How to run update pipeline of the Coronavirus Structural Task Force

The update pipeline begins with the weekly updates on wednesday with Max' part, followed by the updates from Yunyun and Gianluca.
All in all, it looks out for new SARS-CoV and SARS-CoV-2 structures in the PDB or updated versions of already found PDB entries. Afterwards, a few tools run over those new structures and generate some additional data.

Update Part 1: Finding new or revised structures

This part is run first and updates the database as well as the folders in the repository. Next, follows an extensive step by step instruction for running the update:

1. cd to the directory with the git repo and check your branch

```
~/coronavirus_structural_task_force$ git status
```

2. switch to the master branch and download the latest state. Repeat for the update branch (pipeline2_integration)

```
~/coronavirus_structural_task_force$ git checkout master
~/coronavirus_structural_task_force$ git pull
~/coronavirus_structural_task_force$ git checkout pipeline2_integration
~/coronavirus_structural_task_force$ git pull
```

3. Make sure, that you are on pipeline2_integration. Now merge the master into this branch. When you do git status, it tells you the current branch.

```
~/coronavirus_structural_task_force$ git merge master
```

4a. now cd into the folder with the update scripts and launch them. First start it for SARS-CoV-2.

```
~/coronavirus_structural_task_force$ cd utils/Update_pipeline/
~/coronavirus_structural_task_force/utils/Update_pipeline$ python3
tcp_run.py -t SARS-CoV-2
```

This process takes some time and runs the full update for SARS-CoV-2. After termination of the process, check the weekly report generated in the weekly_reports directory. If there is no line with "not_assigned", then the update ran fine and is ready. Otherwise, check step 5.
Also, check if the terminal shows at the end the line "Scan was succesful, no errors found!". If not, check step 6.

4b. Navigate back to the root of the repo and commit and push your changes.

```
~/coronavirus_structural_task_force/utils/Update_pipeline$ cd ../..
~/coronavirus_structural_task_force$ git add .
~/coronavirus_structural_task_force$ git commit -m "weekly update, YYYY-MM-
DD, SARS-CoV-2"
~/coronavirus_structural_task_force$ git push
```

Remember to enter the current date and to state the right taxonomy. If there are not_assigned proteins or other problems, you may state this in the message as well.
In order to push the update, you require an account with access to the repository. The push

command asks for your git username and a password. Recently, the use of passwords was omitted and you require an access token instead.

5. in case of not_assigned proteins, the script was not able to automatically determine to which protein the PDB entry belongs to, probably due to a too short sequence and/or sequence alignment. Skip to the next step, if this is not the case right now.
Checkout the online PDB page of the respective entry. This gives usually good clues to which protein the entry belongs to. Afterwards, you can manually assign the protein with:

```
~/coronavirus_structural_task_force/utils/Update_pipeline$ python3
tcp_manual.py -t SARS-CoV-2 -pdb <pdb-id> -p <protein_name>
```

where -t takes the taxonomy, -pdb the PDB identifier and -p the protein name, which should be exactly (!!!) as the corresponding name of the folder in the repository (spike for example is named “surface_glycoprotein”). Wrong names could destroy the functionality of following scripts.

6. If the regular run did not lead to the line “Scan was succesful, no errors found!”, errors were introduced into the database. This requires a good investigation of someone familiar with the code and shouldn’t be fatal for further scipts along the way. However, if you have to fix this, checkout the code of the scipt “analyze_and_fix_dataframe.py”. Use its functions to first, detect which kind of errors you have exactly and second, to fix directly wrong database entries. In order to fix such errors complelty for the future, you would also have to find out the source of the error.

7. After additional changes, remember to commit and push the changes.
Repeat steps 4a up to 6 for SARS-CoV.

8. If all changes are pushed, checkout to the master and merge all changes. Push the new commits from the merge as well.

```
~/coronavirus_structural_task_force$ git checkout master
~/coronavirus_structural_task_force$ git merge pipeline2_integration
~/coronavirus_structural_task_force$ git push
```