter template repository.

Initial commit · agentolivia/ag

Not That Kind of Agent

Gemini

github.com/agentolivia/agentolivia.github.io/commit/b924e34

Filter files...

- .devcontainer
- .github/workflows
  - deploy.yml
  - update-theme.yml
- .gitignore
- .vscode
- LICENSE
- README.md
- assets
- config/_default
- content
- go.mod
- go.sum

35 files changed +831 -0 lines changed

Top

Search within code

.github/workflows/deploy.yml +105

The Trigger

```
@@ -0,0 +1,105 @@
+ name: Build and deploy
+ on:
+   push:
+     branches:
+       - main
+       - master
+   workflow_dispatch:
+ permissions:
+   contents: read
+   pages: write
+   id-token: write
+ concurrency:
+   group: pages
+   cancel-in-progress: false
+ defaults:
+   run:
+     shell: bash
+ jobs:
+   build:
+     runs-on: ubuntu-latest
+     env:
+       DART_SASS_VERSION: 1.97.1
+       GO_VERSION: 1.25.5
+       HUGO_VERSION: 0.154.2
+       NODE_VERSION: 24.12.0
```

A PROCESS

1. Install a CLI tool locally
2. Run it and tune config/ignore until it's useful
3. Add command(s) to 'scripts' key in package.json
4. Fix files until tool returns 0 failures
5. Commit changes to version control

Set up and test a tool locally before adding it to pipeline.

1ST QUALITY CHECK

Install

```
npm install
--save-dev
prettier
```

Create Config + Ignore

```
.prettierrc
.prettierignore
```

Add command to scripts

```
"format:check":
"prettier --check ."
```

package.json
“scripts”

Install a CLI tool locally. I chose Prettier.

TEST IT LOCALLY

Run it. Refine config until it's useful.

Run the command

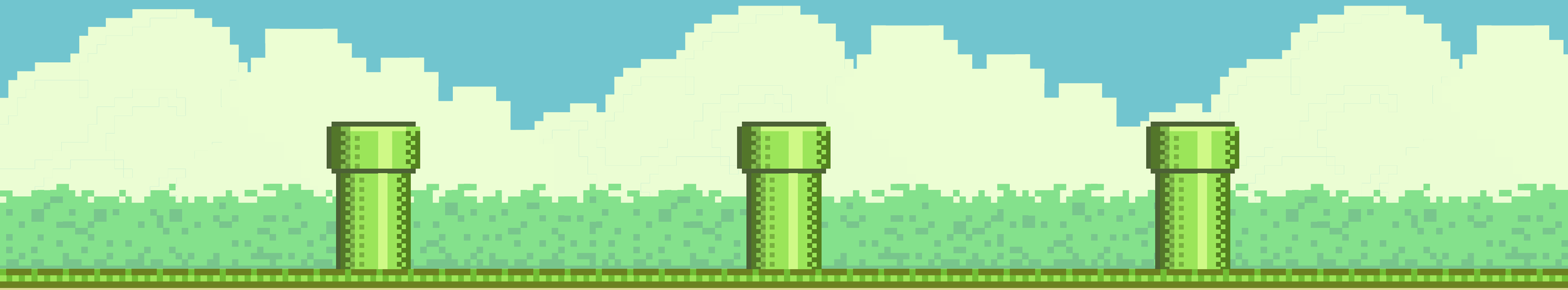
```
npm run
format:check
```

Modify tool config files

```
.prettierrc
.prettierignore
```

What did I need?

```
Node.js
```



ADD A WORKFLOW

Create the file

`.github/workflows (.yml extension required)`

Find an Action that fits

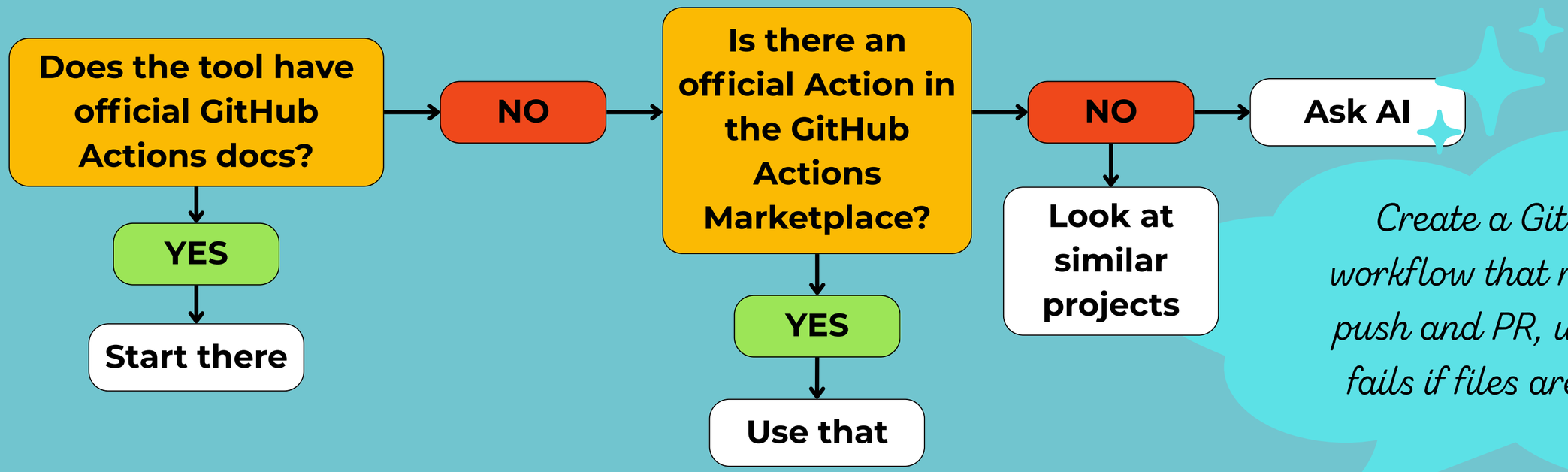
Consult tool docs & GitHub Action Marketplace

Install IDE extension

In VS Code, install GitHub Actions extension

Create a workflow that will run the command.

WORKFLOW STARTER



Create a GitHub Actions workflow that runs Prettier on push and PR, uses Node, and fails if files aren't formatted

In VS Code

1. **git checkout step-2-add-prettier-to-lint**
2. **Open .github/workflows/lint.yml**
3. **Open Timeline to view edit and commit history.**

← Lint

Quality check updates #19

Summary

All jobs

lint

Run details

Usage

Workflow file

lint

succeeded last week in 8s

- Set up job
- Check out code
- Set up Node.js
- Install dependencies
- Run Prettier check

```
1  ▶ Run npm run format:check
4
5  > agentolivia.github.io@1.0.0 format:check
6  > prettier --check .
7
8  Checking formatting...
9  All matched files use Prettier code style!
```

- Run Markdown check
- Post Set up Node.js
- Post Check out code

← Build and deploy

Merge pull request #3 from agentolivia/quality-check-updates #18

Summary

All jobs

- build
- deploy

Run details

- Usage
- Workflow file

Triggered via push last week

agentolivia pushed e96d766 main

Status

Success

Total duration

43s

Artifacts

1

deploy.yml

on: push

Artifacts

Produced during runtime

Name	Size	Digest
github-pages Expired	687 KB	sha256:46da32ba08a3cb620df7f4c1aa39058a56b5

PROGRESS REPORT

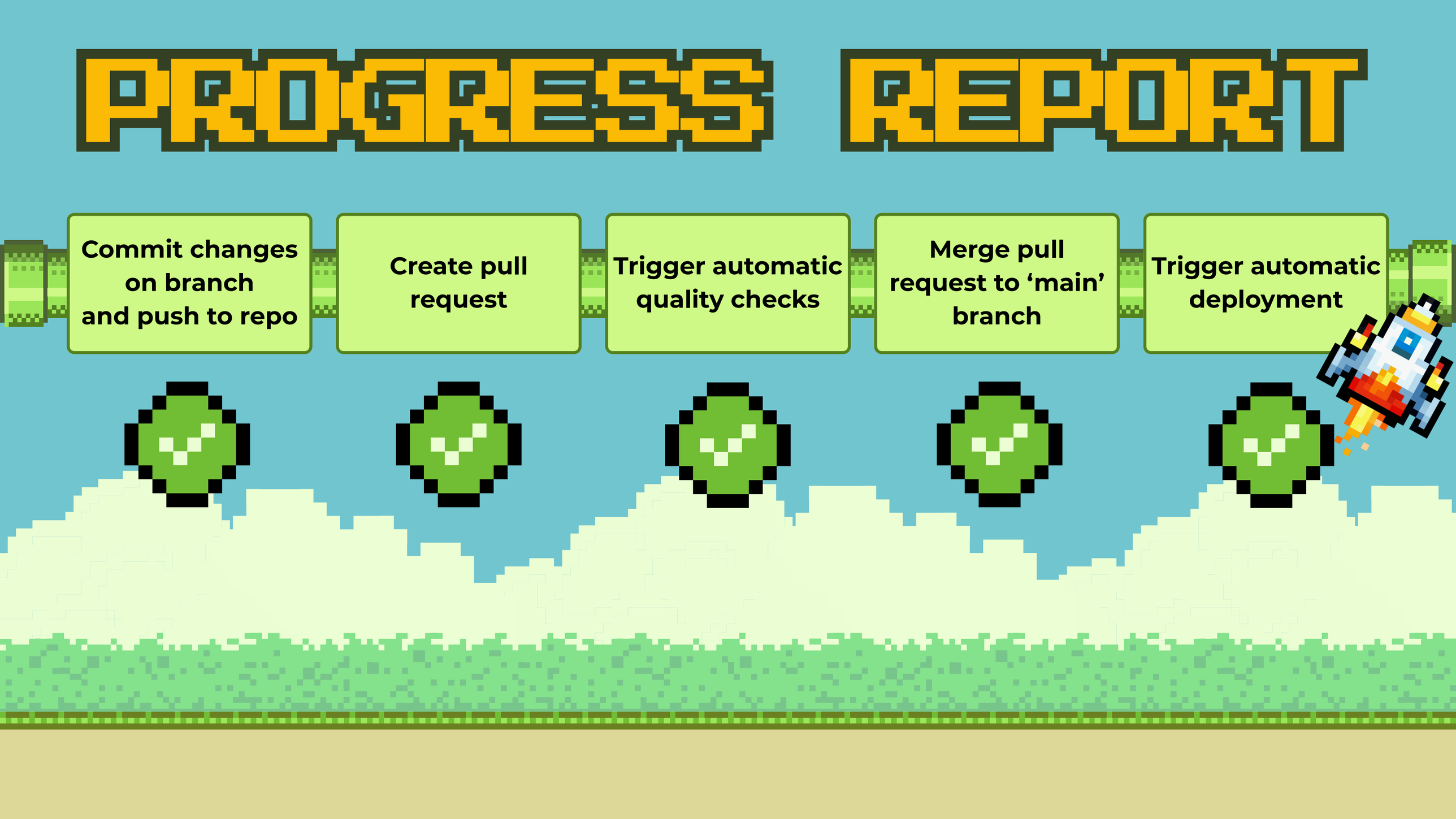
Commit changes on branch and push to repo

Create pull request

Trigger automatic quality checks

Merge pull request to ‘main’ branch

Trigger automatic deployment



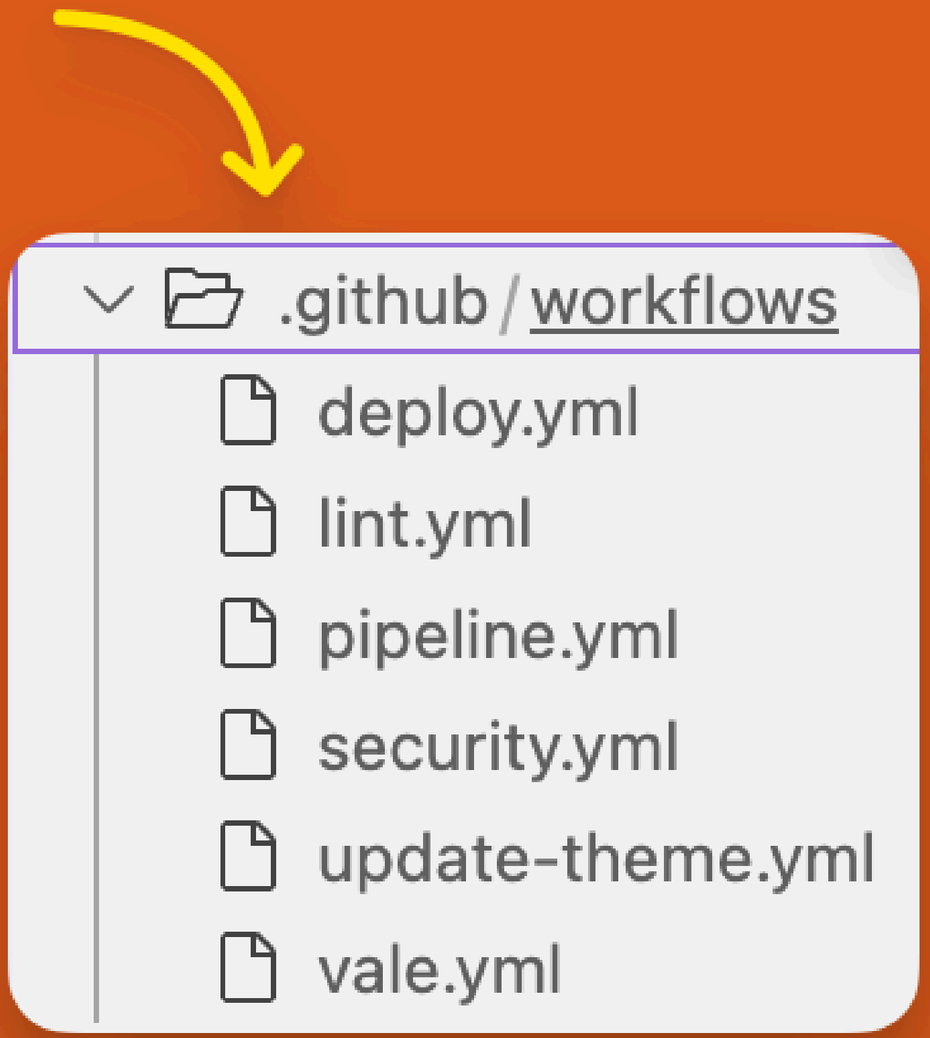
CONCEPT CHECK-IN

THE CONFIG FILE

A version-controlled file in your repo that defines an automation

Describes the environment, triggers, jobs, and steps and how they connect into a pipeline.

Located and named predictably per platform.


.github / workflows
deploy.yml
lint.yml
pipeline.yml
security.yml
update-theme.yml
vale.yml

THE ENVIRONMENT

```yaml
jobs:
  build:
    runs-on: ubuntu-latest
    env:
      DART_SASS_VERSION: 1.97.1
      GO_VERSION: 1.25.7
      HUGO_VERSION: 0.155.3
      NODE_VERSION: 24.12.0
      TZ: America/Los_Angeles
```

.github/workflows/deploy.yml

build

succeeded last week in 27s

Search logs

Set up job

```
1  Current runner version: '2.331.0'
2  ▶Runner Image Provisioner
9  ▼Operating System
10  Ubuntu
11  24.04.3
12  LTS
13 ▶Runner Image
18 ▶GITHUB_TOKEN Permissions
22 Secret source: Actions
23 Prepare workflow directory
24 Prepare all required actions
25 Getting action download info
26 Download action repository 'actions/checkout@v5' (SHA:93cb6efe18208431cddfb8368fd83d5badbf9bfd)
27 Download action repository 'actions/setup-go@v5' (SHA:40f1582b2485089dde7abd97c1529aa768e1baff)
28 Download action repository 'actions/setup-node@v4' (SHA:49933ea5288caeca8642d1e84afbd3f7d6820020)
29 Download action repository 'actions/configure-pages@v5' (SHA:983d7736d9b0ae728b81ab479565c72886d7745b)
30 Download action repository 'peaceiris/actions-hugo@v3' (SHA:75d2e84710de30f6ff7268e08f310b60ef14033f)
31 Download action repository 'actions/cache@v4' (SHA:0057852bfaa89a56745cba8c7296529d2fc39830)
32 Download action repository 'actions/upload-pages-artifact@v3'
   (SHA:56afc609e74202658d3ffba0e8f6dda462b719fa)
33 Getting action download info
34 Download action repository 'actions/upload-artifact@v4' (SHA:ea165f8d65b6e75b540449e92b4886f43607fa02)
35 Complete job name: build
```

```yaml
jobs:
  build:
    runs-on: ubuntu-latest
    env:
      DART_SASS_VERSION: 1.97.1
      GO_VERSION: 1.25.7
      HUGO_VERSION: 0.155.3
      NODE_VERSION: 24.12.0
      TZ: America/Los_Angeles
```

.github/workflows/deploy.yml

.github > workflows > deploy.yml

```yaml
name: Build and deploy
on:
  workflow_call:
permissions:
  contents: read
  pages: write
  id-token: write
concurrency:
  group: pages
  cancel-in-progress: false
defaults:
  run:
    shell: bash
jobs:
  build:
    runs-on: ubuntu-latest
    env: ⋯
    steps: ⋯
  deploy:
    environment: ⋯
    runs-on: ubuntu-latest
    needs: build
    steps: ⋯
```

← Build and deploy

Merge pull request #3 from agentolivia/quality-check-updates #18

Summary

All jobs

- build
- deploy

Run details

- Usage
- Workflow file

Triggered via push last week

agentolivia pushed e96d766 main

Status

Success

Total duration

43s

Artifacts

1

deploy.yml

on: push

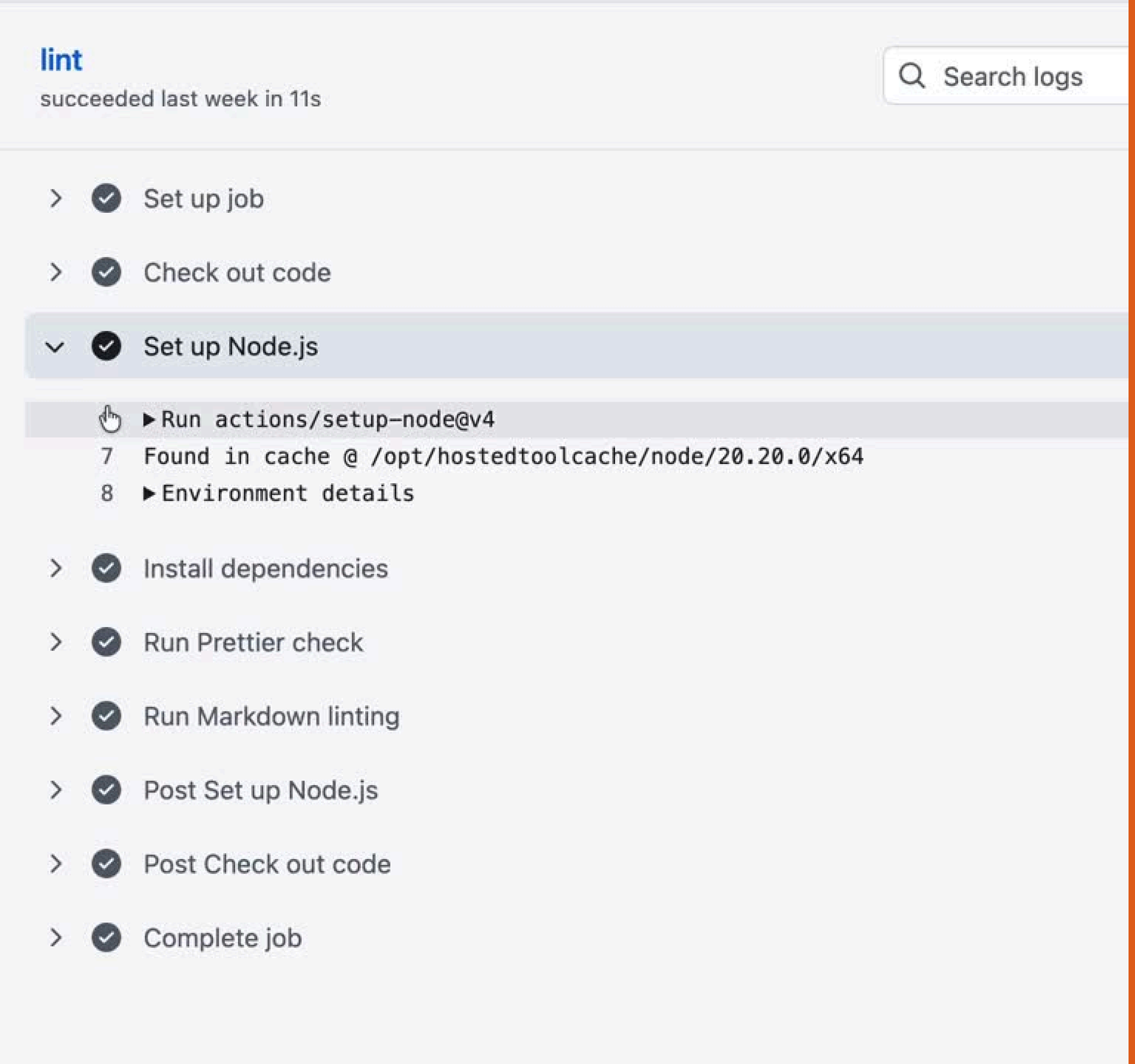
Artifacts

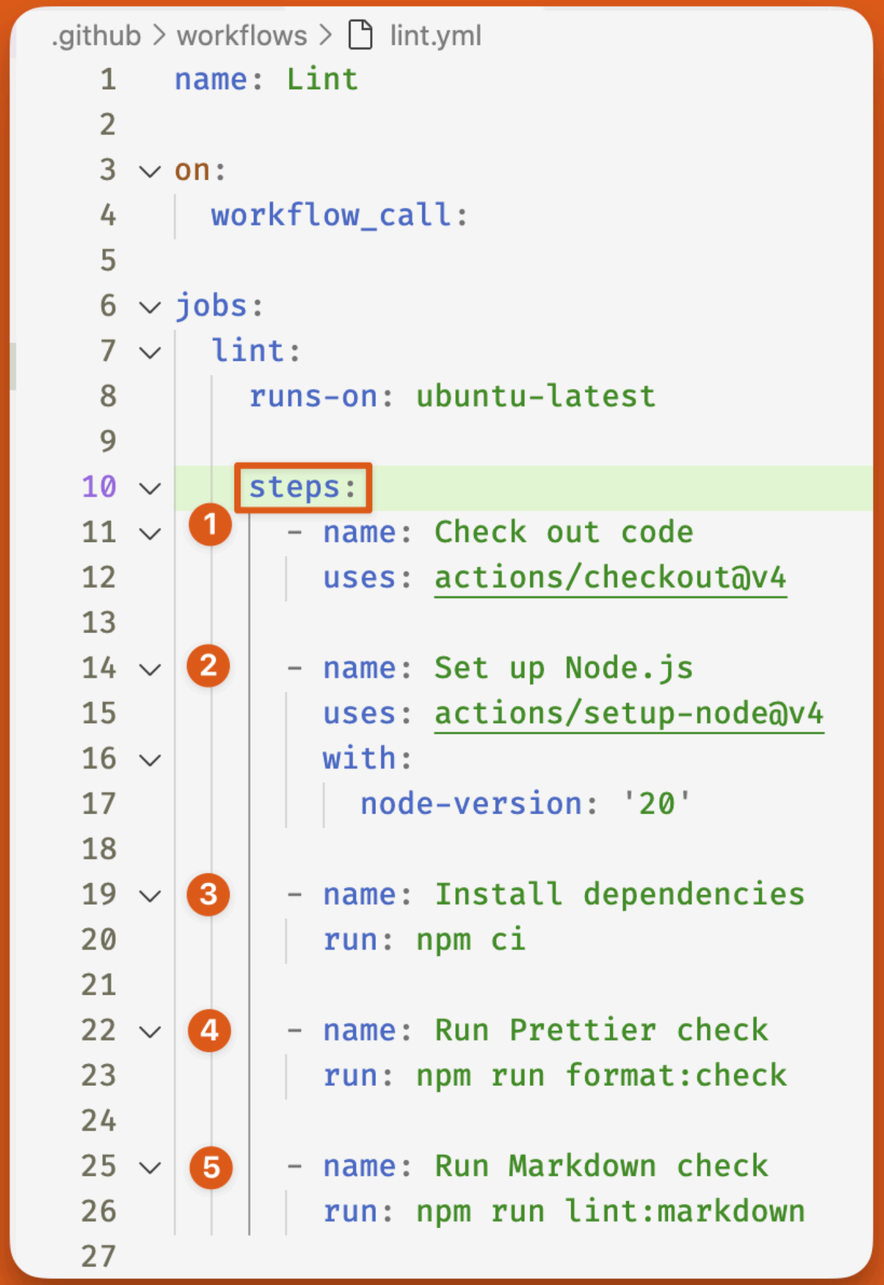
Produced during runtime

Name	Size	Digest
github-pages Expired	687 KB	sha256:46da32ba08a3cb620df7f4c1aa3

.github > workflows > lint.yml

```yaml
name: Lint

on:
  workflow_call:

jobs:
  lint:
    runs-on: ubuntu-latest

    steps:
      - name: Check out code
        uses: actions/checkout@v4

      - name: Set up Node.js
        uses: actions/setup-node@v4
        with:
          node-version: '20'

      - name: Install dependencies
        run: npm ci

      - name: Run Prettier check
        run: npm run format:check

      - name: Run Markdown check
        run: npm run lint:markdown
```


lint
succeeded last week in 11s
Search logs
Set up job
Check out code
Set up Node.js
Run actions/setup-node@v4
7 Found in cache @ /opt/hostedtoolcache/node/20.20.0/x64
8 Environment details
Install dependencies
Run Prettier check
Run Markdown linting
Post Set up Node.js
Post Check out code
Complete job


.github > workflows > lint.yml
1 name: Lint
2
3 on:
4 workflow_call:
5
6 jobs:
7 lint:
8 runs-on: ubuntu-latest
9
10 steps:
11 1 - name: Check out code
12 uses: actions/checkout@v4
13
14 2 - name: Set up Node.js
15 uses: actions/setup-node@v4
16 with:
17 node-version: '20'
18
19 3 - name: Install dependencies
20 run: npm ci
21
22 4 - name: Run Prettier check
23 run: npm run format:check
24
25 5 - name: Run Markdown check
26 run: npm run lint:markdown
27

TRIGGERS

```yaml
on:
  push:
    branches:
      - main
  pull_request:
    branches:
      - main
```

Triggered via push now

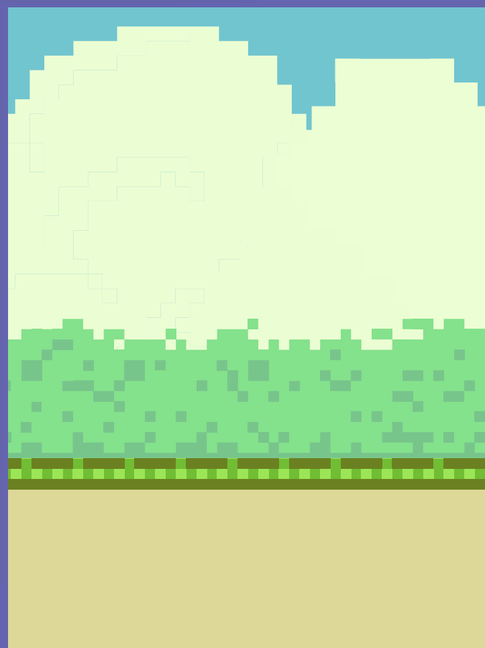
agentolivia pushed 49cedaa main

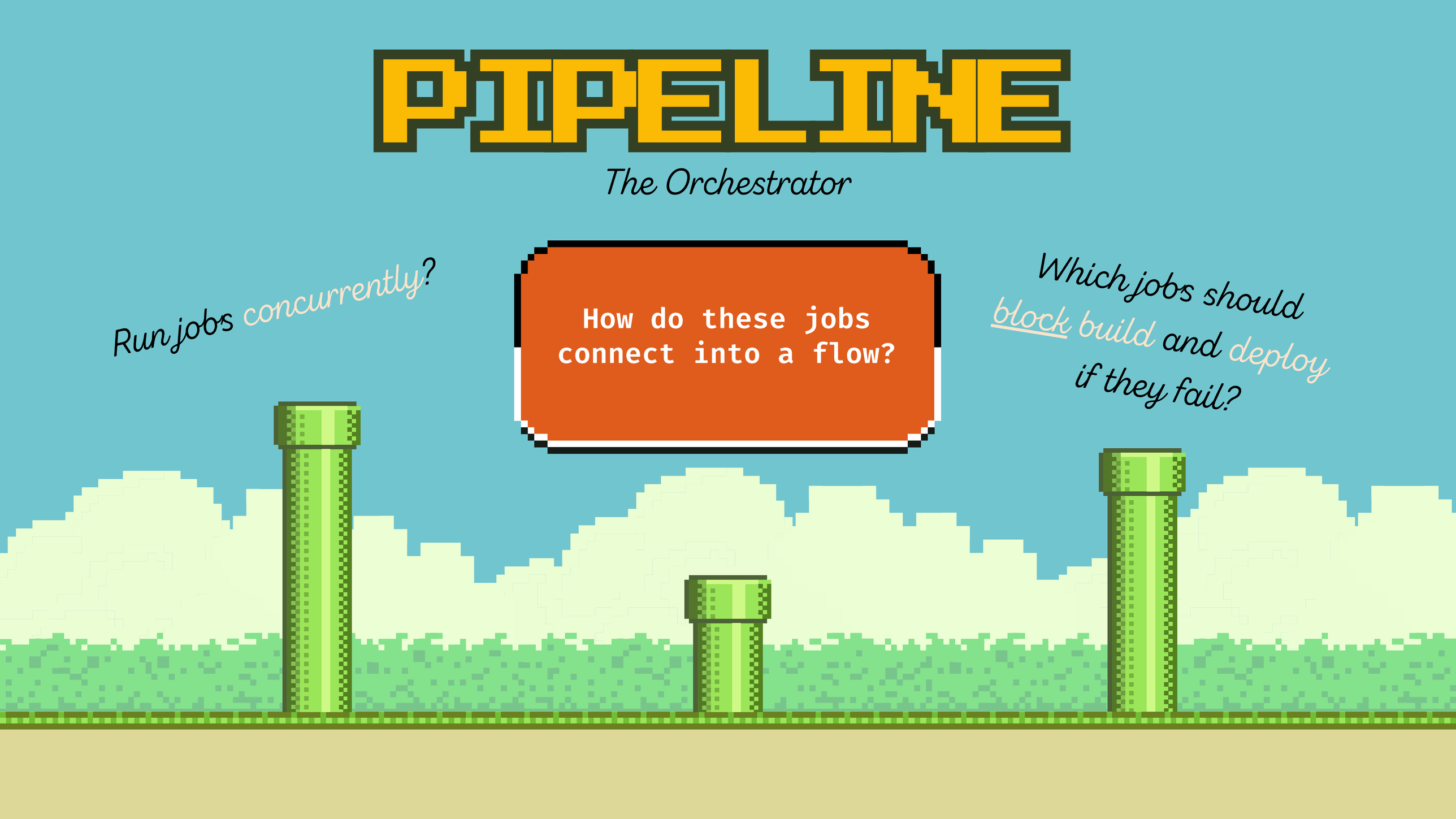
Status	Total duration	Artifacts
In progress	–	–

pipeline.yml

on: push

- lint / lint 10s
- security / security 9s
- deploy / build 8s
- deploy / deploy




PIPELINE
The Orchestrator
Run jobs concurrently?
How do these jobs connect into a flow?
Which jobs should block build and deploy if they fail?

PIPELINE EVOLUTION

deploy.yml

standalone workflow *triggered* by **on: push**

(provided by Hugo Stack Starter repo template)

Added pipeline.yml

orchestrator of child workflows

Updated child workflows

triggers changed from **on: push** / **workflow_dispatch** to **on: workflow_call**

Hardened deploy gate

added least-privilege **permissions** on *deploy* job

Added **if:** condition so *deploy* only fires on **push-to-main**, **not PRs**

Extracted security into its own workflow

npm audit **extracted from deploy** workflow and **added to new security workflow**

security job failures now **block deployment** independently

REPO-WIDE HARDENING

decreased **timeout-minutes**

- a hung job can block other workflows or **cost money / quota**
- detect regressions (you know it shouldn't take this long)

set explicit permissions in each workflow

read only, except *deploy* needs some write perm)

added **persist-credentials: false** to *checkout* actions

- prevents checkout **token** from lingering in the runner (in *.git/config*)
- **removes a credential-exfiltration vector**
- these jobs never use git after checkout anyway

version modernization

- updated env to Node 24
- bumped all action versions to Node 24

supply-chain modernization

- **replaced 3rd-party action**, peaceiris/actions-hugo, with **direct Hugo download**
- added **SHA-256 checksum verification**

CONCEPT: GITHUB TOKENS

GITHUB_TOKEN

automatically-generated credential scoped to your repo

Use principle of least prilvilege in each workflow when setting permissions.

A GITHUB_TOKEN's default permissions are set at the repo/org level. Could be set to read+write to almost everything

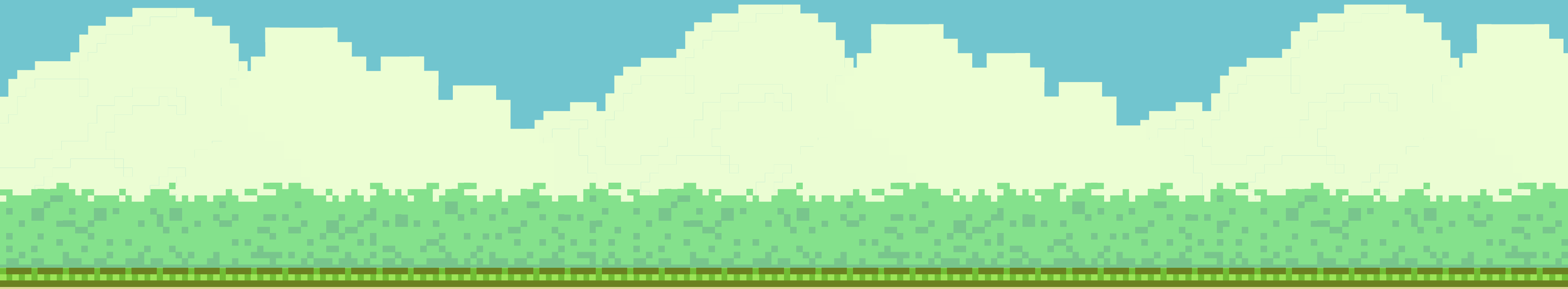
Checkout persists GITHUB_TOKEN to disk (.git/config) as auth header for entire job.

permissions: contents: read

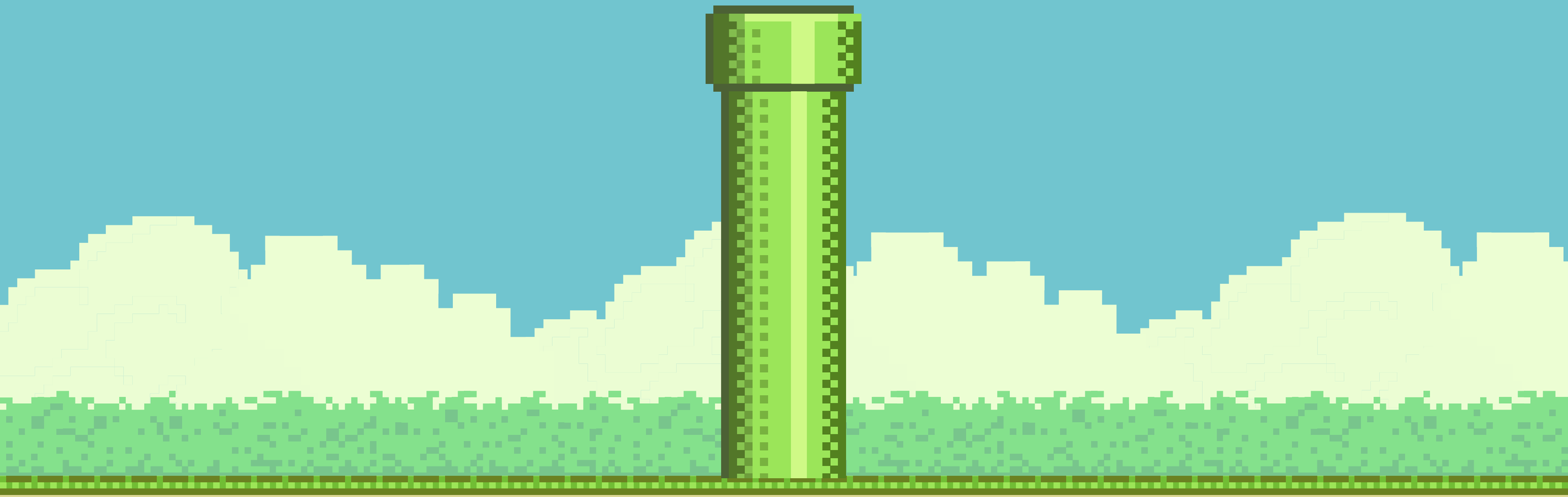
persist-credentials: false

FIRST CHECK

What is **one tool** you **love to use** in your IDE that **you could add to a CI/CD pipeline** that would run **automatically** on every pull request?



FINAL BOSS DEMO




fish
hugo

PROGRESS REPORT

Commit changes on branch and push to repo

Create pull request

Trigger automatic quality checks

Merge pull request to ‘main’ branch

Trigger automatic deployment

Now with orchestration and security hardening!

RECAP

- How I created a sandbox to learn CI/CD concepts
- A process of adding a local CLI tool to a CI/CD check
- Learned A LOT!

Config File!

Environment!

Jobs!

Steps!

Triggers!

The Pipeline FTW!

Security hardening!

RESOURCES

My repo:
https://github.com/agentolivia/agentolivia.github.io

GitHub Actions Marketplace
https://github.com/marketplace?type=actions

GitHub Actions Docs
https://docs.github.com/en/actions

Go forth and...

Build Your First CI/CD PIPELINE

And add a quality check while you're at it!

Let's connect!

Amber Matz *Developer Advocate*

DRUPAL CAMP ASHEVILLE

JULY 11, 2026

Tugboat

https://www.linkedin.com/in/amberhimesmatz/