**Practical Exercise 18 - Introduction to Using Matrices**

**Exercise Description**

**In this practical exercise, our goal is to deepen our knowledge around how to work with matrices.**

Here are the instructions for the exercise:

1. Create a file named 15-matrices.yaml under the .github/workflows folder at the root of your repository.
2. Name the workflow 15 - Working with Matrices.
3. Add the following triggers with event filters and activity types to your workflow:
   1. workflow\_dispatch
4. Add a single job named backwards-compatibility to the workflow.
   1. It should run on ubuntu-latest.
   2. It should use the matrix strategy to define jobs for the following configurations:
      1. Node versions 18.x, 20.x, and 21.x.
      2. Operating systems ubuntu-latest and windows-latest.
   3. It should be named according to the Node version and the OS following the format: <os>-<node-version>
   4. The job should contain two steps:
      1. The first step, named Setup node, should setup Node according to the version from the matrix.
      2. The second step, named Perform some tests, should print a single like "Running tests on OS <OS value here> and NodeJS <Node version here>".
5. Commit the changes and push the code. Trigger the workflow manually from the UI and take a few moments to inspect the result of the workflow run.
6. Extend the 15-matrices.yaml workflow to include the following combinations to the matrix (if you are not sure how to do that, check the **Tips** section below):
   1. Node version 16.x for the OS ubuntu-latest.
   2. An extra tag key with the value of experimental to the combination Node 21.x, OS ubuntu-latest.
7. Additionally, set the fail-fast option under the strategy key to false.
8. Add a new step **after** Setup node, named Fail if experimental, which should run if and only if the tag key equals experimental. The step should simply exit with a non-zero code.
9. Change the step named Perform some tests to, in addition to printing the original message, sleep for 10 seconds.
10. Add a final step **after** Perform some tests, named Upload test results, which should print the message "Uploading test results".
11. Commit the changes and push the code. Trigger the workflow manually from the UI and take a few moments to inspect the result of the workflow run.
12. Now change the fail-fast option to true, commit the changes and push the code. Trigger the workflow manually from the UI and take a few moments to inspect the result of the workflow run. How did the change in the fail-fast configuration impact the workflow run?

**Tips**

**Including and extending specific combinations to the matrix**

To include a specific configuration or extend a specific combination of a matrix, we can use the include option with the following syntax:

1. strategy:
2. matrix:
3. node-version: [18.x, 20.x]
4. os:
5. - ubuntu-latest
6. - windows-latest
7. include:
8. - os: ubuntu-latest
9. tag: linux
10. - os: ubuntu-latest
11. node-version: 21.x

The above example will include the tag key to all combinations that include the ubuntu-latest value for the os key, and it will also add a new combination for ubuntu-latest and node version 21.x (it will not have the 21.x node version set for the windows-latest option). The include option can be quite tricky to understand, so we discuss it in details in the next practical exercise!