

講義・演習環境アプリケーションハンズオン

— 1. nbgraderを用いた課題の手動採点 —

2025年4月23日

ご注意ください

- この資料はテスト環境にて作成しています。
- 実際のハンズオンセミナーの環境とはコース名、コンテンツ名、表示内容が多少異なります。

手動採点の演習内容

- 教師としてログインして課題を作成・公開。
- いったんログアウト。
- 学生としてログインして課題を取得、解答を記入、提出。
- いったんログアウト。
- 教師としてログインして提出物を回収・採点、フィードバックを公開。
- いったんログアウト。
- 学生としてログインしてフィードバックを取得。

※プライベートウィンドウや複数のブラウザを使用して、教師と学生の両方のアカウントでずっとログインしていても構いませんが、セミナーでは1つのブラウザとウィンドウで操作する手順を説明します。

課題の作成・公開（教師）

教師としてMoodleにログイン

- 以下のURLにアクセス。
 - <https://mcj-handson-cwh.vcp-handson.org/moodle/login/index.php>
- 割り当てられたユーザ名とパスワードでログイン。
 - 最初は教師用のアカウントを使用してログイン。

**割り当てられた教師用
アカウントでログイン**

user102t

.....

ログイン




[パスワードを忘れましたか?](#)

いくつかのコースにはゲストアクセスできます

ゲストとしてアクセスする

日本語 (ja) [クッキー通知](#)




demo moodle Home ダッシュボード **マイコース**   教1 [編集モード](#) 

マイコースに移動

ダッシュボード

タイムライン

次の7日 [日付で並べ替える](#) [活動タイプまたは活動名で検索する](#)



アクションを必要とする活動はありません。

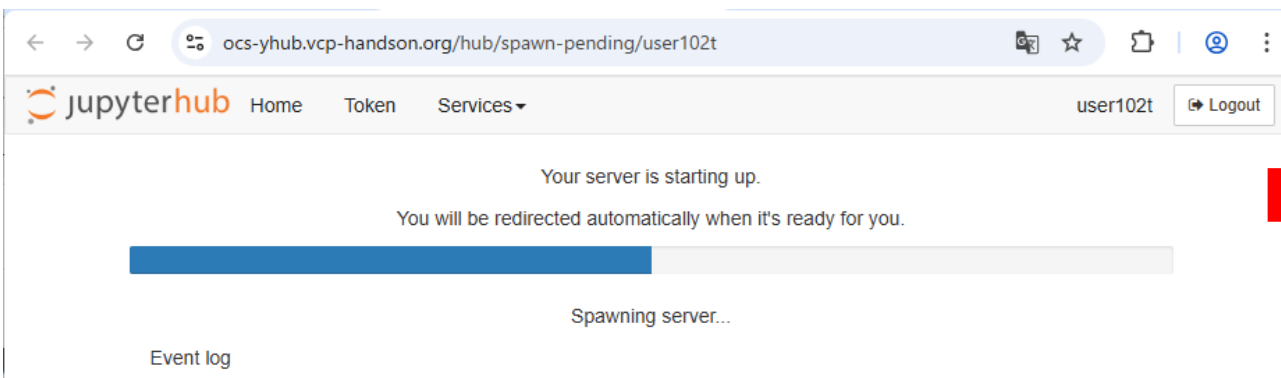
コースに移動してMCJ-CloudHubを起動

- マイコースに自分のユーザ番号に対応したコース) があるので、そのコースアクセス (画面例とは異なるかも)。
- コース内に設置された外部ツールをクリック。

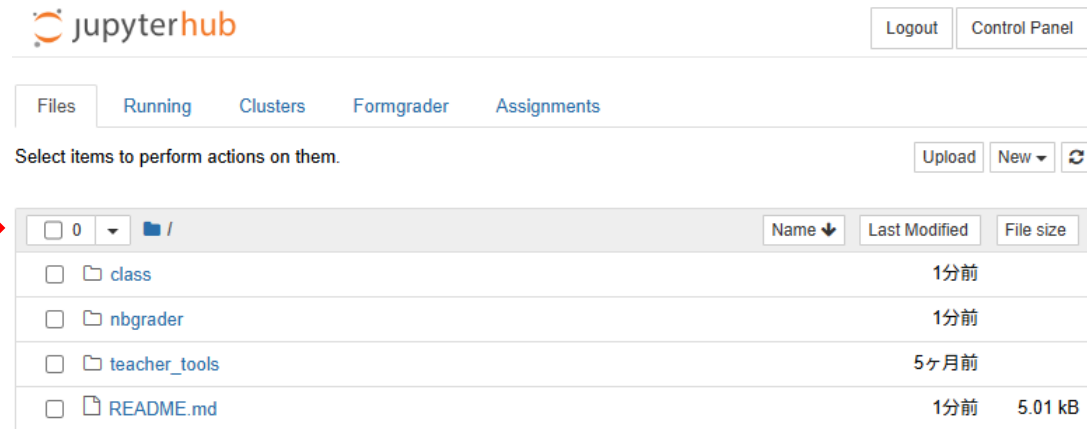
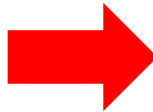
The image shows a two-step process for accessing MCJ-CloudHub from Moodle. On the left, the 'マイコース' (My Courses) page is displayed, showing a list of courses. A red dashed box highlights a course card for 'コース102' (Course 102). A red arrow points from this card to the right, where the 'コース102' (Course 102) page is shown. On this page, the 'MCJ-CloudHub' tool is highlighted with a red dashed box. Red text labels indicate the actions: 'クリックしてコースに移動' (Click to move to course) and 'クリックしてMCJ-CloudHubにアクセス' (Click to access MCJ-CloudHub).

Jupyter環境の起動

- MCJ-CloudHubにアクセスすると、各自のJupyter環境（プログラムの開発・実行環境）が起動。
 - 起動が完了するまで待機。



Jupyter環境が起動するまで待機



起動が完了するとファイルやフォルダの一覧が表示

課題の作成 (1/25)

- 画面上部の「Formgrader」をクリックして課題一覧画面に移動。
- 「Add new assignment...」をクリック。

jupyterhub

クリックして課題一覧画面に移動

Files Running Clusters **Formgrader** Assignments

Select items to perform actions on them.

0 /

- ☐ class
- ☐ nbgrader
- ☐ teacher_tools
- ☐ README.md



Instructions (click to expand)

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions
No data available in table								

+ Add new assignment... クリックして課題の登録を開始

課題の作成 (2/25)

- 課題の名称（今回はassign01）を入力して、「Save」をクリック。
- 課題一覧画面に戻ったら課題の名称（assgin01）をクリック。

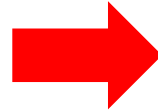
Add New Assignment ×

① 課題の名称を入力

Name

Due date (optional)

② クリックして課題を登録



Instructions (click to expand)

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions
assign01	None	draft						0

課題名をクリック

課題の作成 (3/25)

- 画面右上にある「new」をクリックして、「LC_wrapper」を選択。
- 空のJupyter Notebookファイルが作成される。

The screenshot shows the nbgrader web interface. At the top, there are tabs for 'Files', 'Running', 'Clusters', 'Formgrader', and 'Assignments'. Below these is a message 'Select items to perform actions on them.' and a file browser showing the path '/ nbgrader / course102 / source / assign01'. A red arrow points from the 'New' button in the top right to the 'LC_wrapper' option in the dropdown menu. The dropdown menu also lists 'Python 3 (ipykernel)', 'Text File', 'Folder', and 'Terminal'. Below the file browser, there is a message 'The notebook list is empty.' and a large red arrow pointing down to the Jupyter Notebook interface. The Jupyter Notebook interface shows a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. Below the menu bar is a toolbar with various icons. The main area of the notebook is empty, with a prompt 'In []:' at the top.

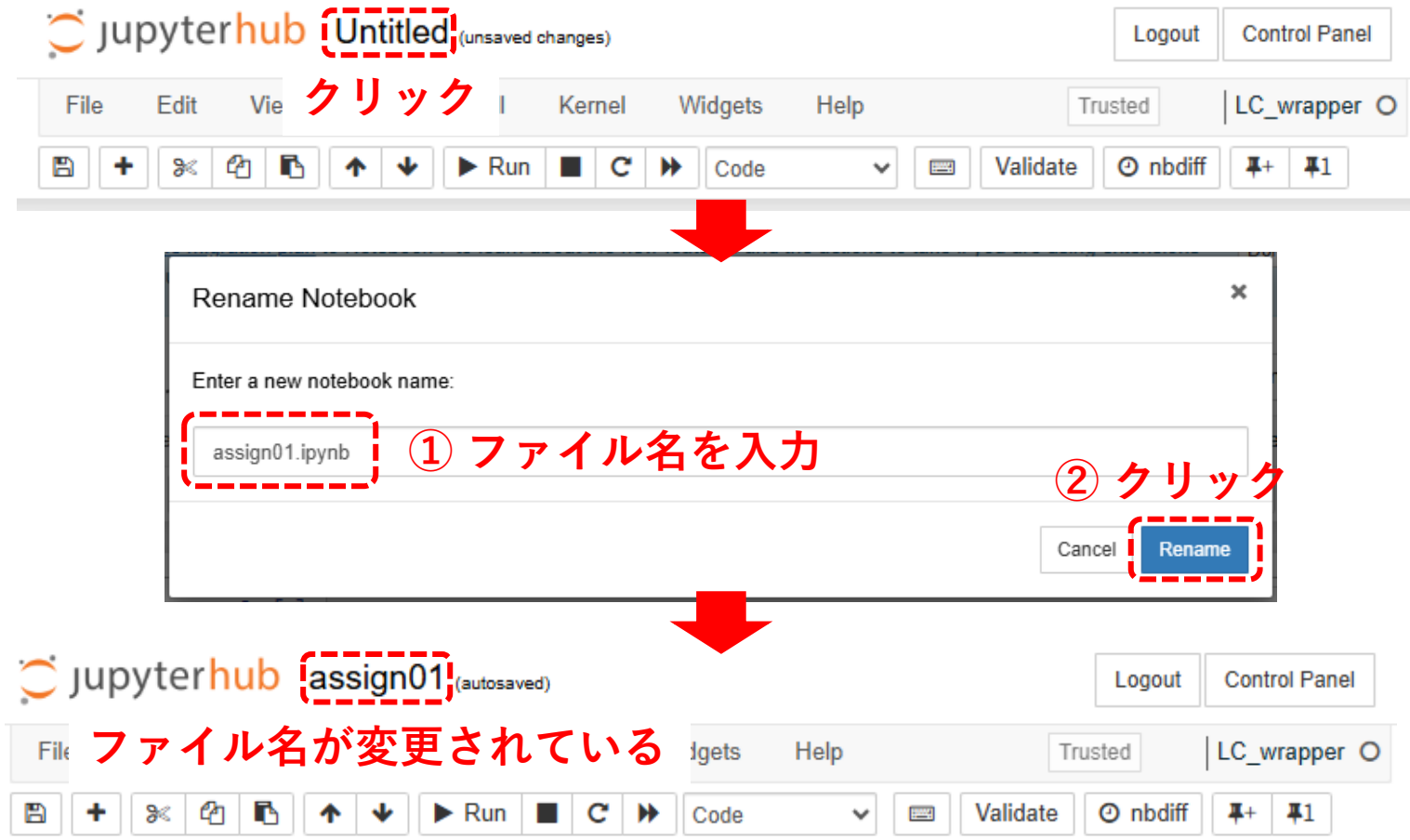
① クリック

② クリック

空のJupyter Notebook
ファイルが作成される

課題の作成 (4/25)

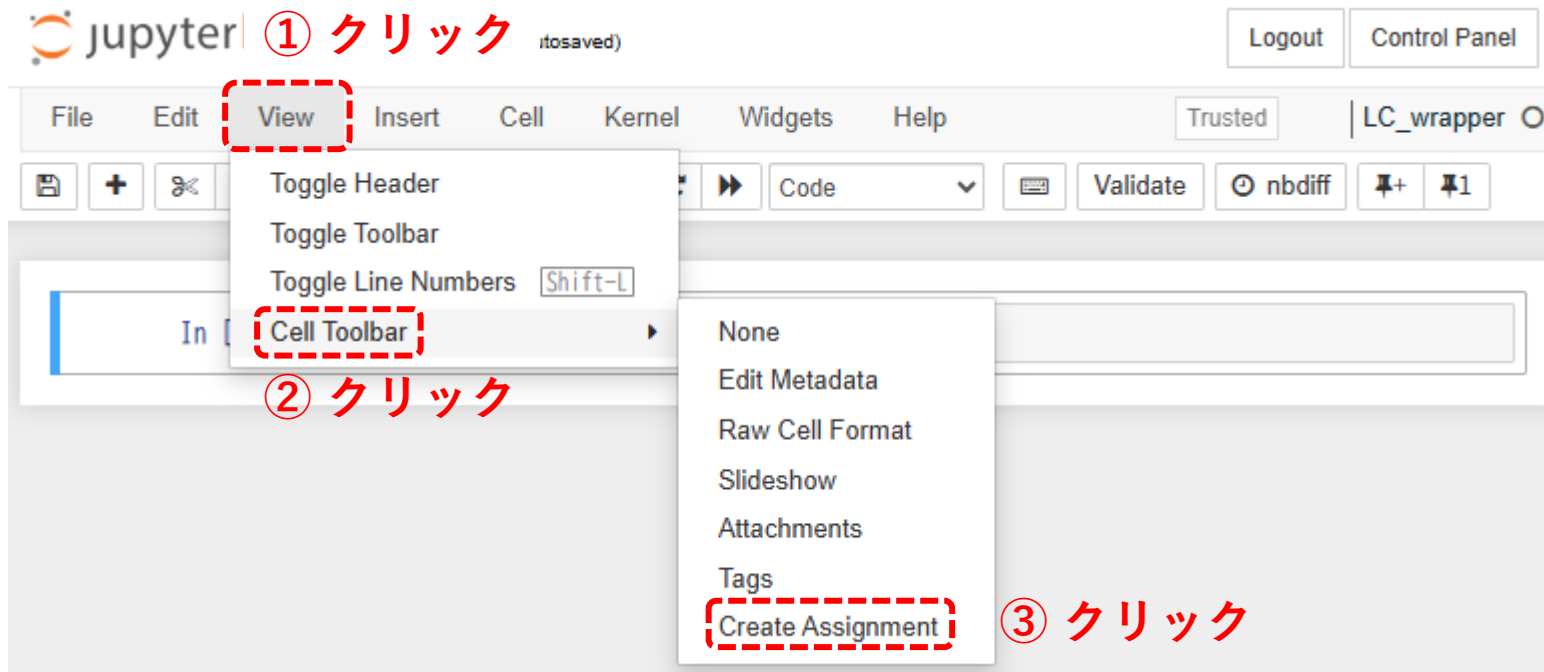
- 画面上部のファイル名（この時点ではUntitled）をクリック。
- ファイル名（assign01.ipynb）を入力して「Rename」をクリック。



今回は課題名とファイル名が同じですが、同じである必要はありません。実際には、異なる名前にしたほうが分かりやすいと思います。

課題の作成 (5/25)

- メニューバーの中から「View」をクリックした後、「Cell Toolbar」、「Create Assignment」を順番に選択。
- セルの右上にリストボックスが表示。

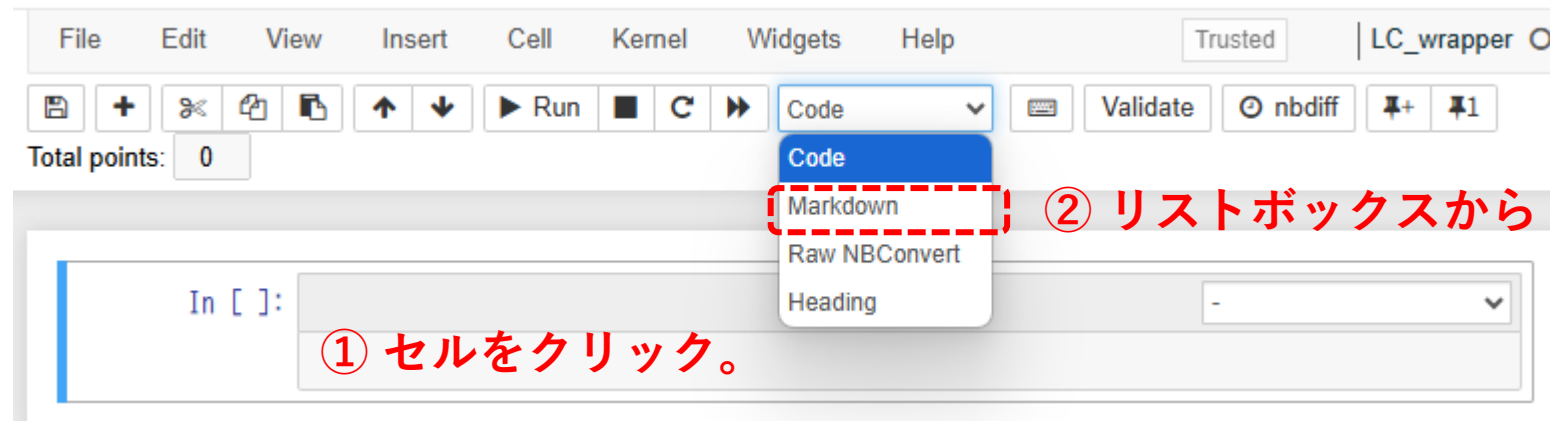


セルの右上にリスト
ボックスが表示



課題の作成 (6/25)

- 最初のセルをクリックしてそこにフォーカスを移動。
- 画面上部のリストボックスをクリックした後、「Markdown」を選択。

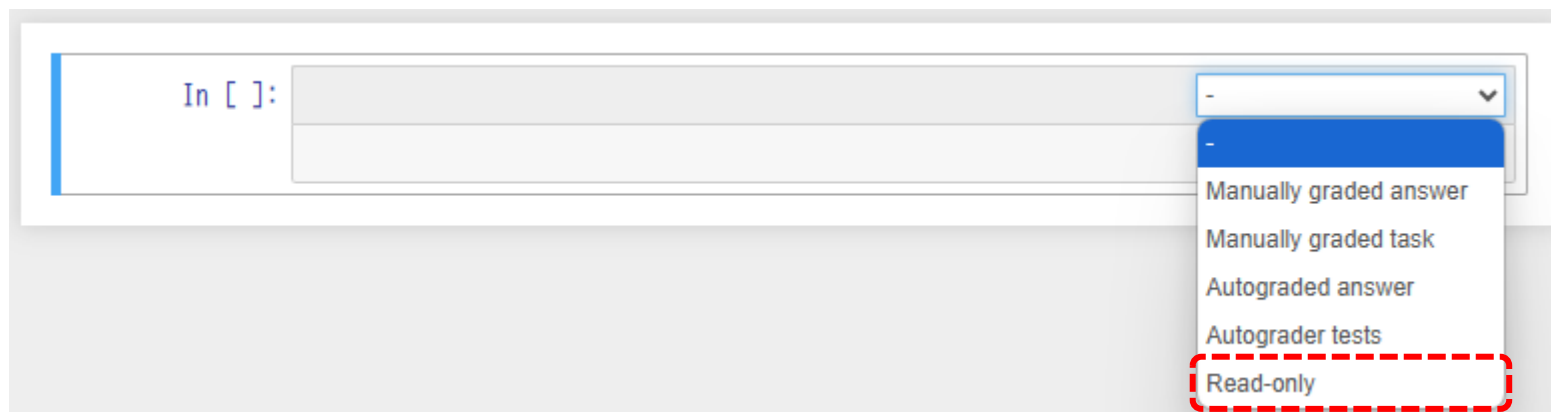


① セルをクリック。

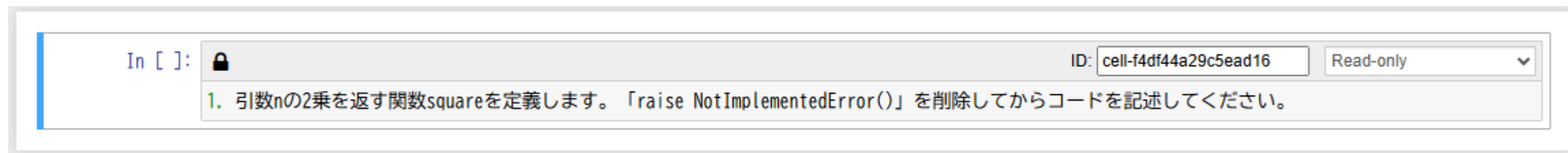
② リストボックスから「Markdown」を選択。

課題の作成 (7/25)

- セルの右上にあるリストボックスの中から「Read-only」を選択。
- セルの中に、以下のような問題文を記述。



① リストボックスから「Read-only」を選択



課題の作成 (8/25)

- 先程のセルを実行（次のセルが作成される）。

① クリックしてセルを実行



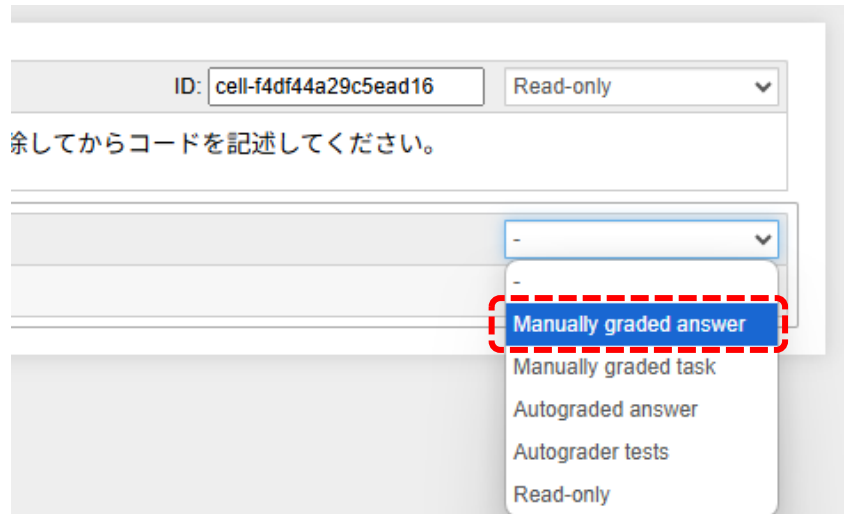
1. 引数nの2乗を返す関数squareを定義します。「raise NotImplementedError()」を削除してからコードを記述してください。

In []:

② 次のセルが作成される

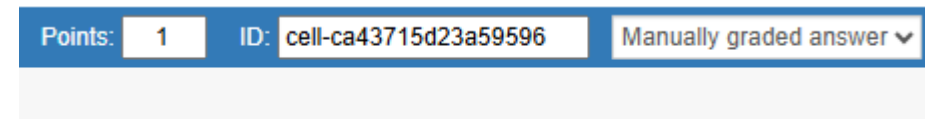
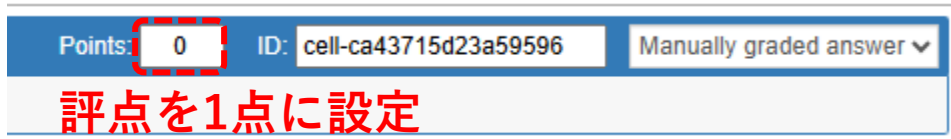
課題の作成 (9/25)

- セルの右上のリストボックスをクリックした後、「Manually graded answer」を選択。



① リストボックスから「Manually graded answer」を選択

- セルに評点の項目が追加されるので、1点に設定。



課題の作成 (10/25)

- セルの中に以下のようなコードを記述した後、セルを実行。



```
In [ ]:
```

```
def square(n):  
    ### BEGIN SOLUTION ###  
    return n*n  
    ### END SOLUTION ###
```

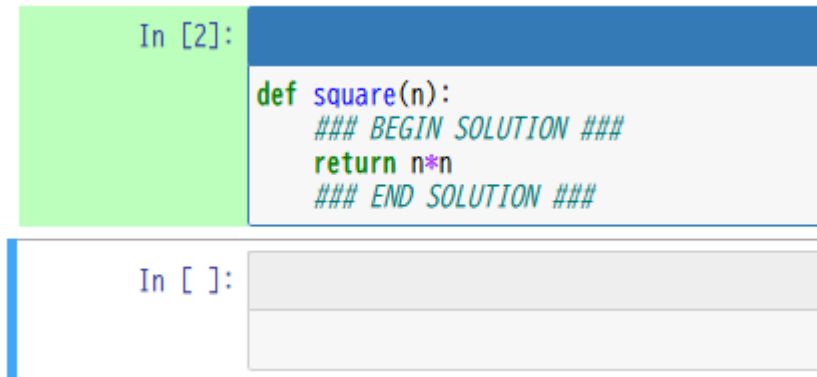
jupyterhub assign01 Last Checkpoint: 1時間前 (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Run

クリックしてセルを実行

- エラーが発生しなければOK (次のセルは自動的に作成される)。



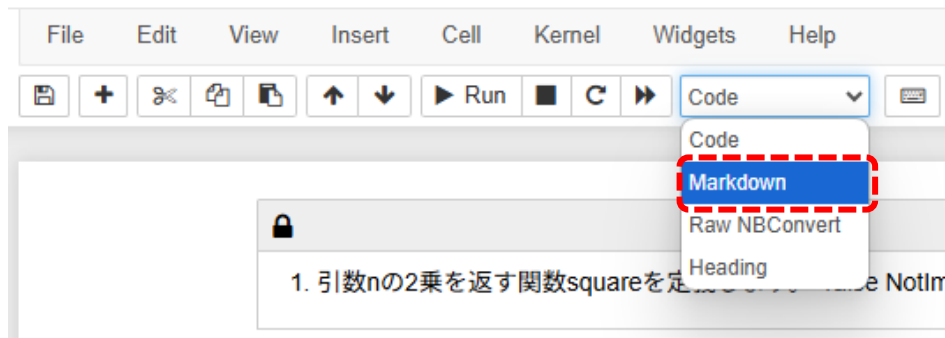
```
In [2]:
```

```
def square(n):  
    ### BEGIN SOLUTION ###  
    return n*n  
    ### END SOLUTION ###
```

```
In [ ]:
```

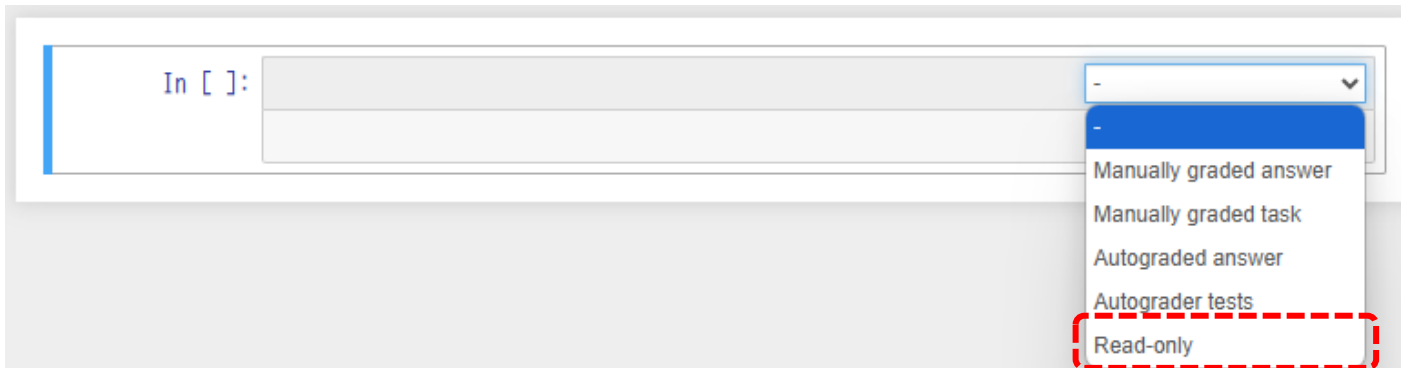
課題の作成 (11/25)

- 作成されたセルをクリックした後、画面上部のリストボックスの中から「Markdown」を選択。



リストボックスから
「Markdown」を選択

- セルの右上にあるリストボックスの中から「Read-only」を選択。

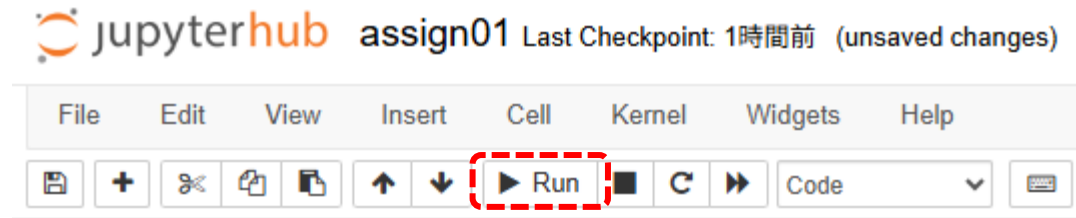


リストボックスから
「Read-only」を選択

課題の作成 (12/25)

- 先程のセルの中に以下のような文章を入力した後、セルを実行。

このセルでプログラムの計算結果を確認してみましょう。



クリックしてセルを実行

- 次のセルが自動的に作成される。

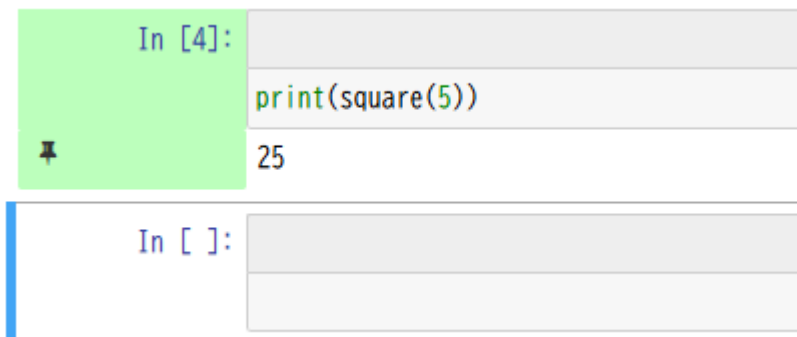


課題の作成 (13/25)

- 作成されたセルの中に以下のようなコードを入力した後、セルを実行。



- 実行結果が表示されたらOK（次のセルが自動的に作成される）。

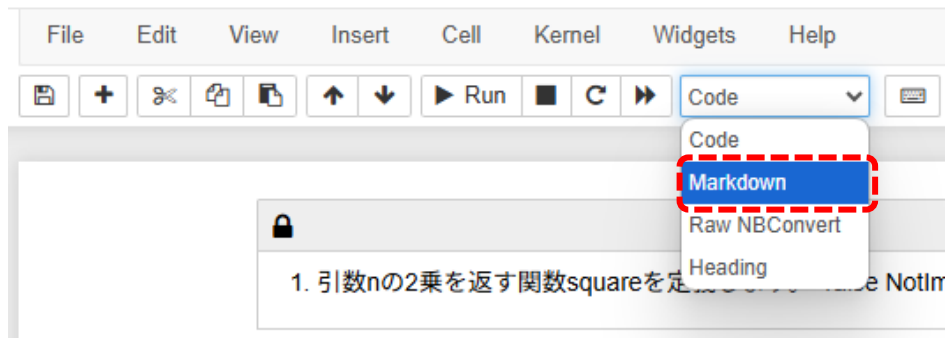


補足

- これまでの操作でプログラムを手動で採点する問題を1問を作成できましたが、今回は第2問として、プログラムではなく文章を入力してもらい、手動で採点するセルを作成します。

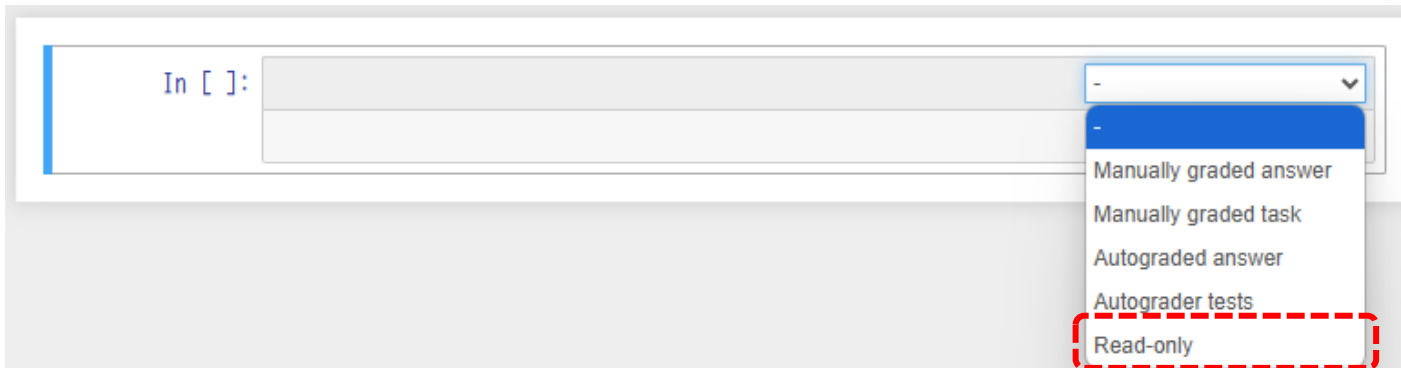
課題の作成 (14/25)

- 先程作成されたセルをクリックした後、画面上部のリストボックスの中から「Markdown」を選択。



リストボックスから
「Markdown」を選択


- セルの右上にあるリストボックスの中から「Read-only」を選択。



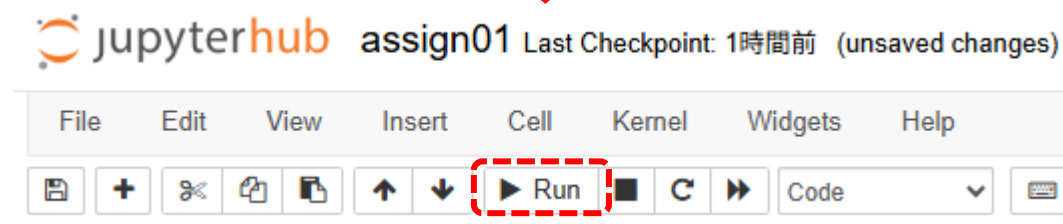
リストボックスから
「Read-only」を選択

課題の作成 (15/25)

- 先程のセルの中に以下のような文章を入力した後、セルを実行。

 ID: cell-4be656d816cc34d2 Read-only ▾

2. 以下に何か感想を入力してください。セルをダブルクリックした後、「YOUR ANSWER HERE」を削除してから感想を入力してください。



クリックしてセルを実行

- 次のセルが自動的に作成される。

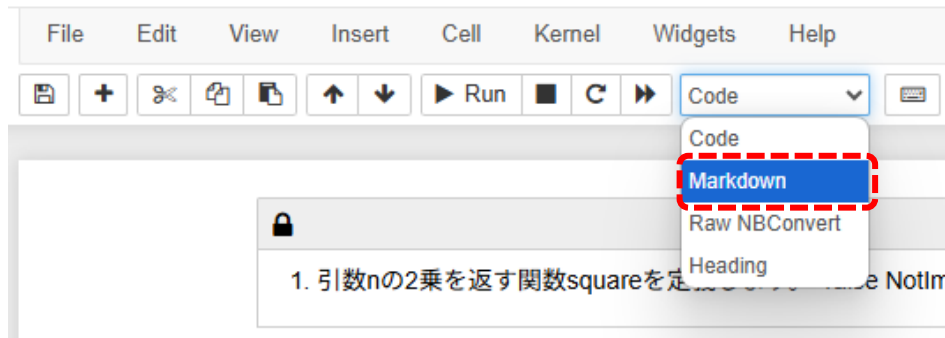
 ID: cell-4be656d816cc34d2 Read-only ▾

2. 以下に何か感想を入力してください。セルをダブルクリックした後、「YOUR ANSWER HERE」を削除してから感想を入力してください。

In []:

課題の作成 (16/25)

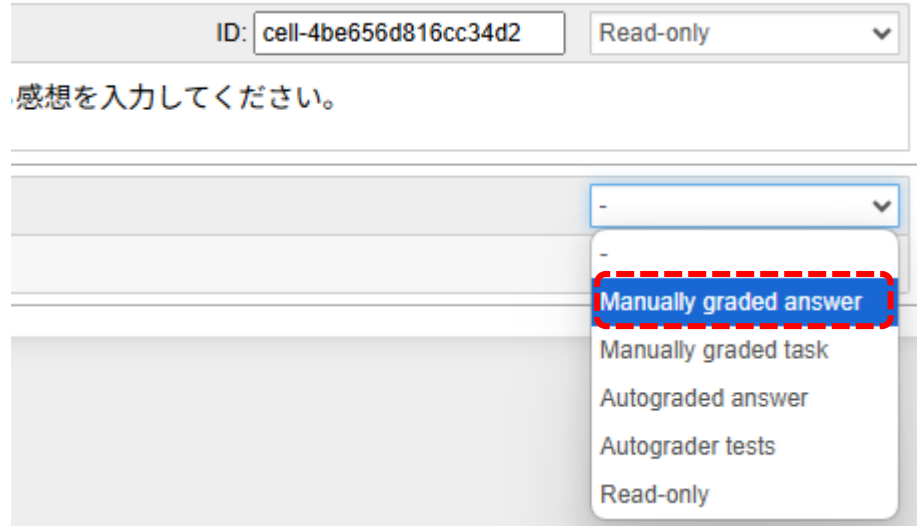
- 作成されたセルをクリックした後、画面上部のリストボックスの中から「Markdown」を選択。



リストボックスから
「Markdown」を選択

課題の作成 (17/25)

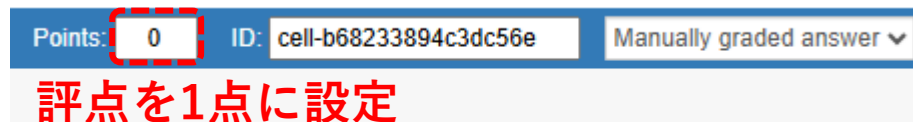
- セルの右上のリストボックスをクリックした後、「Manually graded answer」を選択。



The screenshot shows a form with a header bar containing 'ID: cell-4be656d816cc34d2' and a 'Read-only' dropdown. Below the header is a text input field with the placeholder text '感想を入力してください。'. A dropdown menu is open, showing options: '-', 'Manually graded answer' (highlighted with a red dashed box), 'Manually graded task', 'Autograded answer', 'Autograder tests', and 'Read-only'.

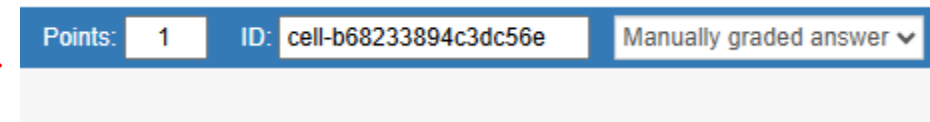
① リストボックスから「Manually graded answer」を選択

- セルに評点の項目が追加されるので、1点に設定。



The screenshot shows a form with a header bar containing 'Points: 0' (the '0' is highlighted with a red dashed box), 'ID: cell-b68233894c3dc56e', and a 'Manually graded answer' dropdown. Below the header is a text input field.

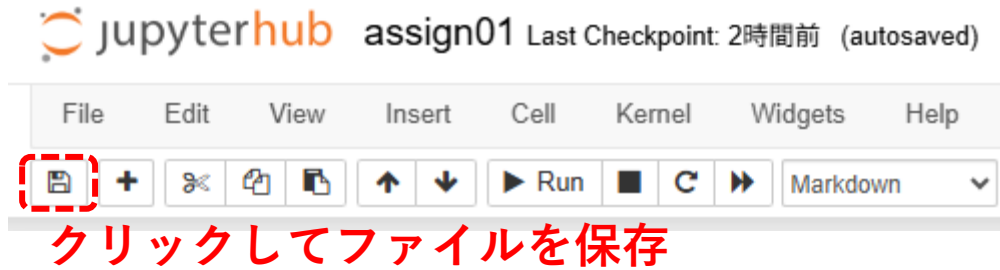
評点を1点に設定



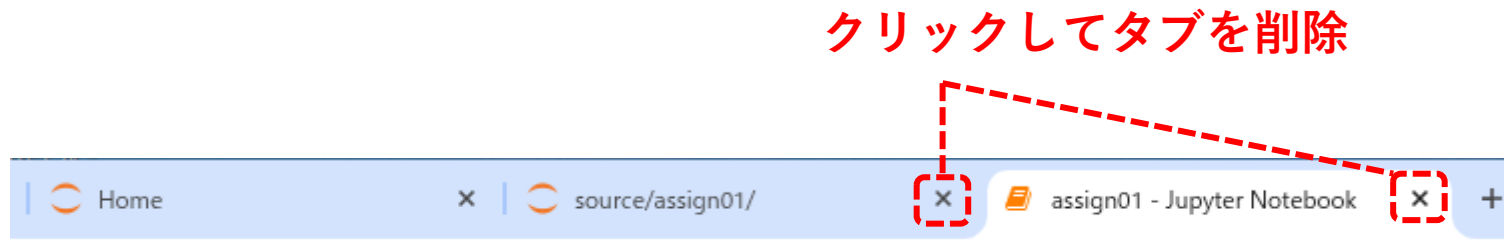
The screenshot shows a form with a header bar containing 'Points: 1', 'ID: cell-b68233894c3dc56e', and a 'Manually graded answer' dropdown. Below the header is a text input field.

課題の作成 (18/25)

- 画面左上のディスクのアイコンをクリックして、ファイルを保存。

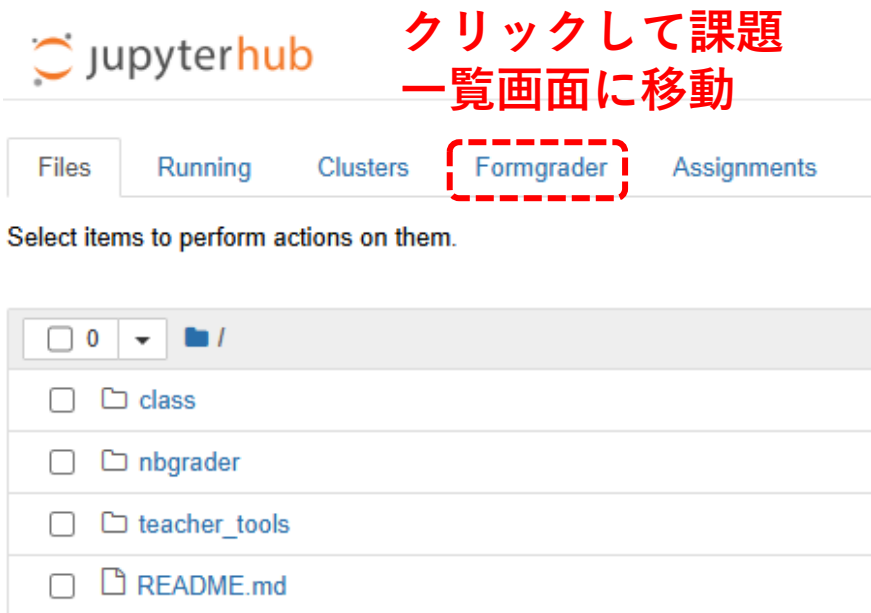


- タブを削除していき、ホームに戻る。



課題の作成 (19/25)

- 画面上部の「Formgrader」をクリックして課題一覧画面に移動。
- 課題「assign01」の行にある「Generate」のアイコンをクリック。

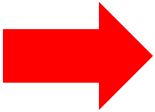
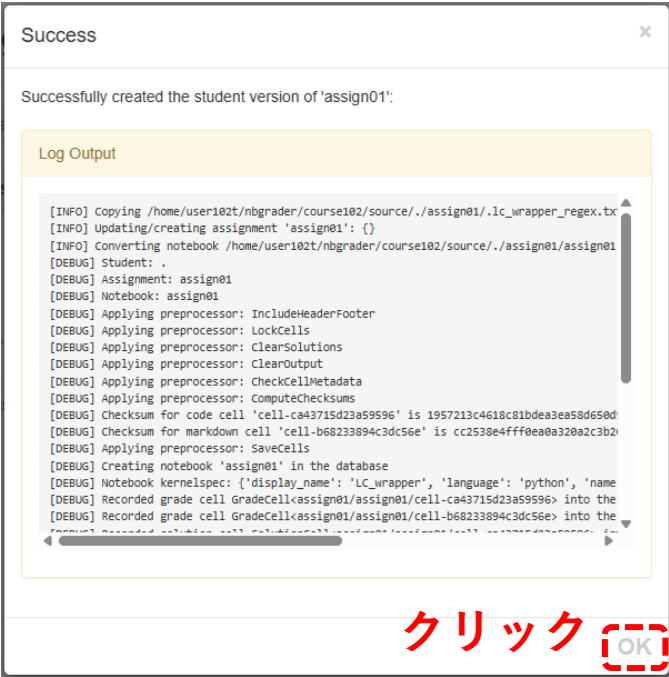


Instructions (click to expand)									
Name ↓	Due Date ↑	Status ↑	Edit	Generate	Preview	Release	Collect	# Submissions	
assign01	None	draft						0	
+ Add new assignment...									


クリックして配布用ファイルを作成。

課題の作成 (20/25)

- 「Success」の画面が表示されたら画面下部の「OK」をクリック。
- 続いて、「Preview」のアイコンをクリック。



Instructions (click to expand)

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions
assign01	None	draft						0
+ Add new assignment...								

クリックして配布用ファイルを表示

課題の作成 (21/25)

- ファイルの一覧が表示されたら、配布用ファイルをクリック。

Select items to perform actions on them.



- ファイルの内容を確認。

1. 引数nの2乗を返す関数squareを定義します。「raise NotImplementedError()」を削除してからコードを記述してください。

```
In [ ]: def square(n):  
        # YOUR CODE HERE  
        raise NotImplementedError()
```

このセルでプログラムの計算結果を確認してみましょう。

```
In [ ]: print(square(5))
```

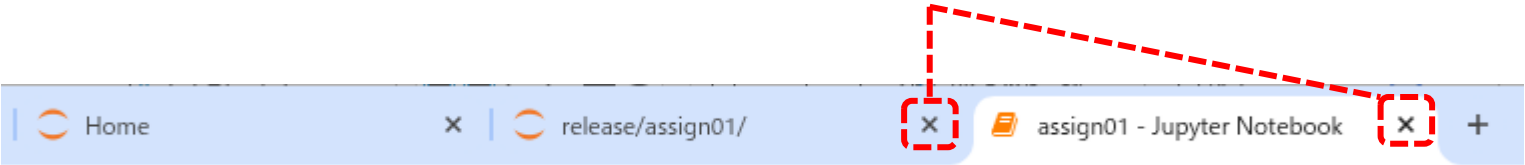
2. 以下に何か感想を入力してください。セルをダブルクリックした後、「YOUR ANSWER HERE」を削除してから感想を入力してください。

YOUR ANSWER HERE

課題の作成 (22/25)

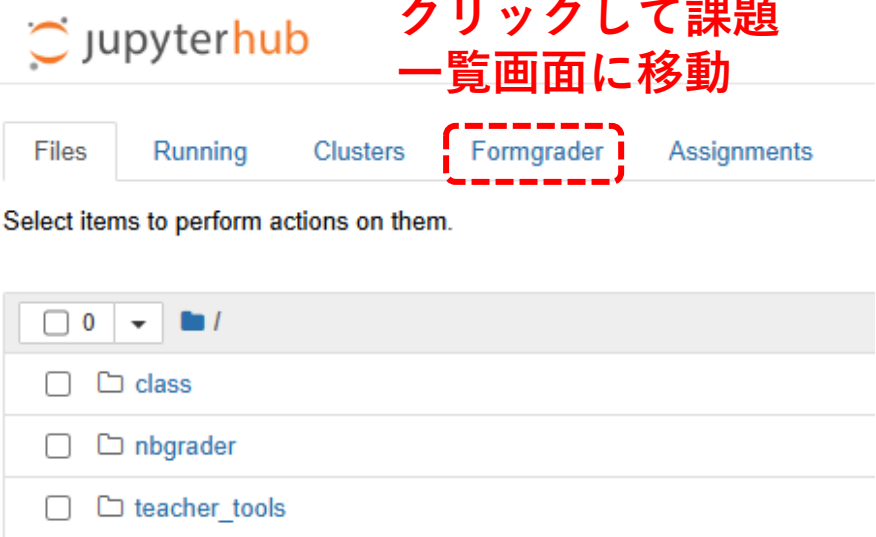
- タブを削除していき、ホームに戻る。

クリックしてタブを削除



- 「assign01」の行の「Release」のアイコンをクリック。

クリックして課題
一覧画面に移動



Instructions (click to expand)									
Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	
assign01	None	draft						0	

クリックして課題の公開を開始

課題の作成 (23/25)

- 「Success」の画面が表示されたら画面下部の「OK」をクリック。



Instructions (click to expand)									
Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	
assign01	None	released						0	

回収アイコンや提出数が表示される

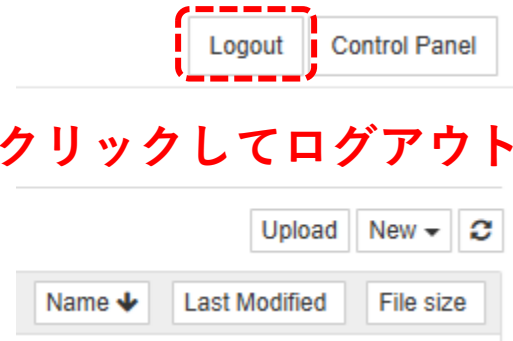
- タブを削除して、ホームに戻る。

クリックしてタブを削除



課題の作成 (24/25)

- 画面右上の「Logout」をクリックしてMCJ-CloudHubからログアウト。



- MCJ-CloudHubのタブを削除。



課題の作成 (25/25)

- Moodleからもログアウト。

① 画面右上のユーザのアイコンをクリック



② クリックしてログアウト

課題の取得・記入・提出（学生）

学生としてMoodleにログイン

- 学生用のアカウントを使用してMoodleにログイン。

**割り当てられた学生用
アカウントでログイン**

user102s

.....

ログイン

[パスワードを忘れましたか?](#)

いくつかのコースにはゲストアクセスできます

ゲストとしてアクセスする

日本語 (ja) [クッキー通知](#)



demo moodle Home ダッシュボード マイコース   学1 [編集モード](#)

マイコースに移動

ダッシュボード

タイムライン

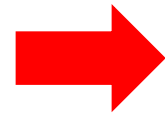
次の7日 [日付で並べ替える](#) [活動タイプまたは活動名で検索する](#)



アクションを必要とする活動はありません。

コースに移動してMCJ-CloudHubを起動

- マイコースに自分のユーザ番号に対応したコース）があるので、そのコースアクセス（画面例とは異なるかも）。
- コース内に設置された外部ツールをクリック。



コース102

コース 参加者 評価 コンビテンシ

▼ 一般

すべてを折りたたむ



アナウンスメント

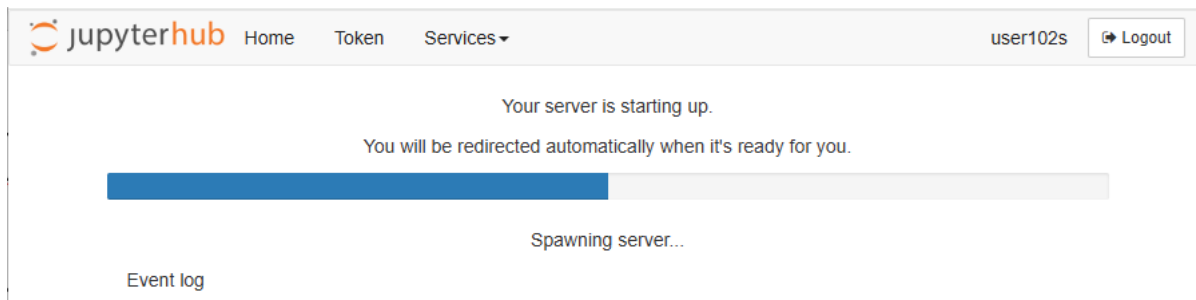


MCJ-CloudHub

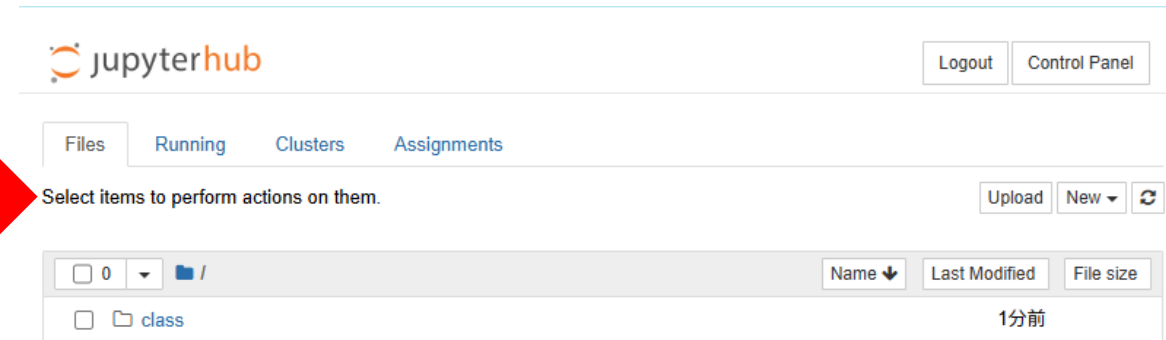
クリックしてMCJ-CloudHubに
アクセス

Jupyter環境の起動

- MCJ-CloudHubにアクセスすると、各自のJupyter環境（プログラムの開発・実行環境）が起動。
 - 起動が完了するまで待機。



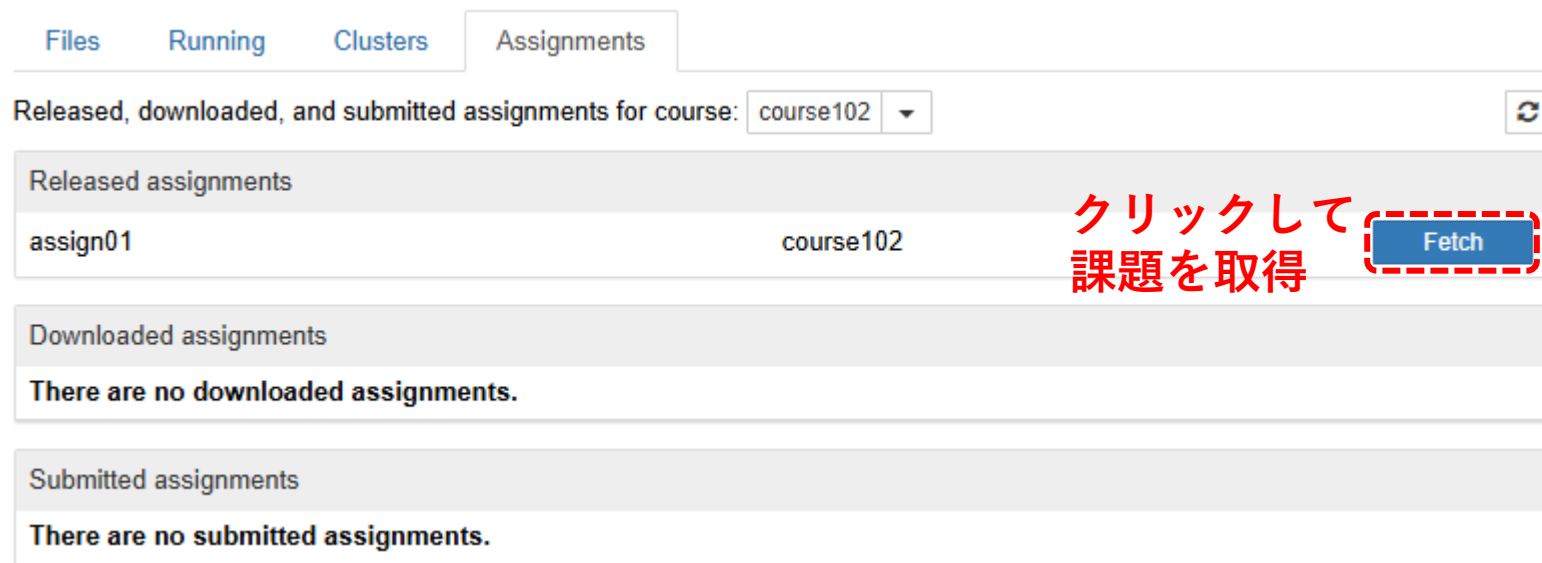
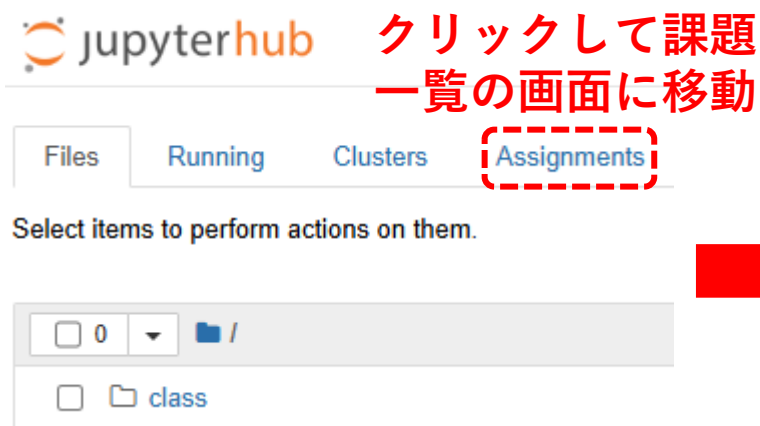
Jupyter環境が起動するまで待機



起動が完了するとファイルやフォルダの一覧が表示

課題の取得 (1/2)

- 画面上部の「Assignments」をクリックして課題一覧画面に移動。
- 「assign01」が公開されているので、「Fetch」をクリック。



課題の取得 (2/2)

- 取得済み課題の中に「assign01」が表示されるので、課題名をクリック。

Released assignments		
There are no assignments to fetch.		

Downloaded assignments		
assign01 ▶	course102	Submit

課題名をクリックして、課題に含まれるファイルを表示

- 課題に含まれるファイルが表示されるので、「assign01」をクリック。

Downloaded assignments		
assign01 ▼	course102	Submit

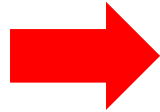
assign01	Validate
----------	----------

ファイル名をクリックして編集を開始

解答の作成 (1/5)

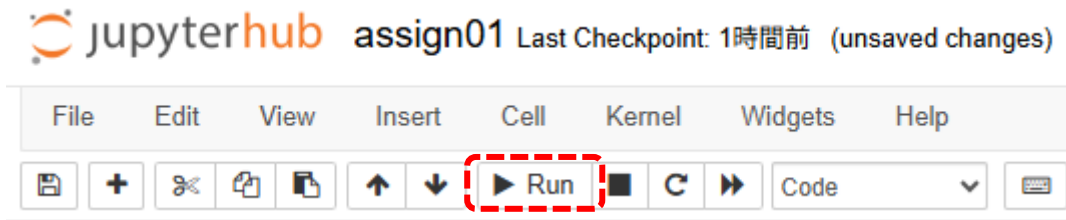
- プログラムを記入する課題の場合、「raise NotImplementedError()」を削除した後に解答を記入。

```
In [ ]: def square(n):  
        # YOUR CODE HERE  
        raise NotImplementedError()
```



```
In [ ]: def square(n):  
        # YOUR CODE HERE  
        return n*n
```

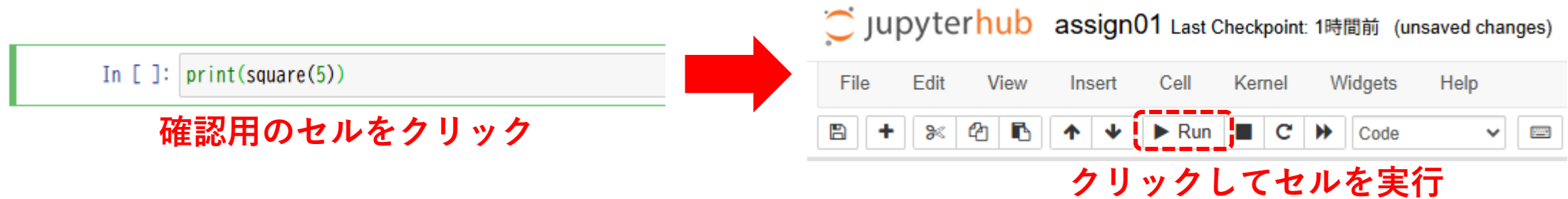
- プログラムを記入できたらセルを実行。



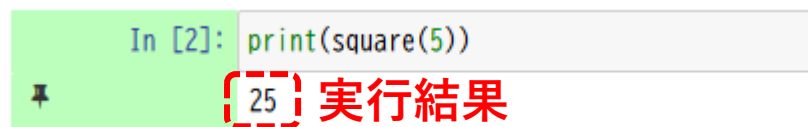
クリックしてセルを実行

解答の作成 (2/5)

- プログラムを実行してエラーが表示されなければ、確認用のセルをクリックした後、そのセルを実行。



- 実行結果の表示に基づいて、解答の正誤を判断。
 - 教師側はそうように確認用のセルを作成すること。



解答の作成 (3/5)

- 文章を入力する課題では、セルをダブルクリックした後、「YOUR ANSWER HERE」を削除。

2. 以下に何か感想を入力してください。セルをダブルクリックした後、「YOUR ANSWER HERE」を削除してから感想を入力してください。

YOUR ANSWER HERE ダブルクリックした後、この文を削除。

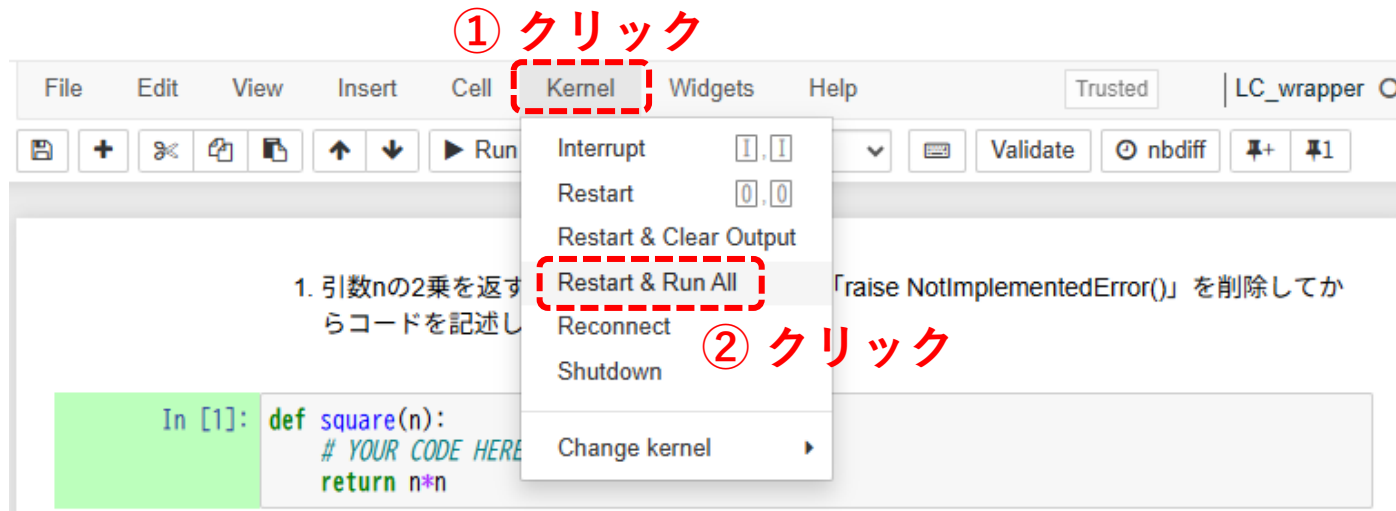
- セルに解答を記入。

2. 以下に何か感想を入力してください。セルをダブルクリックした後、「YOUR ANSWER HERE」を削除してから感想を入力してください。

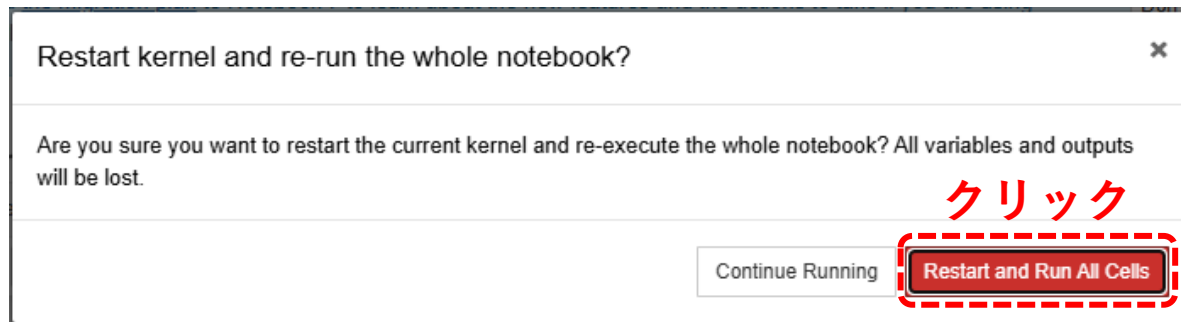
今回の課題は簡単でした。 解答を記入。

解答の作成 (4/5)

- 画面上部のメニューバーの中から「Kernel」をクリックし、

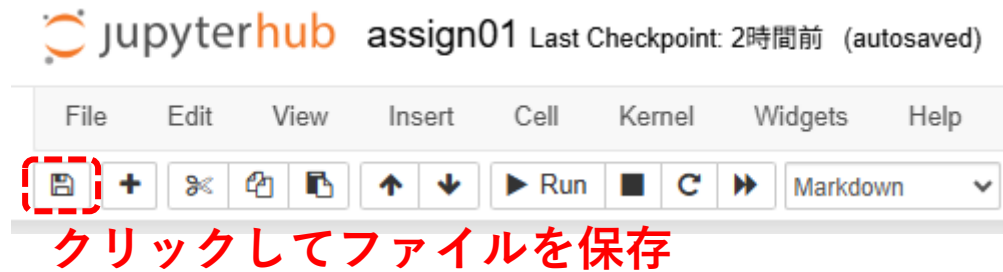


- 「Restart and Run All Cells」をクリック。エラーが発生しなければOK。

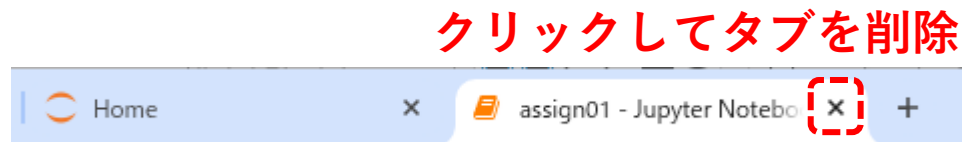


解答の作成 (5/5)

- 画面左上のディスクのアイコンをクリックして、ファイルを保存。

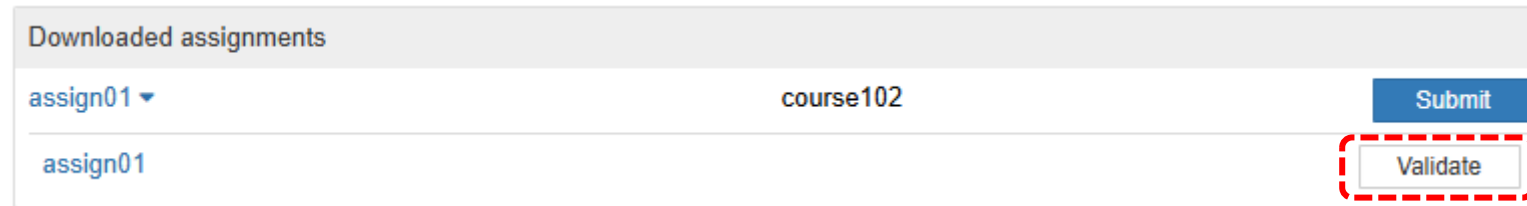


- タブを削除してホームに戻る。



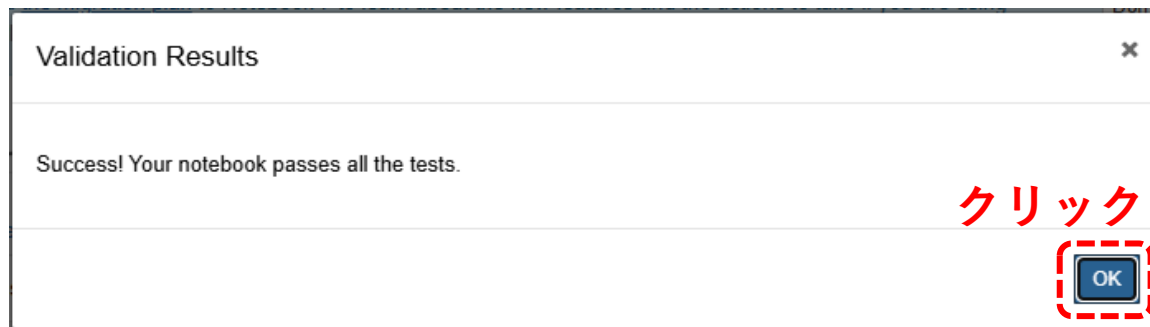
解答の提出 (1/4)

- 課題ファイルについて「Validate」をクリック。



クリックして
ファイルを検証

- エラーが表示されず、以下の画面が表示されたらOK。
 - 「OK」をクリックして元の画面に戻る



クリック

解答の提出 (2/4)

- 課題「assign01」について、「Submit」をクリック。

Downloaded assignments	
assign01 ▼	course102
assign01	

クリックして
解答を提出

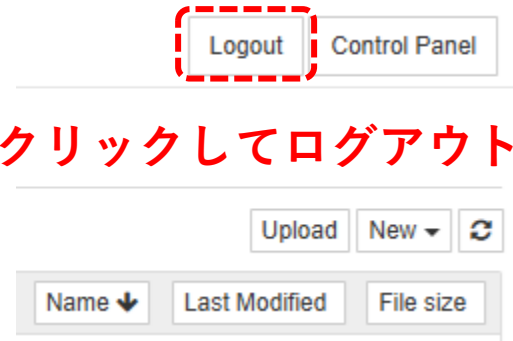
- 提出が完了すると、フィードバックを取得するためのボタンが表示。

Downloaded assignments	
assign01 ▼	course102
assign01	

Submitted assignments	
assign01	course102
2025-04-21 05:39:45 JST	

解答の提出 (3/4)

- 画面右上の「Logout」をクリックしてMCJ-CloudHubからログアウト。



- MCJ-CloudHubのタブを削除。



解答の提出 (4/4)

- Moodleからもログアウト。

① 画面右上のユーザのアイコンをクリック



② クリックしてログアウト

提出物の回収・採点（教師）

教師としてMoodleにログイン

- 教師用のアカウントを使用してMoodleにログイン。

割り当てられた教師アカウントでログイン

[パスワードを忘れましたか?](#)

いくつかのコースにはゲストアクセスできます

日本語 (ja)



demo moodle Home ダッシュボード **マイコース**

マイコースに移動

ダッシュボード

タイムライン

アクションを必要とする活動はありません。

コースに移動してMCJ-CloudHubを起動

- マイコースに自分のユーザ番号に対応したコース) があるので、そのコースアクセス (画面例とは異なるかも)。
- コース内に設置された外部ツールをクリック。

demo moodle Home ダッシュボード マイコース

マイコース

コース概要

すべて ▾ 検索 コース名で並べ替える ▾ カード ▾

コース102
カテゴリ1

クリックして
コースに移動

コース102

コース 設定 参加者 評価 レポート さらに ▾

一般

すべてを折りたたむ

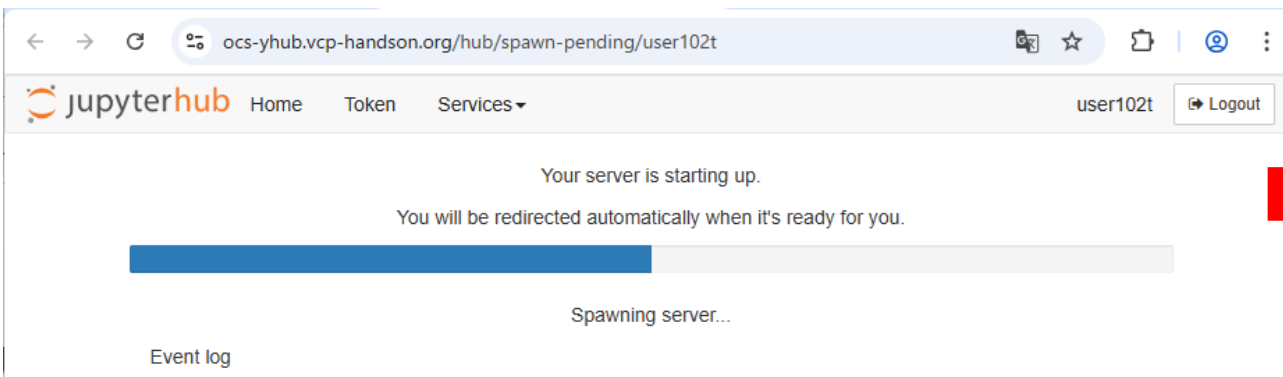
アナウンスメント

MCJ-CloudHub

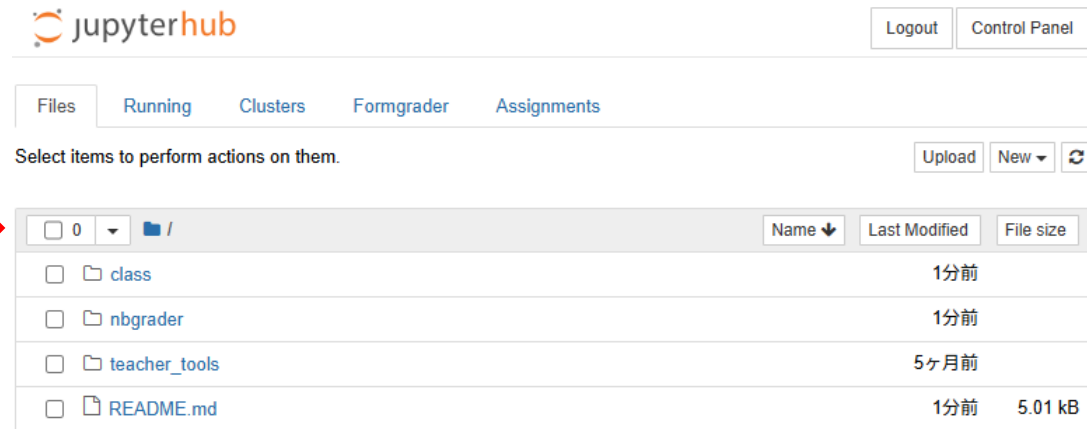
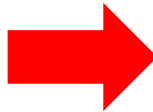
クリックしてMCJ-CloudHubに
アクセス

Jupyter環境の起動

- MCJ-CloudHubにアクセスすると、各自のJupyter環境（プログラムの開発・実行環境）が起動。
 - 起動が完了するまで待機。



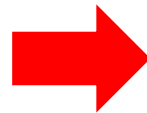
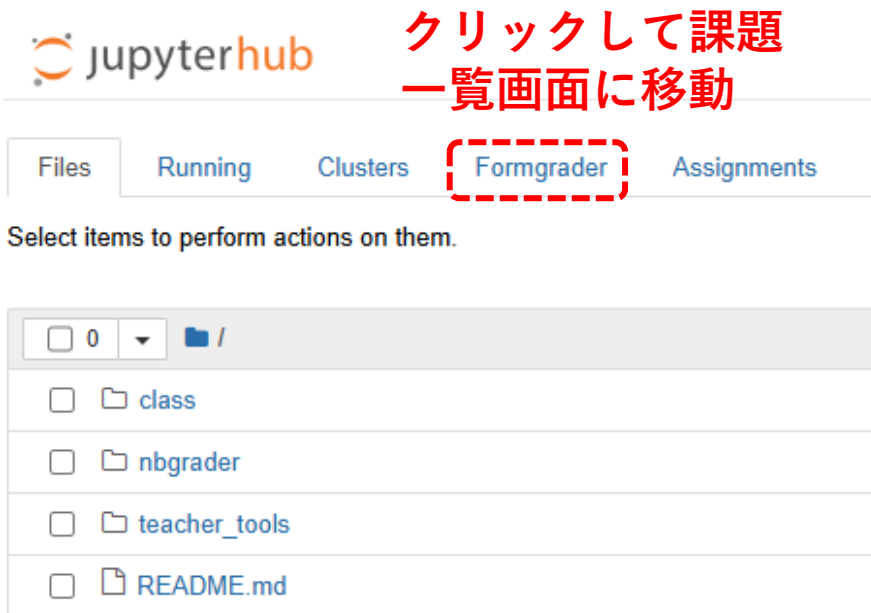
Jupyter環境が起動するまで待機




起動が完了するとファイルやフォルダの一覧が表示

課題の回収 (1/2)

- 画面上部の「Formgrader」をクリックして課題一覧画面に移動。
- 課題「assign01」の行にある回収アイコンをクリック。



Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions
assign01	None	released						0
+ Add new assignment...								

クリックして提出物を回収

課題の回収 (2/2)

- 以下のような画面が表示されたら「OK」をクリック。
- 新規の提出があると、提出数が増える。



Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions
assign01	None	released						1
+ Add new assignment...								

新規の提出があると提出数が増える

課題の手動採点 (1/6)

- 課題の提出数をクリック。

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
assign01	None	released						1		
+ Add new assignment...										

クリックして提出一覧に移動

- (自動採点の問題が無くても) 自動採点を先に実行。

Student Name	Student ID	Timestamp	Status	Score	Autograde	Generate Feedback	Release Feedback
学生, 102	user102s	2025-04-21 05:39:45 JST	needs autograding				

クリックして自動採点を開始

課題の手動採点 (2/6)

- 以下のような画面が表示されたら「OK」をクリック。
- 採点された提出物の状態が「needs manual grading」に変化。

Success

Successfully autograded 'assign01' for student 'user102s'.

Log Output

[INFO] Copying /home/user102t/nbgrader/course102/submitted/user102s/assign01/.lc_wrapper

[INFO] Copying /home/user102t/nbgrader/course102/submitted/user102s/assign01/timestamp.

[INFO] Creating/updating student with ID 'user102s': {}

[INFO] SubmittedAssignment(assign01 for user102s> submitted at 2025-04-20 20:39:45

[INFO] Overwriting files with master versions from the source directory

[INFO] Copying /home/user102t/nbgrader/course102/source/./assign01/.lc_wrapper_regex.txt

[INFO] Sanitizing /home/user102t/nbgrader/course102/submitted/user102s/assign01/assign0:

[INFO] Converting notebook /home/user102t/nbgrader/course102/submitted/user102s/assign0:

[DEBUG] Student: user102s

[DEBUG] Assignment: assign01

[DEBUG] Notebook: assign01

[DEBUG] Applying preprocessor: ClearOutput

[DEBUG] Applying preprocessor: DeduplicateIds

[DEBUG] Applying preprocessor: OverwriteKernelspec

[DEBUG] Source notebook kernelspec: {'display_name': 'LC_wrapper', 'language': 'python'.


[DEBUG] Submitted notebook kernelspec: {'display_name': 'LC_wrapper', 'language': 'pyth

[DEBUG] Overwriting submitted notebook kernelspec: {'display_name': 'LC_wrapper', 'lang

[DEBUG] Applying preprocessor: OverwriteCells

[DEBUG] Applying preprocessor: CheckCellMetadata

クリック[OK]

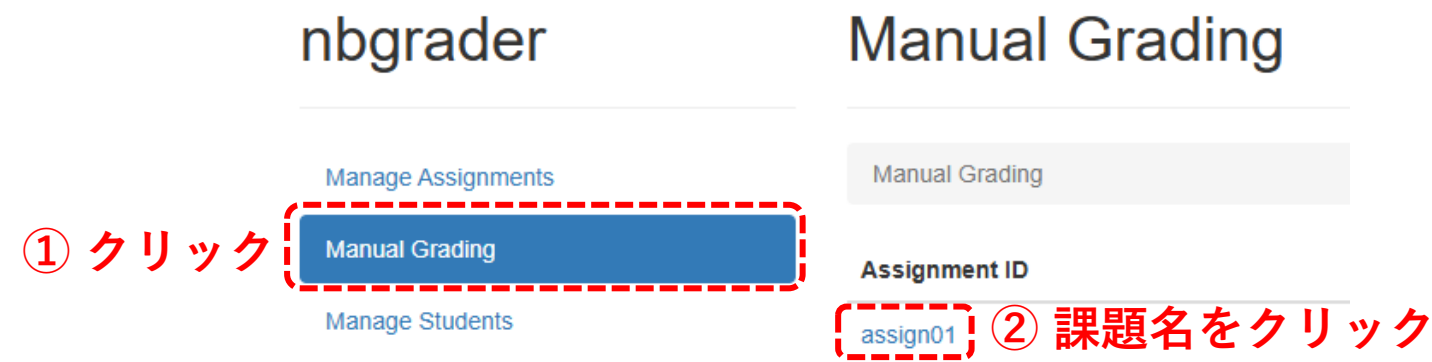


Student Name	Student ID	Timestamp	Status	Score	Autograde	Generate Feedback	Release Feedback
学生, 102	user102s	2025-04-21 05:39:45 JST	needs manual grading	0 / 2	⚡	💬	✉

提出物の状態が「needs manual grading」に変化

課題の手動採点 (3/6)

- 画面左側の「Manual Grading」をクリックした後、課題名（ここでは assign01）をクリック。



- ファイル名（ここではassign01）をクリック

Notebook ID	Avg. Score	Avg. Code Score	Avg. Written Score	Avg. Task Score
assign01	0 / 2	0 / 1	0 / 1	0 / 0

ファイル名をクリック

課題の手動採点（4/6）

- 提出物の一覧画面が表示されたら、採点する提出物をクリック。
 - 今回は1つしかありませんが。

Submission	Overall	Code	Written	Task	Needs	Tests	
ID	Score	Score	Score	Score	Manual	Failed?	Flagged?
Submission #1	0 / 2	0 / 1	0 / 1	0 / 0	✓		

クリックして採点を手動開始

- 最初の問題の評点とコメント（任意）を記入。

In [1]:

Student's answer

Full creditNo credit

/ 1.0 +

0

 (extra credit)

```
def square(n):  
    # YOUR CODE HERE  
    return n*n
```

Type any comments here (supports Markdown and MathJax)



In [1]:

Student's answer

Full creditNo credit

1

 / 1.0 +

0

 (extra credit)

```
def square(n):  
    # YOUR CODE HERE  
    return n*n
```

よくできました。

① 評点を記入

② コメントを記入（任意）

課題の手動採点 (5/6)

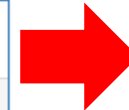
- 第2問の評点とコメント（任意）を記入。

Student's answer

Full credit No credit / 1.0 + (extra credit)

今回の課題は簡単でした。

Type any comments here (supports Markdown and MathJax)



Student's answer

Full credit No credit / 1.0 + (extra credit)

今回の課題は簡単でした。

① 評点を記入

次回はもう少し難しい問題にしてみましょう。

② コメントを記入（任意）

- 画面上部の「next」をクリック。
 - 採点対象が前や次の提出物に移動するときに採点結果が保存。

Manual Grading / assign01 / assign01 / Submission #1

Next →

クリックして次の提出物に移動

課題の手動採点 (6/6)

- 採点対象の提出物をすべて採点し終わると元の画面に戻ってくる。

Submission	Overall	Code	Written	Task	Needs	Tests	
↓↑ ID	↓↑ Score	↓↑ Score	↓↑ Score	↓↑ Score	Manual	Failed?	↓↑ Flagged? ↓↑
Grade?							
👁 Submission #1	2 / 2	1 / 1	1 / 1	0 / 0			

- 画面左側の「Manage Assignments」をクリックして、課題一覧画面に移動。

nbgrader

Manage Assignments

クリック

Manual Grading

Manage Students

フィードバックの作成・公開（1/5）

- 課題「assign01」のフィードバック作成のアイコンをクリック。

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
assign01	None	released						1		

+ Add new assignment...

クリックしてフィードバックを作成

- 以下のような画面が表示されたら「OK」をクリック。

Success

Successfully generated feedback of 'assign01'.

Log Output

```
[INFO] Copying /home/user102t/nbgrader/course102/autograded/user102s/assign01/.lc_wrapp
[INFO] Copying /home/user102t/nbgrader/course102/autograded/user102s/assign01/timestam
[INFO] Converting notebook /home/user102t/nbgrader/course102/autograded/user102s/assign
[DEBUG] Student: user102s
[DEBUG] Assignment: assign01
[DEBUG] Notebook: assign01
[DEBUG] Template paths:
/opt/conda/lib/python3.11/site-packages/nbgrader/server_extensions/formgrader/t
/opt/conda/share/jupyter/nbconvert/templates/classic
/opt/conda/share/jupyter/nbconvert/templates/base
/home/user102t/.local/share/jupyter
/home/user102t/.local/share/jupyter/nbconvert/templates
/home/user102t/.local/share/jupyter/nbconvert/templates/compatibility
/opt/conda/share/jupyter
/opt/conda/share/jupyter/nbconvert/templates
/opt/conda/share/jupyter/nbconvert/templates/compatibility
/usr/local/share/jupyter
/usr/local/share/jupyter/nbconvert/templates
/usr/local/share/jupyter/nbconvert/templates/compatibility
```

OK

クリック

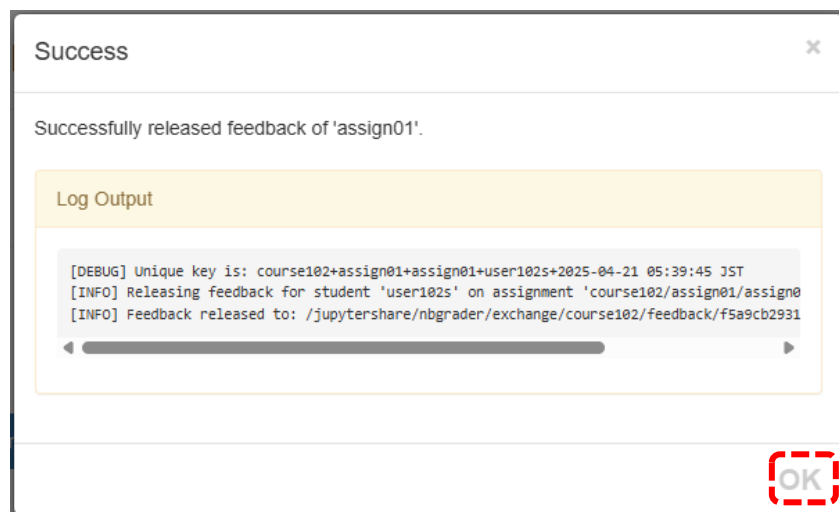
フィードバックの作成・公開 (2/5)

- 課題「assign01」のフィードバック公開のアイコンをクリック。

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
assign01	None	released						1		
+ Add new assignment...										

クリックしてフィードバックを公開

- 以下のような画面が表示されたら「OK」をクリック。



クリック

フィードバックの作成・公開 (3/5)

- タブを削除して、ホームに戻る。

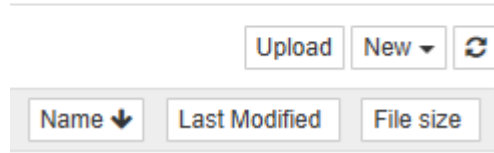
クリックしてタブを削除



- 画面右上の「Logout」をクリックしてMCJ-CloudHubからログアウト。



クリックしてログアウト



フィードバックの作成・公開 (4/5)

- MCJ-CloudHubのタブを削除。

クリックしてタブを削除



フィードバックの作成・公開 (5/5)

- Moodleからもログアウト。

① 画面右上のユーザのアイコンをクリック



② クリックしてログアウト

フィードバックの取得（学生）

学生としてMoodleにログイン

- 学生用のアカウントを使用してMoodleにログイン。

**割り当てられた学生用
アカウントでログイン**

user102s

.....

ログイン

[パスワードを忘れましたか?](#)

いくつかのコースにはゲストアクセスできます

ゲストとしてアクセスする

日本語 (ja) [クッキー通知](#)



demo moodle Home ダッシュボード マイコース [通知](#) [メッセージ](#) 学1 [編集モード](#)

マイコースに移動

ダッシュボード

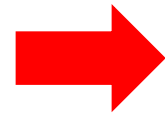
タイムライン

次の7日 [日付で並べ替える](#) [活動タイプまたは活動名で検索する](#)

アクションを必要とする活動はありません。

コースに移動してMCJ-CloudHubを起動

- マイコースに自分のユーザ番号に対応したコース) があるので、そのコースアクセス (画面例とは異なるかも)。
- コース内に設置された外部ツールをクリック。



コース102

コース 参加者 評価 コンビテンシ

▼ 一般

すべてを折りたたむ



アナウンスメント

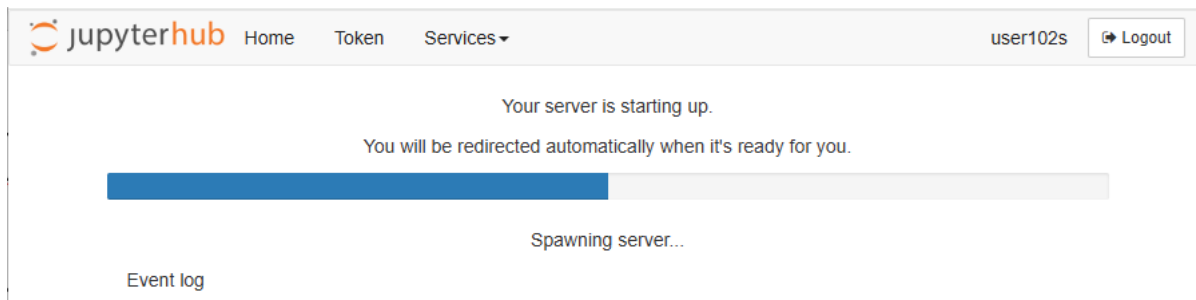


MCJ-CloudHub

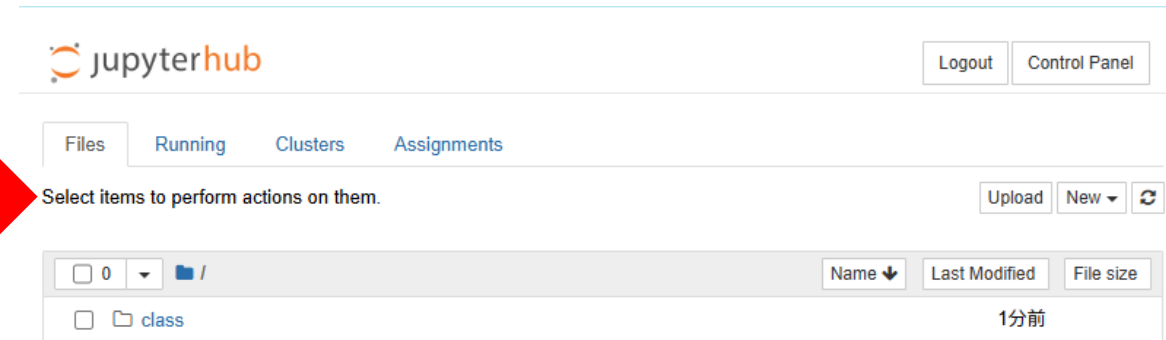
クリックしてMCJ-CloudHubに
アクセス

Jupyter環境の起動

- MCJ-CloudHubにアクセスすると、各自のJupyter環境（プログラムの開発・実行環境）が起動。
 - 起動が完了するまで待機。



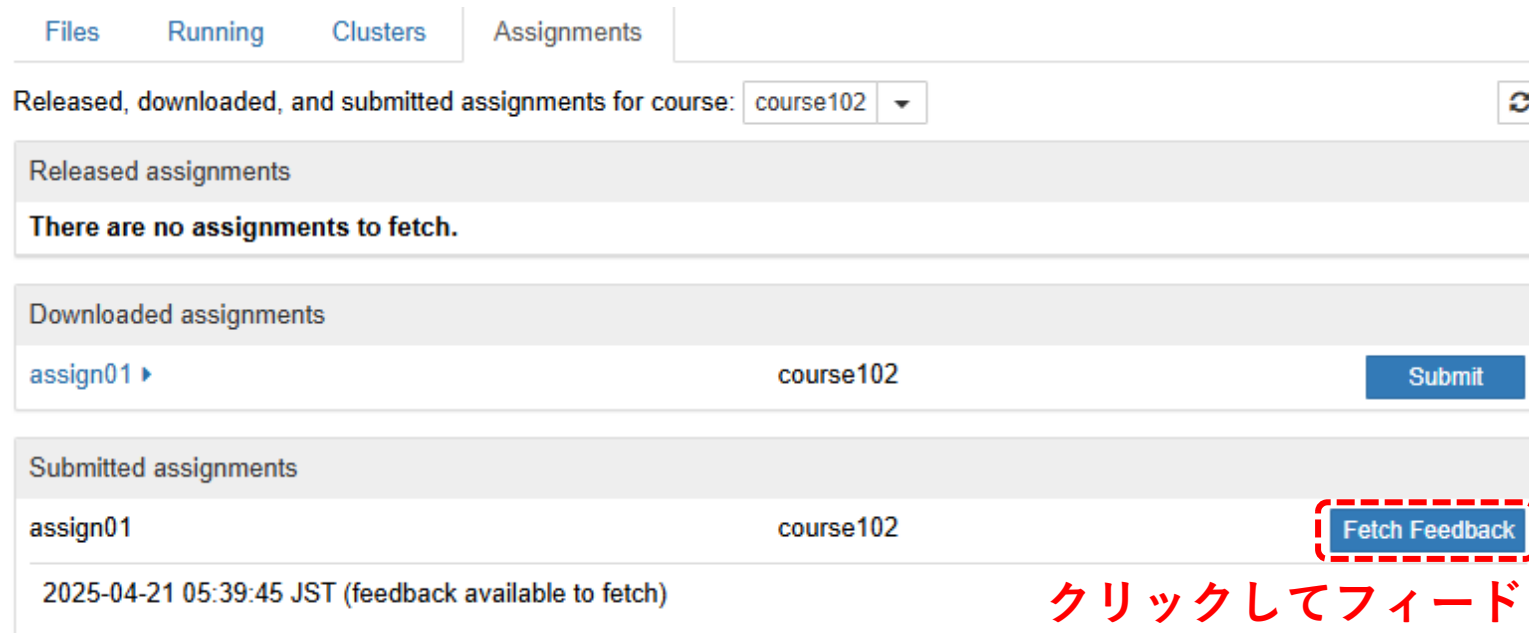
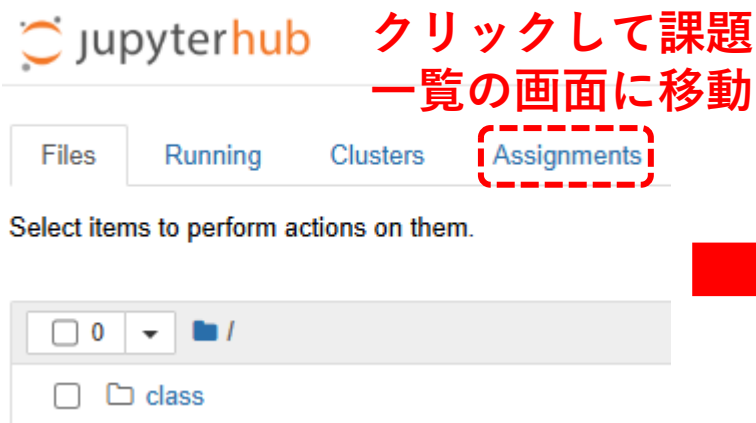
Jupyter環境が起動するまで待機



起動が完了するとファイルやフォルダの一覧が表示

フィードバックの取得 (1/6)

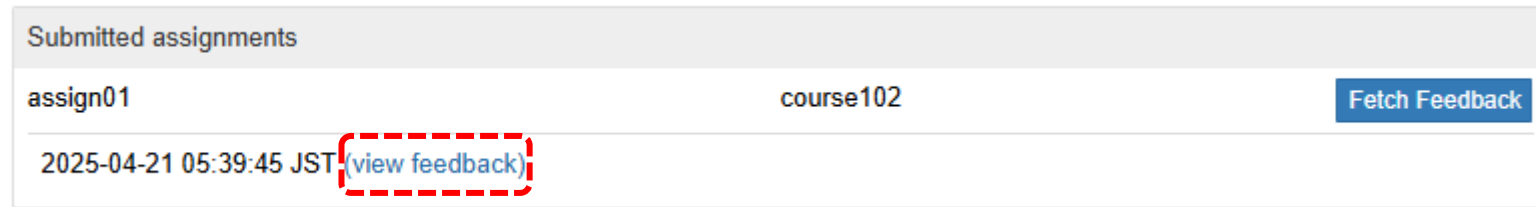
- 画面上部の「Assignments」をクリックして課題一覧画面に移動。
- 「assign01」のフィードバックが公開されているので、「Fetch feedback」をクリック。



クリックしてフィードバックを取得

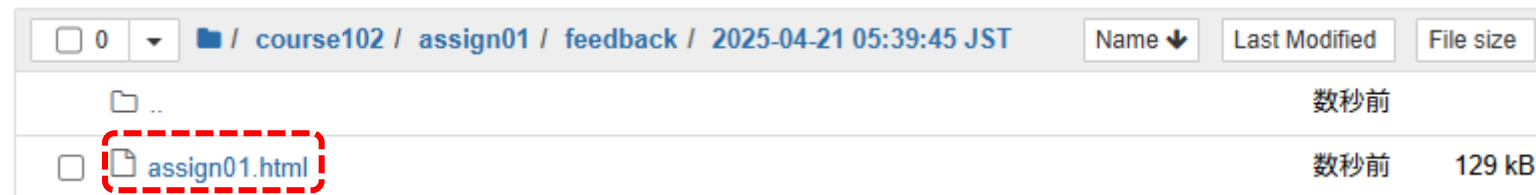
フィードバックの取得 (2/6)

- フィードバックが公開されている場合、「view feedback」というリンクが作成されるので、これをクリック。



クリックしてフィードバックを表示

- ファイル名をクリックしてフィードバックの内容を確認。



クリックしてフィードバックを表示

フィードバックの取得 (3/6)

- フィードバックの内容を確認。

assign01 (Score: 2.0 / 2.0)

- [1. Coding free-response \(Score: 1.0 / 1.0\)](#)
- [2. Comment](#)
- [3. Written response \(Score: 1.0 / 1.0\)](#)
- [4. Comment](#)

1. 引数nの2乗を返す関数squareを定義します。「raise NotImplementedError()」を削除してからコードを記述してください。

In [1]:

Student's answer

Score: 1.0 / 1.0 (Top)

```
def square(n):  
    # YOUR CODE HERE  
    return n*n
```

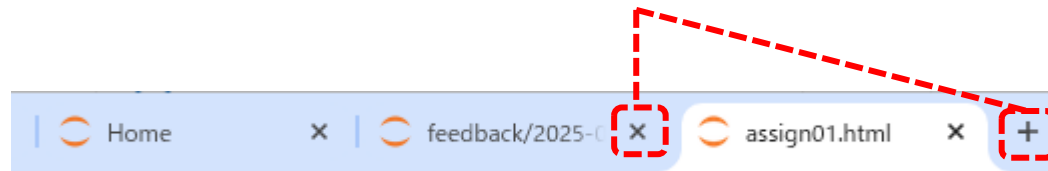
Comments:

よくできました。

フィードバックの取得 (4/6)

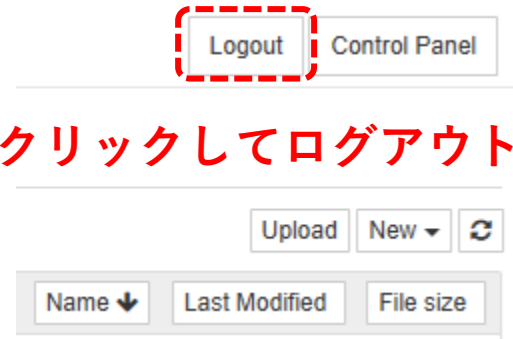
- タブを削除していき、ホームに戻る。

クリックしてタブを削除



フィードバックの取得 (5/6)

- 画面右上の「Logout」をクリックしてMCJ-CloudHubからログアウト。



- MCJ-CloudHubのタブを削除。



フィードバックの取得 (6/6)

- Moodleからもログアウト。

① 画面右上のユーザのアイコンをクリック



② クリックしてログアウト

アンケートへのご協力をお願い

- まずは課題の手動採点の操作や演習についてアンケートを行いますので、ご協力いただけますと幸いです。