GitHub

Ep2: GitHub Actions 소개

Presented by 엄지혜 @wisdeom | 프로젝트 듣보잡

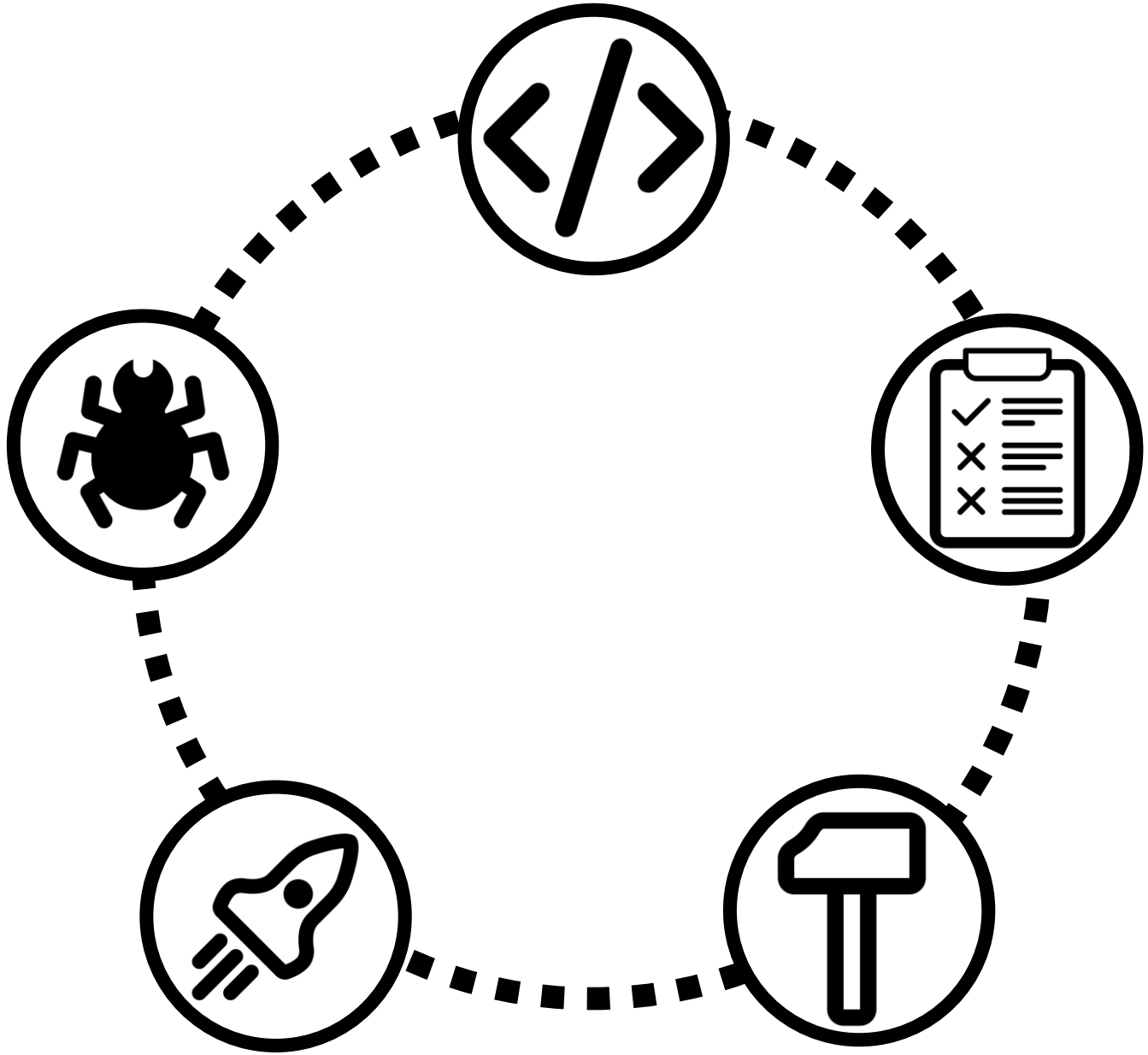
여러분들이
오늘 알게 될 것들 :

CI/CD가 무엇인지 이해할 수 있다.

GitHub Actions으로 troubleshooting!

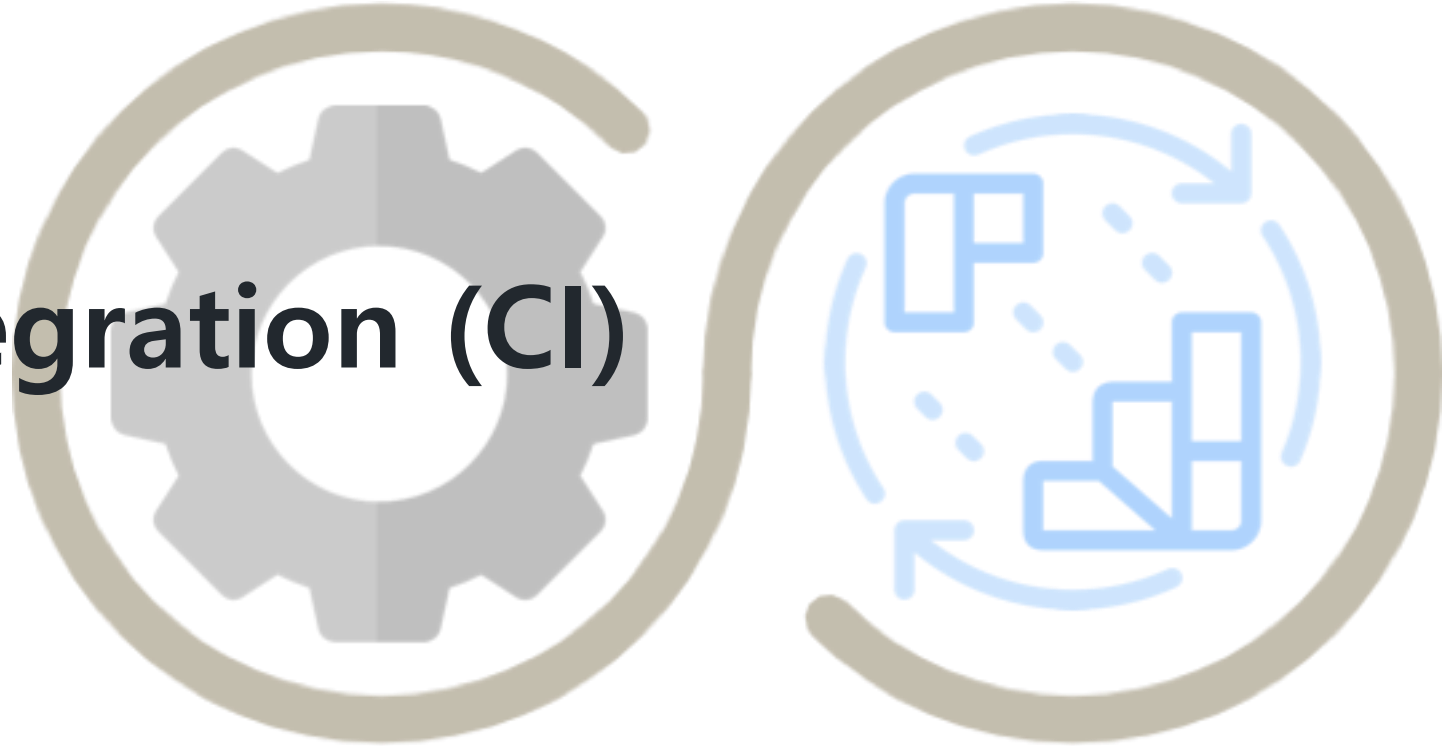
GitHub Actions 템플릿을 사용할 수 있다.

CI/CD가 무엇인가요?



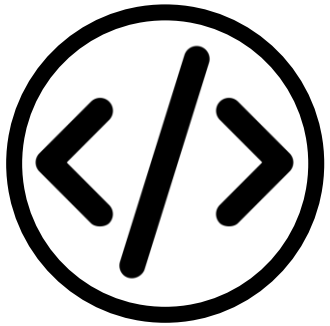
Continuous Integration (CI)

지속적인 통합

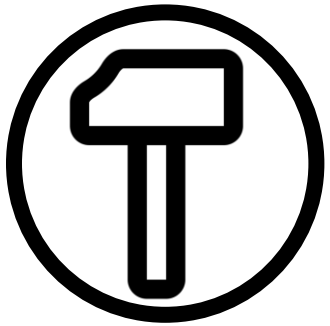


Continuous Delivery / Deployment (CD)

지속적인 제공 / 배포



코드



빌드

테스트

배포

개발 환경 → 운영 환경

Continuous Delivery 지속적인 제공

코드

빌드

테스트

배포

개발 환경 → 운영 환경

Continuous Deployment 지속적인 배포

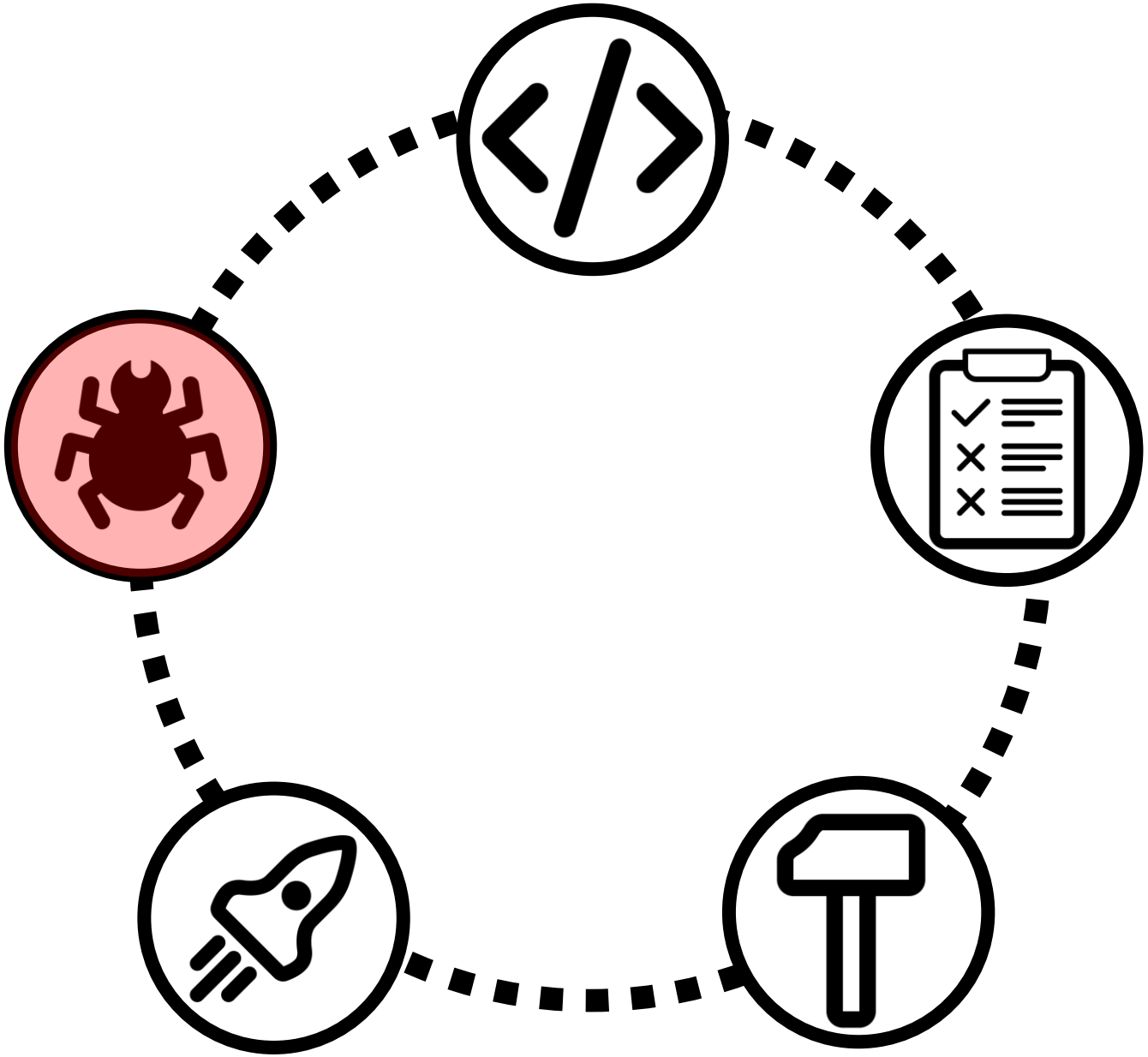
코드

빌드

테스트

배포

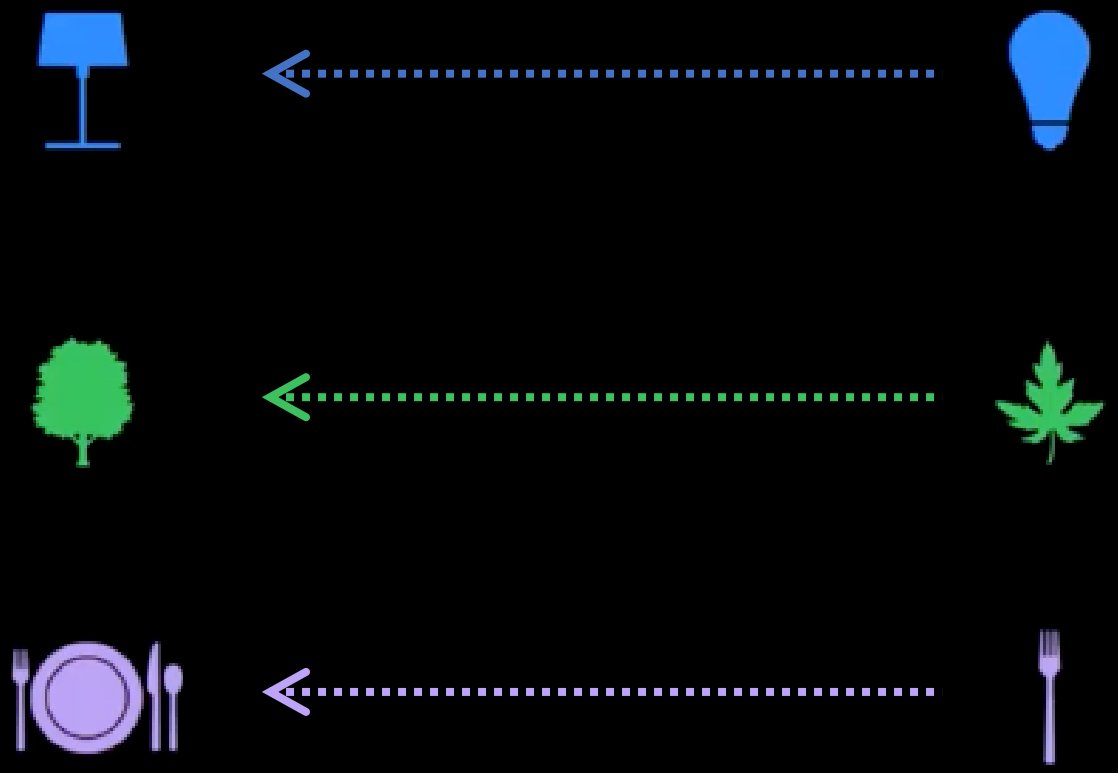
개발 환경 → 운영 환경

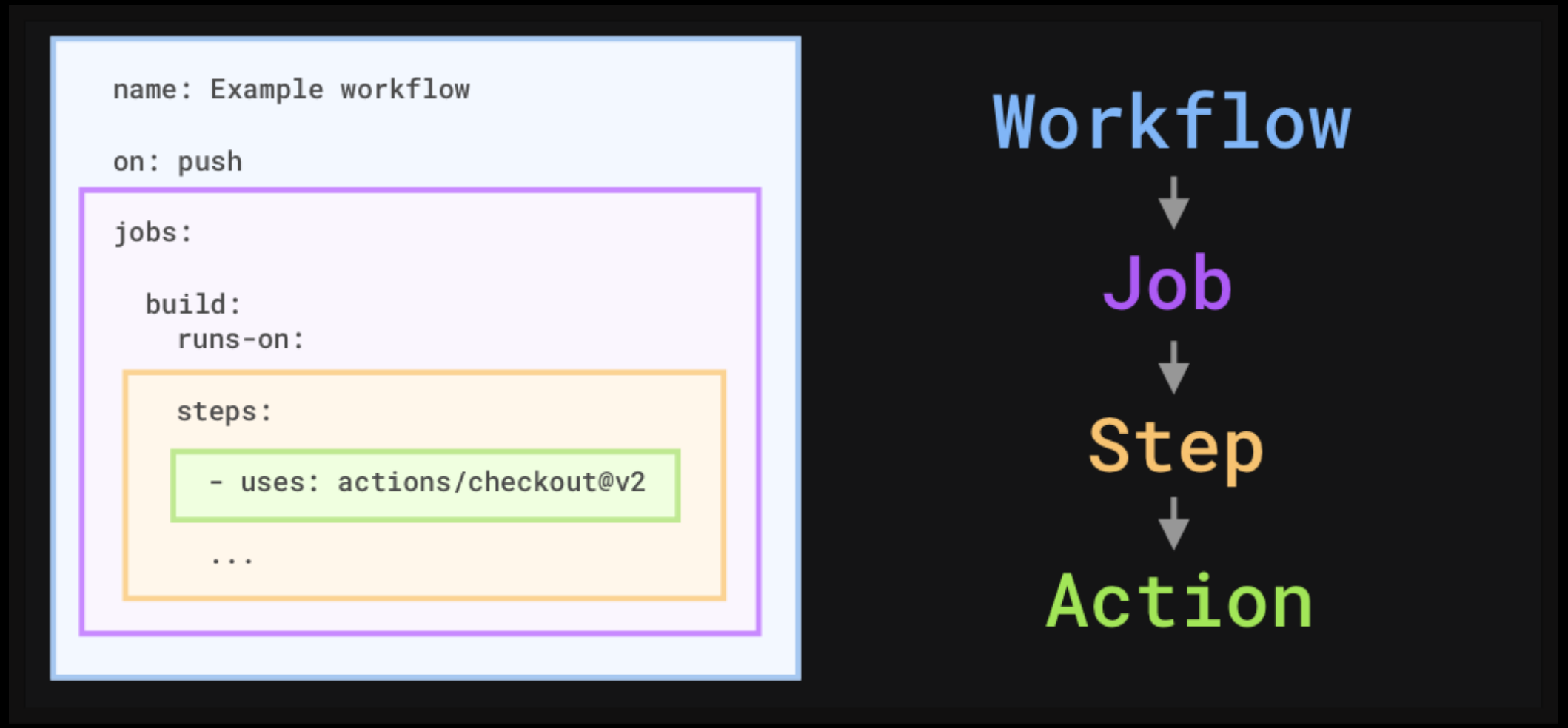


GitHub Actions으로 Troubleshooting!

✍ GitHub Actions 템플릿을 사용할 수 있다.

Workflows > Actions




name: Example workflow
on: push
jobs:
build:
runs-on:
steps:
- uses: actions/checkout@v2
...
Workflow
Job
Step
Action

Choose a workflow template

Build, test, and deploy your code. Make code reviews, branch management, and issue triaging work the way you want. Select a workflow template to get started.

Skip this and set up a workflow yourself →

Workflows made for your JavaScript repository Suggested

Publish Node.js Package

By GitHub Actions

Publishes a Node.js package to npm and GitHub Packages.

Set up this workflow

```
npm ci
npm test
```

Node.js

By GitHub Actions

Build and test a Node.js project with npm.

Set up this workflow

```
npm ci
npm run build --if-present
```

템플릿 기본 구조 설명: 'on:'

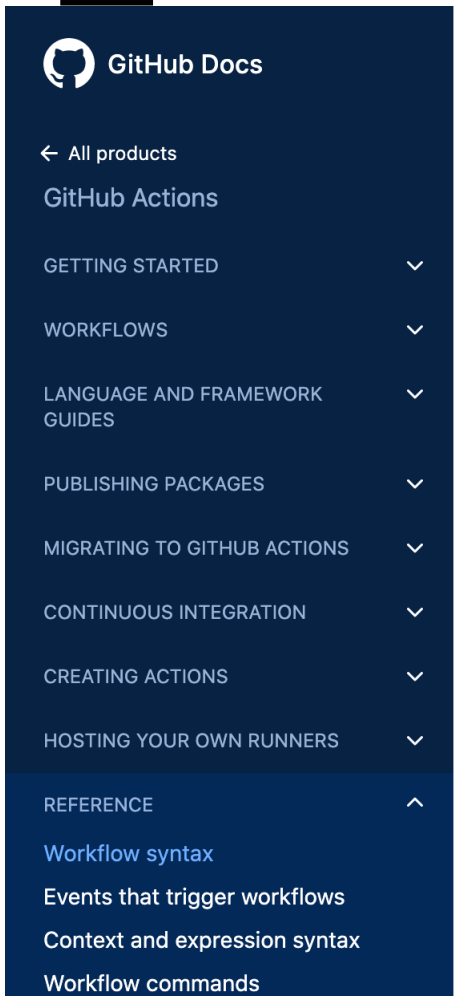
29 .github/workflows/node.js.yml

```
@@ -0,0 +1,29 @@
+ # This workflow will do a clean install of node dependencies, build the source code and run tests
+ # For more information see: https://help.github.com/actions/language-and-framework-guides/using-no
+
+ name: Node.js CI
+
+ on:
```

This conversation was marked as resolved by **wisdeom** Show conversation

```
+   push:
+     branches: [ master ]
+   pull_request:
+     branches: [ master ]
+
+ jobs:
```

템플릿 기본 구조 설명: 'on:' 필드



on

Required The name of the GitHub event that triggers the workflow. You can provide a single event `string`, `array` of events, `array` of event `types`, or an event configuration `map` that schedules a workflow or restricts the execution of a workflow to specific files, tags, or branch changes. For a list of available events, see "Events that trigger workflows."

Example using a single event

```
# Trigger on push
on: push
```

Example using a list of events

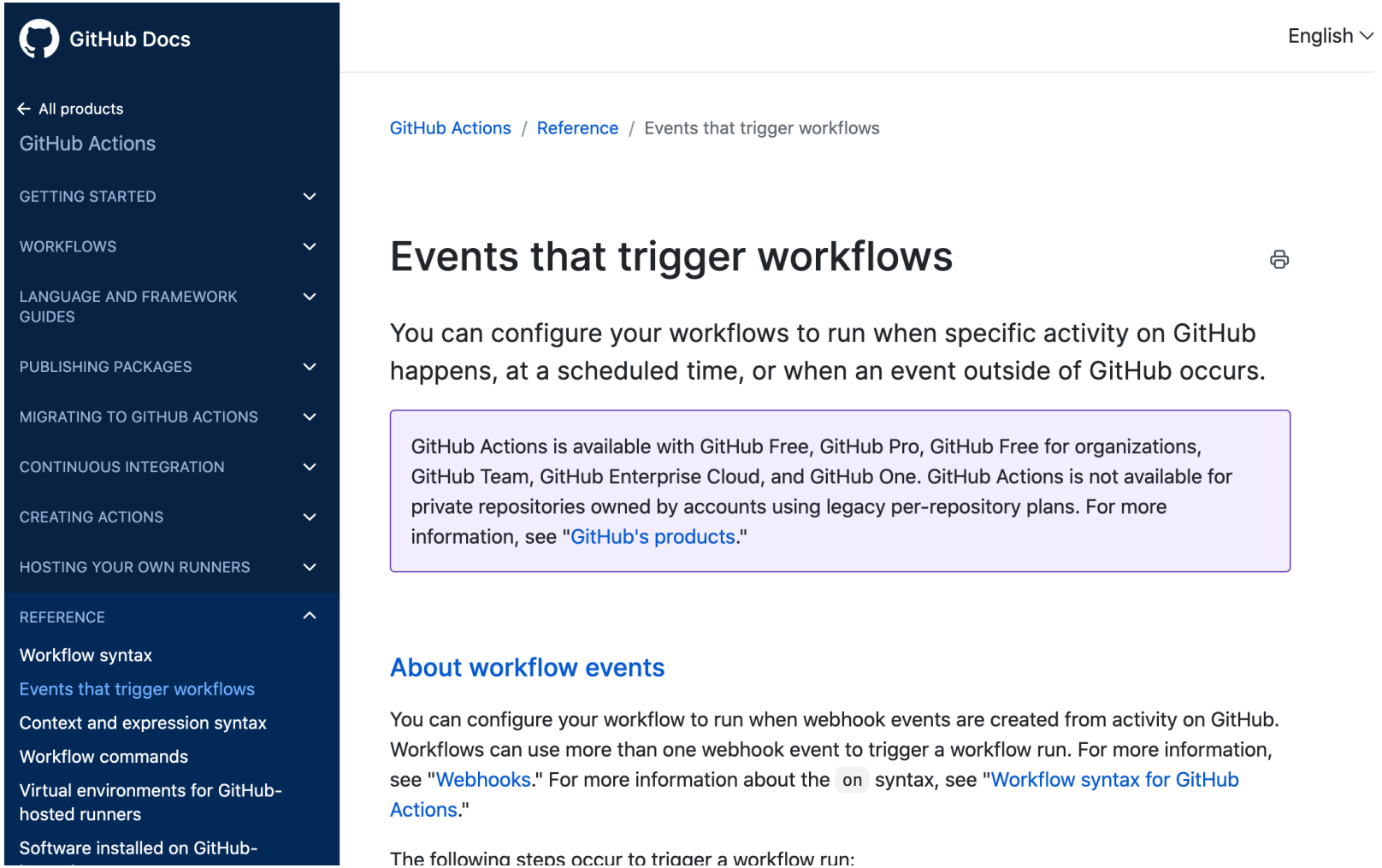
```
# Trigger the workflow on push or pull request
on: [push, pull_request]
```

Example using multiple events with activity types or configuration

If you need to specify activity types or configuration for an event, you must configure each event separately. You must append a colon (`:`) to all events, including events without configuration.

```
on:
  # Trigger the workflow on push or pull request
```

템플릿 기본 구조 설명: 이벤트 트리거



템플릿 기본 구조 설명: ‘jobs’ 필드 ‘runs-on’ 설명

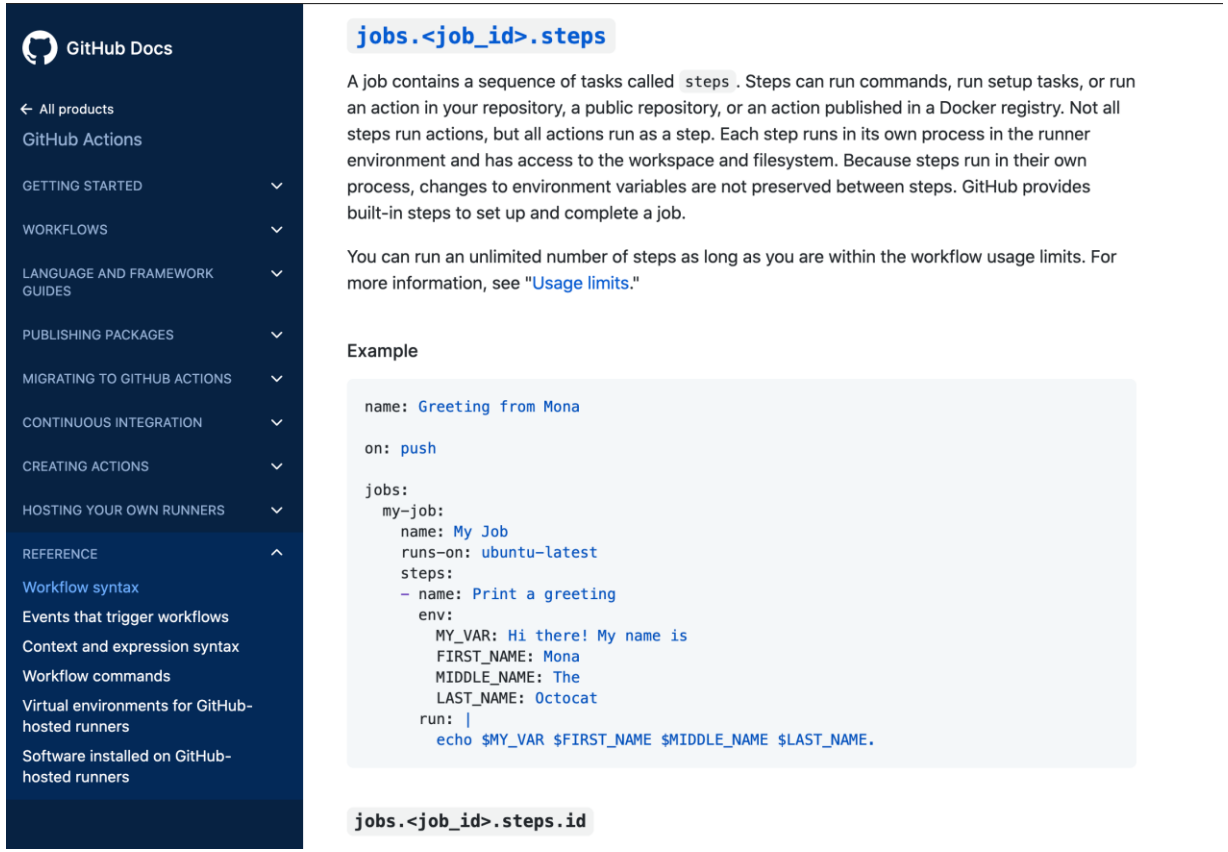
```
12  + jobs:
```

This conversation was marked as resolved by **wisdeom**

```
13  +   build:
14  +
15  +     runs-on: ubuntu-latest
16  +
17  +     strategy:
18  +       matrix:
19  +         node-version: [10.x, 12.x, 14.x]
20  +
21  +     steps:
```

템플릿 기본 구조 설명: ‘jobs’ 필드 ‘runs-on’ 설명

Job 을 더 잘 쓰는 방법 예시 데모
: https://docs.github.com/en/actions/reference/workflow-syntax-for-github-actions#jobsjob_idsteps



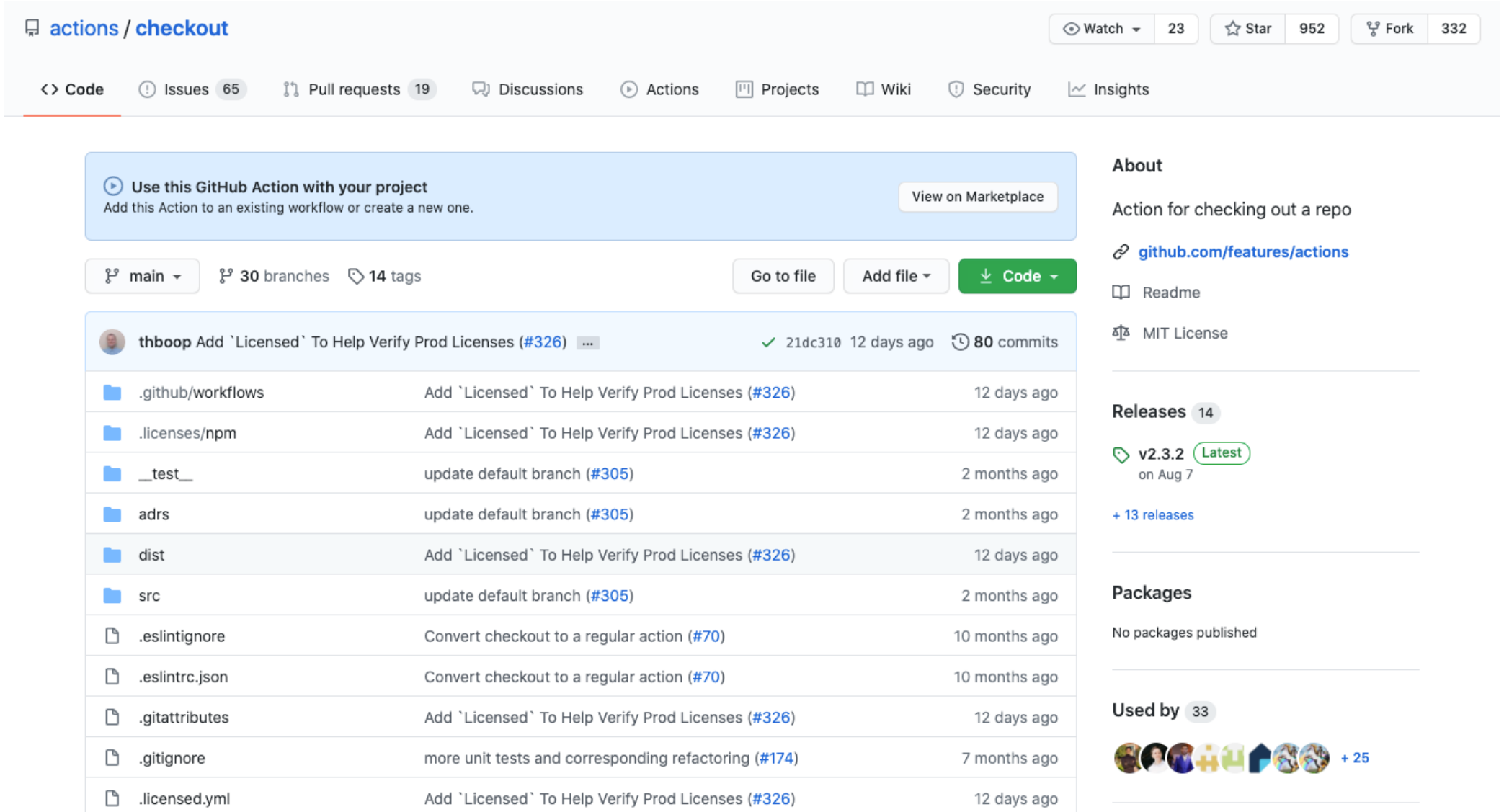
템플릿 기본 구조 설명: checkout

```
21  +    steps:
```

his conversation was marked as resolved by **wisdeom**

```
22  +    - uses: actions/checkout@v2
23  +    - name: Use Node.js ${{ matrix.node-version }}
24  +      uses: actions/setup-node@v1
25  +      with:
26  +        node-version: ${{ matrix.node-version }}
27  +    - run: npm ci
```

템플릿 기본 구조 설명: checkout



템플릿 기본 구조 설명: run

```
22  +     - uses: actions/checkout@v2
23  +     - name: Use Node.js ${{ matrix.node-version }}
24  +       uses: actions/setup-node@v1
25  +       with:
26  +         node-version: ${{ matrix.node-version }}
27  +     - run: npm ci
```

This conversation was marked as resolved by **wisdeom** Show conversation

```
28  +     - run: npm run build --if-present
29  +     - run: npm test
```

템플릿 기본 구조 설명: run

`jobs.<job_id>.steps.run`

Runs command-line programs using the operating system's shell. If you do not provide a `name`, the step name will default to the text specified in the `run` command.

Commands run using non-login shells by default. You can choose a different shell and customize the shell used to run commands. For more information, see "Using a specific shell."

Each `run` keyword represents a new process and shell in the runner environment. When you provide multi-line commands, each line runs in the same shell. For example:

- A single-line command:

```
- name: Install Dependencies
  run: npm install
```

- A multi-line command:

```
- name: Clean install dependencies and build
  run: |
    npm ci
    npm run build
```

Using the `working-directory` keyword, you can specify the working directory of where to run the

여러분들이
오늘 알게 **된** 것들 :

CI/CD가 무엇인지 이해할 수 있다.

GitHub Actions으로 troubleshooting!

GitHub Actions 템플릿을 사용할 수 있다.

다음 시간에는 :

GitHub Actions 워크플로우를 내 입맛에 맞게 작성할 수 있다.

<요구 사항 3가지 해결하기 >

OS 종류와 **Node 버전**이 다양해요.

빌드할 때마다 **테스트**를 실행해서 불편해요.

다른 job에서 **빌드 아티팩트** 접근이 안돼요.