**Practical Exercise 08 - Environment Variables**

**Exercise Description**

**In this practical exercise, our goal is to explore in details how to work with environment variables in workflows.**

Here are the instructions for the exercise:

1. Create a file named 08-variables.yaml under the .github/workflows folder at the root of your repository.
2. Name the workflow 08 - Using Variables.
3. Add the following triggers with event filters and activity types to your workflow:
   1. push
   2. workflow\_dispatch
4. Define two environment variables at the workflow level:
   1. WORKFLOW\_VAR, should contain the value 'I am a workflow env var'
   2. OVERWRITTEN, should contain the value 'I will be overwritten'
5. Add a single job named echo to the workflow:
   1. The job should run on ubuntu-latest.
   2. Define two environment variables at the job level:
      1. JOB\_VAR, should contain the value 'I am a job env var'
      2. OVERWRITTEN, should contain the value 'I have been overwritten at the job level'
6. Add a step named Print Env Variables:
   1. Define two environment variables at the step level:
      1. STEP\_VAR, should contain the value 'I am a step env var'
      2. step\_var2, should contain the value 'I am another step var'
   2. The step should print the following information:
      1. "Step env var: <retrieve the value of STEP\_VAR here>"
      2. "Step env var 2: <retrieve the value of step\_var2 here>"
      3. "Job env var: <retrieve the value of JOB\_VAR here>"
      4. "Workflow env var: <retrieve the value of WORKFLOW\_VAR here>"
      5. "Overwritten: <retrieve the value of OVERWRITTEN here>"
7. Add another step named Overwrite Job Variable:
   1. Define a single environment variable at the step level:
      1. OVERWRITTEN, should contain the value 'I have been overwritten at the step level'
   2. The step should print a single line containing "Step env var: <retrieve the value of OVERWRITTEN here>"
8. Commit the changes and push the code. Take a few moments to inspect the output of the workflow run. How were the different values of the environment variables overwritten?
9. Change the workflow triggers to contain only workflow\_dispatch to prevent this workflow from running with every push and pollute the list of workflow runs.

**Tips**

**Retrieving the values of environment variables**

When running shell scripts, we can retrieve the values of environment variables in two ways:

1. The first one is to use the env context: echo "Env variable value: ${{ env.MY\_VARIABLE }}"
2. The second one is to directly access the variable with the dollar-sign syntax: echo "Env variable value: $MY\_VARIABLE"