GitHub Actions Tutorial — Lesson Notes

What is GitHub Actions?

GitHub Actions is a CI/CD platform built into GitHub. It automates workflows like testing, building, and deploying code every time you push or create a pull request.

Core Concepts

Concept	Description
Workflow	An automated process defined in a YAML file
Event	A trigger (push, pull_request, schedule, etc.)
Job	A set of steps that run on the same runner
Step	An individual task (run a command or use an action)
Runner	A server that executes the workflow

Basic Workflow

```yaml
# .github/workflows/ci.yml
name: CI Pipeline

on:
  push:
    branches: [main]
  pull_request:
    branches: [main]

jobs:
  test:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4

      - name: Set up Python
        uses: actions/setup-python@v5
        with:
```

```
          python-version: "3.11"

      - name: Install dependencies
        run: |
          pip install -r requirements.txt

      - name: Run tests
        run: pytest --verbose
```

Deploy Workflow

```
deploy:
  needs: test
  runs-on: ubuntu-latest
  if: github.ref == 'refs/heads/main'
  steps:
    - uses: actions/checkout@v4
    - name: Deploy to server
      run: ssh user@server "cd /app && git pull && systemctl
      restart app"
      env:
        SSH_KEY: ${{ secrets.DEPLOY_KEY }}
```

Secrets & Environment Variables

- Store sensitive data in **Settings → Secrets and variables → Actions**.
- Reference in workflows: `${{ secrets.MY_SECRET }}`.
- Never hardcode API keys or passwords in workflow files.

Key Takeaways

1. Workflows live in `.github/workflows/` as YAML files.
2. Use `needs:` to chain jobs (e.g., deploy only after tests pass).
3. Leverage the Actions Marketplace for pre-built actions.
4. Use branch protection rules to require CI checks before merging.
