Events triggers a workflow to run.

Events:

Repository Events: Push, Pull request, Open/Close Issue, Merge Branch.

External Events: Sending a Post request to a Rest API

Scheduled Times: Everyday at 6 PM

Manually: Pressing a Button.

Workflow can contains one or more Jobs.

Each Job is going to run in Runner Machine.

A Job can contain one or more steps. These steps can be run directly in runner machine or inside a docker container. Steps can be simple commands, actions, containers.

Each Job can run independent or parallel. Because each job will run on it’s own Runner Machine.

We can run Job each after another, if they are depend on each other.

Runner Machine:

Github Hosterd Runners (Windows/Linux/MacOS with tools installed)

Self Hosted Runners (More Responsibilty/Customization)

Larger Runners (Hosted by Github with more RAM & CPU)

**Create First-workflow**

First create a file under .github/workflows/<your file name>.yml. And insert below in it.

name: My First Workflow

on: [push]

jobs:

run-shell-commands:

runs-on: ubuntu-latest

steps:

* name: echo a string

run: echo “Hello world”

* name: Multiline Command

run: |

node -v

npm -v

**Parallel and Dependent Jobs**

In below example, our **dependent-job** job will execute once **run-shell-commands** job finish it’s work.

name: First Workflow

on: [push]

jobs:

run-shell-commands:

runs-on: ubuntu-latest

steps:

- name: Echo a string

run: echo "Hello World"

- name: Multiline Commands

run: |

node -v

npm -v

parallel-job-macos:

runs-on: macos-latest

steps:

- name: View SW version

run: sw\_vers

dependent-job:

runs-on: windows-latest

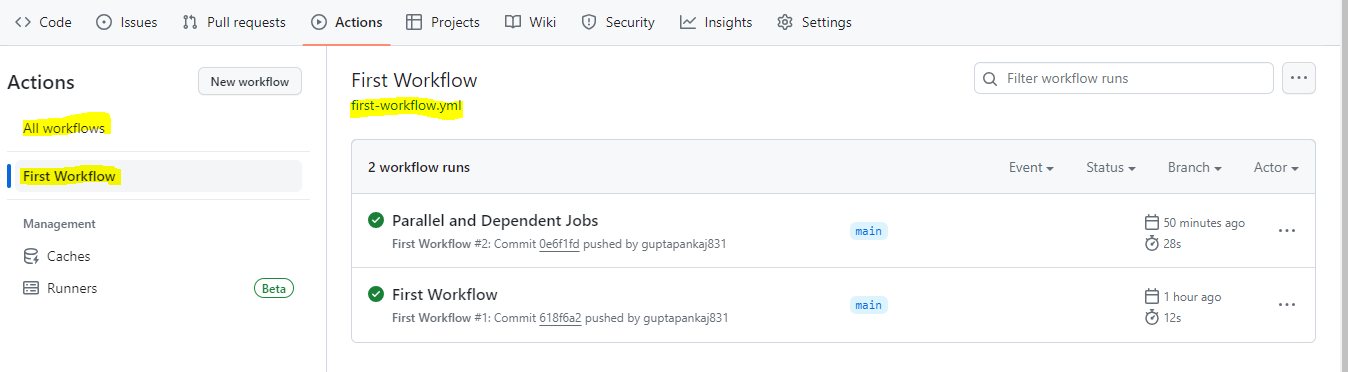
needs: run-shell-commands

steps:

- name: Echo a string

run: Write-Output "Windows String"

Under Action tab, bydefault it shows us all workflows history. But if you want to see history of specific workflow, then select that workflow.



We can manage Workflow Run through WebUI and GitHub CLI.

To see workflow run history of specific workflow using GitHub CLI, use command as

gh workflow view “<workflow name>”

To see all workflows

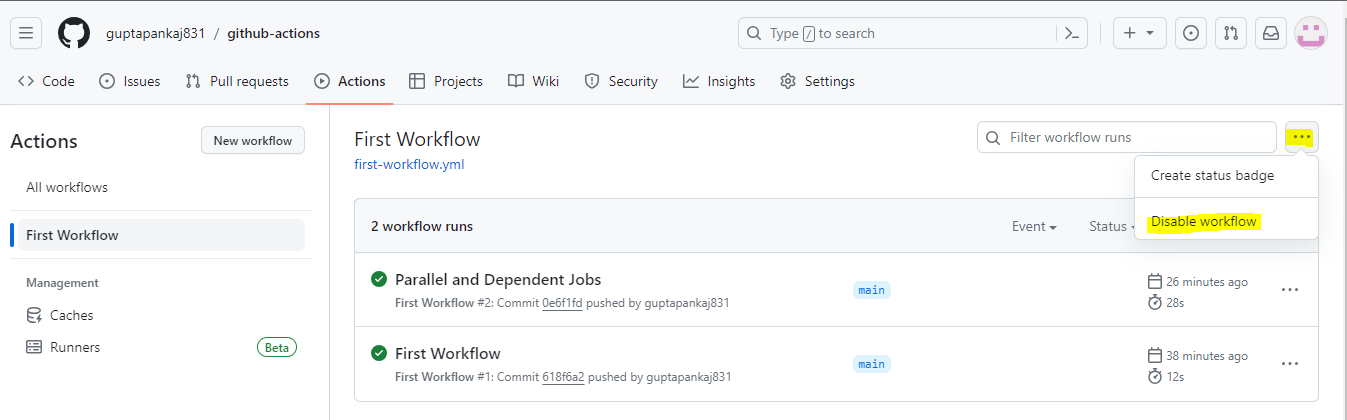
gh wokflow list

To Enable workflow

gh workflow enable “<workflow name>”

If you want to skip the workflow to run because of any reason, then you can disable the workflow. Once we disable the workflow, workflow will never run. To run it, we have to enable it.

Once workflow is disable, and your push something. Then You will see a commit log, but your workflow will not run and you will not see any workflow run history.



But If you want to skip the workflow for certain commits, then mention anyone of string in commit message.

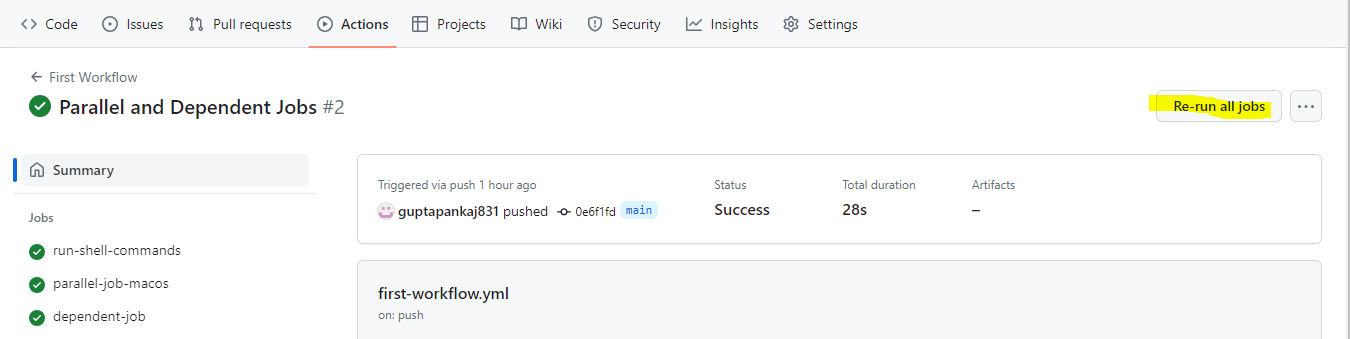
git commit -am “<your msg> [skip ci]”

git commit -am “<your msg> [ci skip]”

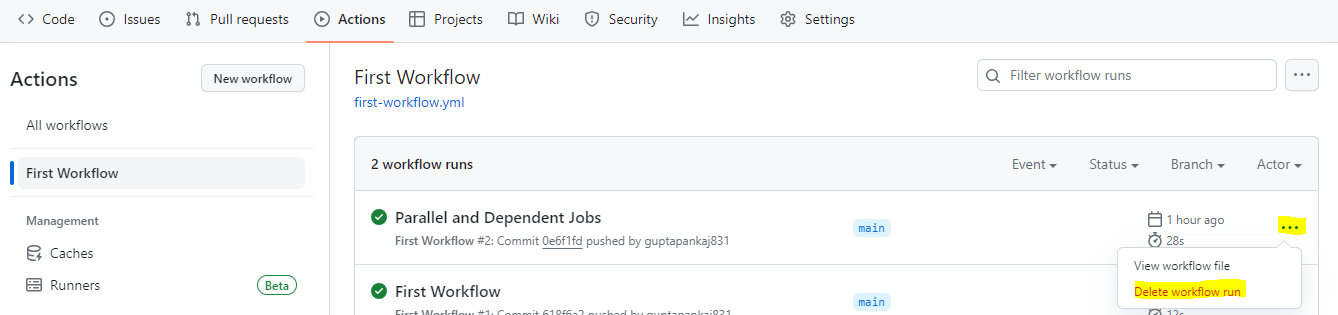
git commit -am “<your msg> [no ci]”

git commit -am “<your msg> [skip actions]”

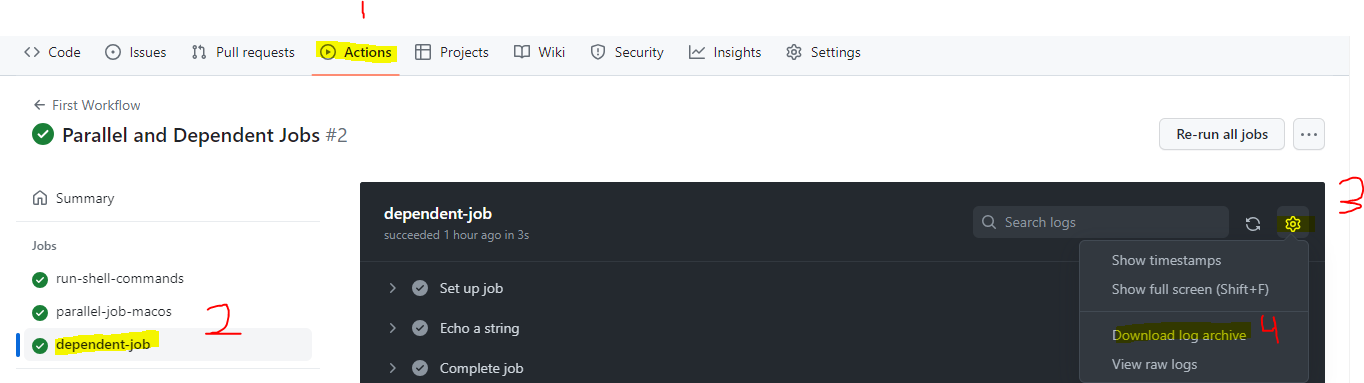
If you want to re-run the job, then select the job and on right-top you will see “Re-Run Jobs” button. From there, select options



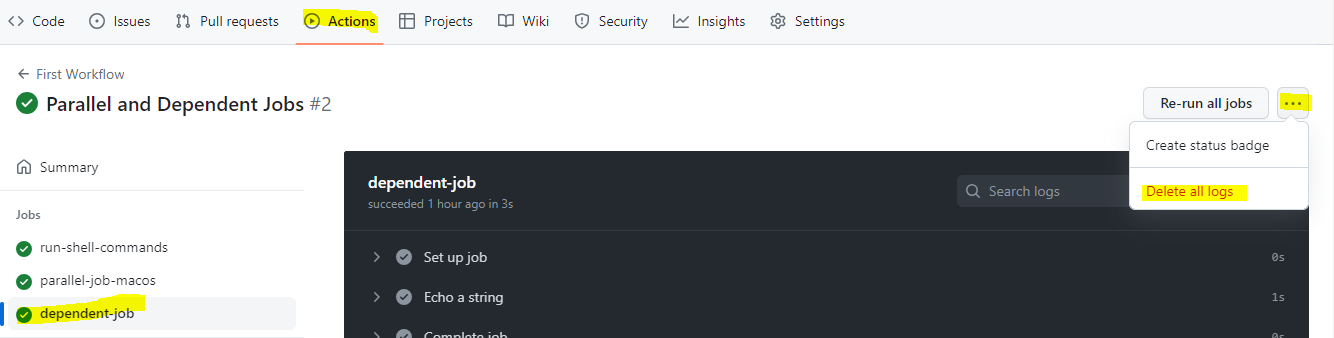
We can delete workflow run from by selecting workflow run and delete it



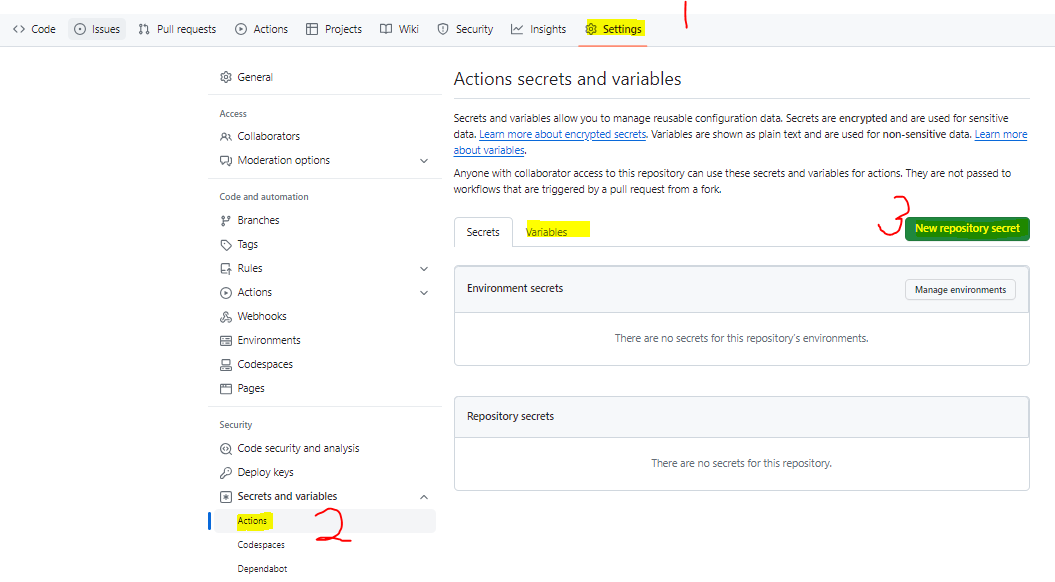
We can download log of a job



We can delete all logs of a job



We can create Secrets and Variables for Actions to be used later in workflow



**Section 2: Events that trigger workflows**

**Section 3: Expressions, Contexts, Functions, Environment Variables & Secrets**