

# Praxis: Github Actions

---

## 1. CI/CD Konzepte:

- Erkläre den Unterschied zwischen kontinuierlicher Integration (CI) und kontinuierlicher Bereitstellung (CD).
- Warum ist automatisierte CI/CD-Integration wichtig?

## 2. GitHub Actions Workflow:

- Was ist ein GitHub Actions Workflow?

## 3. Workflow Datei (YAML):

- Erkläre die grundlegende Struktur einer GitHub Actions Workflow-Datei (YAML).
- Welche Schlüsselwörter werden in der Workflow-Datei verwendet?

## 4. Unit Testing in React:

- Warum sind Unit-Tests in React-Anwendungen wichtig?
- Wie führt man Unit-Tests in einer React-Anwendung aus?

## 5. AWS CLI Profil:

- Wie überprüft man vorhandene AWS CLI-Profile?

## 6. AWS S3-Bucket mit Terraform:

- Welche Konfigurationen sind für die AWS S3-Website notwendig?

## 7. GitHub Actions und AWS-Keys:

- Warum ist es wichtig, AWS-Keys sicher zu behandeln?
- Wie fügt man AWS-Keys als Secrets in einem GitHub Repository hinzu?

## 8. GitHub Actions und Node.js:

- Wie wird Node.js in einem GitHub Actions Workflow verwendet?

## 9. Terraform-Version in GitHub Actions:

- Wie installiert man die richtige Terraform-Version in einem GitHub Actions Workflow?

## 10. GitHub Actions - Terraform Init und Apply:

- Wie führt man `terraform apply` in einem Workflow aus?

## 11. AWS S3-Bucket Deployment:

- Welchen Befehl verwendet man, um Dateien in ein S3-Bucket zu synchronisieren?

## 12. GitHub Actions - Verwendung von Aktionen:

- Wie verwendet man eine externe Aktion in einem GitHub Actions Workflow?
- Wo findet man eine Sammlung von Aktionen die von anderen Usern definiert wurden?
- Gibt es bestimmte Aktionen, die häufig in CI/CD-Workflows verwendet werden?