# This is a basic workflow that is manually triggered

name: MAIN TEST

# Controls when the workflow will run

on:

# Triggers the workflow on push or pull request events but only for the master branch

push:

branches: [ main ]

pull\_request:

branches: [ main ]

# Allows you to run this workflow manually from the Actions tab

workflow\_dispatch:

# A workflow run is made up of one or more jobs that can run sequentially or in parallel

jobs:

# This workflow contains a single job called "build"

build:

# The type of runner that the job will run on

runs-on: ubuntu-latest

# Steps represent a sequence of tasks that will be executed as part of the job

steps:

# Checks-out your repository under $GITHUB\_WORKSPACE, so your job can access it

- uses: actions/checkout@v4

#Set up the Node.js 20

- name: Set up JDK 20

uses: actions/setup-java@v4

with:

java-version: '20'

distribution: 'temurin'

#node-version: '20'

# list contents or directory

- name: List contents

run: ls

# Compiles classes to test

- name: compile classes

working-directory: ./src

run: javac -cp "../testresources/junit-platform-console-standalone-1.6.0.jar" ./practicalassignment/PracticalAssignment.java ./practicalassignment/Student.java "../test/practicalassignment/StudentTest.java"

# list contents of src to verify that classess compiled

- name: List contents of src directory after compile

working-directory: ./src/practicalassignment

run: ls

# copy compiled classes to test directory

- name : copy compiled classes to testresources directory

working-directory: ./src/practicalassignment

run: cp -t "../../test/practicalassignment" PracticalAssignment.class Student.class

#copy compiled tests

- name : copy compiled Test classes

working-directory: ./build/test/classes/practicalassignment

run: cp -t "../../../../testresources/" StudentTest.class

# list contents or directory

- name: List contents test package

working-directory: ./test/practicalassignment

run: ls

#run Message Class tests and output report

- name: run Messages Class tests and create report

working-directory: ./test

run: |

ls

java -Djava.awt.headless=true -jar "../testresources/junit-platform-console-standalone-1.6.0.jar" -cp "." --select-class practicalassignment.StudentTest