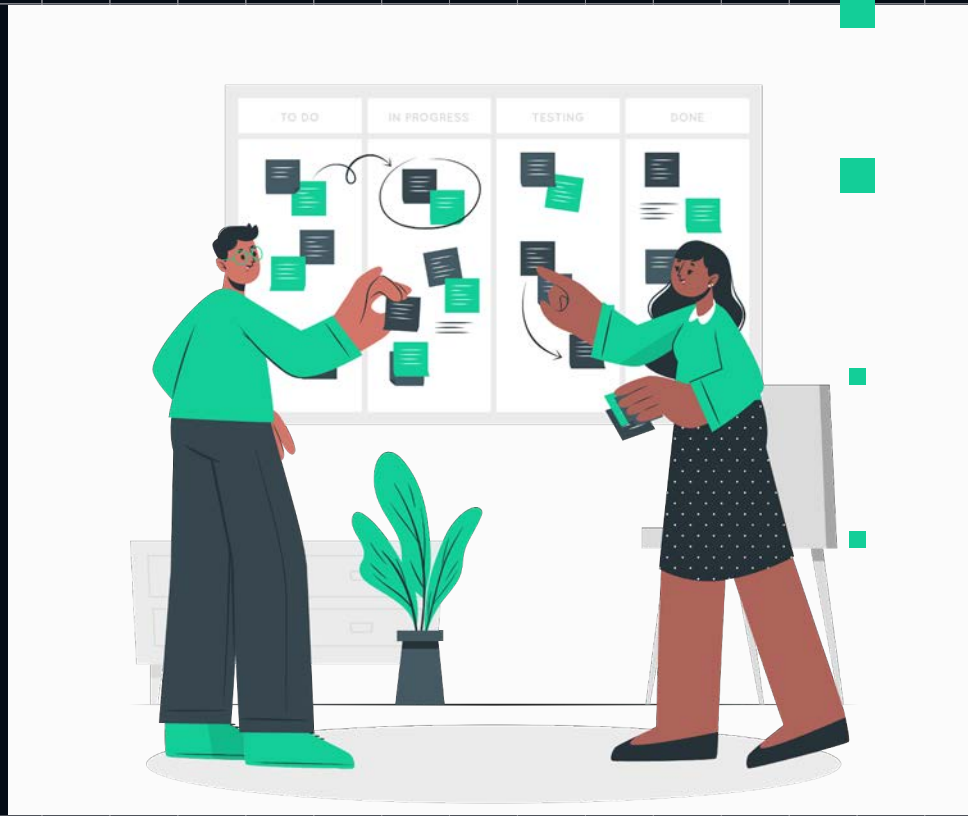
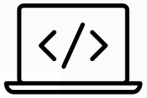
Building a CI/CD Pipeline

Byron Sommardahl

This Segment



Continuous Delivery



Coding

Code Review

Merge/
Integrate

Compile/Lint

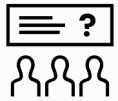
Run Unit
Tests

Run
Integration
Tests

Continuous Delivery

Verify Dependency Security

Deploy to Test Environment



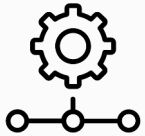
Team Test

Deploy to Client Test Environment

Client Test

Create Infrastructure

Continuous Delivery



Configure Infrastructure

Deploy to Production

Smoke Test in Production

Rollbacks

Promoting to Production

Celebration

Continuous Delivery

Continuous Everything!

Continuous Delivery

Continuous Integration

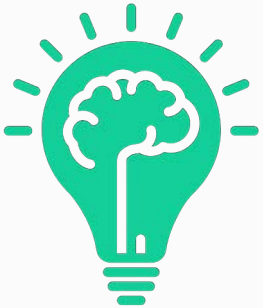
Continuous Deployment

Continuous Everything!

Continuous
Delivery

Continuous
Integration

Continuous
Deployment



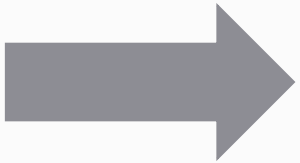
Continuous Delivery

Continuous
Delivery

Continuous
Integration

Continuous Delivery

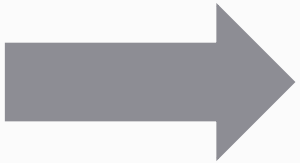
Continuous Integration



Continuous Deployment

Continuous Delivery

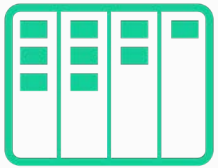
Continuous Integration



Continuous Deployment

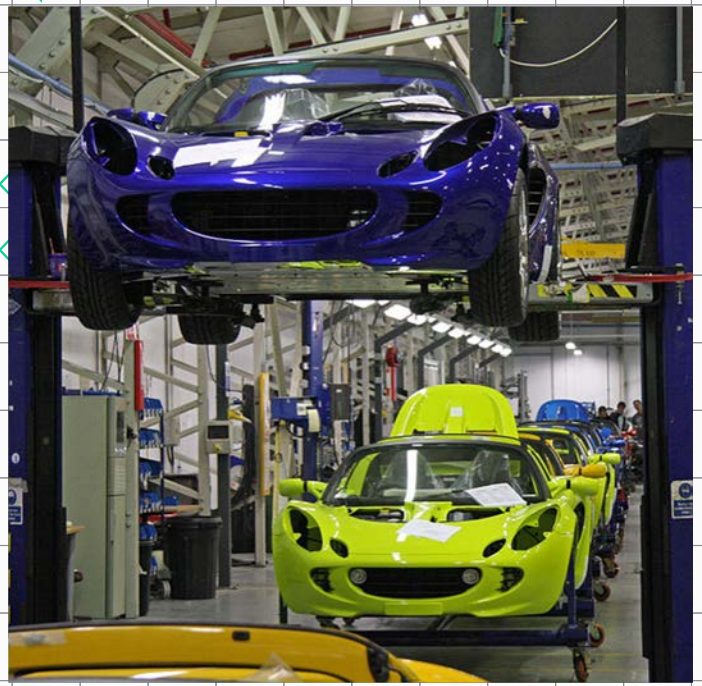
Continuous Delivery

Continuous Integration



Development Process

Continuous Deployment



Toyota Manufacturing

Continuous improvement is a process, continuously and intentionally moving in the direction of “True North”.

Toyota Manufacturing

Continuous improvement is a process, continuously and intentionally moving in the direction of “True North”.

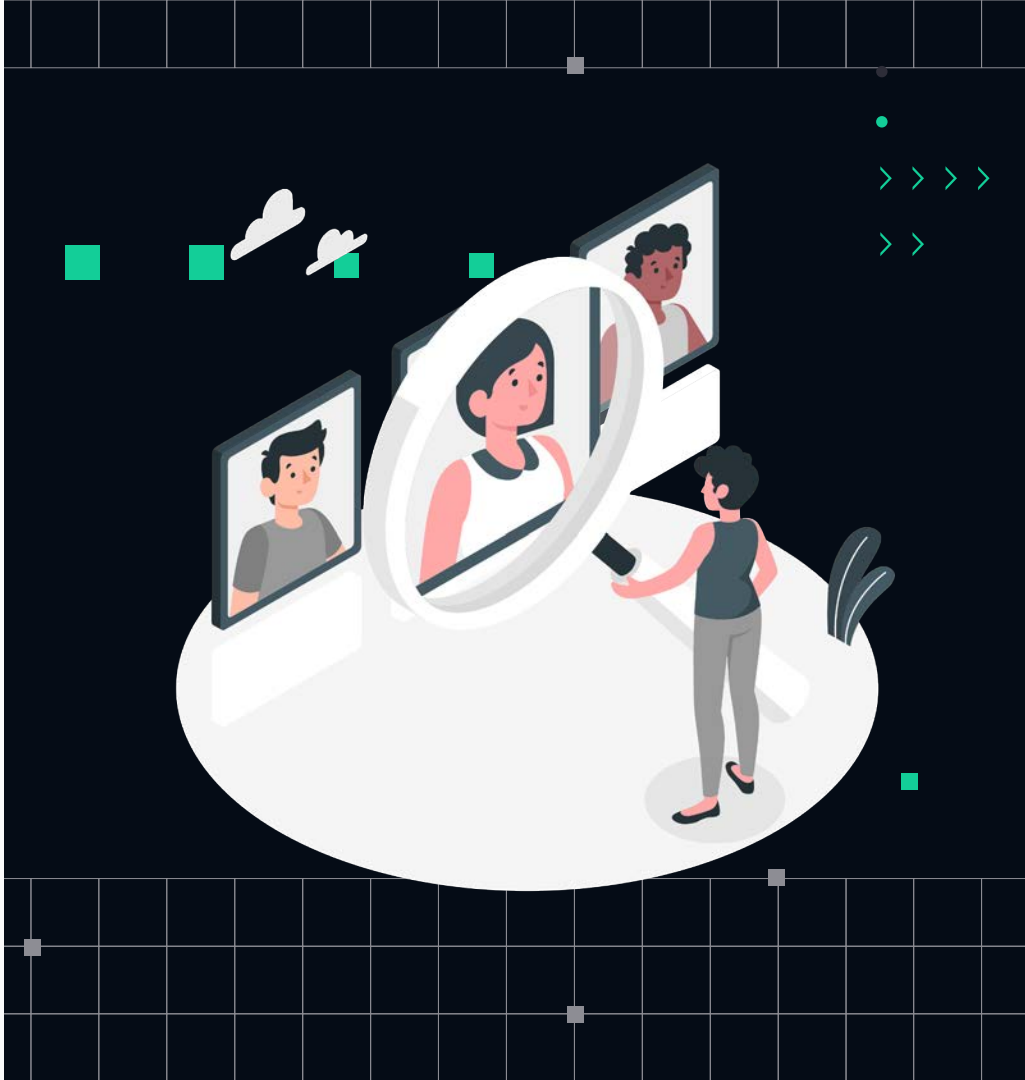
Influencing Culture

10

1 million tiny, consistent, intentional pushes.

“Tell me how you measure me and I will tell you how I will behave.”

- Eli Goldratt



Common Metrics:

- Mean Time to Recovery (MTTR)
- Lead time
- Delivery rate
- Cycle Time
- Test Coverage
- Bug Escape Rate
- Etc

So what?? Who cares?

Tech People

Business People

So what?? Who cares?

Tech People

- Quality of life
- Pride in work
- Desire to help organization

Business People

So what?? Who cares?

Tech People

- Quality of life
- Pride in work
- Desire to help organization

Business People

- Better bottom-line
- Competitive in market
- Less turn-over

Business Value

Increase Revenue

Increased profits, more sales, new features.

Protect Revenue

Keep revenue flowing, preserve customer engagement.

Reduce Costs

Lower server costs, less manpower.

Avoid Costs

Reduce costs, cut expenses.

Influential Metrics

Influential Metrics

Influential Metrics

Influential Metrics

Influential Metrics

bit.ly/devops-quiz

Take the quiz. See how your organization measures up!

Student Poll

According to the “DORA DevOps Quick Check” (bit.ly/devops-quiz), how does your organization rank?

A. Low
B. Medium
C. High
D. Elite

10 Minute Break

Building a CI/CD Pipeline

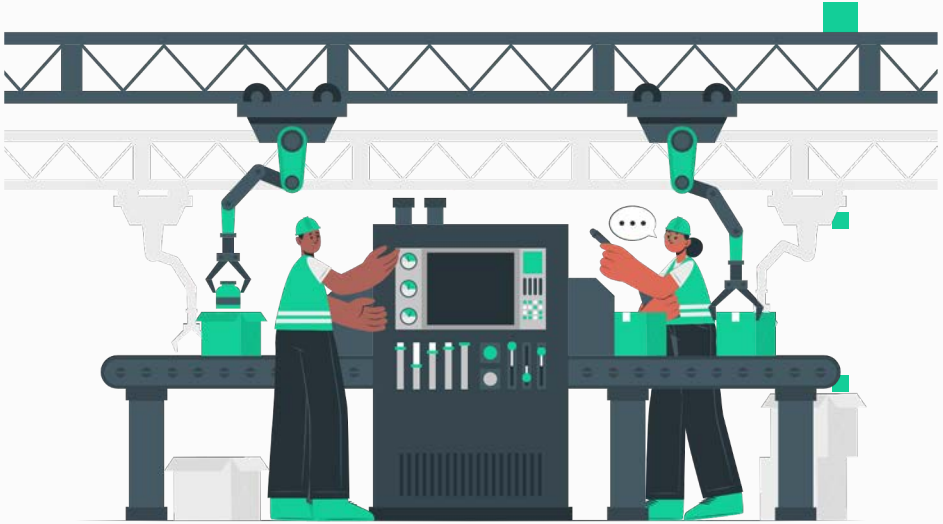
Byron Sommardahl

This Segment

01 **Benign Scripts**
Maintainable scripts that “do no harm”.

02 **YAML Notation**
How to write value YAML.

03 **Vendor Syntax**
Navigating the differences.



Benign Scripts

A necessary evil, but don’t necessarily need to be evil.


{
"name": "codex-academy-website",
"version": "0.1.0",
"private": true,
"scripts": {
"serve": "cross-env NODE_ENV=development vue-cli-service serve",
"build": "vue-cli-service build",
"test:unit": "vue-cli-service test:unit",
"lint": "vue-cli-service lint",
"deploy": "gh-pages -d dist",
"test": "jest --coverage",
"predeploy": "cross-env NODE_ENV=production npm run build",
"storybook:build": "vue-cli-service storybook:build -c config/storybook",
"storybook": "vue-cli-service storybook:serve -p 6006 -c config/storybook",
"smoke": "ts-node e2e/smoke.e2e.ts"
}
}
npm

Benign Scripting

Script Files

Slice scripts into logical groups using files.

Benign Scripting

Script Files

Slice scripts into logical groups using files.

One Language

Choose a language and stick with it to keep your team together.

Benign Scripting

Script Files

Slice scripts into logical groups using files.

One Language

Choose a language and stick with it to keep your team together.

SRP

Do one thing and do it really well!

Benign Scripting

Script Files

Slice scripts into logical groups using files.

One Language

Choose a language and stick with it to keep your team together.

SRP

Do one thing and do it really well!

Fail Fast, Fail Hard

Embrace failure and bring it to the forefront.

YAML?

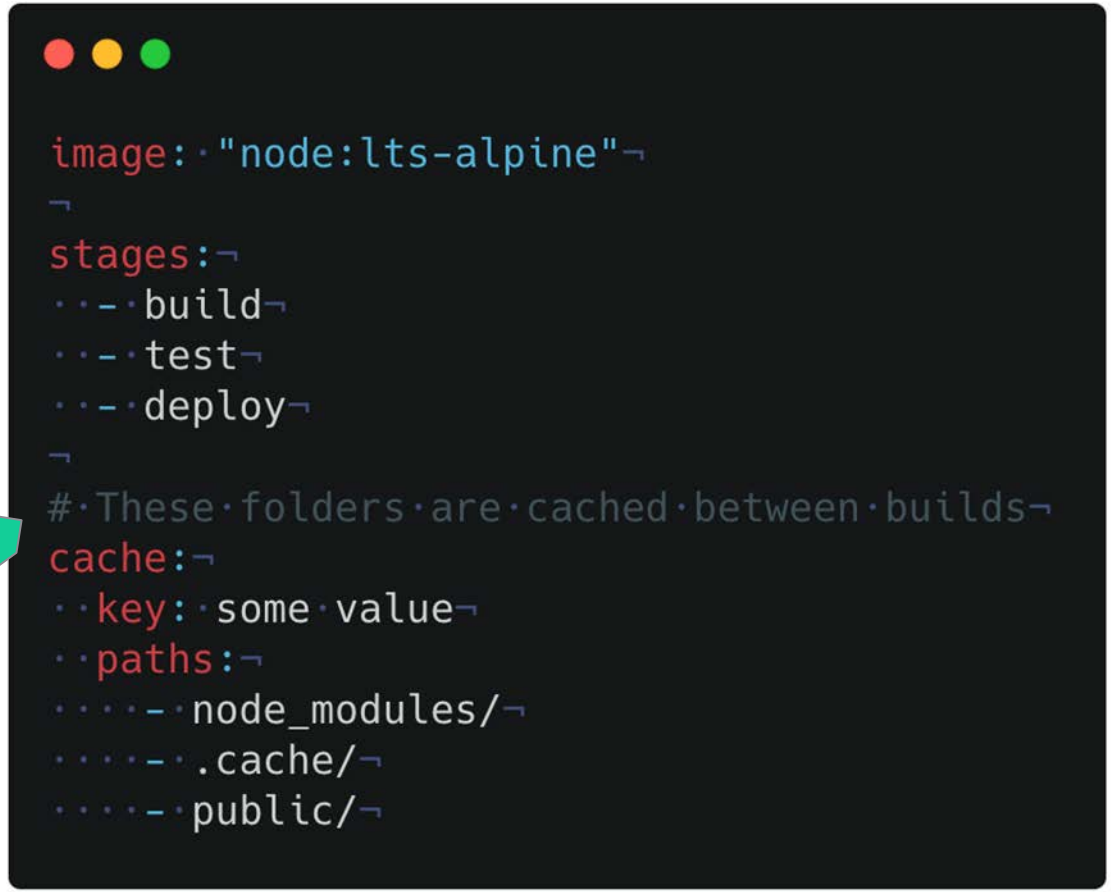
“Human-readable data-serialization language”

YAML Spacing


image: "node:lts-alpine"
stages:
- build
- test
- deploy
These folders are cached between builds
cache:
key: some value
paths:
- node_modules/
- .cache/
- public/

YAML Keys

YAML Comments

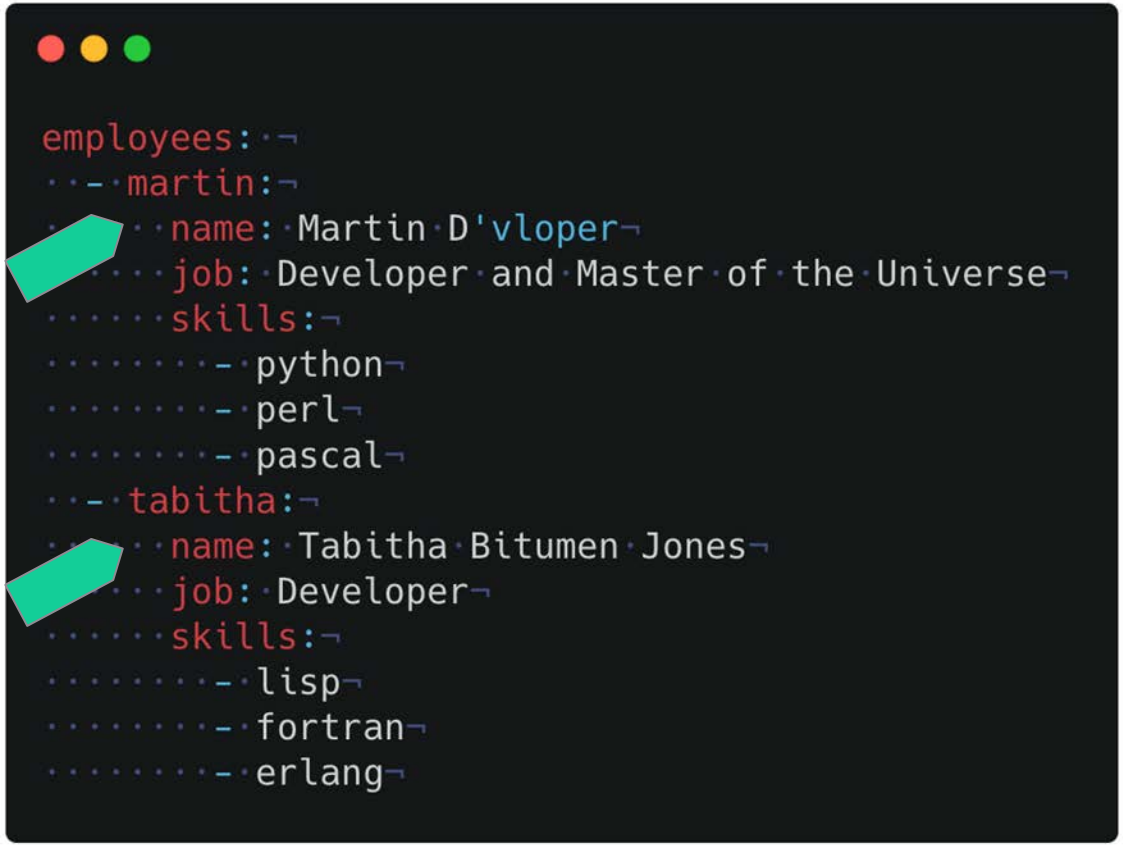


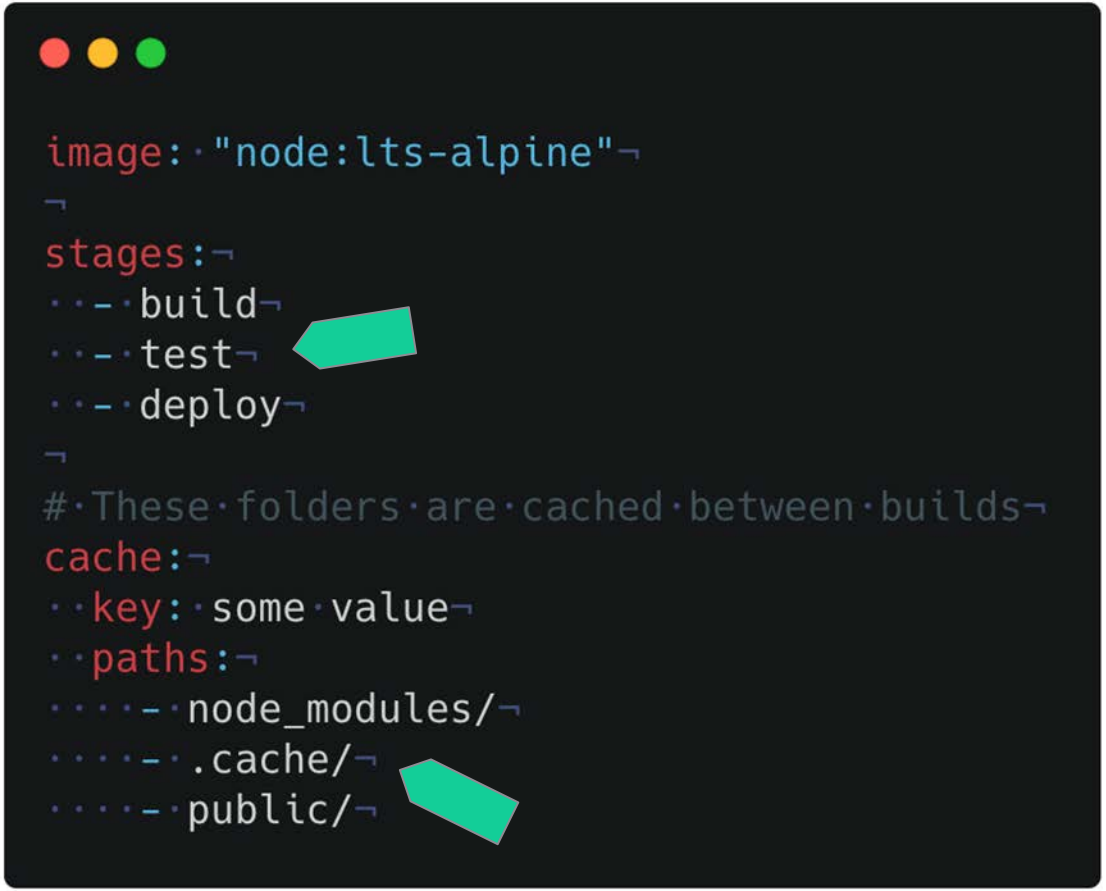
```yaml
image: "node:lts-alpine"

stages:
  - build
  - test
  - deploy

# These folders are cached between builds
cache:
  key: some value
  paths:
    - node_modules/
    - .cache/
    - public/
```


employees:
- martin:
name: Martin D'vloper
job: Developer and Master of the Universe
skills:
- python
- perl
- pascal
- tabitha:
name: Tabitha Bitumen Jones
job: Developer
skills:
- lisp
- fortran
- erlang

YAML Scalar

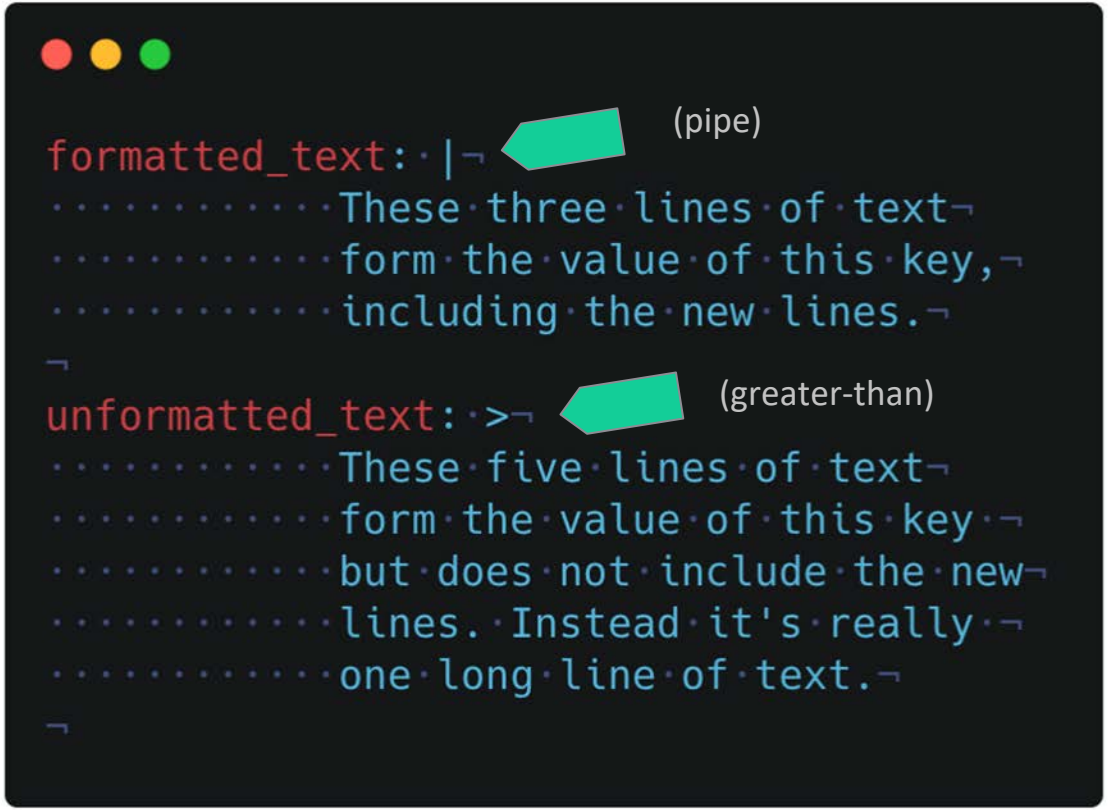
YAML Dictionaries





YAML Collections

YAML Multi-Line



YAML?

“Human-readable data-serialization language”

CI/CD Solutions Abound

GitHub Actions

Leading cloud-based git repository provider with added CI/CD functionality with “Actions”.

GitLab CI/CD

Open-source, community-driven git repository provider with built-in CI/CD facilities.

CircleCI

Popular cloud-based CI/CD provider. Requires 3rd party git provider.

CI/CD Solutions Solve the Same Problems

- Triggering
- Importing Code
- Running Scripts
- Raising Alerts

CI/CD Solutions Abound

GitHub Actions

Leading cloud-based git repository provider with added CI/CD functionality with “Actions”.

GitLab CI/CD

Open-source, community-driven git repository provider with built-in CI/CD facilities.

CircleCI

Popular cloud-based CI/CD provider. Requires 3rd party git provider.

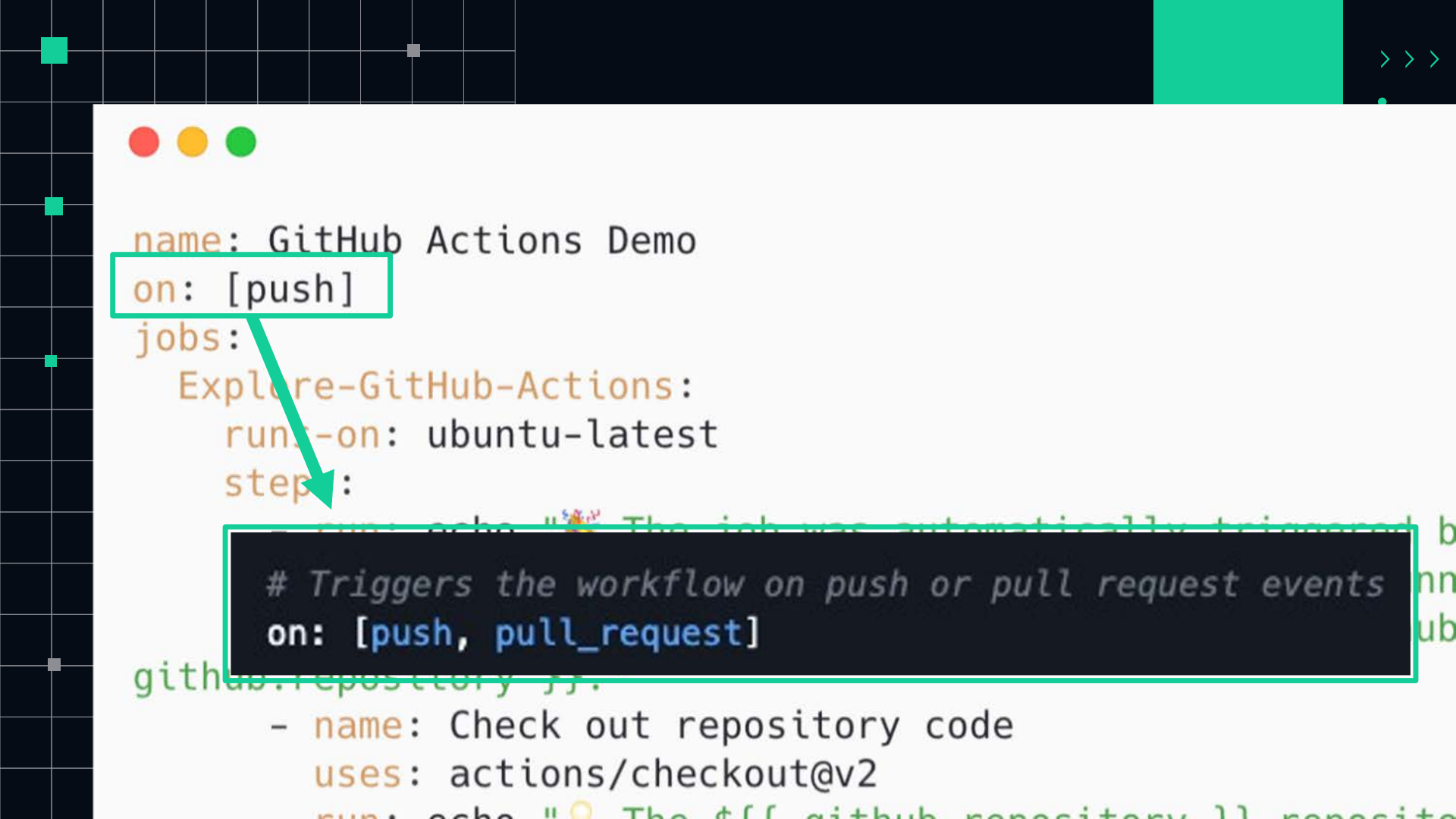
bit.ly/github-actions-quickstart

Workflows Folder

```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered by a ${{ github.event_name }} event."
      - run: echo "🐧 This job is now running on a ${{ runner.os }} server hosted by GitHub!"
      - run: echo "🔎 The name of your branch is ${{ github.ref }} and your repository is ${{
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} repository has been cloned to the runner."
      - run: echo "🖥️ The workflow is now ready to test your code on the runner."
      - name: List files in the repository
        run: |
          ls ${{ github.workspace }}
      - run: echo "🍏 This job's status is ${{ job.status }}."
```

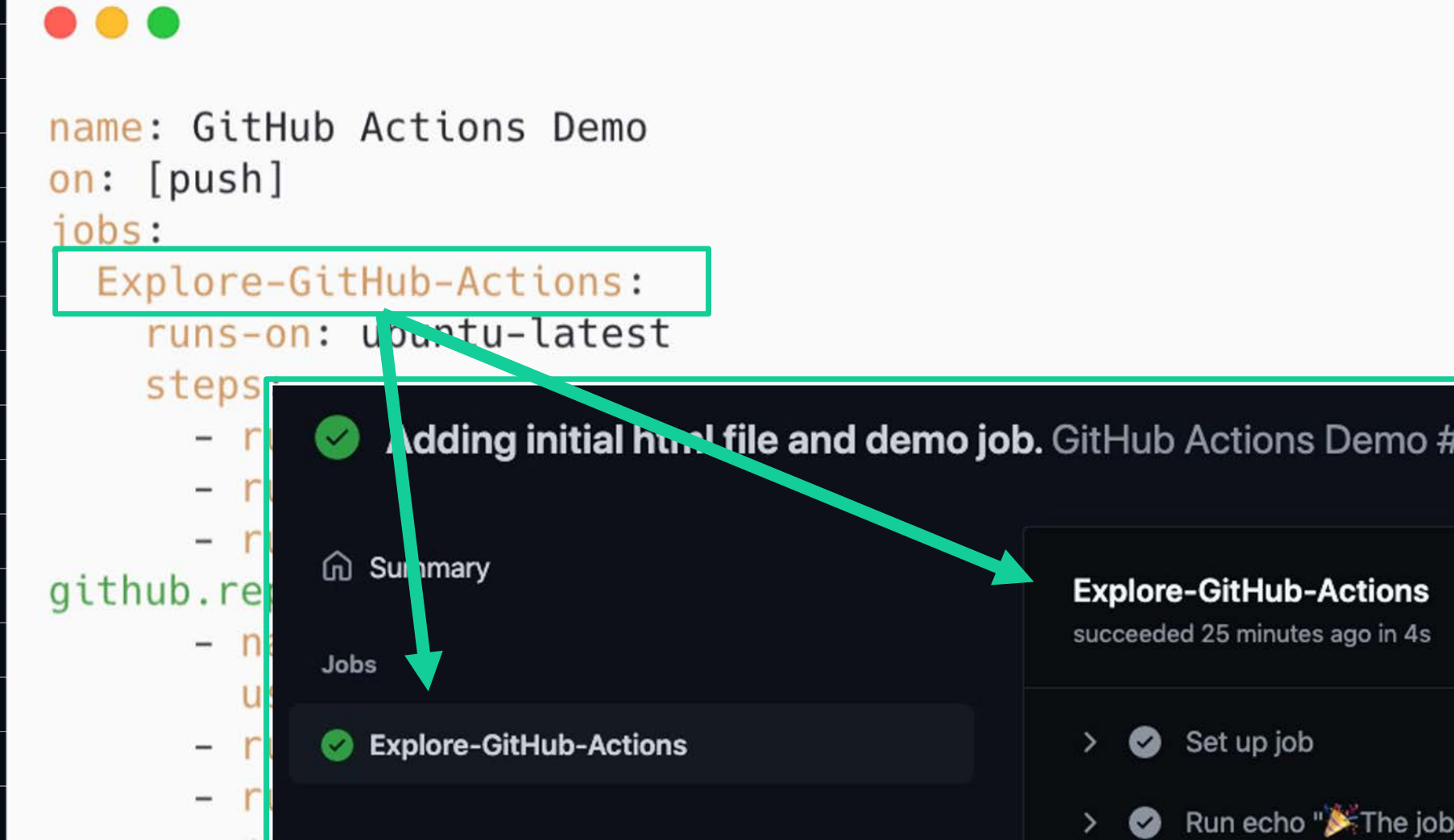
```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
```

```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} repositc
```

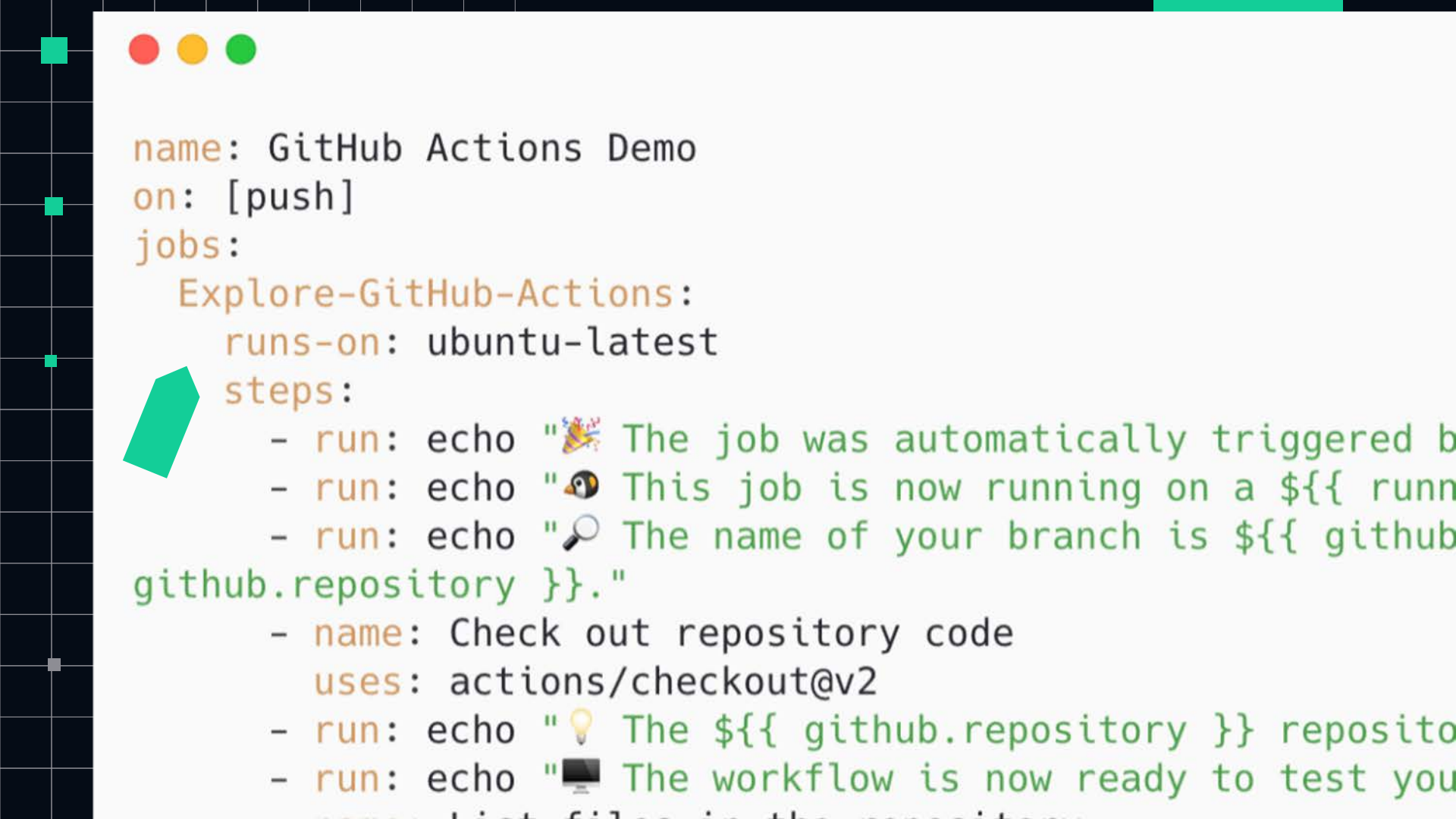

name: GitHub Actions Demo
on: [push]
jobs:
Explore-GitHub-Actions:
runs-on: ubuntu-latest
steps:
Triggers the workflow on push or pull request events
on: [push, pull_request]
- name: Check out repository code
uses: actions/checkout@v2

```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} repositc
```

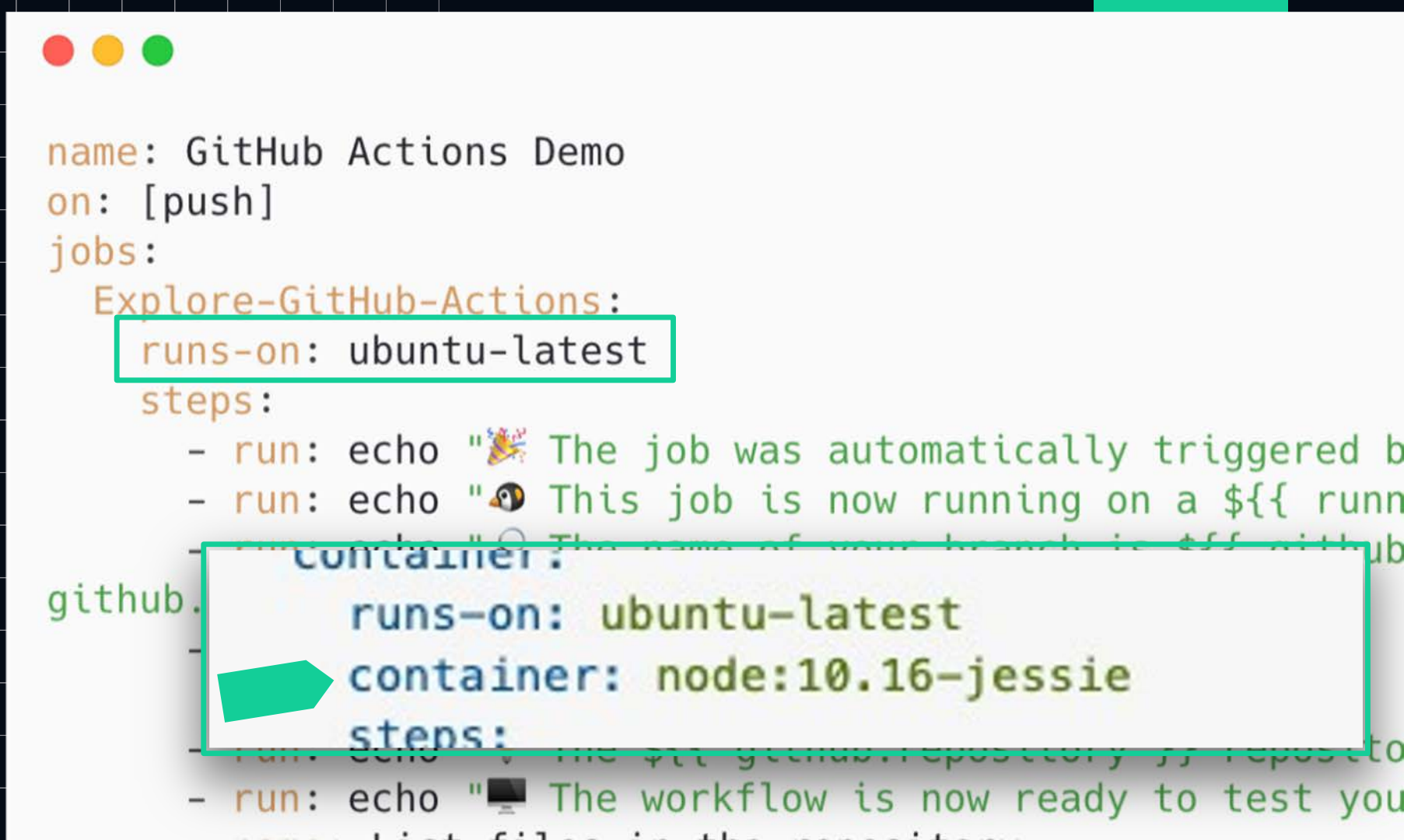
```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} repositc
```


name: GitHub Actions Demo
on: [push]
jobs:
Explore-GitHub-Actions:
runs-on: ubuntu-latest
steps:
Adding initial html file and demo job. GitHub Actions Demo #
Summary
Jobs
Explore-GitHub-Actions
Explore-GitHub-Actions
succeeded 25 minutes ago in 4s
Set up job
Run echo "The job

```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} reposito
      - run: echo "🖥️ The workflow is now ready to test you
```


name: GitHub Actions Demo
on: [push]
jobs:
Explore-GitHub-Actions:
runs-on: ubuntu-latest
steps:
- run: echo "🎉 The job was automatically triggered b
- run: echo "🐧 This job is now running on a ${{ runn
- run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
- name: Check out repository code
uses: actions/checkout@v2
- run: echo "💡 The ${{ github.repository }} reposito
- run: echo "🖥️ The workflow is now ready to test you

```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} reposito
      - run: echo "🖥️ The workflow is now ready to test you
```


name: GitHub Actions Demo
on: [push]
jobs:
Explore-GitHub-Actions:
runs-on: ubuntu-latest
steps:
- run: echo "🎉 The job was automatically triggered b
- run: echo "🐧 This job is now running on a ${{ runn
runs-on: ubuntu-latest
container: node:10.16-jessie
steps:
- run: echo "🖥️ The workflow is now ready to test you

```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} reposito
      - run: echo "🖥️ The workflow is now ready to test you
```

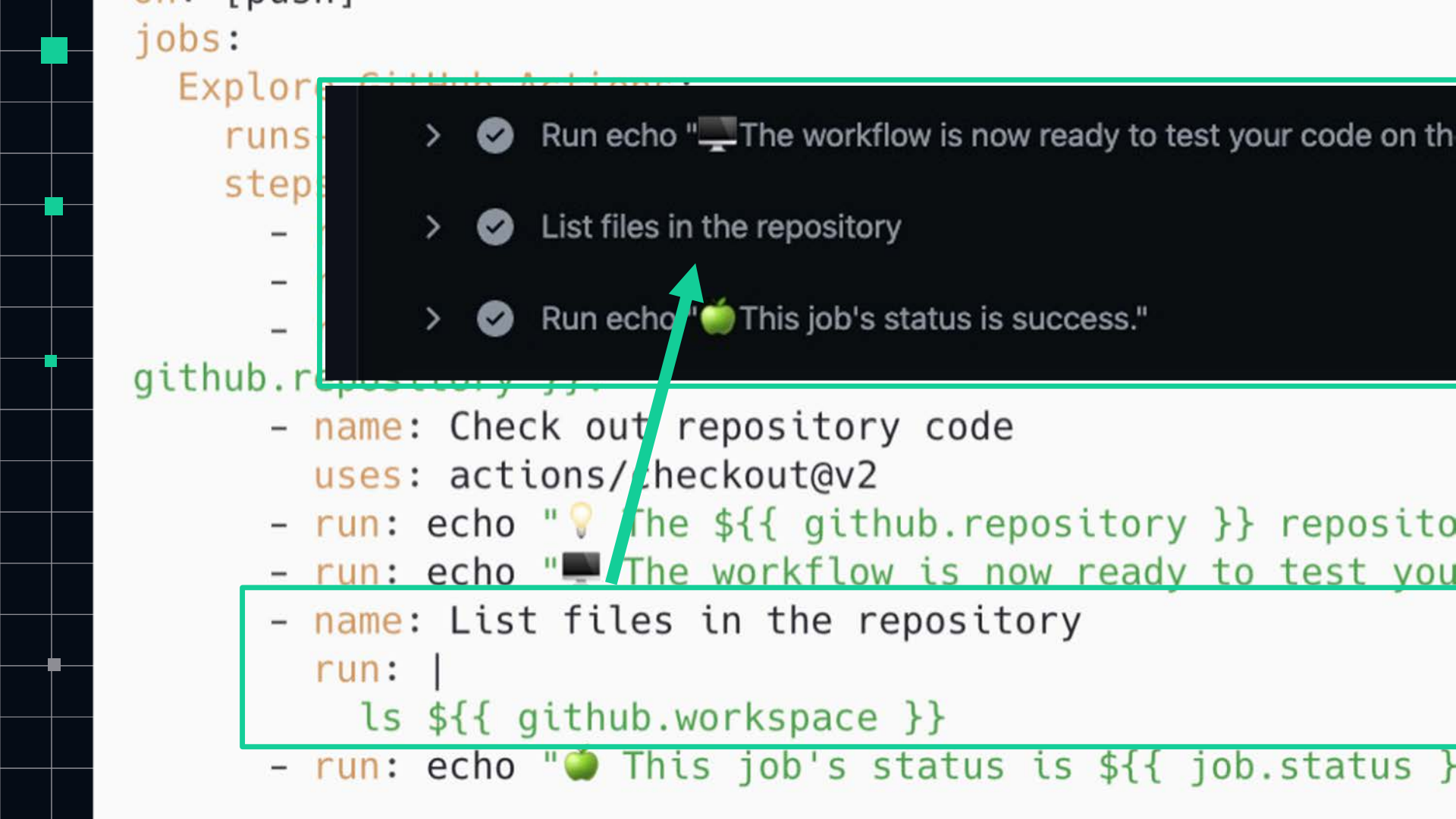
```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} reposito
      - run: echo "🖥️ The workflow is now ready to test you
      - name: List files in the repository
        run: |
          ls ${{ github.workspace }}
```

```
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} reposito
      - run: echo "🖥️ The workflow is now ready to test you
      - name: List files in the repository
        run: |
          ls ${{ github.workspace }}
      - run: echo "🍏 This job's status is ${{ job.status }
```

```
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} reposito
      - run: echo "🖥️ The workflow is now ready to test you
      - name: List files in the repository
        run: |
          ls ${{ github.workspace }}
      - run: echo "🍏 This job's status is ${{ job.status }
```

```
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} reposito
      - run: echo "🖥️ The workflow is now ready to test you
      - name: List files in the repository
        run: |
          ls ${{ github.workspace }}
      - run: echo "🍏 This job's status is ${{ job.status }
```

```
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} reposito
      - run: echo "🖥️ The workflow is now ready to test you
      - name: List files in the repository
        run: |
          ls ${{ github.workspace }}
      - run: echo "🍏 This job's status is ${{ job.status }
```


jobs:
Run echo "🖥️The workflow is now ready to test your code on th
List files in the repository
Run echo "🍏This job's status is success."
- name: Check out repository code
uses: actions/checkout@v2
- run: echo "🖥️ The workflow is now ready to test you
- name: List files in the repository
run: |
ls ${{ github.workspace }}
- run: echo "🍏 This job's status is ${{ job.status }

```
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered b
      - run: echo "🐧 This job is now running on a ${{ runn
      - run: echo "🔎 The name of your branch is ${{ github
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} reposito
      - run: echo "🖥️ The workflow is now ready to test you
      - name: List files in the repository
        run: |
          ls ${{ github.workspace }}
      - run: echo "🍏 This job's status is ${{ job.status }
```

```yaml
name: GitHub Actions Demo
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered by a ${{ github.event_name }} event."
      - run: echo "🐧 This job is now running on a ${{ runner.os }} server hosted by GitHub!"
      - run: echo "🔎 The name of your branch is ${{ github.ref }} and your repository is ${{
github.repository }}."
      - name: Check out repository code
        uses: actions/checkout@v2
      - run: echo "💡 The ${{ github.repository }} repository has been cloned to the runner."
      - run: echo "🖥️ The workflow is now ready to test your code on the runner."
      - name: List files in the repository
        run: |
          ls ${{ github.workspace }}
      - run: echo "🍏 This job's status is ${{ job.status }}."
```

Exercise

Repair and test my YAML file for errors.

YAML: bit.ly/bad-yaml

TEST:yamllint.com

10 Minute Break

Building a CI/CD Pipeline

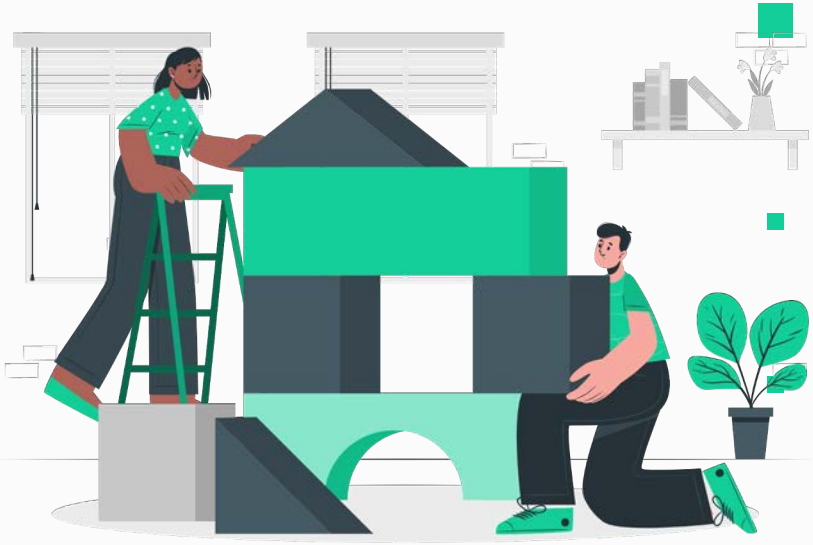
Byron Sommardahl

This Segment

01 **Feedback Loop**
The shorter the better.

02 **CI Components**
Recognizing the pieces that make continuous integration.

03 **The Artifact**
The goal of any CI build!



bit.ly/codex-site

Continuous Integration

Checkout

Pull from source control.

Compile

Make source code usable for runtime.

Unit Test

Make sure code does what it's supposed to do.

Audit

Make sure there aren't any obvious vulnerabilities.

Analyze

Verify source code adheres to team standards.

Artifact

Package the code for deployment.

“Talk is cheap.

Show me the code.”

10 Minute Break

Building a CI/CD Pipeline

Byron Sommardahl

This Segment

01 **Human Error**

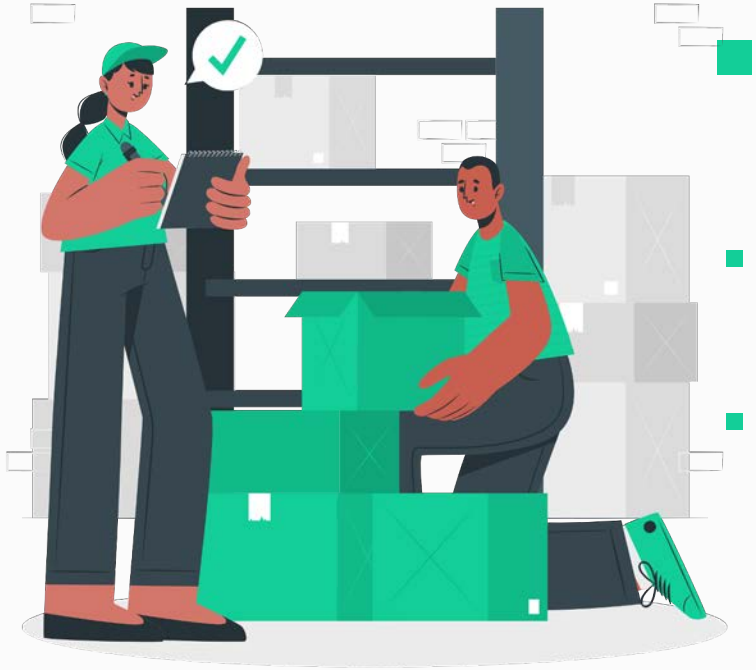
Take humans out of the equation.

02 **Deploy Features**

Get working software into users' hands.

03 **Sanity Check**

Smoke tests for the win!



Continuous Deployment

Create

Create any infrastructure needed for the app.

Configure

Configure infrastructure so that it works properly.

Deploy

Move executables into place so they can be used.

Verify

Check to make sure things work with smoke tests.

Rollback

Revert changes and clean up infrastructure on failure.

Alert

Let the team know of any failure so they can react.

Back to the CODE!

Stretch Goals

Rollback

Save the world after attempting to kill it.

Reusable Actions

Discover and utilize reusable actions.

IaC

Give infrastructure and configuration citizenship.

Strategies

Utilize deployment strategies for smarter outcomes.

Thanks!

Byron Sommardahl
byron@sommardahl.com
calendly.com/sommardahl