

講義・演習環境アプリケーション ハンズオン

— nbgraderを用いた演習課題の配布・回収・採点 —

齊藤智也（山口大学情報基盤センター）

本日の内容

■課題の作成・配布・回収・採点(1)

- 課題の作成・配布(教員側)
- 課題の取得・提出(学生側)
- 課題の回収・採点(教員側)
- フィードバックの閲覧(学生側)

■課題ファイルを作成する際の注意点

■課題に関する各種操作

■課題の作成

- 自由に課題を作成していただきます。

資料内の画面例について

- ハンズオン環境では各サーバのOSの言語モードが英語に設定されているため、Jupyter関連のメニュー等がすべて英語で表示されます。
- 資料内の画面例は、OSの言語モードが日本語の環境で作成されたため、Jupyter関連のメニュー等が日本語で表示されています。

タイムアウトに伴う操作エラーについて

- 教師ユーザ及び学生ユーザのJupyter環境は、しばらく操作しないでいるとタイムアウトによりログアウトします（画面にはその旨が表示されません）。
- ログアウトした状態で各種操作を行うとエラーが表示されます。
 - ・ その場合には再度、MoodleコースからMCJ-CloudHubのリンクをクリックしてください。

GitHub上から必要なファイルをダウンロード

4

■以下のURLから必要なファイル等をダウンロードします。

- <https://github.com/nii-gakunin-cloud/handson>
- 「2024年度ハンズオン」の配下にある「講義・演習環境アプリケーション ハンズオン(演習)」の箇所からダウンロードします。

ハンズオンのMoodleサイトにアクセス

■各自、事前に通知されたURLにアクセスしてください。

Example University Moodle Demo にログインする

ユーザ名

① ユーザ名を入力

パスワード

② パスワードを入力

ログイン

③ ボタンをクリック

[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) ▾

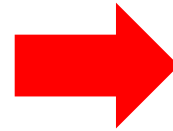
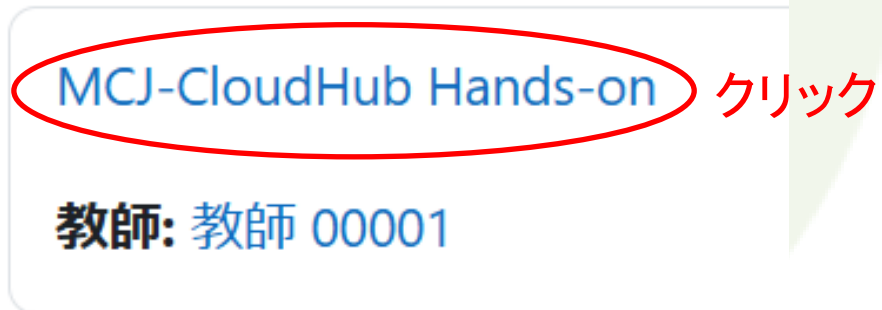
クッキー通知

■まずは教師としてログイン

- ユーザ名: tea + 00001
- パスワード: 通知されたもの

Moodleコースへのアクセス

- Moodleコースへのリンクをクリック。
- コース画面が表示されたら, MCJ-CloudHubへのリンクをクリック。



Jupyter環境（プログラムの開発・実行環境）の起動

教師ユーザ

7

■しばらく待つと、ユーザ向けのJupyter環境が起動。

The screenshot displays the JupyterHub web interface. At the top, the header includes the JupyterHub logo, navigation links for 'Home' and 'Token', a user count of '126', and a 'Logout' button. A central message states 'Your server is starting up. You will be redirected automatically when it's ready for you.' Below this is a progress bar that is partially filled with blue. Under the progress bar, the text 'Spawning server...' is visible. To the left of the progress bar is an 'Event log' section. A large red arrow points from the 'Spawning server...' text down to a blue notification banner. The banner contains an 'UPDATE' notice about migrating to Notebook 7, with a 'Don't show anymore' button. Below the banner, the JupyterHub logo is repeated, followed by 'Logout' and 'Control Panel' buttons. A navigation bar shows tabs for 'Files', 'Running', 'Clusters', and 'Assignments'. Below this is a section for selecting items to perform actions on, with buttons for 'Upload', 'New', and a refresh icon. A table header shows columns for 'Name', 'Last Modified', and 'File size'. The table is currently empty, with a message 'The notebook list is empty.' at the bottom.

jupyterhub Home Token 126 Logout

Your server is starting up.
You will be redirected automatically when it's ready for you.

Spawning server...

Event log

UPDATE Read [the migration plan](#) to Notebook 7 to learn about the new features and the actions to take if you are using extensions - Please note that updating to Notebook 7 might break some of your extensions. Don't show anymore

jupyterhub Logout Control Panel

Files Running Clusters Assignments

Select items to perform actions on them. Upload New ↻

☐ 0 / Name ↓ Last Modified File size

The notebook list is empty.

課題の作成(1/2)

■「Formgrader」をクリック。

- ・ 新たなタブが開いてFormgraderの画面が表示される。

■「Add new assignment」をクリック。



① クリック

Files Running Clusters **Formgrader** Assignments

Select items to perform actions on them.

☐ 0 /

nbgrader

Manage Assignments

Manual Grading

Manage Students

Manage Assignments

Assignments

Instructions (click to expand)

Name Due Date Status Edit Generate

+ Add new assignment...

② クリック

課題の作成(2/2)

- 課題名を入力して、「Save」をクリック。
 - ・課題名は半角英数字が良い。
 - ・今回のセミナーでは提出期限の設定は省略。
- 課題一覧の中に、作成した課題が追加される。

Add New Assignment

① 課題名を入力

Name

ex01

Due date (optional)



yyyy/mm/dd --:--

② クリック

Save

Cancel



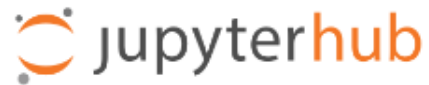
Name	Due Date	Status	Edit	Generate	Preview	Release
ex01	None	draft				
+ Add new assignment...						

課題の作成(3/5)

教師ユーザ

10

- 本来は、課題一覧の画面で課題名をクリックすると配布ファイル置き場（フォルダ）にジャンプするが、現在のバージョンでは不具合がある。
- 以前のタブに戻り、課題のファイル置き場に移動する。
 - 「nbgrader/handson202408/source/ex01」



ログアウト

Control Panel

ファイル

実行中

クラスタ

Formgrader

Assignments

Nbextensions

アクションを実行する為のアイテムを選択して下さい。

アップロード

新規 ▼



0



/ nbgrader / handson202408 / source / ex01

名前 ▼

最終変更時刻

ファイルサイズ



..

数秒前

The notebook list is empty.

課題ファイルのアップロード(1/2)

■課題のファイル置き場に、先程ダウンロードしたファイル「01-basic.ipynb」をアップロードする。

- この画面で新規ファイルの作成も可能だが、今回は省略。

ファイル 実行中 クラスタ Formgrader Assignments Nbextensions

アクションを実行する為のアイテムを選択して下さい。

アップロード 新規 ◯

0 ▾ / nbgrader / handson202408 / source / ex01 名前 ↓ 最終変更時刻 ファイルサイズ

The notebook list is empty.

.. 数秒前

ファイル 実行中 クラスタ Formgrader Assignments Nbextensions

アクションを実行する為のアイテムを選択して下さい。

アップロード 新規 ◯

0 ▾ / nbgrader / handson202408 / source / ex01 名前 ↓ 最終変更時刻 ファイルサイズ

The notebook list is empty.

.. 数秒前

01-basic.ipynb Upload Cancel

右上にある「Upload」をクリックするとファイル選択画面が開き、必要なファイルを選択する。

ファイルの右側に表示される「Upload」をクリックすると、アップロードの処理が確定する。

課題ファイルのアップロード(2/2)

■同様に、「02-basic.ipynb」もアップロードする。

ファイル 実行中 クラスタ Formgrader Assignments Nbextensions

アクションを実行する為のアイテムを選択して下さい。

アップロード 新規 刷新

0 / nbgrader / handson202408 / source / ex01

名前	最終変更時刻	ファイルサイズ
..	数秒前	
01-basic.ipynb	数秒前	5.95 kB

右上にある「Upload」をクリックするとファイル選択画面が開き、必要なファイルを選択する。

ファイル 実行中 クラスタ Formgrader Assignments Nbextensions

アクションを実行する為のアイテムを選択して下さい。

アップロード 新規 刷新

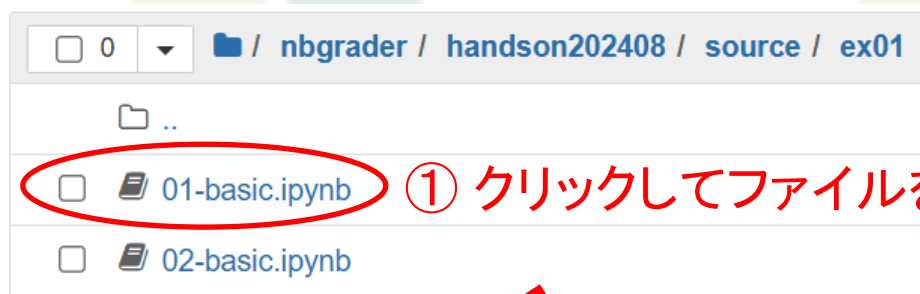
0 / nbgrader / handson202408 / source / ex01

名前	最終変更時刻	ファイルサイズ
02-basic.ipynb		
..	数秒前	
01-basic.ipynb	1分前	5.95 kB

ファイルの右側に表示される「Upload」をクリックすると、アップロードの処理が確定する。

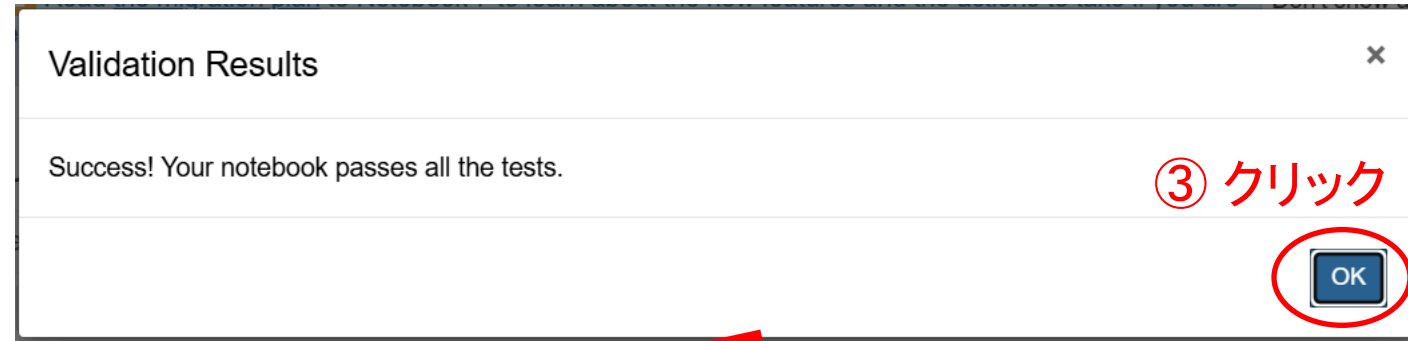
課題ファイルの確認(1/2)

- アップロードしたファイル「01-basic.ipynb」を開いた後、メニューバーにある「Validate」をクリックする。

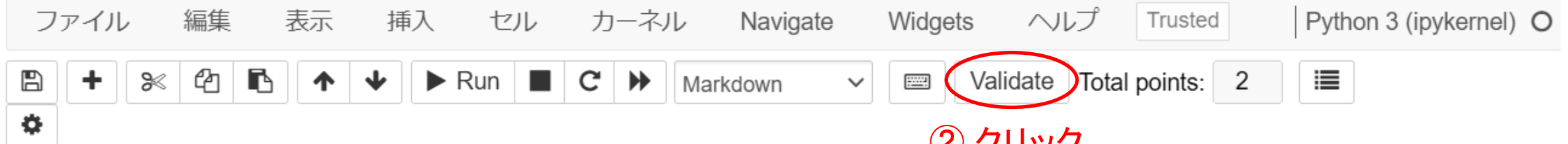


① クリックしてファイルを開く

jupyterhub 01-basic (autosaved)



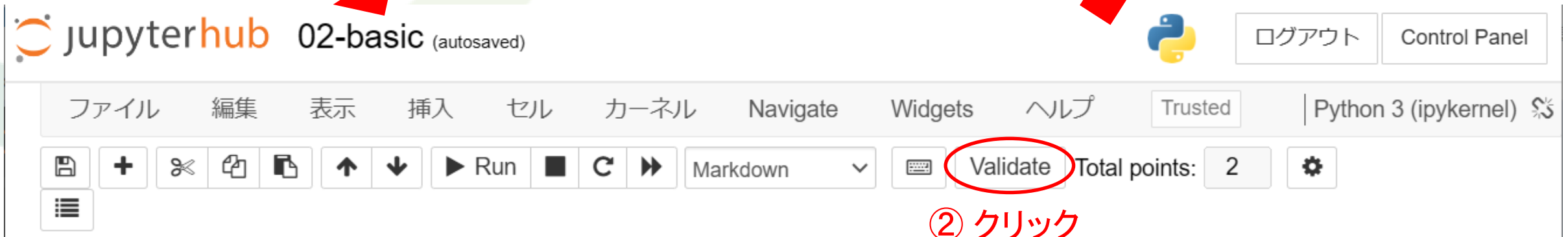
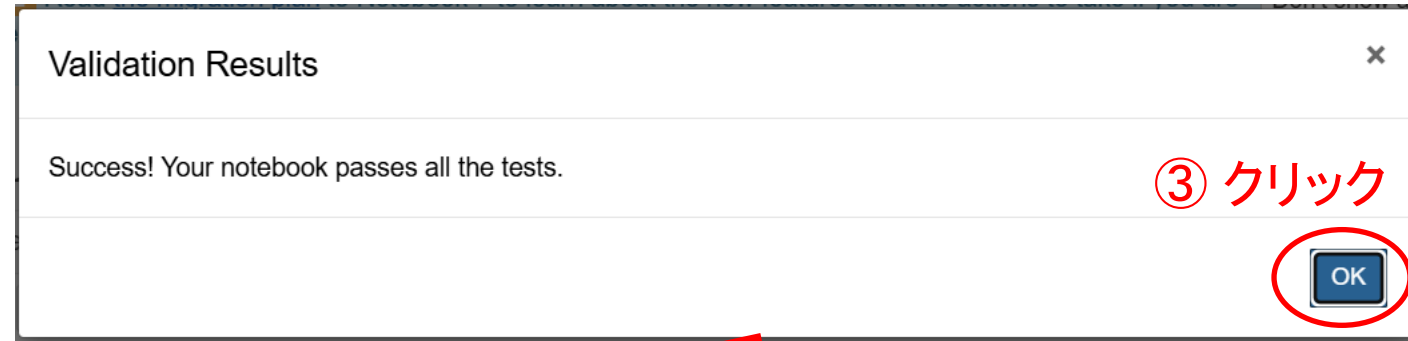
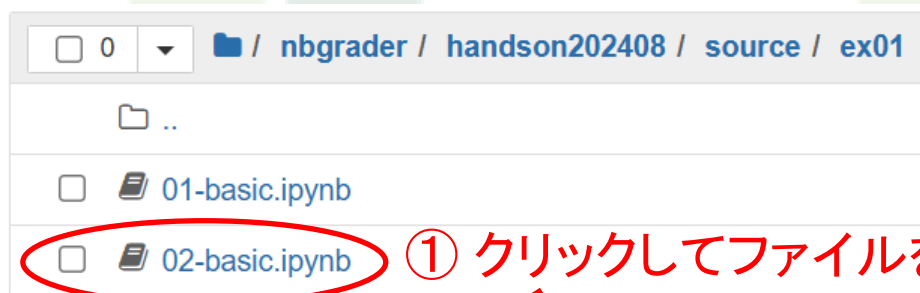
③ クリック



② クリック

課題ファイルの確認(2/2)

- 同様に「02-basic.ipynb」を開いた後、メニューバーにある「Validate」をクリックする。



「01-basic.ipynb」の内容(1/6)

■最初のほうにあるセルはMarkdown用のセル。

- Markdown記法を利用して説明文などを記述。
- nbgraderにおけるセル種別としては「-」を選択(採点等に影響しない通常のセル)。

The screenshot shows the Jupyter Notebook interface. The top toolbar includes menus like 'ファイル', '編集', '表示', '挿入', 'セル', 'カーネル', 'Navigate', 'Widgets', and 'ヘルプ'. The 'セル' (Cell) menu is open, and 'Markdown' is selected, indicated by a red circle. Below the toolbar, the notebook content is displayed. The first cell is titled '1 関数の練習' (1 Function Practice) and contains the text '今回は関数の練習です。' (This time is practice for functions). The second cell is titled '1.1 級数の和を求める関数' (1.1 Function to find the sum of a series) and contains text about a 'sum' function and a note to remove 'raise NotImplementedError()' and replace it with a program. Both cells have a dropdown menu in the top right corner, with the '-' option selected, also indicated by red circles.

対象のセルを選択した状態で、リストから「Markdwodn」を選択すると、Markdown用のセルになる。

セルの右上にあるリストから「-」を選択する。

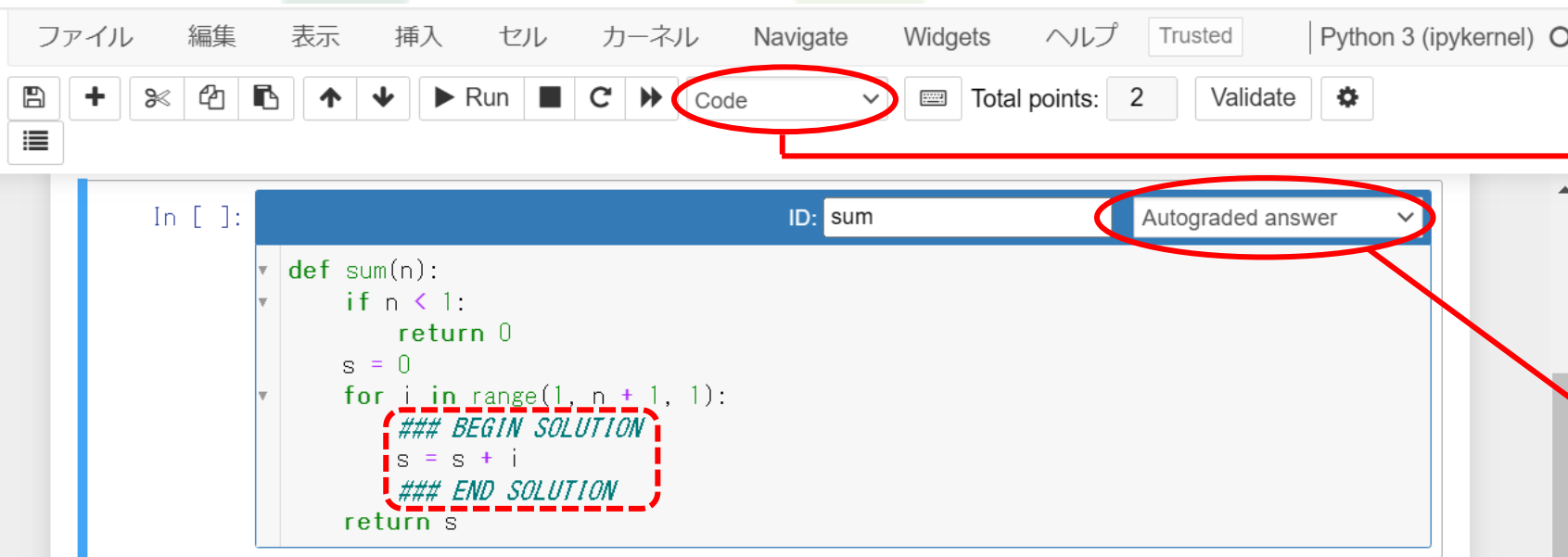
「01-basic.ipynb」の内容(2/6)

■自動採点のためのコード記述用のセル(プログラムの回答欄)。

- Jupyter Notebookのセル種別としては「Code」を選択。
- nbgraderのセル種別としては「Autograded answer」を選択。

■解答記入箇所を「### BEGIN SOLUTION」と「### END SOLUTION」で挟む。

- 受講生への配布ファイルでは「# YOUR CODE HERE」に置き換えられる。



対象のセルを選択した状態で、リストから「Code」を選択すると、コード用のセルになる。

セルの右上にあるリストから「-」を選択する。

「01-basic.ipynb」の内容 (3/6)

■受講生向けに確認用のセルを配置。

- Markdown用のセルを使用して説明を記述。
- コード用のセルを使用して、実行結果を確認する場所を作成。

Markdown用のセル

以下のセルを利用して、sum関数の入出力を確認しましょう。

In []:

コード用のセル

```
# 例えば print(sum(4)) のような文を実行して表示内容を確認してみる  
print(sum(4))
```

受講生はこのセルを実行すると、先ほどの関数sumの動作を確認できる。

「01-basic.ipynb」の内容(4/6)

■自動採点用の評価セルを配置。

- Jupyter Notebookのセル種別は「Code」を選択。
- nbgraderのセル種別は「Autograder tests」を選択。

■セル内のコードを実行し、エラーや例外が発生しなければ評点を付与する。

評点(教師が手動で設定)

In []:

```
# このセルには何も記入しないでください。  
### BEGIN HIDDEN TESTS  
assert sum(1) == 1  
assert sum(4) == 10  
assert sum(5) == 15  
assert sum(0) == 0  
assert sum(-1) == 0  
### END HIDDEN TESTS
```

Points: 1 ID: correct_sum Autograder tests

「#### BEGIN HIDDEN TESTS」と「#### END HIDDEN TESTS」で挟まれた箇所は、受講生への配布ファイルでは表示されない。

「01-basic.ipynb」の内容 (5/6)

- `assert` 文では、条件式の評価が `False` の場合には例外を発生させる。

```
assert sum(4) == 10
```

「`sum(4)`」の戻り値が10の場合に `True`。

```
assert type(n) is int
```

変数 `n` のデータ型が `int` の場合に `True`。

```
assert "University" in s
```

変数 `s` に文字列「`University`」が含まれている場合に `True`。

```
assert squares(2) == [1, 4]
```

「`squares(2)`」の戻り値が配列の `[1, 2]` の場合に `True`。

「01-basic.ipynb」の内容 (6/6)

In [1]:

ID: squares

Autograded answer

```
def squares(n):  
    if n < 1:  
        raise ValueError("引数nには自然数を渡すべきです。")  
    ### BEGIN SOLUTION  
    return [i ** 2 for i in range(1, n + 1)]  
    ### END SOLUTION
```

引数の例外処理

- 2問目では、今回は評点を設定していないが、例外的な入力への反応をチェックするセルを配置。

In []:



Points: 0

ID: squares_invalid_input

Autograder tests

```
# このセルには何も記入しないでください。  
### BEGIN HIDDEN TESTS  
try:  
    squares(0)  
except ValueError:  
    pass  
else:  
    raise AssertionError("did not raise")  
  
try:  
    squares(-4)  
except ValueError:  
    pass  
else:  
    raise AssertionError("did not raise")  
### END HIDDEN TESTS
```

どちらもコード用のセル

- 引数の例外処理の箇所も受講生に解答させる場合には、このセルに評点を設定して、設問通りの例外処理を記述しているかどうかにも評価に含めることが可能。

補足: セルのIDについて

■nbgraderにおけるセル種別として「-」以外を選択すると、セルのIDを設定可能になる。

- 通常はランダムな値が設定されているが、手動で変更可能になる。
- 例外やエラーが発生したセルを分かりやすくする効果がある。
- 受講生ごとの実行のログを収集して分析したい場合に便利。
 - MCJ-CloudHubではログの収集・可視化の機能も実装予定。
 - 現在は各受講生のホームディレクトリの配下にログファイルを作成する機能のみ。



コードの自動採点用のセルについて

■配布ファイルでは、解答を記入させる箇所が以下のように置き換わる。

```
In [ ]: ID: sum Autograded answer ▼  
  
def sum(n):  
    if n < 1:  
        return 0  
    s = 0  
    for i in range(1, n + 1, 1):  
        ### BEGIN SOLUTION  
        s = s + i  
        ### END SOLUTION  
    return s
```



```
In [ ]: def sum(n):  
    if n < 1:  
        return 0  
    s = 0  
    for i in range(1, n + 1, 1):  
        # YOUR CODE HERE  
        raise NotImplementedError()  
    return s
```

セルの実行時に「NotImplementedError」が発生するので、このエラーを発生させる文を削除した後に解答を記入するよう、直前のセルにおいて説明が必要。

「02-basic.ipynb」の内容(1/3)

■最初のほうにあるセルは題材になるコード例と解答への説明文。

- 受講生向けの配布ファイルでは、自由記述のセルの枠が表示されず、その位置にセルがあるかどうか分かりにくいため、説明が必要。

In [3]:

コード用のセル

ID: code

Read-only

```
# 次のコードについて考えてみましょう。  
def f(x):  
    if x == 0 or x == 1:  
        return x  
    return f(x - 1) + f(x - 2)
```

Markdown用のセル

ID: part1_summary

Read-only

A. この関数fの働き、計算内容、使い方について解答しましょう。

「YOUR ANSWER HERE」をダブルクリックすると入力可能になるので、「YOUR ANSWER HERE」を削除した後に解答を記入してください。

「Read-only」のセルでは、受講生がそれらのセルを書き換えても、採点時には元の内容に戻る。

「02-basic.ipynb」の内容(2/3)

■自由記述のセルを配置。

- Jupyter Notebookのセル種別は「Markdown」を選択。
- nbgraderのセル種別は「Manually graded answer」を選択。

Markdown用のセル

Points:	1	ID:	part1_answer	Manually graded answer ▼
<p>この関数は、関数の再起呼び出しを使用してフィボナッチ数列を計算します。引数が 0 や 1 の場合、関数はそれぞれ 0 または 1 を返します。それ以外の場合、この関数は $x-1$ と $x-2$ を引数としてそれぞれ自身を呼び出し、それらの戻り値を加算したものを返します。</p>				

自由記述のセルの場合、受講生向けの配布ファイルではセルの内容が空欄になる。

「02-basic.ipynb」の内容 (3/3)

■コードの手動採点のセルを配置。

- Jupyter Notebookのセル種別は「Code」を選択。
- nbgraderのセル種別は「Manually graded answer」を選択。

コード用のセル

In []: Points: 1 ID: part2_answer Manually graded answer ▾

```
print(f(10))
```

■受講生向けの配布ファイルでは以下のように置き換えられるので、直前にMarkdown用のセルを配置して説明を記述。

In []: ▾

```
# YOUR CODE HERE  
raise NotImplementedError()
```

課題ファイルを閉じる(1/2)

- Webブラウザのタブのうち、課題ファイルを開いているタブを両方とも閉じる。
- ファイル・エクスプローラのタブにおいて、「01-basic.ipynb」を選択した後、上部に表示される「Shutdown」をクリックする。

① チェックボックスをクリック

② ボタンをクリック

The screenshot shows the nbgrader file explorer interface. At the top, there is a toolbar with buttons: 複製 (Copy), シャットダウン (Shutdown), 表示 (View), 編集 (Edit), and a trash icon. The 'Shutdown' button is highlighted with a red box. Below the toolbar, the breadcrumb path is / nbgrader / handson202408 / source / ex01. The file list shows two files: 01-basic.ipynb and 02-basic.ipynb. The checkbox for 01-basic.ipynb is checked, and the Shutdown button is highlighted with a red box. A red arrow points from the Shutdown button to the next screenshot.

名前	最終変更時刻	ファイルサイズ
01-basic.ipynb	Running 9分前	5.95 kB
02-basic.ipynb	Running 8分前	3.61 kB

↓

名前	最終変更時刻	ファイルサイズ
01-basic.ipynb	11分前	5.95 kB
02-basic.ipynb	Running 10分前	3.61 kB

課題ファイルを閉じる(2/2)

■同様の操作により、「02-basic.ipynb」もシャットダウンする。

① チェックボックスをクリック

② ボタンをクリック

The screenshot shows the nbgrader interface with the file '02-basic.ipynb' selected. The 'シャットダウン' button is highlighted with a red box, and a red arrow points to it from the text '② ボタンをクリック'. The checkbox for '02-basic.ipynb' is also highlighted with a red circle, with a red arrow pointing to it from the text '① チェックボックスをクリック'. A large red arrow points down to the next screenshot.

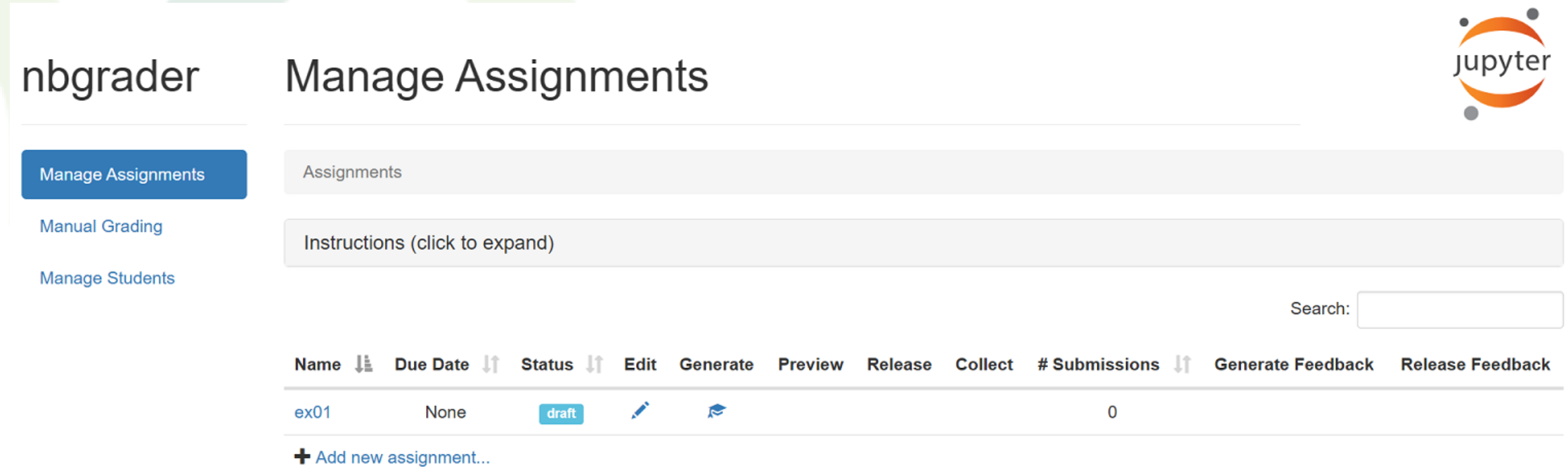
名前	最終変更時刻	ファイルサイズ
..	数秒前	
01-basic.ipynb	16分前	5.95 kB
02-basic.ipynb	Running 14分前	3.61 kB

↓

名前	最終変更時刻	ファイルサイズ
..	数秒前	
01-basic.ipynb	17分前	5.95 kB
02-basic.ipynb	16分前	3.61 kB

配布ファイルの生成(1/2)

■「Formgrader」の画面(タブ)に移動。



nbgrader Manage Assignments

Assignments

Instructions (click to expand)

Search:



Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
ex01	None	draft						0		

+ Add new assignment...

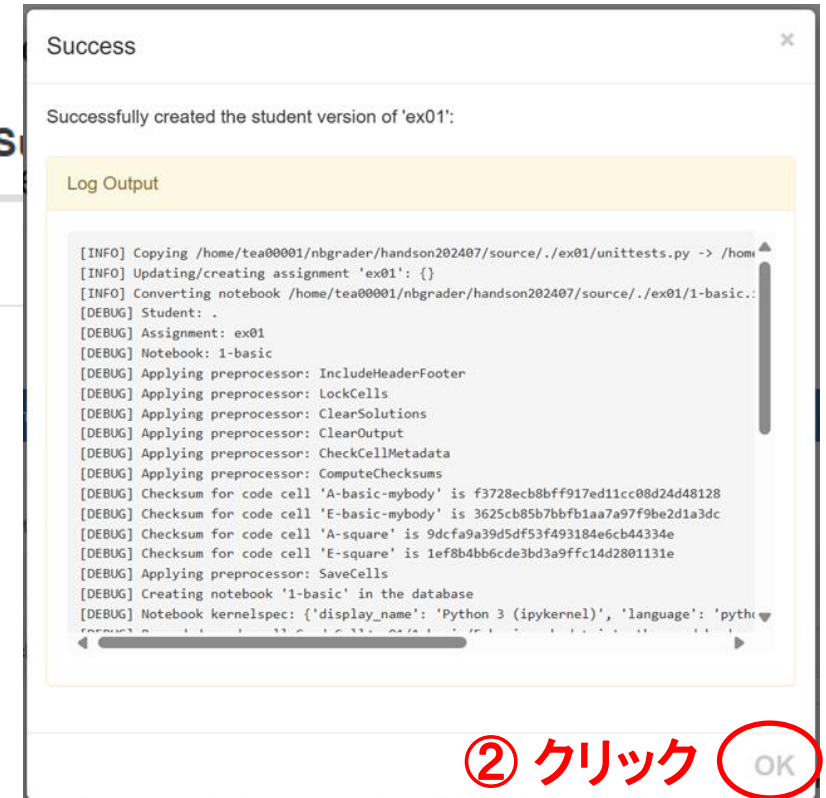
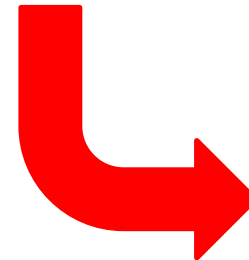
■「Formgrader」のタブを閉じてしまっている場合は、ファイルエクスプローラのタブに置いて再度「Formgrader」をクリック。

配布ファイルの生成(2/2)

- 課題一覧の画面において、「Generate」のアイコン(帽子の絵)をクリック。
- ログの画面が表示される(画面下部のOKをクリックすると閉じる)。
- エラー等が発生していなければ配布が可能になる。

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	S
ex01	None	draft						
+ Add new assignment...								

① クリック



課題の配布開始

■課題一覧の画面において、「Release」のアイコン(雲の絵)をクリック。

The screenshot illustrates the process of releasing an assignment. The top part shows the assignment 'ex01' in 'draft' status. A red circle highlights the 'Release' icon (a cloud with an upward arrow), with the text '① クリック' (Click) next to it. A red arrow points from this icon to a modal dialog box. The dialog box, titled 'Success', contains the message 'Successfully released 'ex01'.' and a 'Log Output' section with the following text: [INFO] Source: /home/t-saito/nbgrader/handson202408/release/./ex01, [INFO] Destination: /jupytershare/nbgrader/exchange/handson202408/outbound/ex01, [INFO] Released as: handson202408 ex01. A red circle highlights the 'OK' button in the dialog, with the text '② クリック' (Click) next to it. Below the dialog, the assignment 'ex01' is shown in 'released' status, with the 'released' label circled in red. The bottom text states: 課題のステータスが「draft」から「released」に変化する。

Name	Due	Status	Edit	Generate	Preview	Release	Collect	Submissions	Feedback	Feedback
ex01	None	draft						0		

+ Add new assignment...

Name	Due	Status	Edit	Generate	Preview	Release	Collect	Submissions	Feedback	Feedback
ex01	None	released						0		

+ Add new assignment...

学生側のWebブラウザ画面の準備

学生ユーザ

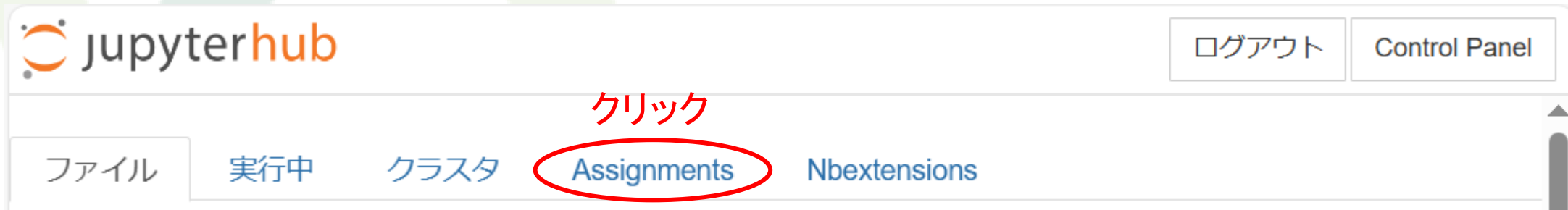
31

- 別のWebブラウザを起動し、先ほどと同じMoodleサイトにログイン。
 - ・ ユーザ名には「stu00001」を使用。
- ログイン後にはハンズオン用のコースに移動。
- 外部ツールを利用して「MCJ-CloudHub」にログイン。

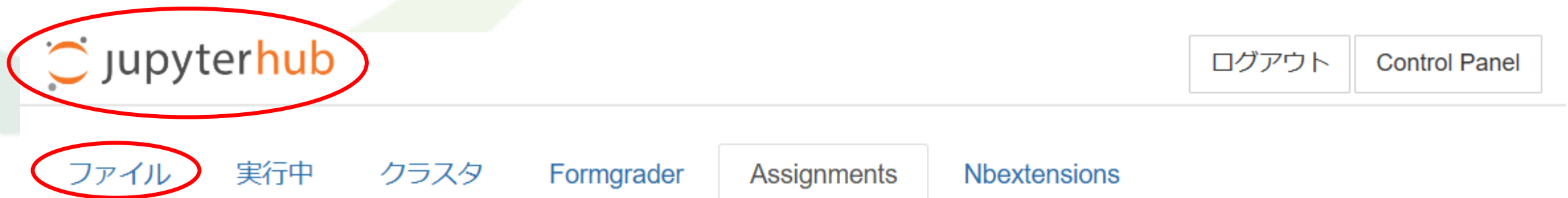
- 課題ファイルが公開済みになった後に受講生がMCJ-CloudHubにログインすると、課題の取得が可能。

課題ファイルの取得(1/2)

■ファイル・エクスプローラの画面(タブ)において「Assignments」をクリック。



■元の画面に戻る場合は「File」やJupyterHubのロゴをクリック。



課題ファイルの取得(2/2)

■課題名の右側にある「Fetch」をクリック。

- 課題を取得できると、下段に課題名のリンクが表示。

ファイル 実行中 クラスタ Assignments Nbextensions

Released, downloaded, and submitted assignments for course: handson202408

Released assignments

ex01	handson202408	Fetch
------	---------------	-------

クリック

ファイル 実行中 クラスタ Assignments Nbextensions

Released, downloaded, and submitted assignments for course: handson202408

Released assignments

There are no assignments to fetch.

Downloaded assignments

ex01 ▶	handson202408	Submit
--------	---------------	--------

課題名のリンクが表示される

課題への取り組み(1/6)

- 課題名をクリックすると、ファイルの一覧が表示される。
- ファイル名をクリックすると、そのファイルが開く。

The diagram illustrates the process of opening a Jupyter Notebook through a series of three screenshots connected by red arrows.

Step 1: A box titled "Downloaded assignments" shows a list with "ex01" circled in red. A red circle around "ex01" is labeled "クリック" (Click).

Step 2: The box shows "ex01" expanded into a dropdown menu. The option "01-basic" is circled in red and labeled "クリック" (Click). "02-basic" is also visible below it.

Step 3: A screenshot of the Jupyterhub interface for "01-basic (autosaved)". The interface includes a top bar with "jupyterhub", "01-basic (autosaved)", and a "Python 3 (ipykernel)" kernel. Below the top bar is a menu bar with options like "ファイル", "編集", "表示", "挿入", "セル", "カーネル", "Navigate", "Widgets", and "ヘルプ". A toolbar contains icons for file operations and a "Run" button. The main area displays Japanese instructions: "課題を提出する前に、すべてが期待通りに実行されることを確認してください。最初に、**カーネルを再起動してください** (メニューバーの中から、「カーネル」→「再起動」の順に選択)。続いて、**すべてのセルを実行してください** (メニューバーの中から、「セル」→「全てを実行」の順に選択)。
課題に取り組むときは、`YOUR CODE HERE` もしくは "YOUR ANSWER HERE" と書かれている箇所をすべて記入してください。
提出の前に、以下にあなたの氏名 (NAME) と共同作業 (COLLABORATORS) も記入してください。"

課題への取り組み(2/6)

■ 解答を記入し終わったら、コードを記述するセルに続いて、確認用のセルも実行してみる(すべてのセルをまとめて実行しても構わない)。

- 以下は2問目の例。
- 「raise NotImplementedError ()」を削除した後に解答を記入。

```
In [1]: def squares(n):  
        if n < 1:  
            raise ValueError("引数nには自然数を渡すべきです。")  
        # YOUR CODE HERE  
        return [i ** 2 for i in range(1, n + 1)]
```

① 実行

```
In [3]: # 例えば print(squares(10)) のような文を実行して表示内容を確認してみる  
        print(squares(10))
```

② 実行

[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

実行結果を確認

課題への取り組み(3/6)

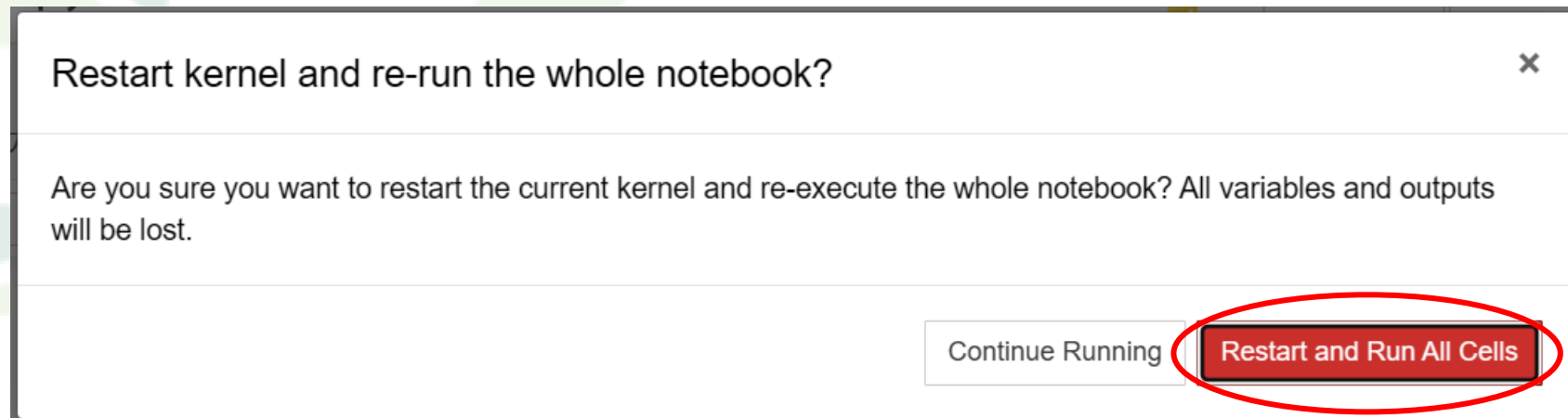
学生ユーザ 36

■ファイルが完成したら、カーネルを再起動した後、すべてのセルを実行する。

- Jupyter Notebookのメニューでは「早送り」のようなボタンをクリックする。
- 確認画面が表示されたら「Restart and Run All Cells」をクリック。



① クリック



② クリック

課題への取り組み(4/6)

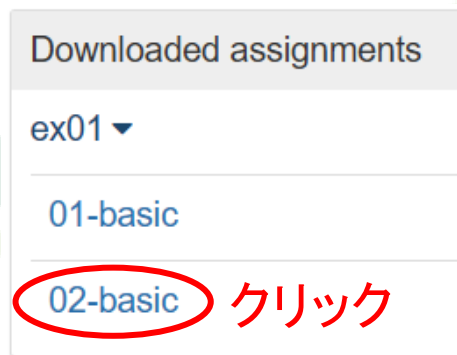
■ファイルを保存する。



クリック

■そのファイルのブラウザ・タブを閉じる。

■続いて、「02-basic.ipynb」を開く。



課題への取り組み(5/6)

■自由記述ではセルの枠が表示されないため、入力欄の位置が分からない。

- 「YOUR ANSWER HERE」をダブルクリック。

A. この関数fの働き、計算内容、使い方について解答しましょう。

「YOUR ANSWER HERE」をダブルクリックすると入力可能になるので、「YOUR ANSWER HERE」を削除した後に解答を記入してください。

YOUR ANSWER HERE

① ダブルクリック



A. この関数fの働き、計算内容、使い方について解答しましょう。

「YOUR ANSWER HERE」をダブルクリックすると入力可能になるので、「YOUR ANSWER HERE」を削除した後に解答を記入してください。

YOUR ANSWER HERE

② 「YOUR ANSWER HERE」を削除した後に解答を記入

課題の取り組み(6/6)

学生ユーザ

39

- プログラムを記入するセルでは、「raise NotImplementedError()」を削除した後に解答を記入。

B. この関数fに引数として整数の「10」を渡し、その結果を画面に表示するコードを解答しましょう。

以下の「raise NotImplementedError()」を削除した後、その位置に適切なプログラムを記入してください。

```
In [ ]: # YOUR CODE HERE  
print(squares(10))
```

- すべての解答を記入できたら、カーネルを再起動してすべてのセルを実行。
- その後、ファイルを保存し、ブラウザのタブを閉じる。

課題の提出

- 課題名の右側にある「Submit」をクリック。
- 提出が完了すると、フィードバックを取得するボタンが表示される。

The image shows a two-part screenshot of a web interface. The top part, titled 'Downloaded assignments', shows a table with columns for assignment name, ID, and action. The first row is 'ex01' with ID 'handson202408' and a blue 'Submit' button circled in red with the Japanese text 'クリック' (Click) next to it. Below this are two rows for '01-basic' and '02-basic', each with a green 'Validate' button. A large red arrow points down to the bottom part of the screenshot. The bottom part shows the 'Submitted assignments' section. It has a table with 'ex01' and ID 'handson202408', now with a blue 'Fetch Feedback' button. Below the table, the timestamp '2024-08-19 08:19:37 JST' is displayed.

Downloaded assignments		
ex01 ▼	handson202408	Submit
01-basic		Validate
02-basic		Validate

↓

Submitted assignments		
ex01	handson202408	Fetch Feedback

2024-08-19 08:19:37 JST

提出物の回収(1/2)

教師ユーザ

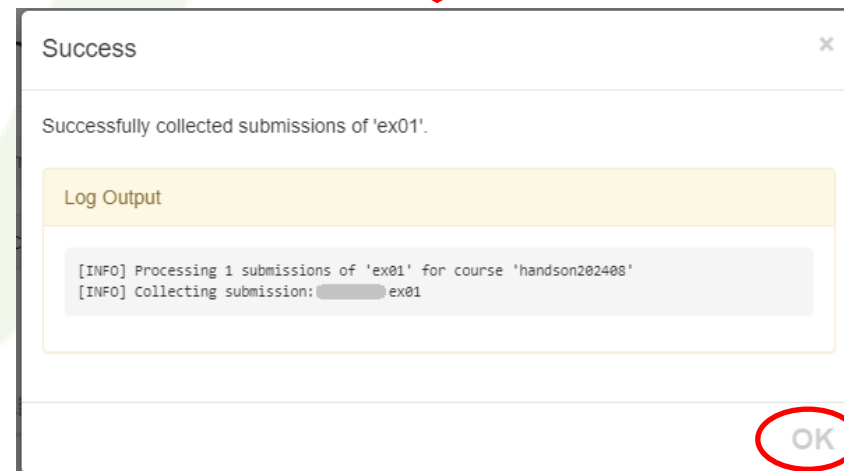
41

■教師側の画面において、課題の回収(Collect)のボタンをクリック。

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
ex01	None	released						0		








+ Add new assignment...

クリック



クリック

■新たに提出物が回収されると、提出数が増加している。

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
ex01	None	released						1		

+ Add new assignment...

提出物の自動採点(1/2)

- 課題一覧の画面において、対象となる課題の提出数をクリック。
- 提出状況の一覧が表示されたら、対象の受講生について「Autograde」のアイコン(稲妻マーク)をクリック。

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
ex01	None	released						1		

+ Add new assignment...

① クリック

Student Name	Student ID	Timestamp	Status	Score	Autograde	Generate Feedback	Release Feedback
		2024-8-19 8:19:37 JST	needs autograding				

② クリック

Success

Successfully autograded 'ex01' for student ' '.

Log Output

```
[DEBUG] Skipping non-executing cell 7
[DEBUG] Executing cell:
# YOUR CODE HERE
print(squares(10))
[DEBUG] msg_type: status
[DEBUG] content: {'execution_state': 'busy'}
[DEBUG] msg_type: execute_input
[DEBUG] content: {'code': '# YOUR CODE HERE\nprint(squares(10))', 'execution_count': 3}
[DEBUG] msg_type: error
[DEBUG] content: {'traceback': ['\x1b[0;31m-----
[DEBUG] msg_type: status
[DEBUG] content: {'execution_state': 'idle'}
[DEBUG] Destroying zmq context for <jupyter_client.blocking.client.BlockingKernelClient
[DEBUG] Applying preprocessor: LimitOutput
[DEBUG] Applying preprocessor: SaveAutoGrades
[DEBUG] Applying preprocessor: AssignLatePenalties
[DEBUG] Applying preprocessor: CheckCellMetadata
[INFO] Writing 5664 bytes to /home/ /nbgrader/handson202408/autograded/ /ex
[INFO] Setting destination file permissions to 444
```

③ クリック OK

提出物の自動採点(2/2)

- Jupyterのターミナル上で下記のようなコマンドを実行することにより、未評価の提出物の一括採点も可能。

```
teacher1@c7e37e1cfa2e:~$ cd nbgrader/handson202408/  
teacher1@c7e37e1cfa2e:~/nbgrader/handson202408$ nbgrader autograde "ex01"
```

- 「ファイルあたりの実行時間 × 提出数」の時間がかかるので、しばらく放置しておくしかない。
- メモリが不足すると、途中で採点の動作が中断されるため、ユーザごとのメモリ割り当て量の設定に注意が必要。

提出物の手動採点(1/3)

- 提出物のステータスが「need manual grading」の場合は、手動採点が必要。
- 画面左側の「Manual Grading」をクリックした後、対象の課題名をクリック。

The screenshot displays the submission management interface. At the top, a table lists submission details. A red arrow points from the 'needs manual grading' status in the table to the 'Manual Grading' button in the left sidebar. Below the sidebar, a table lists assignments, with 'ex01' circled in red.

Student Name	Student ID	Timestamp	Status	Score	Autograde	Generate Feedback	Release Feedback
[Redacted]	[Redacted]	2024-8-19 8:19:37 JST	needs manual grading	2 / 4	⚡	💬	✉

Manual Grading

Manage Assignments

Manual Grading

Manage Students

Search:

Assignment ID	Due Date	Submissions	Score
ex01	None	1	2 / 4

提出物の手動採点(2/3)

- ファイルの一覧が表示されたら、手動採点が必要なファイル名をクリック。
- 受講生一覧が表示されるので、対象の受講生をクリック。
 - ・デフォルトでは各受講生の氏名は伏せられている。

Notebook ID	Avg. Score	Avg. Code Score	Avg. Written Score	Avg. Task Score	Needs Manual Grade?
01-basic	2 / 2	2 / 2	0 / 0	0 / 0	
02-basic	0 / 2	0 / 1	0 / 1	0 / 0	✓

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

② クリック

提出物の手動採点(3/3)

- 採点対象のセルについて、評点を入力する。
- 必要に応じてコメントの記入も可能。

提出物や課題ファイルの一覧の画面に戻る

前の受講生

← Prev

Manual Grading / ex01 / 02-basic / Submission #1

Next →

次の受講生

Student's answer

Full credit

No credit

1

/ 1.0

+

0

(extra credit)

再帰呼び出しを使ってフィボナッチ数を求める関数です。

できればもう少し説明を書いてほしかったのですが、まあ良いでしょう。

コメント

B. この関数fに引数として整数の「10」を渡し、その結果を画面に表示するコードを解答しましょう。

以下の「raise NotImplementedError()」を削除した後、その位置に適切なプログラムを記入してください。

In [3]:

Student's answer

Full credit

No credit

0

/ 1.0

+

0

(extra credit)

```
# YOUR CODE HERE
print(squares(10))
```

使用する関数名を間違えていますね。

コメント

評点を入力

採点が完了すると...

- 課題「ex01」について提出状況の一覧(受講生の一覧)を表示すると、採点が完了した者のステータスが「graded」に設定されている。

① クリック

Manage
Assignments

Manual Grading

Manage Students

Assignments

Instructions (click to expand)

Search:

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
ex01	None	released						1		
+ Add new assignment...										

② クリック

Student Name	Student ID	Timestamp	Status	Score	Autograde	Generate Feedback	Release Feedback
		2024-8-19 8:19:37 JST	graded	3 / 4			

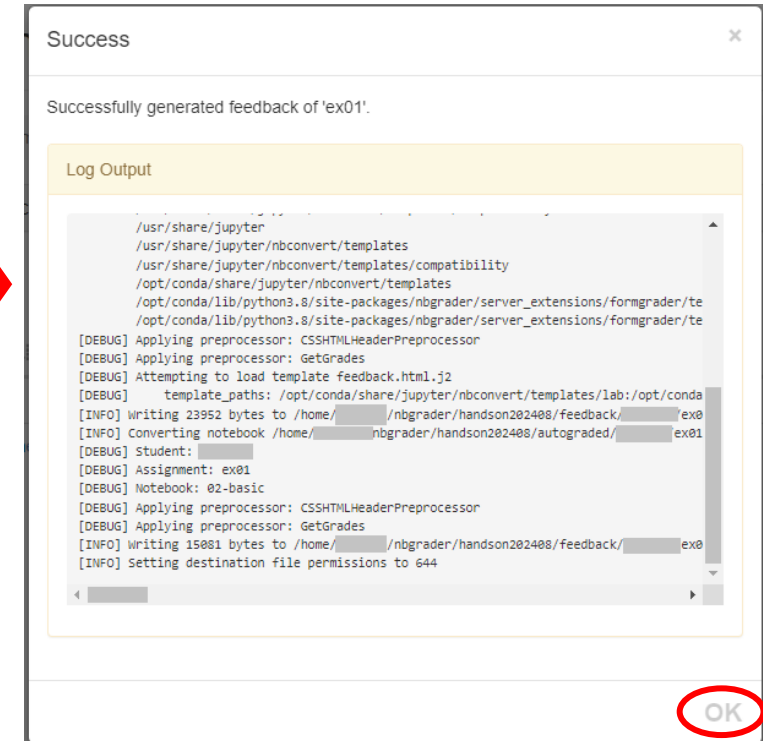
フィードバックの作成

- 課題一覧の画面において、対象の課題の右側にある「Generate Feedback」アイコン(吹き出しマーク)をクリックするとフィードバックが生成される。

Name	Due Date	Status	Edit	Generate	Preview	Release	Collect	# Submissions	Generate Feedback	Release Feedback
ex01	None	released						1		

+ Add new assignment...

① クリック



② クリック

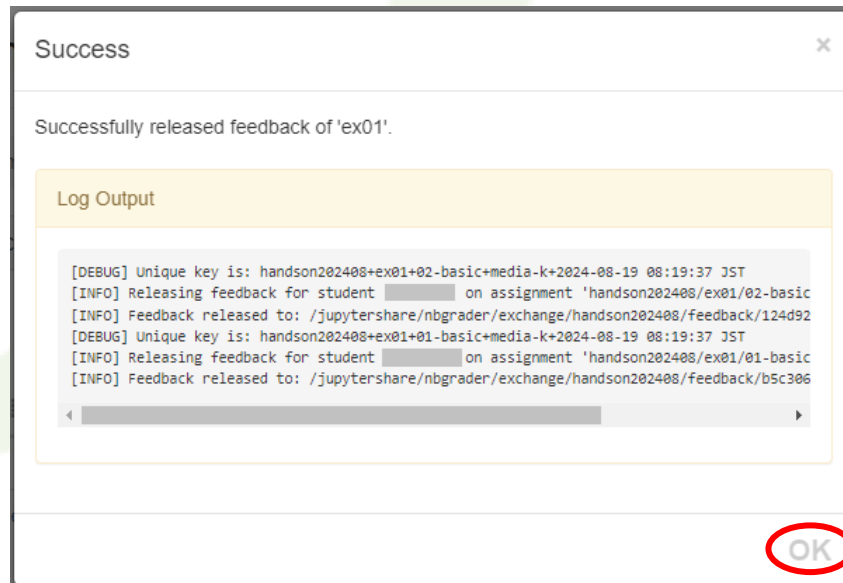
フィードバックの公開

- 課題一覧の画面において、対象の課題の右側にある「Release Feedback」アイコン(手紙マーク)をクリックするとフィードバックが受講生に公開される。

Name ↓↑	Due Date ↓↑	Status ↓↑	Edit	Generate	Preview	Release	Collect	# Submissions ↓↑	Generate Feedback	Release Feedback
ex01	None	released						1		

+ Add new assignment...

① クリック



② クリック

フィードバックの取得

- 学生側のWebブラウザにおいて、「Assignments」の画面に移動。
- 対象の課題について、「Fetch Feedback」をクリック。

① クリック

ファイル 実行中 クラスタ **Assignments** Nbextensions

Released, downloaded, and submitted assignments for course: handson202408

Released assignments

There are no assignments to fetch.

Downloaded assignments

ex01 ▶	handson202408	Submit
--------	---------------	--------

Submitted assignments

ex01	handson202408	Fetch Feedback
2024-08-19 08:19:37 JST (feedback available to fetch)		

② クリック

フィードバックの閲覧

■「view feedback」というリンクをクリック。

Submitted assignments

ex01	handson202408	Fetch Feedback
2024-08-19 08:19:37 JST (view feedback) ① クリック		

↓

0 / handson202408 / ex01 / feedback / 2024-08-19 08:19:37 JST

	名前 ↓	最終変更時刻	ファイルサイズ
...		数秒前	
<input type="checkbox"/>	01-basic.html	2分前	25.1 kB
<input type="checkbox"/>	02-basic.html	2分前	16 kB

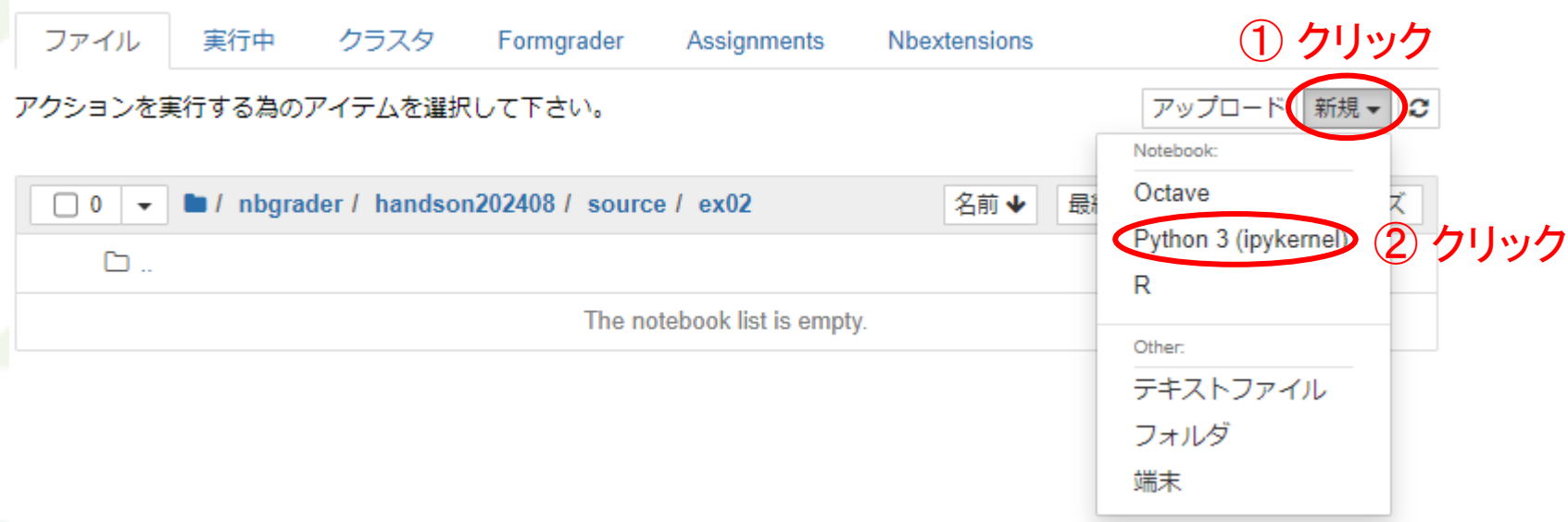
HTMLファイルの中で採点結果やコメントを閲覧可能

課題ファイルを作成する際の注意点(1/2)

教師ユーザ

53

- 課題ファイルの配置場所において、「新規」→「Python3」の順にクリックすると、その場で新規の課題ファイルを作成可能。

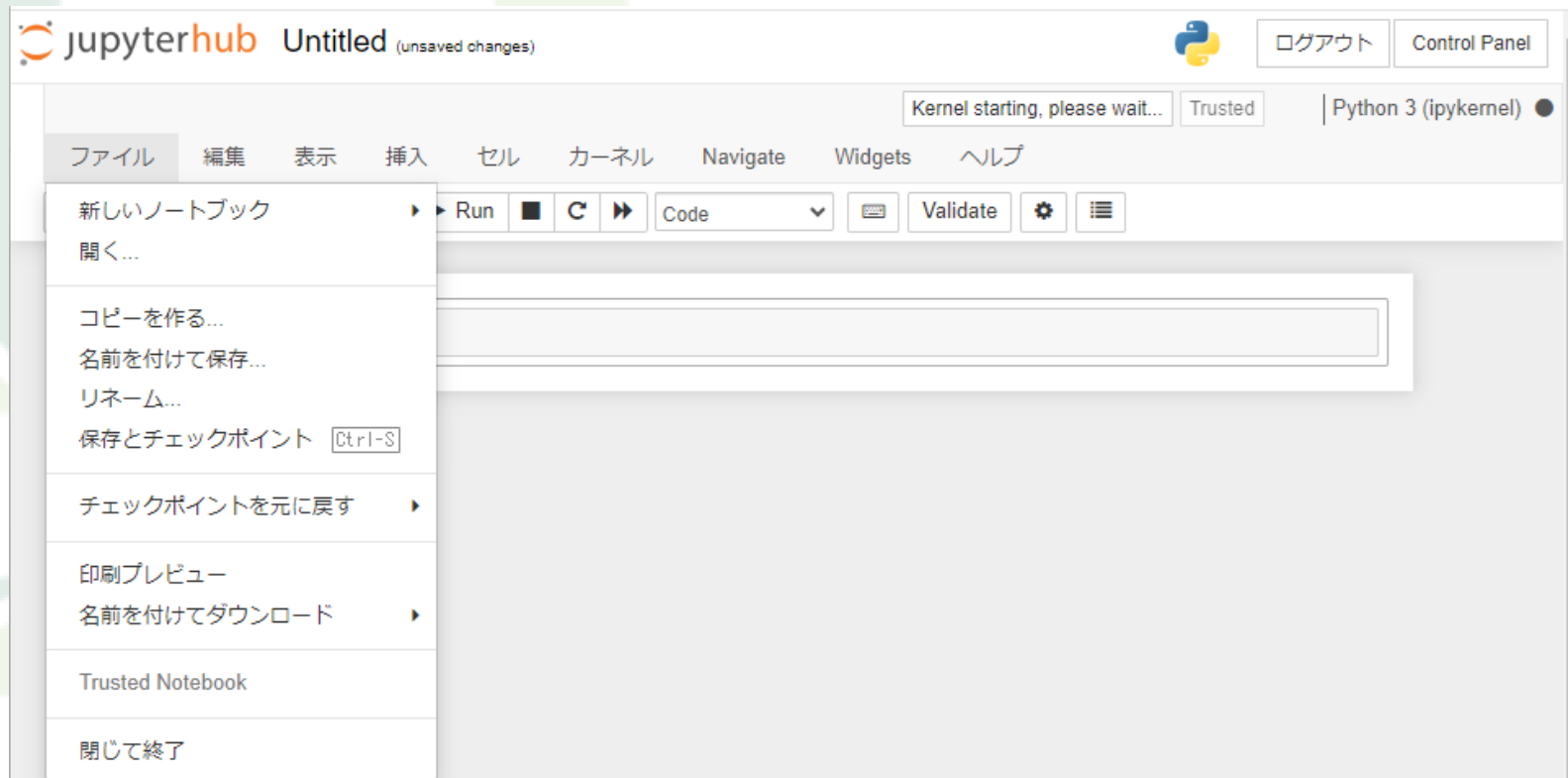


課題ファイルを作成する際の注意点(2/3)

教師ユーザ

54

- 最初は「Untitled.ipynb」といったファイル名で作成される。
- 「ファイル」メニューから「名前を付けて保存」を選択すると、「Untitled.ipynb」と、名前を付けたファイルの両方がフォルダに残る(どちらも配布対象になってしまう)。

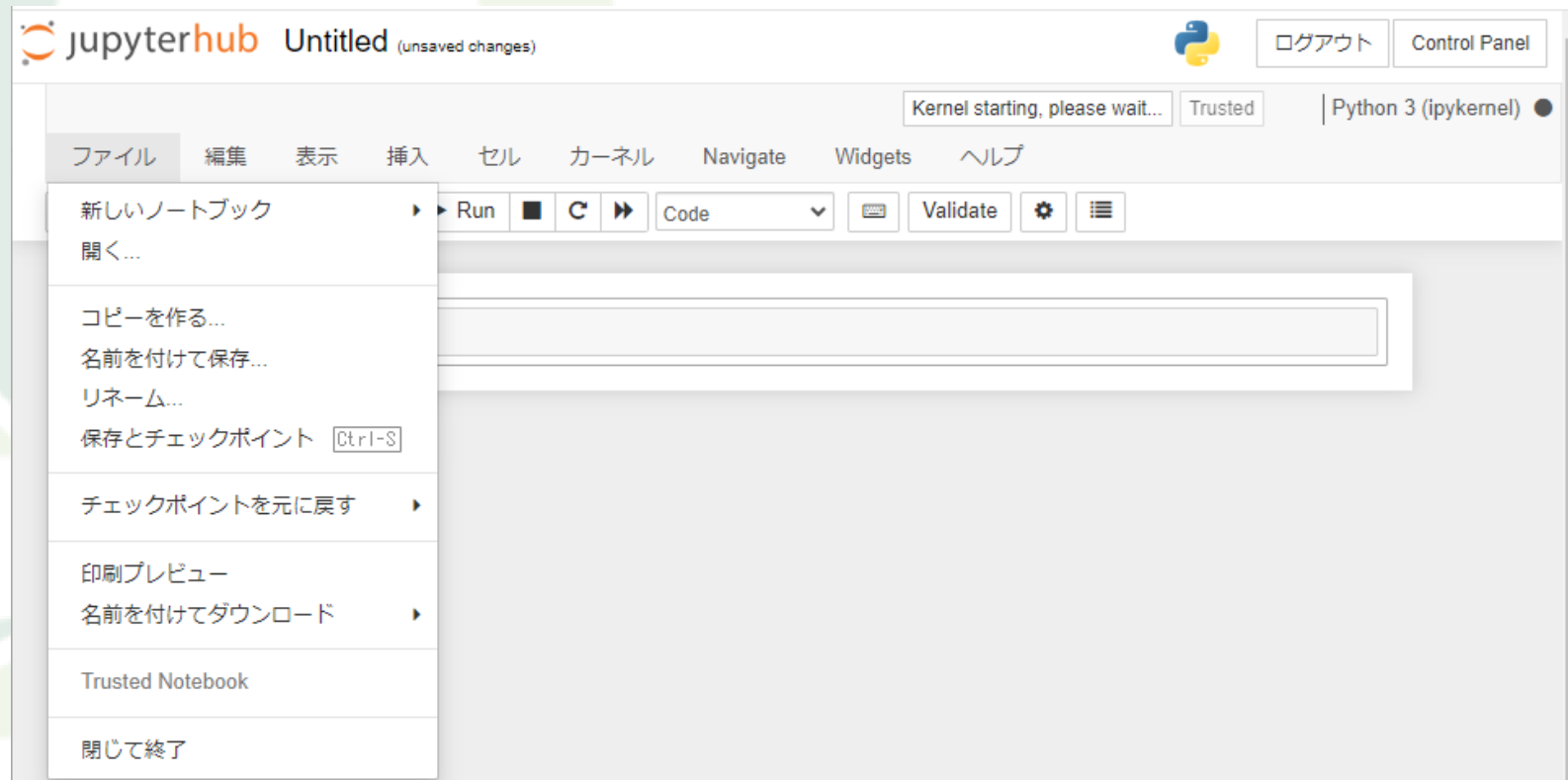


課題ファイルを作成する際の注意点(3/3)

教師ユーザ

55

- 最初に「ファイル」メニューの中から「リネーム」を選択し、ファイル名を変更しておいたほうが良い。



その他の注意点

- 提出物が存在する課題については、課題の内容を修正できない。
 - ・ 課題ファイルの内容を修正しても、配布ファイルの作成時にエラーになる。
- このような状況で課題の内容を修正する場合、既存の課題を削除した後、新たに課題を作成・配布するしかない。

課題を削除したい場合

■Jupyterのターミナル上で下記のようなコマンドを実行。

■最初に、課題を非公開に戻す。

```
teacher1@c7e37e1cfa2e:~$ nbgrader list --remove <課題名>
```

■続いて、課題をデータベースから削除する。

```
teacher1@c7e37e1cfa2e:~$ nbgrader db assignment remove <課題名>
```

課題の作成など

■皆さん、自由に課題の作成・配布・提出などを行っていきましょう。

まだ資料を作成中

- 受講生が新たにMoodleコースを作成した際に、MCJ-CloudHubへのリンク（外部ツール）をコース内に貼り付ける手順について解説が必要。