**Practical Exercise 34 - Managing Concurrency at the Job Level**

**Exercise Description**

**In this practical exercise, our goal is to get familiar with managing concurrency in workflow runs.**

Here are the instructions for the exercise:

1. Create a file named 19-2-concurrency.yaml under the .github/workflows folder at the root of your repository.
2. Name the workflow 19 - 2 - Managing Concurrency.
3. Add the following triggers to your workflow:
   1. workflow\_dispatch
4. Add a first job named ping-with-concurrency to the workflow.
   1. It should run on ubuntu-latest.
   2. It should have a concurrency key added to it, with the group parameter containing the value <retrieve the workflow name here>-<retrieve the git ref here>.
   3. It should contain two steps:
      1. The first step should checkout the code using the appropriate third-party action.
      2. The second step, named Ping URL, should use our docker-ping-url custom action and:
         1. Provide a valid but unreachable URL as the url parameter.
         2. Set the max\_trials parameter to 20.
         3. Set the delay parameter to 5.
5. Add a second job named ping-without-concurrency to the workflow.
   1. It should run on ubuntu-latest.
   2. Do not include any concurrency configuration to the job.
   3. It should contain two steps:
      1. The first step should checkout the code using the appropriate third-party action.
      2. The second step, named Ping URL, should use our docker-ping-url custom action and:
         1. Provide a valid but unreachable URL as the url parameter.
         2. Set the max\_trials parameter to 20.
         3. Set the delay parameter to 5.
6. Commit the changes and push the code. Trigger the workflow several times from the UI (with a few seconds between the triggers) and take a few moments to inspect the result of the workflow run. How did concurrency influence the execution of the workflow and of the job with concurrency?