GitHub Actions

This a platform to automate developer workflows

What are workflows?

github : platform for open source projects
* lot of developer. * publicly available
users can make issues so that
contributers could fix them
the contributer with will fix the bug
and make a pull request
↓
Then we review the pull request.
↓
is the bug fixed.
↓
merge to master branch ← this done
on the next release

then

merged → Test → Build →
code   Deployment

This a workflow example.

github actions is used to automate
when something happens IN or TO
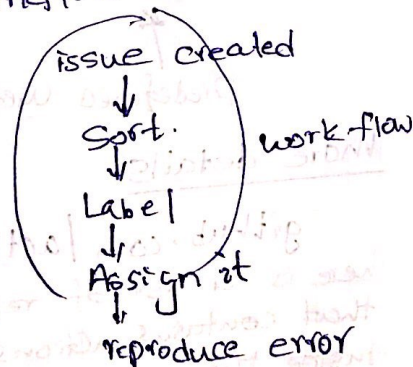your repository
↑
these are called github events.
. - pull requests
  - issues
  - contributor joined
  - merging pull request
    . to the master branch

How github actions automate these workflows?

1) Listen to events

2) Trigger workflow
     ex: when   issue created
                      ↓
                   Sort.   work flow
                      ↓
                   Label
                      ↓
                   Assign it
                      ↓
                reproduce error

CI/CD with Github Actions ②

- Most common workflow for your repository

```
┌────────┐     ┌──────┐     ┌───────┐
│ Commit │ ──→ │ TEST │ ──→ │ BUILD │
│  Code  │     └──────┘     └───────┘
└────────┘                      │
                                ↓
                            ┌──────┐
                            │ PUSH │
                            └──────┘
                                │
                                ↓
                          ┌─────────┐
                          │ DEPLOY  │
                          └─────────┘
```
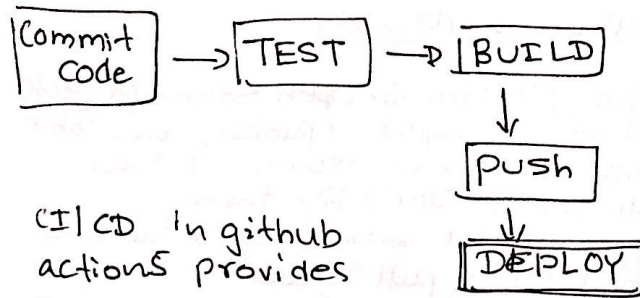
CI/CD In github actions provides an enviroment to connect different tools & to deploy the app without the requirement for the user to install libraries and plugins

Steps

① push the local code to a github repository

② In the actions tab and use a template which matches the tools you need to complete the task

one you select the template a work flow file will be created

Work flow file format.

name :   [optional] (java ver...)

On :   [required] (events)

jobs : [required] ← set of steps that should be executed

    Steps :
      uses: actions/checkout @ v2
           ↑
      Predefined useful commands

More details

github.com/actions
here is a list of repositories that contains actions.
Inside the action repository
there is a action.yml file with
... that will happen.

actions could have versions (3)
as well

steps :
    uses : actions / checkout@v2
    ↗   ↗
attribute to use   version
an action

action to setup java :

    name : Setup JDK 1·8·
    uses : actions / setup - java @X1
    with :
        java - version? 1·8

to run some command like a
linux command we use run: attribute

ex: ~~run~~
    run : chmod +X gradlew

    run: ·/gradlew build

example
    when such a set work flow is
present if there is a pull request
then the workflow will be execut
and details could be viewed.

<u>where does this happen?</u>

This runs on git hub server.

each job runs in a fresh VE

The jobs are executed in parallel
but we may sometimes need to
wait say to publish.

    publish :
        needs: build·

under jobs :
    build :
        runs-on: ubuntu-latest··
            * window /macOS

        strategy:
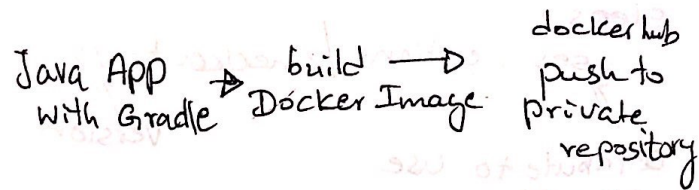   matrix : } attributes to run
            os: } on different OS
            os: [ubuntu-latest, windows-
                latest, macsOS-lates].

Build Docker Image & push to Private repository.

Java App ➜ build ⟶ docker hub
with Gradle   Docker Image   push to
                              private
                              repository

In the workflow file run
the necessary commands to
acess the docker repository

→ This is a common process
So there are alread defined
actions that we can use

name: Build and Push Docker Image
uses: mr-smithers-excellect/docker-
                              build-push
                              @v4.
With:
    image: repo/appName.
    registry: docker.io.
    username: ${{ secrets. Docker_
                        userName}}
    password: ${{secrets. Docker_
                        password}}

these can be added in git hub
repository settings

settings ⟶ secrets.
    Nam | Docker _username |
    password
            | - - - |

*Note if OS = ubuntu then docker
is pre-installed.

Now the java application will appear
* in docker repository.