Step-by-step hands-on document to demonstrate custom actions in GitHub Actions

**Step 1: Create a custom action**

1. Create a new repository to store your custom action.
2. Create a new file in the repository called action.yml or action.yaml.
3. In the action.yml file, define the metadata for your action. This includes the name, description, inputs, outputs, and steps of the action.
4. Commit and push the action.yml file to your repository.

**Step 2: Use your custom action in a workflow**

1. Create a new workflow file in your repository.
2. In the workflow file, add a step to use your custom action.
3. Set the inputs and outputs of the action, if necessary.
4. Commit and push the workflow file to your repository.

**Step 3: Trigger the workflow**

1. Push a change to your repository.
2. The workflow will be triggered automatically.
3. You can view the progress of the workflow on the Actions page of your repository.

Example

The following is an example of a custom action that prints a message to the console:

CUSTOM ACTION YAML

name: Print message

description: Prints a message to the console.

inputs:

message:

description: The message to print.

required: true

outputs:

none:

description: This action does not produce any outputs.

runs:

using: "node16"

main: "index.js"

The following is an example of a workflow file that uses the Print message action:

WORKFLOW YAML

name: Print message workflow

on:

push:

branches:

- main

jobs:

build:

runs-on: ubuntu-latest

steps:

- uses: ./print-message@main

with:

message: "Hello, world!"

To trigger the workflow, simply push a change to the main branch of your repository.

INDEX.JS:

const { core } = require('@actions/core');

async function run() {

const message = core.getInput('message');

console.log(message);

}

run();