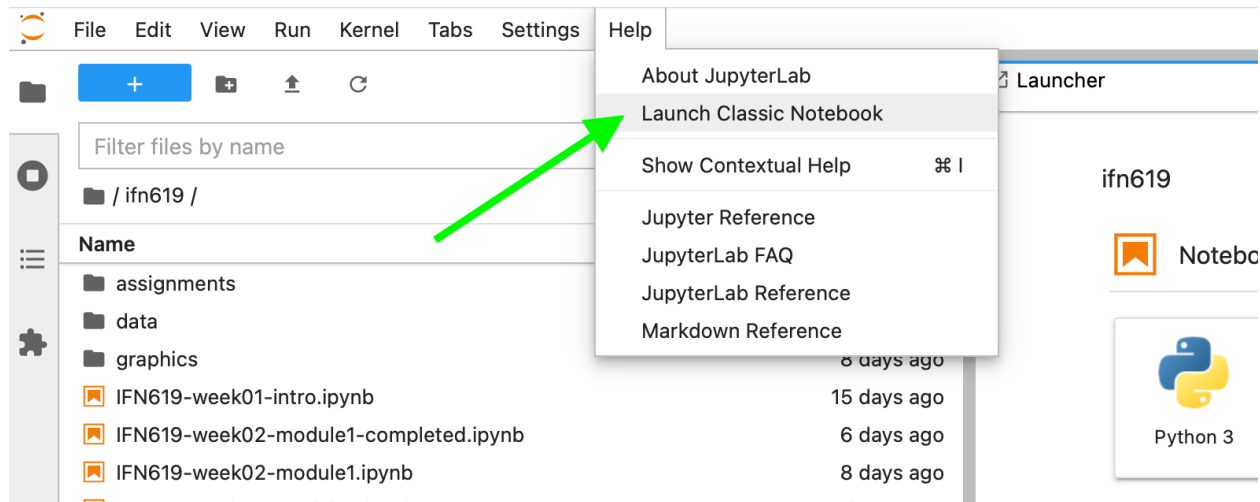
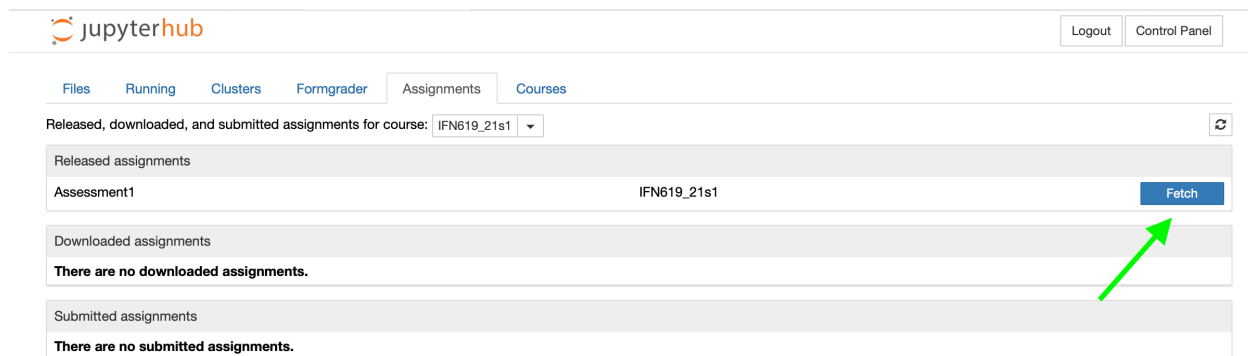


Instructions for IFN619 Assessment 1 Part A

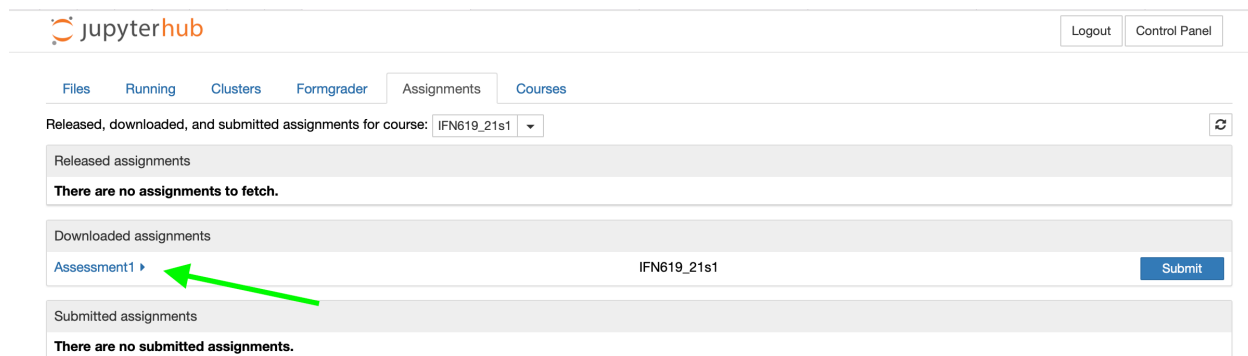
1. Log into your Jupyter environment, and select **Launch Classic Notebook** from the **Help** menu.



2. In the **Released assignments** section, click **Fetch**



3. Click on the assignment name to show the assignment, and then click on the assignment link to open it



Released, downloaded, and submitted assignments for course: IFN619_21s1

Released assignments

There are no assignments to fetch.

Downloaded assignments

Assessment1

IFN619_21s1

Submit

/home/jovyan/Assessment1/Assessment1_PartA_1.ipynb

Validate

Submitted assignments

There are no submitted assignments.

- The Assignment will open as a Jupyter notebook which you can edit and run cells. Ensure that you don't add any cells or your notebook might fail in the grading process. Ensure that you read the instructions carefully, add your answers under the **# YOUR CODE HERE** and run the cells to check your answers. Note, replace `raise NotImplementedError()` with your solution.

- Assessment 1 - Part A

Answer the questions below to demonstrate your understand of foundational techniques. Each question relates to a technique that has been addressed in the first 4 modules.

Each question will be marked separately, so if you are unsure about a question, simply try to answer it with your best possible attempt. You can run the test cells (after each answer cell) to check for simple errors.

DO NOT ADD OR REMOVE ANY CELL BLOCKS. PROVIDE YOUR ANSWERS IN THE INDICATED SPACES ONLY, AND DO NOT DELETE THE ### LINES.

1. Read the instructions carefully

1. Read a CSV file

After importing the required dataframes library as `pd`, read the contents of file `mountain_heights.csv` into a dataframe called `mountain_heights`. Use the mountain code for the index of the dataframe. Using a dataframe property, output the dataframe's rows and columns in the format `(rows,cols)`.

This solution requires 3 lines of code.

In []: # YOUR CODE HERE

`raise NotImplementedError()`

2. Replace this with your own code and run the cell

3. Run the following cell to do basic checks

In []: "Check that the dataframe has been loaded and rows and columns output"
`assert(type(mountain_heights)==type(pd.DataFrame()))`
`assert(_==(10,4))`

- When you are certain that everything is OK, make sure your notebook is saved and return to the assignments tab. Click **validate** to check that your notebook is OK, and then click **Submit**

Released, downloaded, and submitted assignments for course: IFN619_21s1

Released assignments

There are no assignments to fetch.

Downloaded assignments

Assessment1

IFN619_21s1

Submit

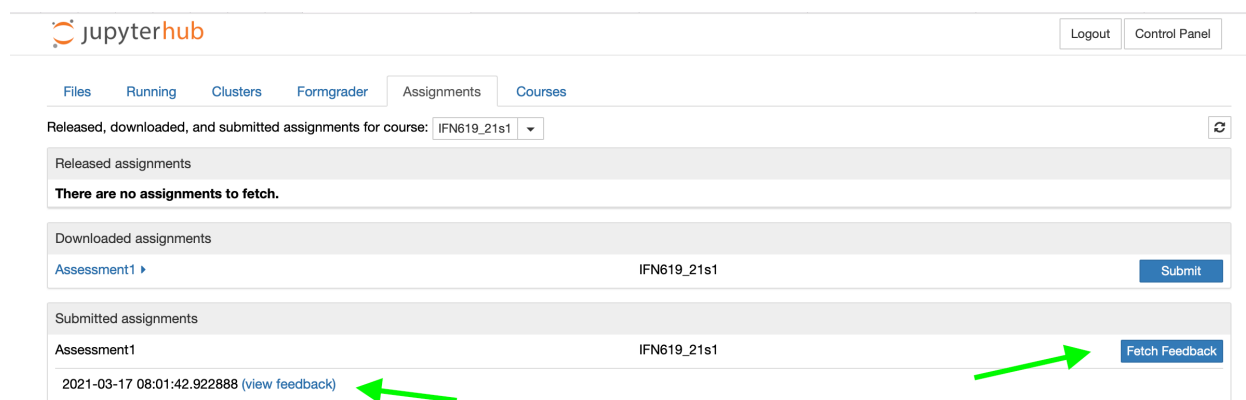
/home/jovyan/Assessment1/Assessment1_PartA_1.ipynb

Validate

Submitted assignments

There are no submitted assignments.

6. A member of the teaching team will run the autograder on your notebook (during the next run of grading). After this has occurred, click **Fetch Feedback**. If your notebook has been graded, then a **(view feedback)** link will appear beside your submission date. Click this link to open your feedback.



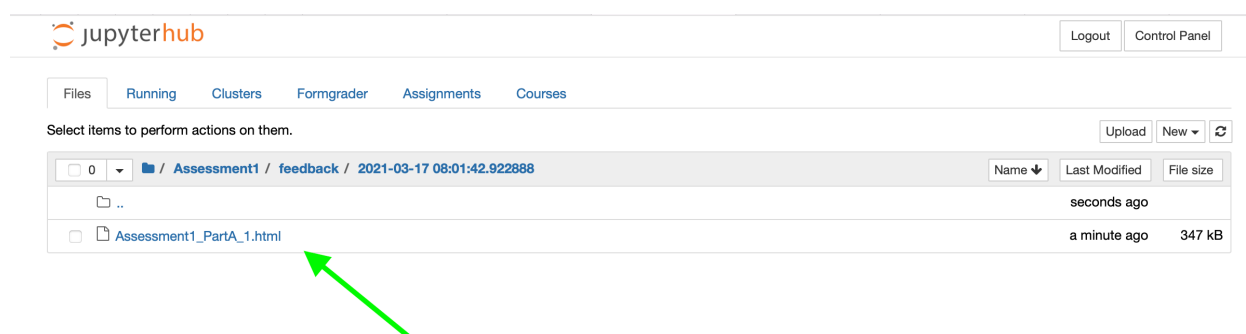
The screenshot shows the JupyterHub interface with the 'Assignments' tab selected. It displays a table of submitted assignments for the course 'IFN619_21s1'. The table has columns for the assignment name, submission date, and a 'Fetch Feedback' button. A green arrow points to the '(view feedback)' link next to the submission date '2021-03-17 08:01:42.922888'.

Released assignments
There are no assignments to fetch.

Downloaded assignments
Assessment1

Submitted assignments
Assessment1
2021-03-17 08:01:42.922888 (view feedback)

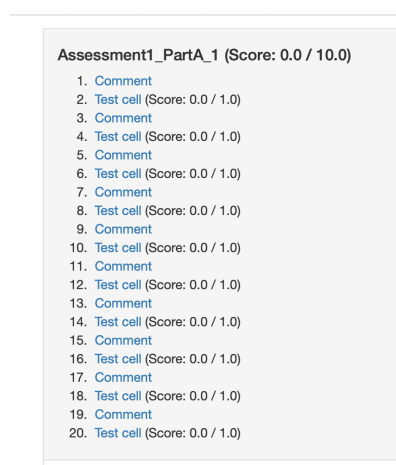
7. Your feedback is in a webpage which you can open from your Jupyter environment



The screenshot shows the JupyterHub interface with the 'Files' tab selected. It displays a file browser view of the 'Assessment1' directory. A green arrow points to the file 'Assessment1_PartA_1.html'.

Name	Last Modified	File size
Assessment1_PartA_1.html	a minute ago	347 kB

8. The feedback consists of a summary with your total score up the top and the scores for the 10 questions



The screenshot shows the feedback page for 'Assessment1_PartA_1' with a score of 0.0 / 10.0. It lists 20 items, each with a score of 0.0 / 1.0. The items are: 1. Comment, 2. Test cell, 3. Comment, 4. Test cell, 5. Comment, 6. Test cell, 7. Comment, 8. Test cell, 9. Comment, 10. Test cell, 11. Comment, 12. Test cell, 13. Comment, 14. Test cell, 15. Comment, 16. Test cell, 17. Comment, 18. Test cell, 19. Comment, 20. Test cell.

Assessment1_PartA_1 (Score: 0.0 / 10.0)

1. Comment
2. Test cell (Score: 0.0 / 1.0)
3. Comment
4. Test cell (Score: 0.0 / 1.0)
5. Comment
6. Test cell (Score: 0.0 / 1.0)
7. Comment
8. Test cell (Score: 0.0 / 1.0)
9. Comment
10. Test cell (Score: 0.0 / 1.0)
11. Comment
12. Test cell (Score: 0.0 / 1.0)
13. Comment
14. Test cell (Score: 0.0 / 1.0)
15. Comment
16. Test cell (Score: 0.0 / 1.0)
17. Comment
18. Test cell (Score: 0.0 / 1.0)
19. Comment
20. Test cell (Score: 0.0 / 1.0)

9. You can scroll down and look at any errors that have occurred with your answers, and view the tests that the grader uses to check your answers. You can then use this info to fix your code.

1. Read a CSV file

After importing the required dataframes library as `pd`, read the contents of file `mountain_heights.csv` into a dataframe called `mountain_heights`. Use the mountain code for the index of the dataframe. Using a dataframe property, output the dataframe's rows and columns in the format `(rows,cols)`.

This solution requires 3 lines of code.

```
In [1]: Student's answer (Top)

# YOUR CODE HERE
raise NotImplementedError()

Comments:
No response.
```

```
-----
NotImplementedError                                Traceback (most recent call last)
<ipython-input-1-15b94d1fa268> in <module>
      1 # YOUR CODE HERE
----> 2 raise NotImplementedError()

NotImplementedError:
```

```
In [2]: Grade cell: q1-tests Score: 0.0 / 1.0 (Top)

"Check that the dataframe has been loaded and rows and columns output"
assert(type(mountain_heights)==type(pd.DataFrame()))
assert(_==(10,4))
```

10. If you have errors, fix your code and resubmit following the same process again.
If you received 10/10, congratulations! You have finished part A.