LU13.L01 - GitHub Workflow

Musterlösung zu LU13.A01 - GitHub Workflow.

.github/workflows/tests.yml

```yaml
name: Tests und Lint

on:
  push:
    branches-ignore:          # bei jedem Push, ausser auf main und master
      - main
      - master

jobs:
  test:
    runs-on: ubuntu-latest

    steps:
      - name: Code auschecken
        uses: actions/checkout@v4

      - name: Python einrichten
        uses: actions/setup-python@v5
        with:
          python-version: '3.12'

      - name: Abhängigkeiten installieren
        run: |
          python -m pip install --upgrade pip
          pip install pytest pylint
          if [ -f requirements.txt ]; then pip install -r requirements.txt;
fi

      - name: Unit Tests ausführen
        run: pytest -v

      - name: Code mit pylint prüfen
        run: pylint $(git ls-files '*.py') --fail-under=8
```

Erläuterungen

Abschnitt	Bedeutung
`on: push: branches-ignore`	Der Workflow startet bei jedem Push, ausser auf `main` und `master`. Alternativ: `branches: ['**']` zusammen mit `branches-ignore` ist nicht erlaubt, deshalb nur `branches-ignore`.

Abschnitt	Bedeutung
`runs-on: ubuntu-latest`	Der Job läuft auf einer virtuellen Maschine mit Ubuntu, die GitHub bereitstellt.
`actions/checkout`	Kopiert den Code des Repositorys auf die virtuelle Maschine. Ohne diesen Schritt sind keine Dateien vorhanden.
`actions/setup-python`	Installiert die gewünschte Python-Version.
`pip install`	Installiert pytest, pylint und die Pakete aus `requirements.txt`.
`pytest -v`	Führt alle Tests aus. Schlägt ein Test fehl, endet der Schritt mit einem Fehlercode und der Workflow wird rot.
`pylint … –fail-under=8`	Prüft alle Python-Dateien im Repository. Der Schritt schlägt fehl, wenn die Bewertung unter 8 von 10 liegt.

Kontrolle

1. Den Workflow auf einem neuen Branch (z.B. `workflow-test`) committen und pushen.
2. Im Reiter **Actions** erscheint ein neuer Lauf. Jeder Schritt lässt sich aufklappen und zeigt die Ausgabe.
3. Einen Test absichtlich fehlschlagen lassen und erneut pushen: Der Lauf wird rot und GitHub zeigt den fehlgeschlagenen Schritt.
4. Ein Push auf `main` startet keinen Lauf.

Mit `strategy: matrix: python-version: ['3.11', '3.12', '3.13']`
lassen sich die Tests mit mehreren Python-Versionen gleichzeitig ausführen
(siehe GitHub-Doku «Building and testing Python»).

M450-LU13, M450-C2A

Marcel Suter

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/modul/m450/learningunits/lu13/loesungen/workflow

Last update: **2026/11/17 17:46**