Mentoring session exercises

Analysis preservation workshop Jan 20 2023

Docker

- **Python3.11 (easy):** You want to try out one of the new features of python3.11, but don't have it installed locally. Find a docker container that contains python3.11 and start python.
  - For example, try out that you can import the new tomllib

- **Build a docker file (easy):** Write your own image that builds on python3.11 and contains the `uproot` library
  - **Optional:** Most python applications have a `requirements.txt` that states all requirements (such as `uproot`). Create a dummy `requirements.txt`, copy it in your container and install all required packages with `pip3 install -r requirements.txt`.

Apptainer/Singularity

- **Easy:** Repeat one of the docker exercises on your cluster with Apptainer/Singularity using an interactive session

- **Medium:** Do the same with a definition file.
  - Option: Use the `%runscript` directive to print out the `uproot` version when the container runs

- **Hard:** Perform the CMS example analysis in a single execution using a definition file. Save the plots in the execution directory.
  - Hint: keep in mind where to store intermediate files.

- **Hard:** Set a Jupyter Notebook with the environment for running the analysis example, and execute it on the cells.

GitHub actions / Gitlab CI

- **From zero to hero (medium):** Start a new repository, add a main.py with a trivial function, a `test_main.py` with a trivial test and add CI that calls `pytest` to run the test
  - Option 1: Test on multiple python versions
  - Option 2: Create a file in your python script and upload it as an artifact (optionally as a second step)

- **LaTeX documents (GitHub; hard):** Create a simple LaTeX document and compile it using [github-action-for-latex]
- **Build a docker container (GitHub; hard):** Pick a docker image and build it in the CI using [build-and-push-docker-images]