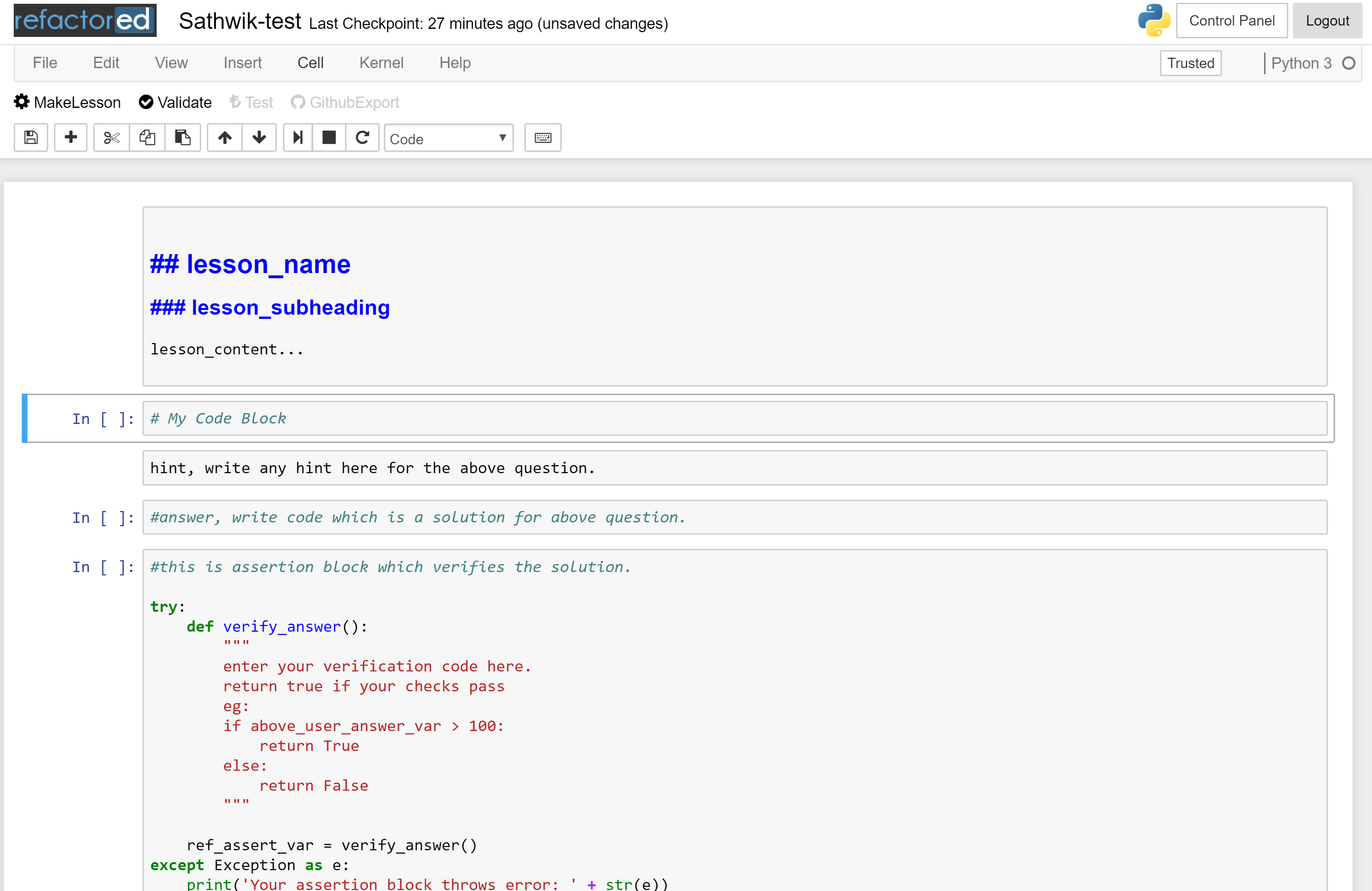


**Format for the Hackathon:**

The courses on Refactored follow a specific format. Every course is designed keeping this format in mind. Therefore, it is imperative for students to follow this format while submitting their notebook. The format is detailed below:

**The ‘Refactored’ Format for Lesson Building:**

Each lesson on Refactored follows a particular format in order to be seamlessly incorporated into product build. When you submit your notebook, kindly make sure that you have followed the format as shown below:

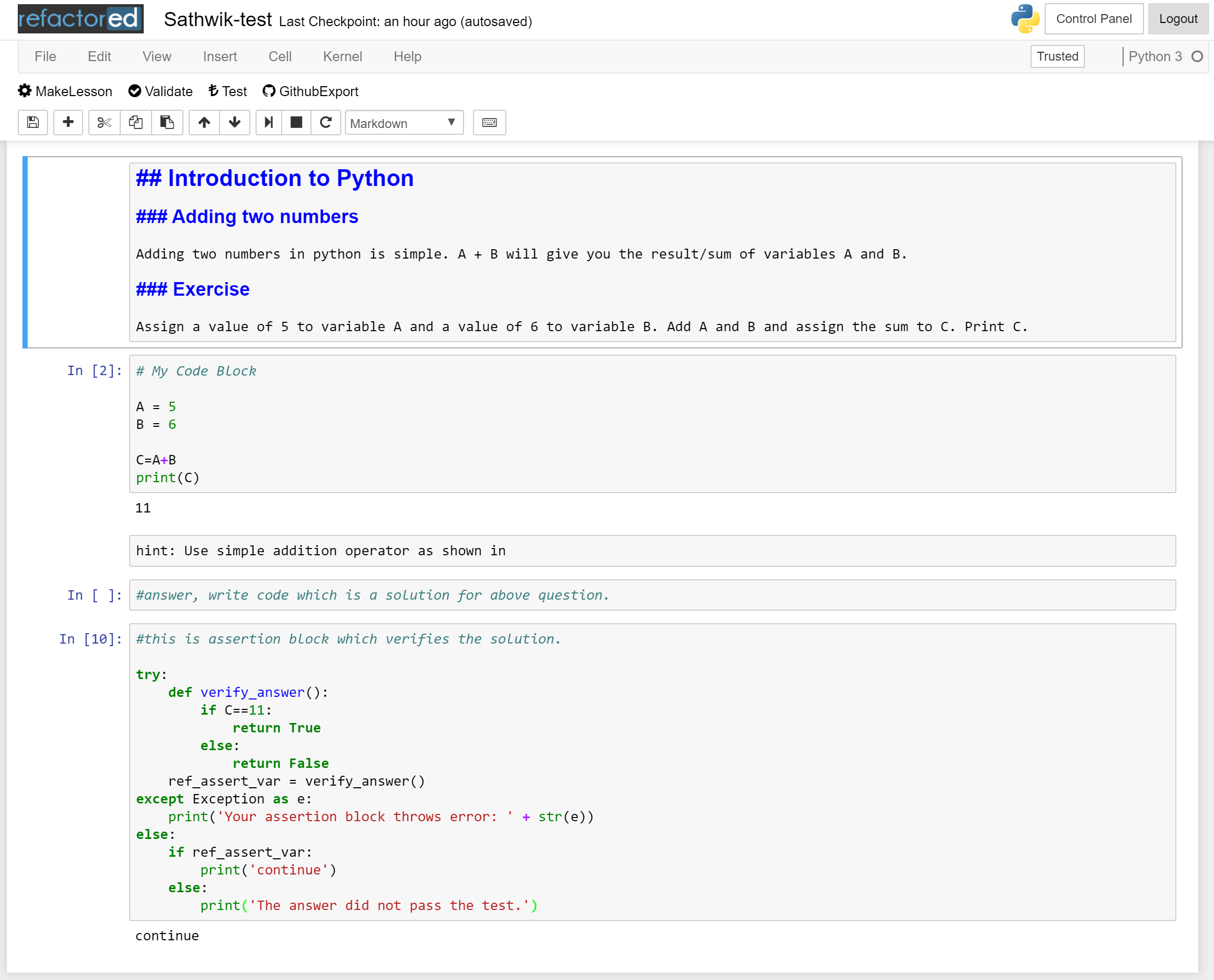


The above image shows the ‘FIVE-BLOCK’ structure followed for each lesson in a notebook. As seen above, you will have to present your insight in the form of a story. The aim is to present and explain your findings as if it was a lesson on Refactored.ai. You need to present it as if you are explaining it to a student. To understand better, consider yourself to be the person building content on Refactored.ai and you want students to solve this dataset, so you present them with content, explain your content, then present them with an exercise, give them hint to the exercise, provide them a block where they can write their code and then verify their code with an assertion block.

**FIVE-BLOCK structure:**

* **1st block – Content Block:** should be the description or analysis you are trying to do or an explanation. This block should end with presenting a question in the form of an exercise. The format and features to be followed in this block are detailed under the “Content format” header in the following section.
* **2nd block – Code Block:** This is your work area for this Hackathon. It is the cell in which all your code to solve the exercise resides in. On the refactored platform, this block will be the console, where the user enters his code.
* **3rd block – Hint Block:** is a Markdown cell, you need to write a hint in here for the question you are solving.
* **4th block – Solution Block:** is where the ideal solution for the exercise will reside in. The code given in this block will be shown as the solution (or key) on the platform, when the user is not able to solve the given exercise. For this hackathon, you may leave this cell empty as the solution code is already presented by you in the Code block.
* **5th block – Assertion Block:** The assertion block is where you need to verify the code students have provided (as described in the paragraph above). More details will be provided on this later.

For example:



**Content Format – The format within 1st Block**:

In Jupyter notebook, we will use "code" to write in code and "markdown" to write comments or explain what the code is about. This feature can be accessed as shown below:

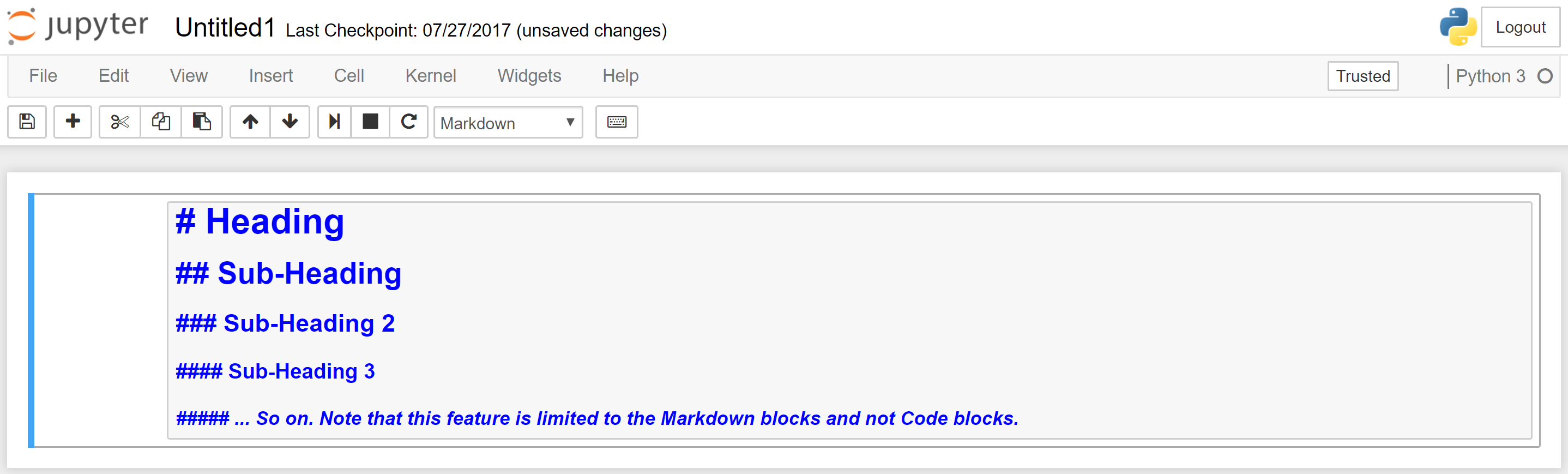
A screenshot of a cell phone

Description generated with very high confidence

**The 1st block – The Content Block should be a ‘Markdown’ Block** (also referred to as Markdown cell). The markdown cell allows features to format text as headings, embed HTML tags and media objects and also Latex codes to display mathematical formulas.

All insights derived from the dataset should be visualized and presented in the form of a lesson as seen on Refactored.ai. The format to specify section headers is as follows:

* Use single hash(#) for main title.
* Use double hash(##) for sub headings, for ex if you want to show that the section is about visualization, then use (## Data Visualization) in markdown
* All other subheadings follow from three hashes and more.
* Use html break tags (<br>) to provide more space between each lesson.



[Use this link](http://nestacms.com/docs/creating-content/markdown-cheat-sheet) to know more about the features in Markdown.

For example, students performing Exploratory data analysis of the datasets should present their finding as following:

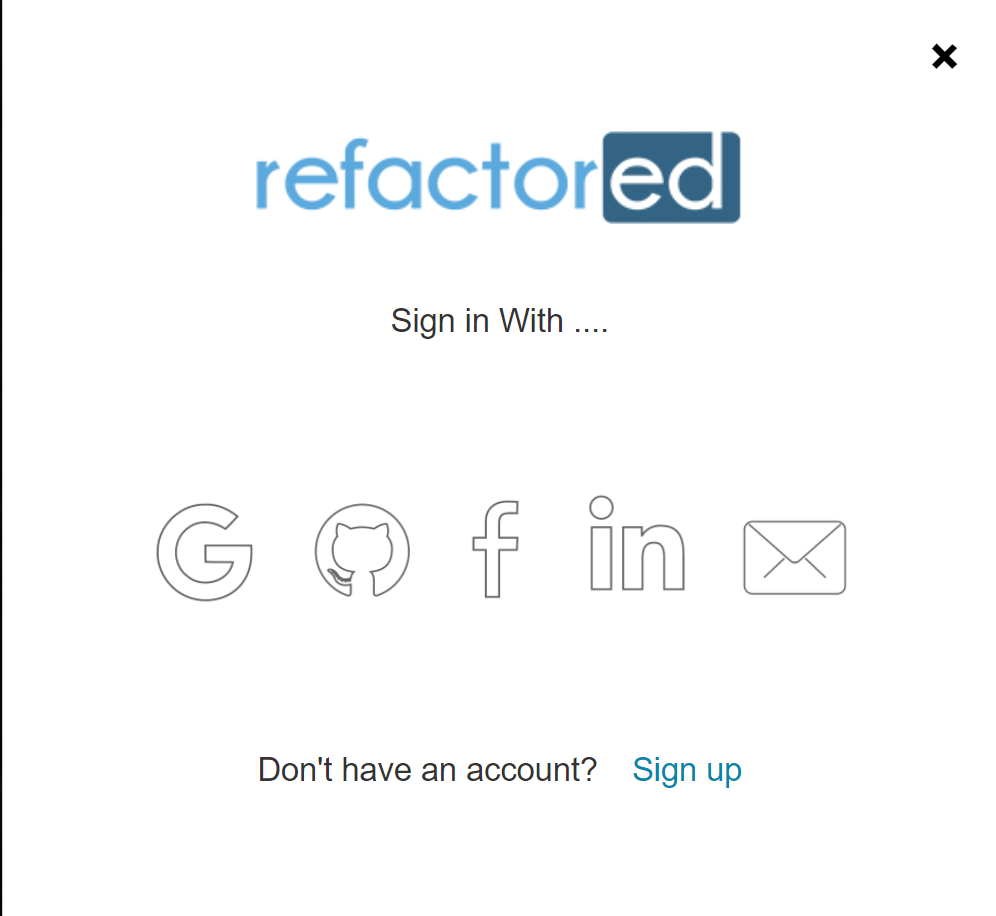
<main title> #Exploratory Data Analysis

<sub heading> ##Data Visualization

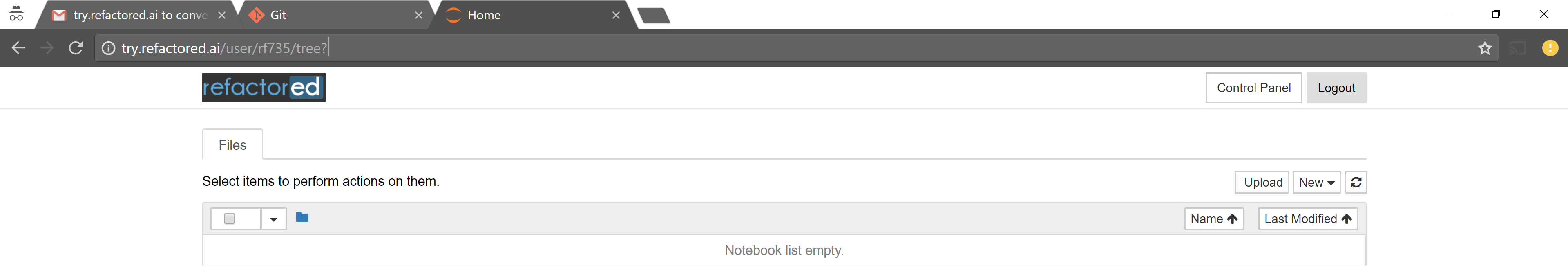
<sub heading> ###Relation between a and b variable

**Accessing the Jupyter Interface through Refactored:**

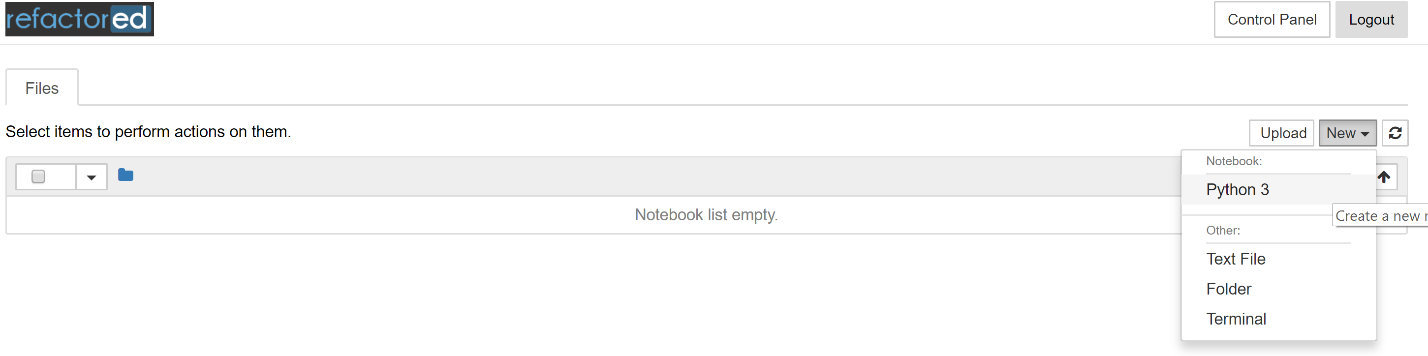
* Go to the link: [try.refactored.ai](http://try.refactored.ai/)
* You are then asked to login into the Refactored platform



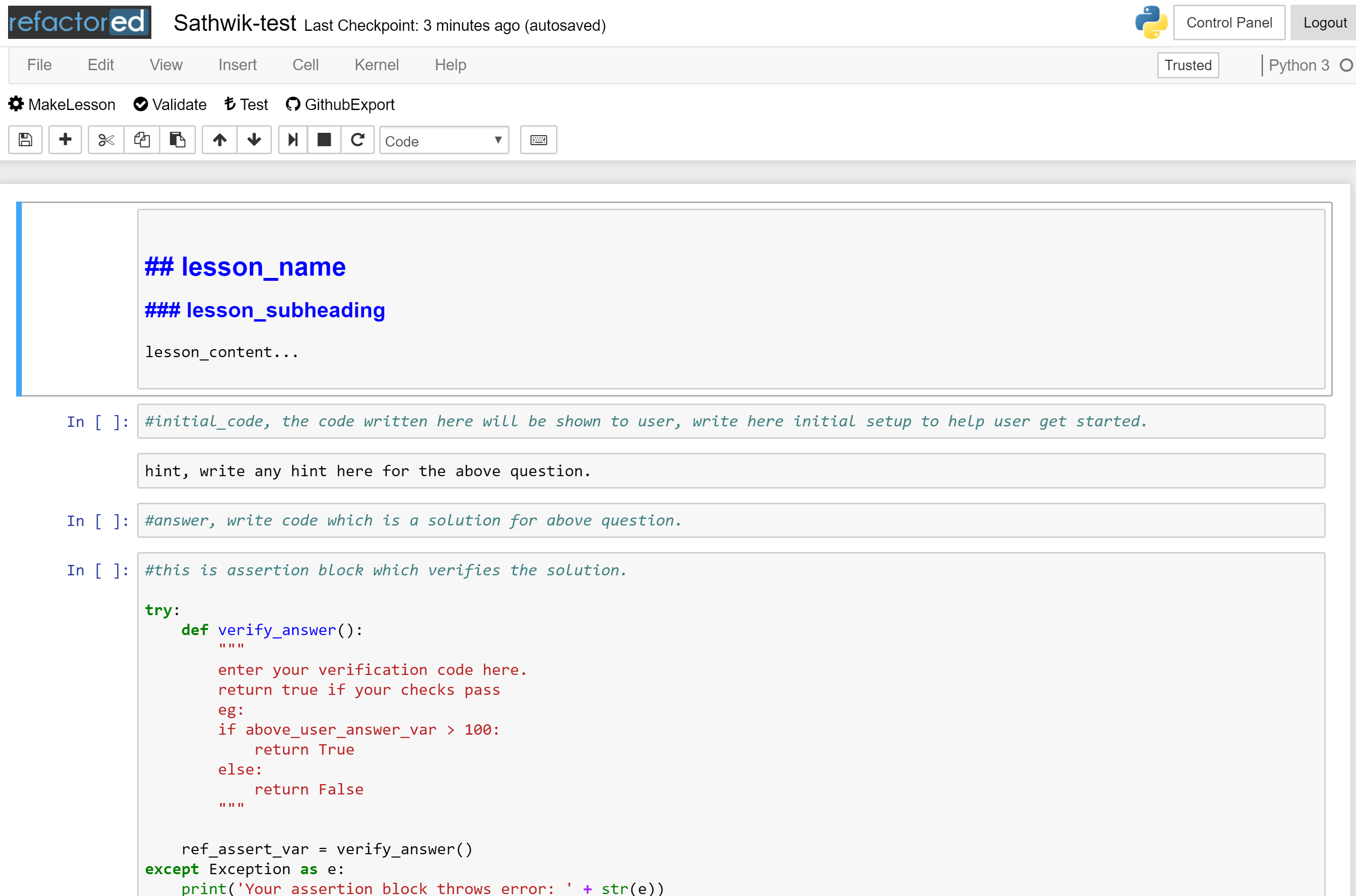
* After logging in you will see the home screen of the Jupyter Notebook interface via Refactored



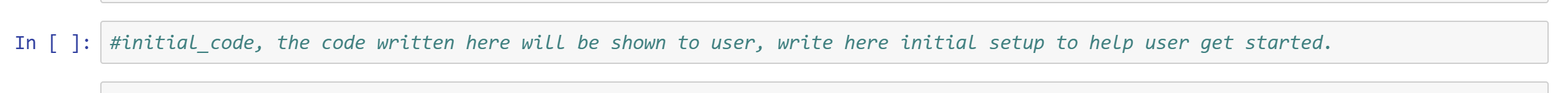
* Create a new Python 3 notebook. Rename the notebook adhering to the file naming convention. Name your lesson as <YourTeamName>\_UTDFall17.



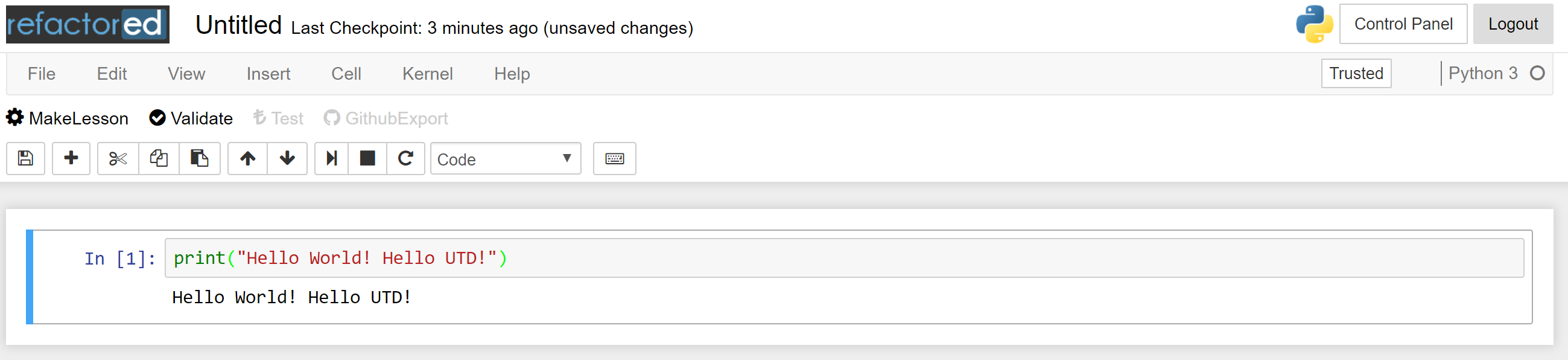
* The default notebook opens with the below format



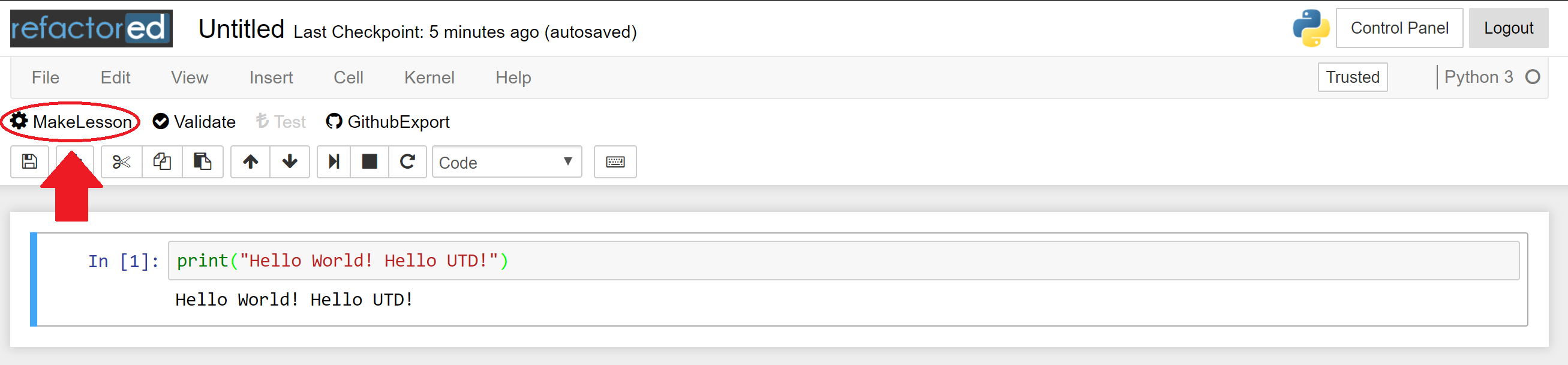
* You may begin writing code (to solve the datasets) in the second block – The code block.



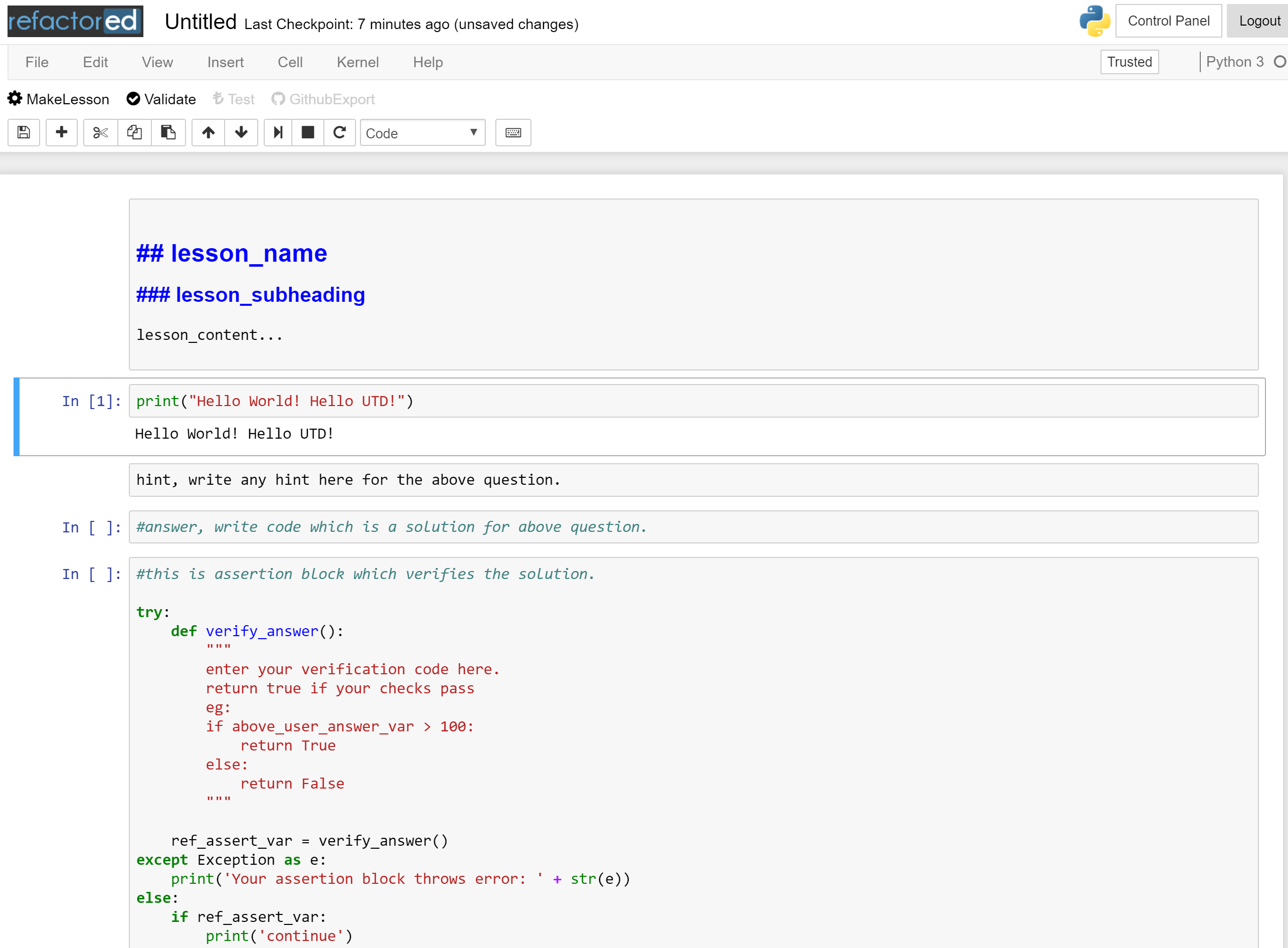
* Feel free to add code blocks and work towards the solution. Keep saving your work frequently. Once you are completed with your solution, **Make each code block into a Lesson**. The way you do it is explained below.
* Say you have written some code in a cell.



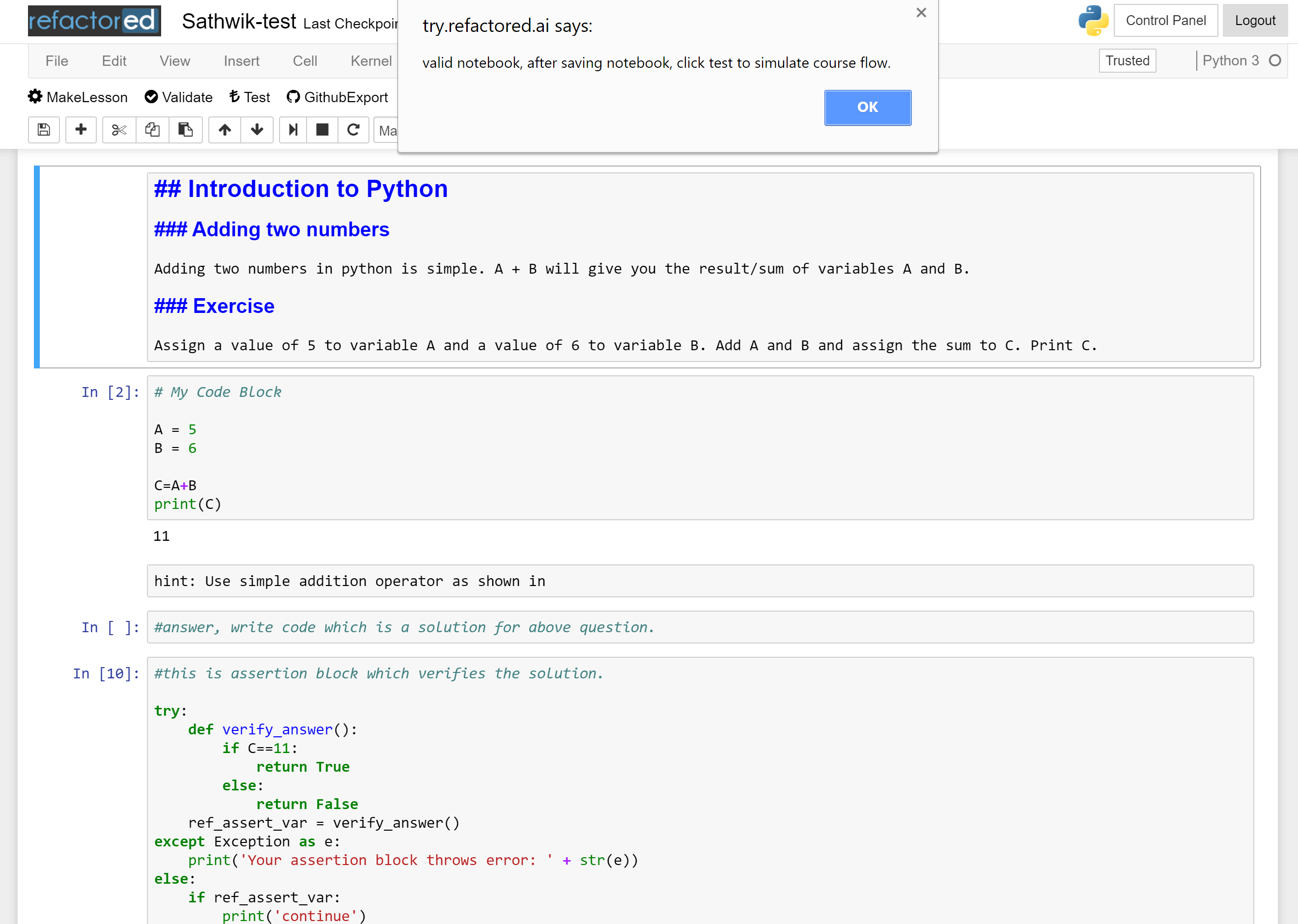
* While your code block is selected, click on Make Lesson Button to the top left side of the screen.



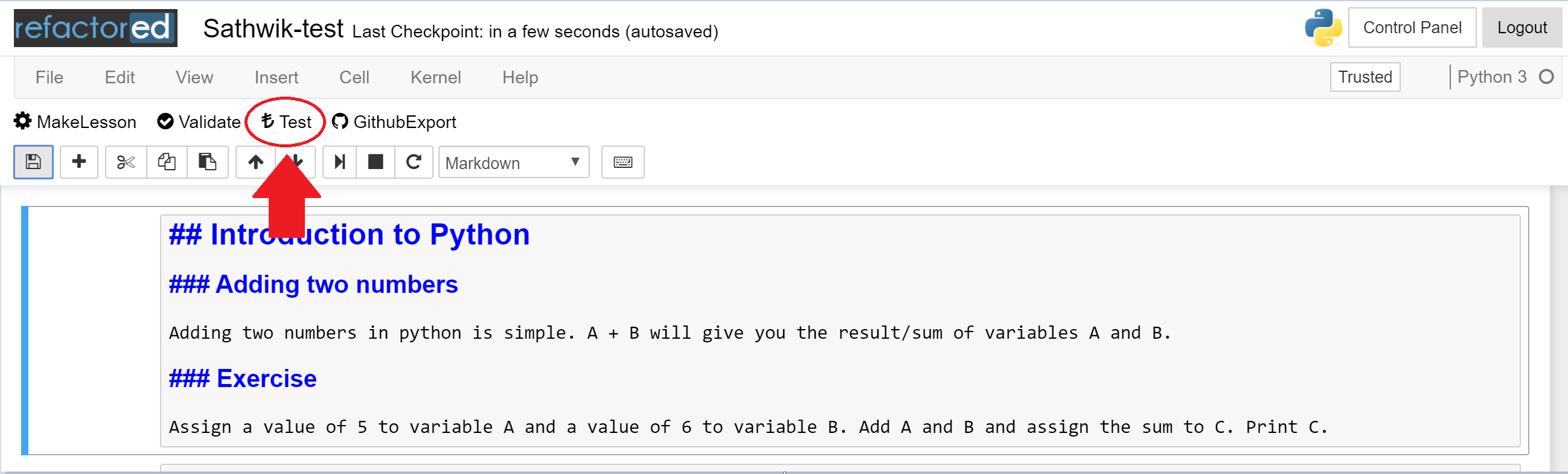
* Clicking on Make Lesson will add 4 blocks (the content, hint, solution and assertion block). The content block would be added above your code block and the remaining three blocks would be added below your code block.



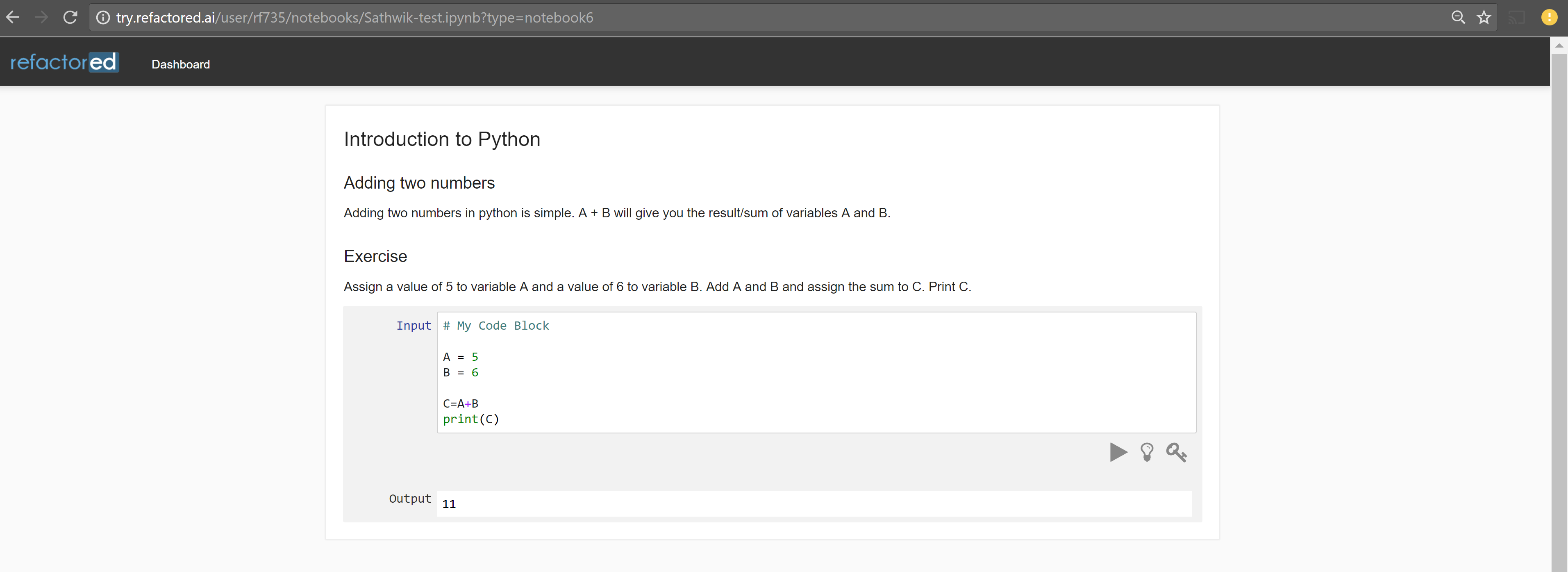
* You may add verification code in assertion block to verify the values of variables used in your exercise. This is to ensure whether the user has accurately solved the exercise, created all necessary variables and arrived at the right solution.
* Keep saving your progress frequently by clicking on the “Save me” icon right below the “Make Lesson” button. Once you complete with making you solution and lesson, click on the “Validate” button next to the “Make Lesson” button. If your notebook is maintaining the correct format, it will pass Validation and the below message will appear.



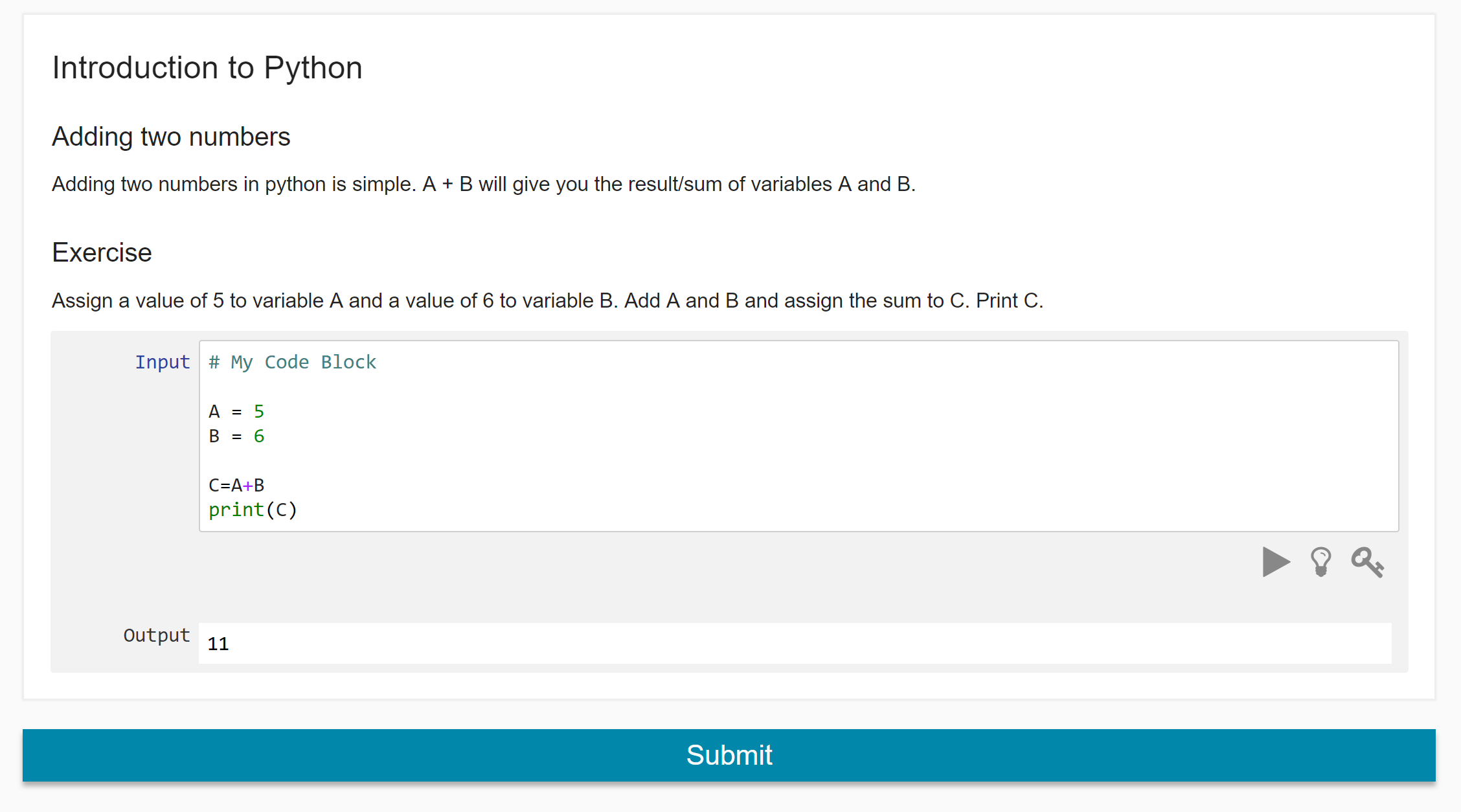
* After saving the notebook, you can simulate the workflow of your notebook on the refactored platform by clicking on the test button



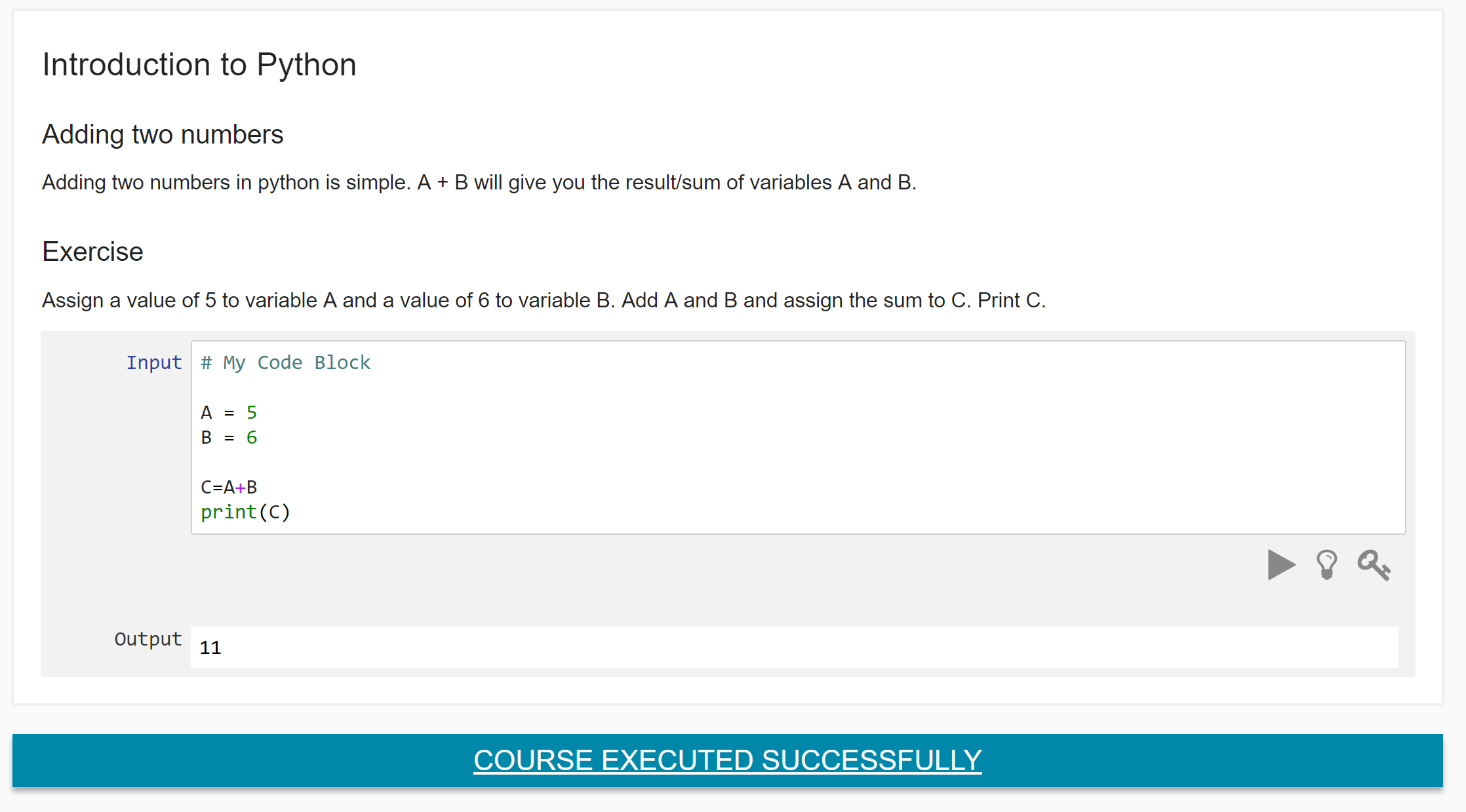
* The test button opens up the Refactored platform screen and displays your lesson.



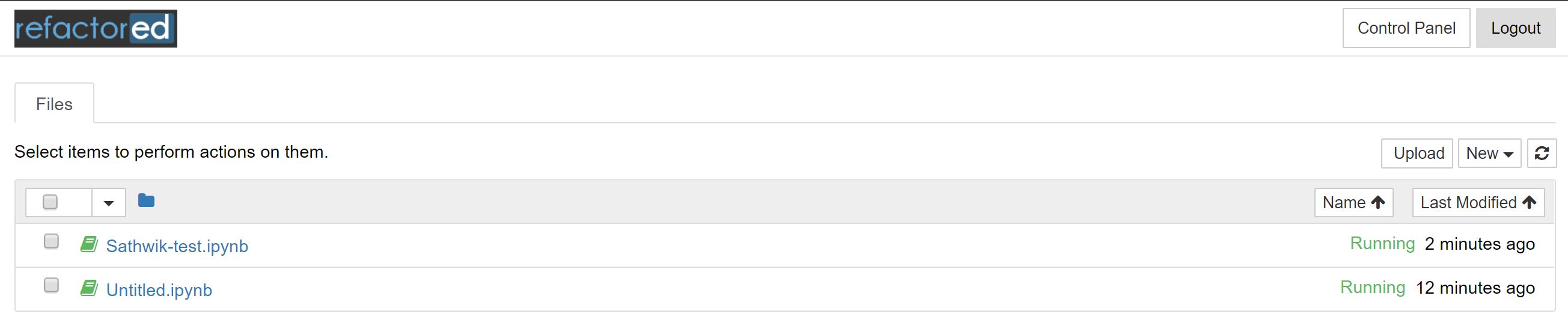
* You can test out the various buttons to check if your hint, key block and code are all running fine. Once your code runs successfully and passes the assertion conditions you would have set in the assertion block (5th block), a submit button would appear.

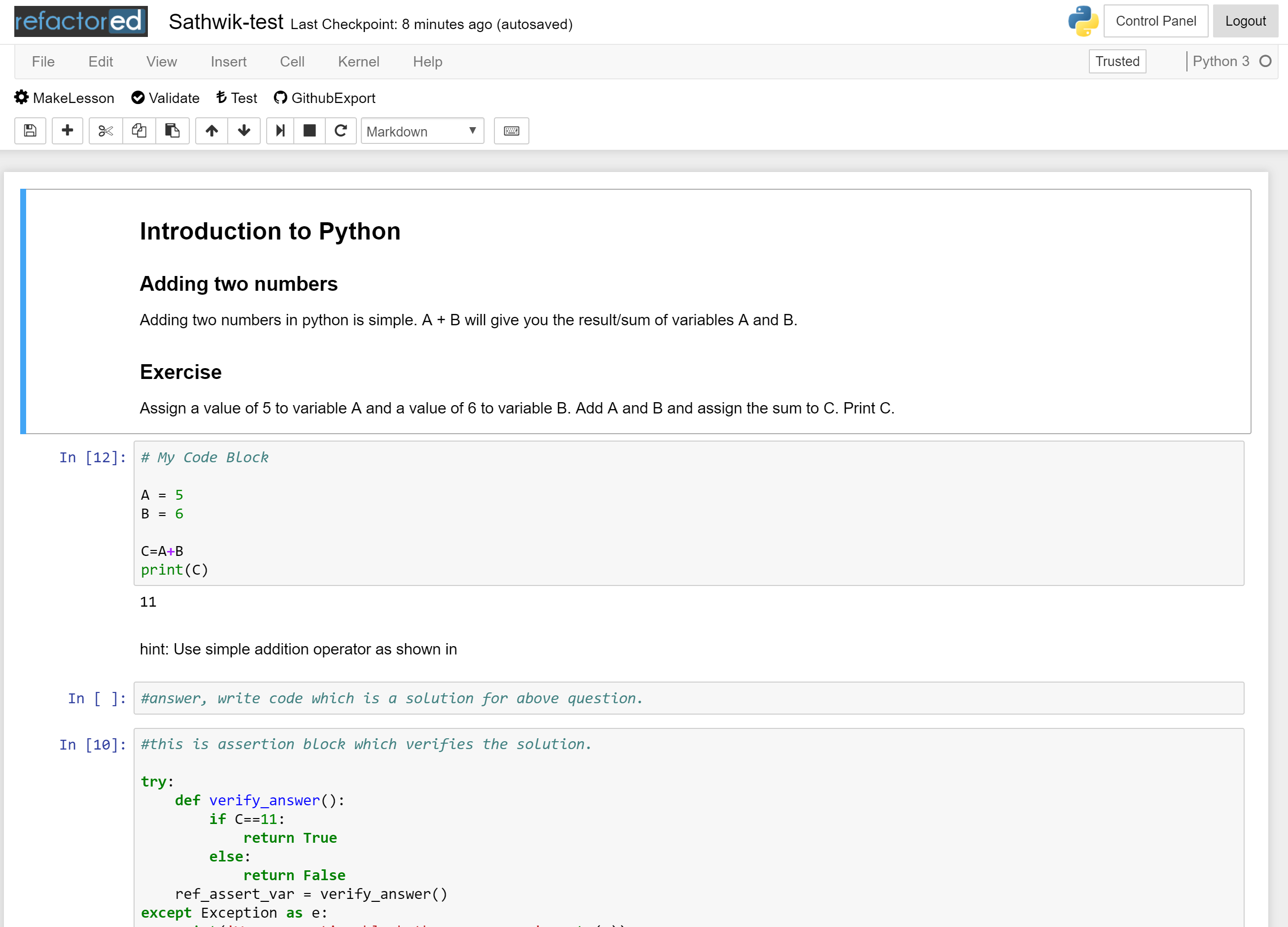


* Clicking on Submit will change the button to “COURSE EXECUTED SUCCESSFULLY” if everything passes.

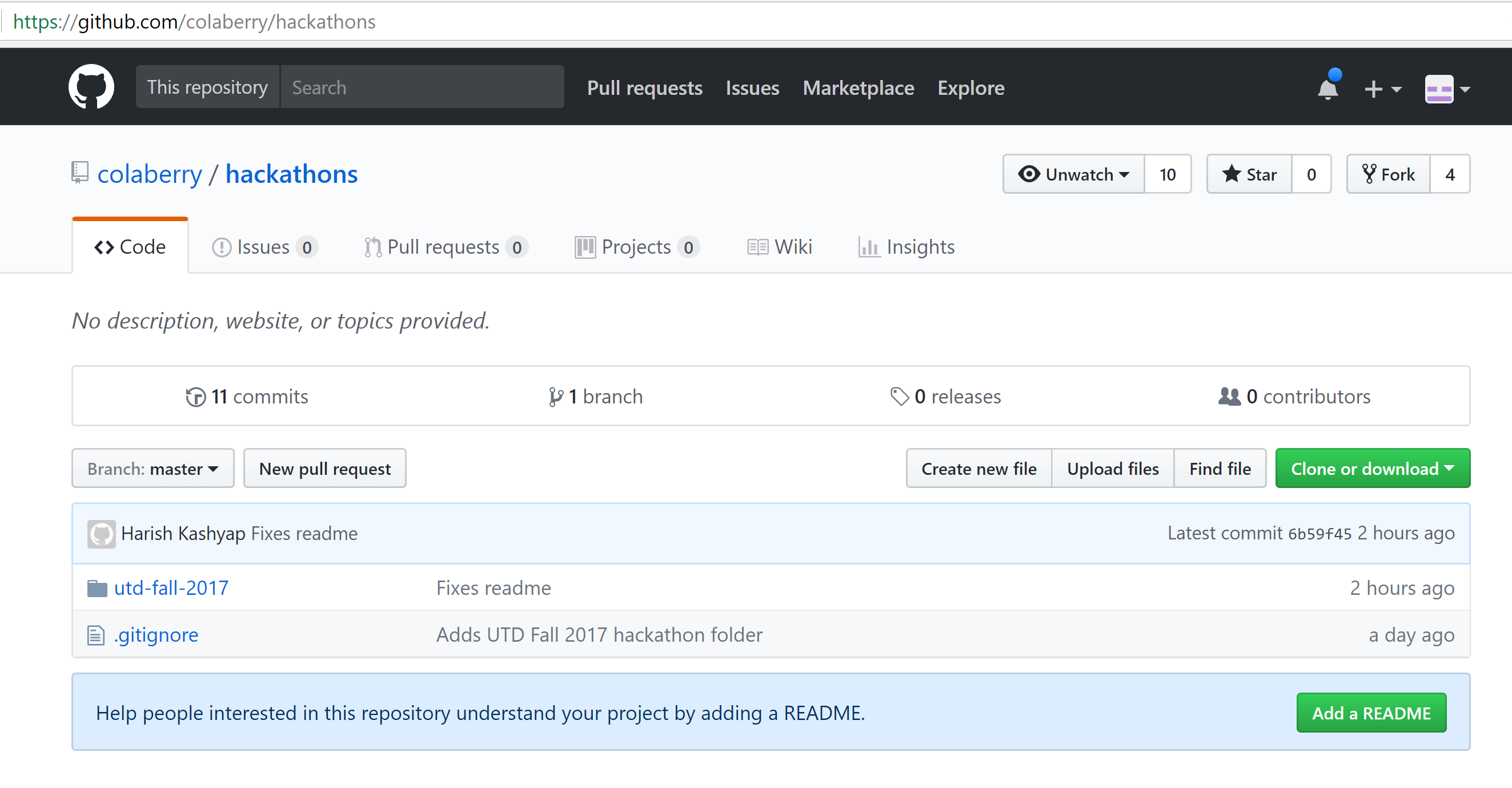


* Go back to the Jupyter Interface you Initially started with and re-open your notebook.

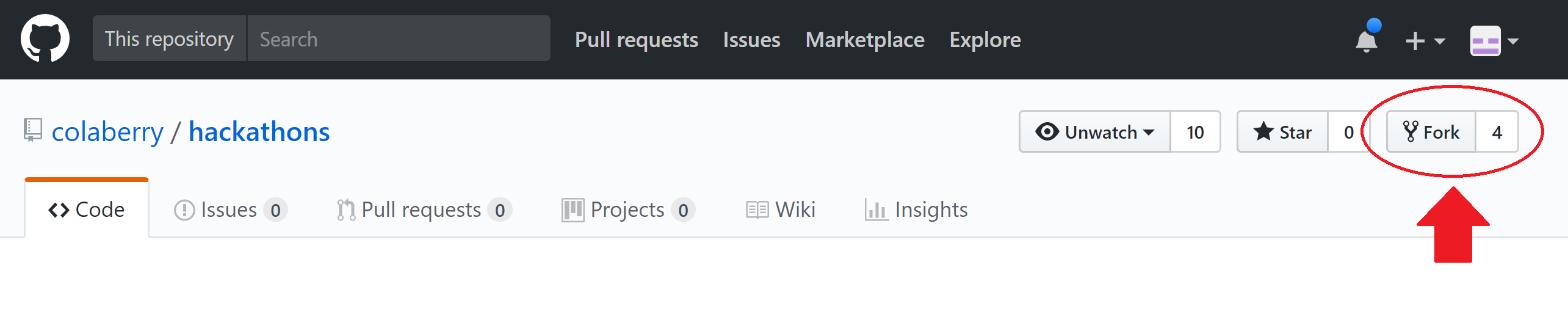




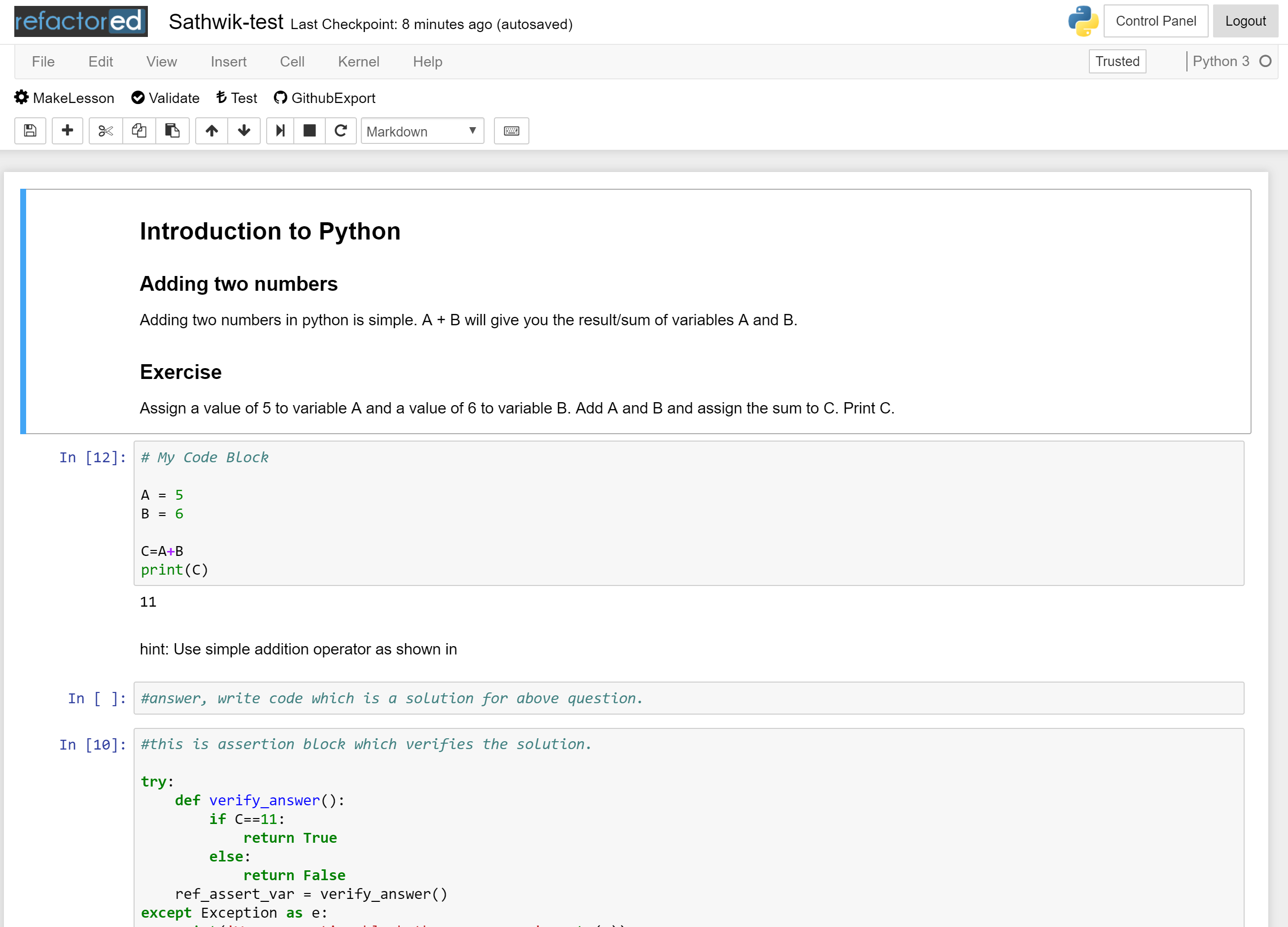
* Open your web browser and go to Github. On Github go to the Colaberry/Hackathons repository (Link: <https://github.com/colaberry/hackathons>)



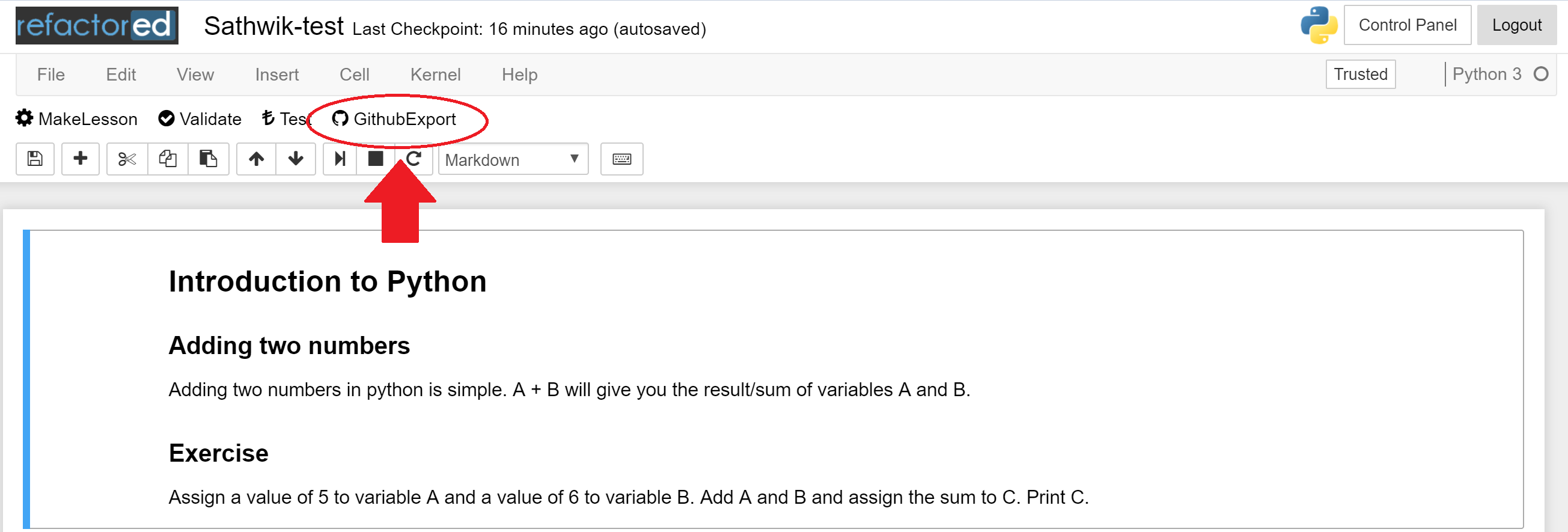
* On the top right side there is a “Fork” button. Click on it to create a copy of this repository in your personal Github account. This also creates a link, through which work you create in Hackathons repository in your personal account can be added to the Master Hackathons repository in Colaberry’s organizational account.



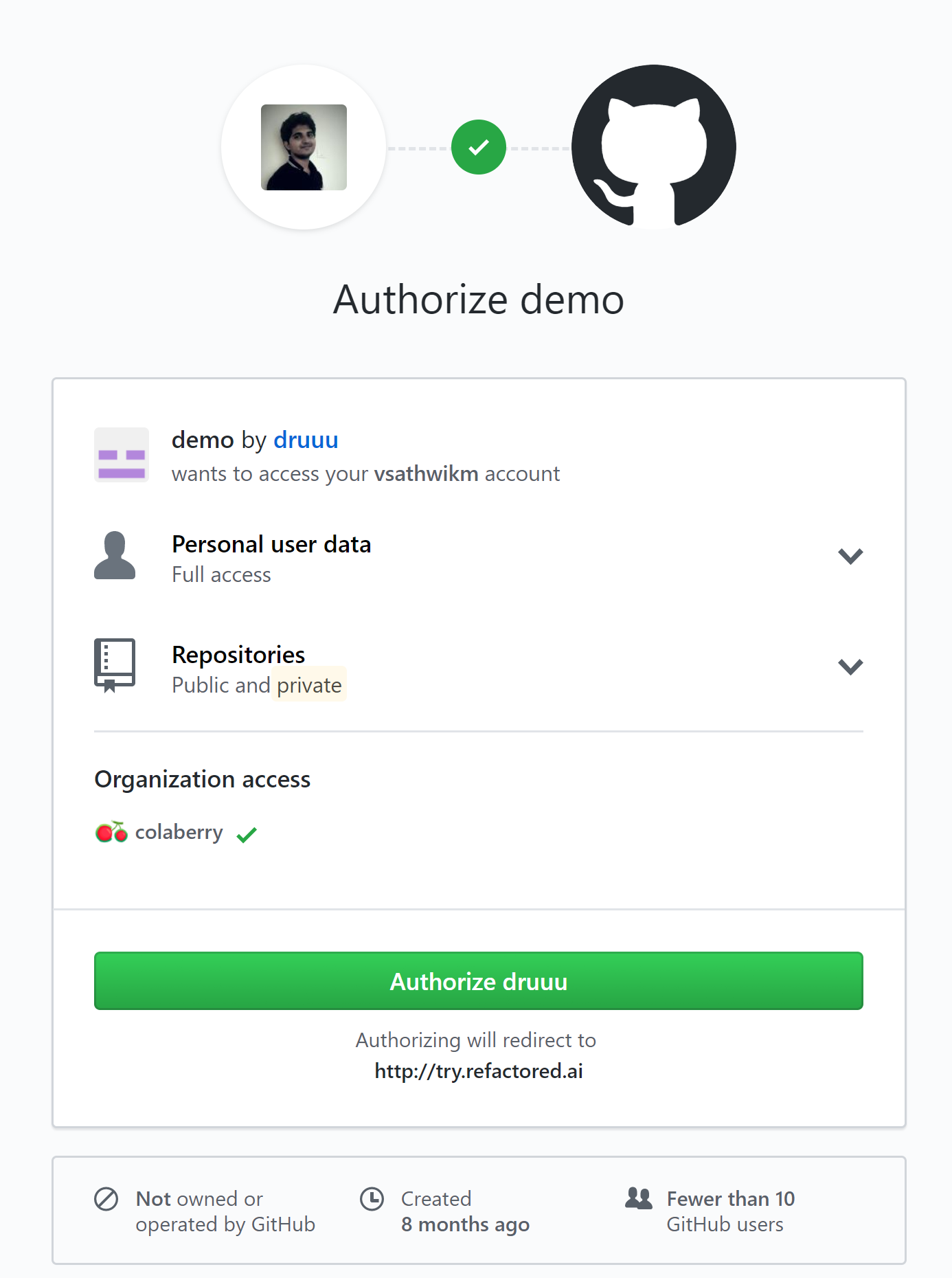
* Come back to the Jupyter Interface and your notebook



* Click on the “GithubExport” button



* You will see a request on the browser screen. This request is from Colaberry’s admin. The request is for Colaberry to seek permission to add work from (the forked) Hackathon repository on your personal account to Hackathon master repository on Github. Click on “Authorize druuu” button to give Colaberry access to your work. This enables a link to showcase your work on Colaberry’s Refactored platform.



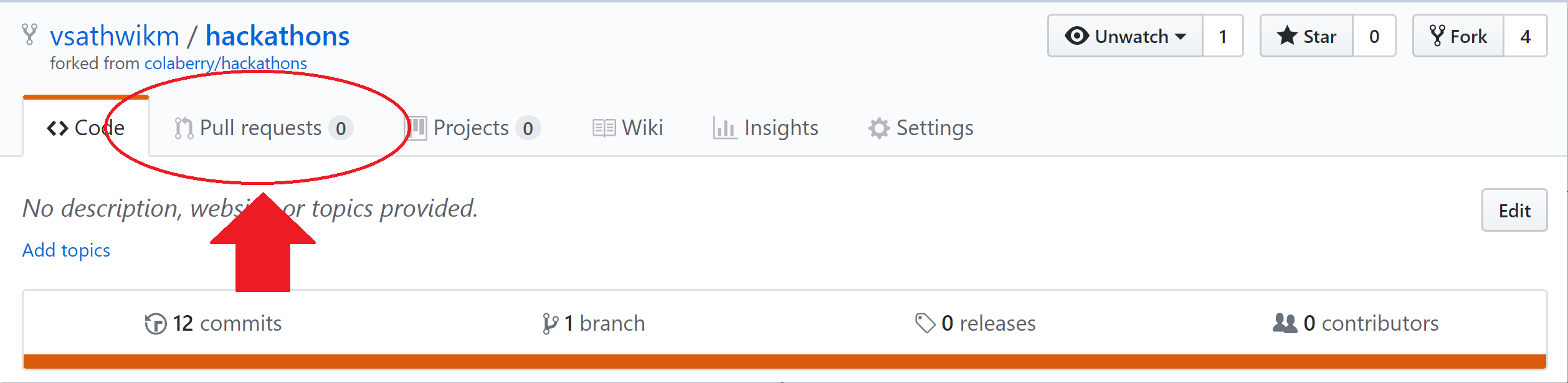
* Once you authorize, the screen shows you the list of all repositories in your personal github account and asks you to “Select a Repository”. Make sure you select the Hackathons repository that you have earlier forked from Colaberry in this step.



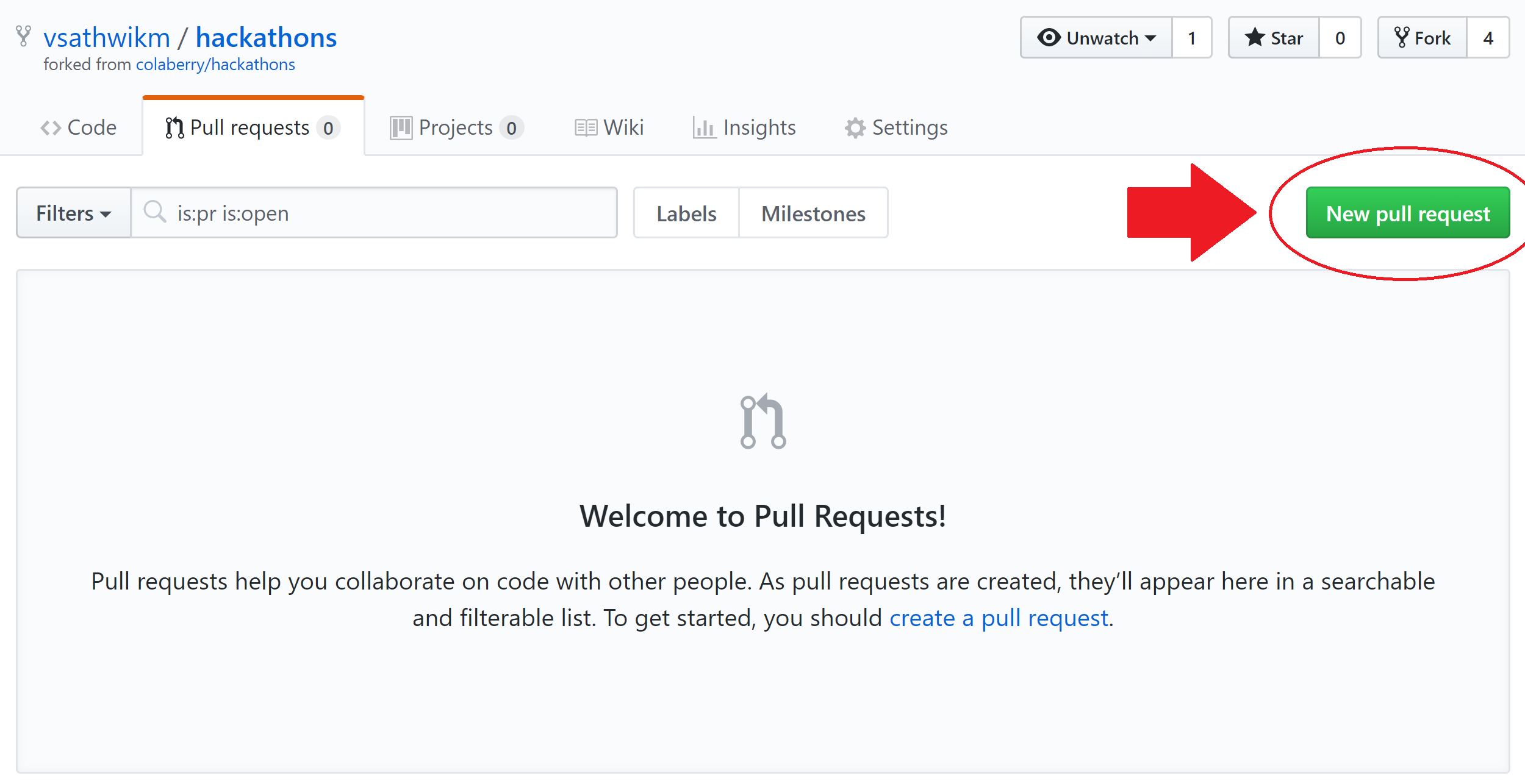
* The notebook will then be successfully exported to your hackathons repository and you will be redirected to your notebook. Now go to github and view hackathons repository in your personal account to view the exported notebook.



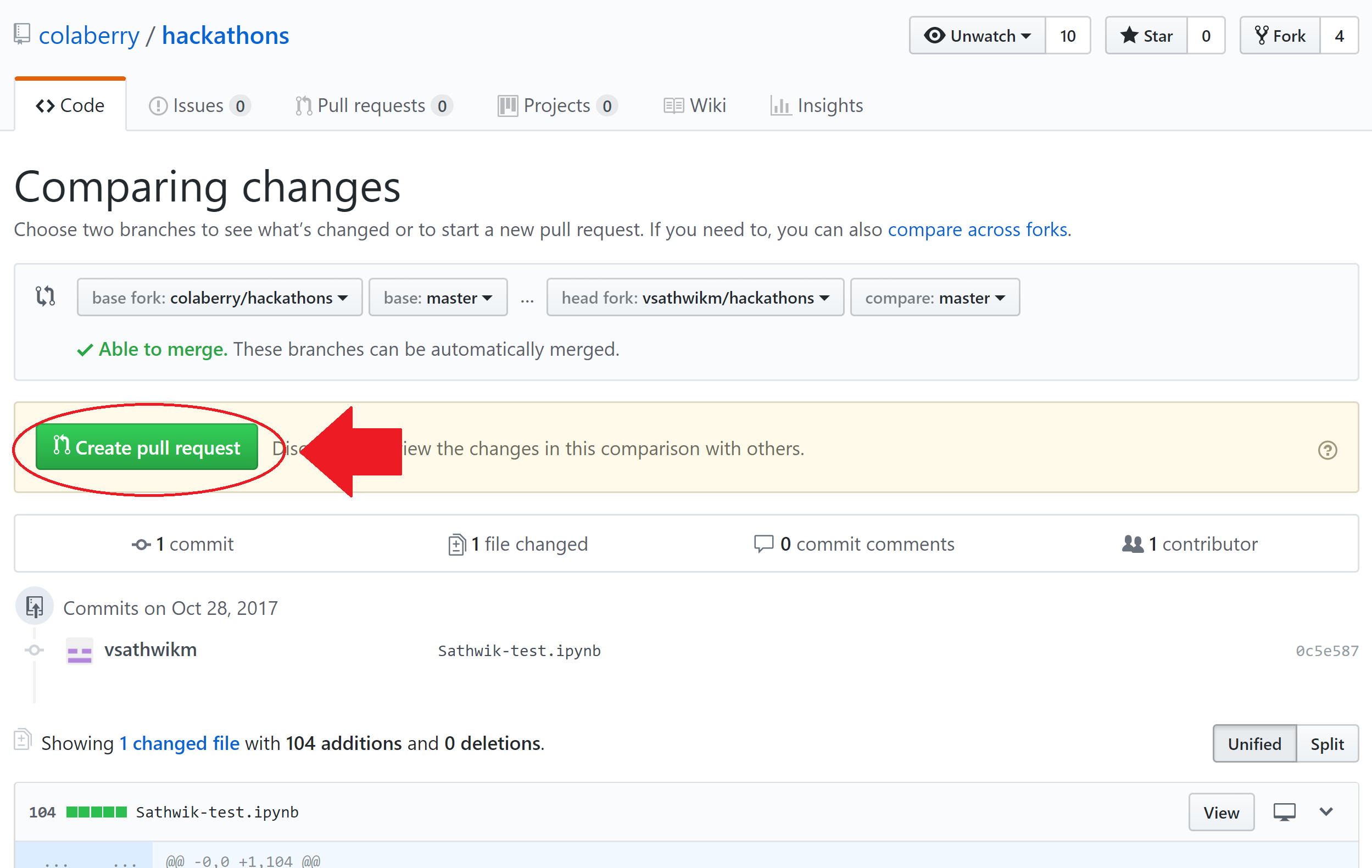
* Now click on the “Pull requests” tab



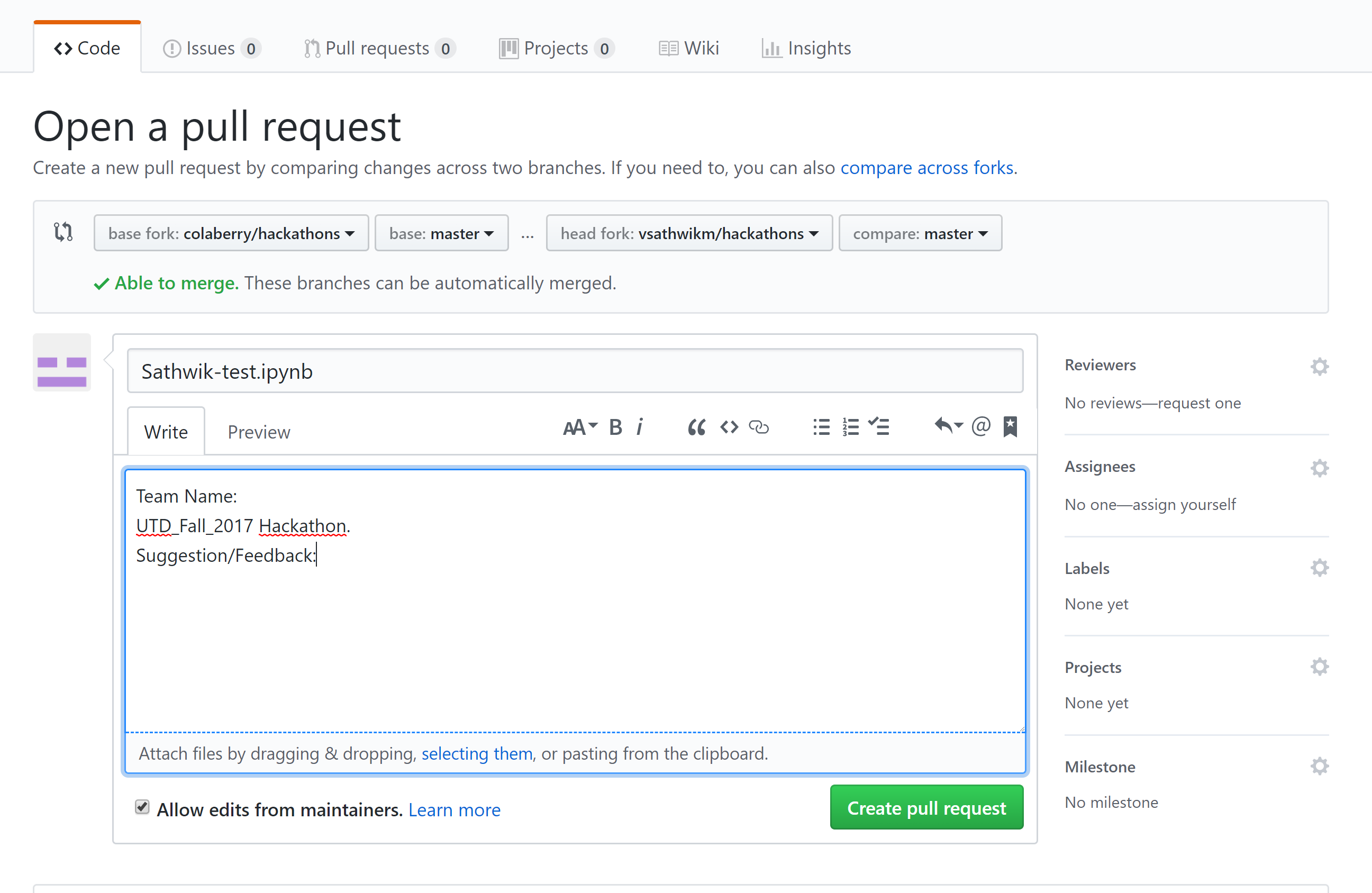
* In the next page, click on “New Pull Request”



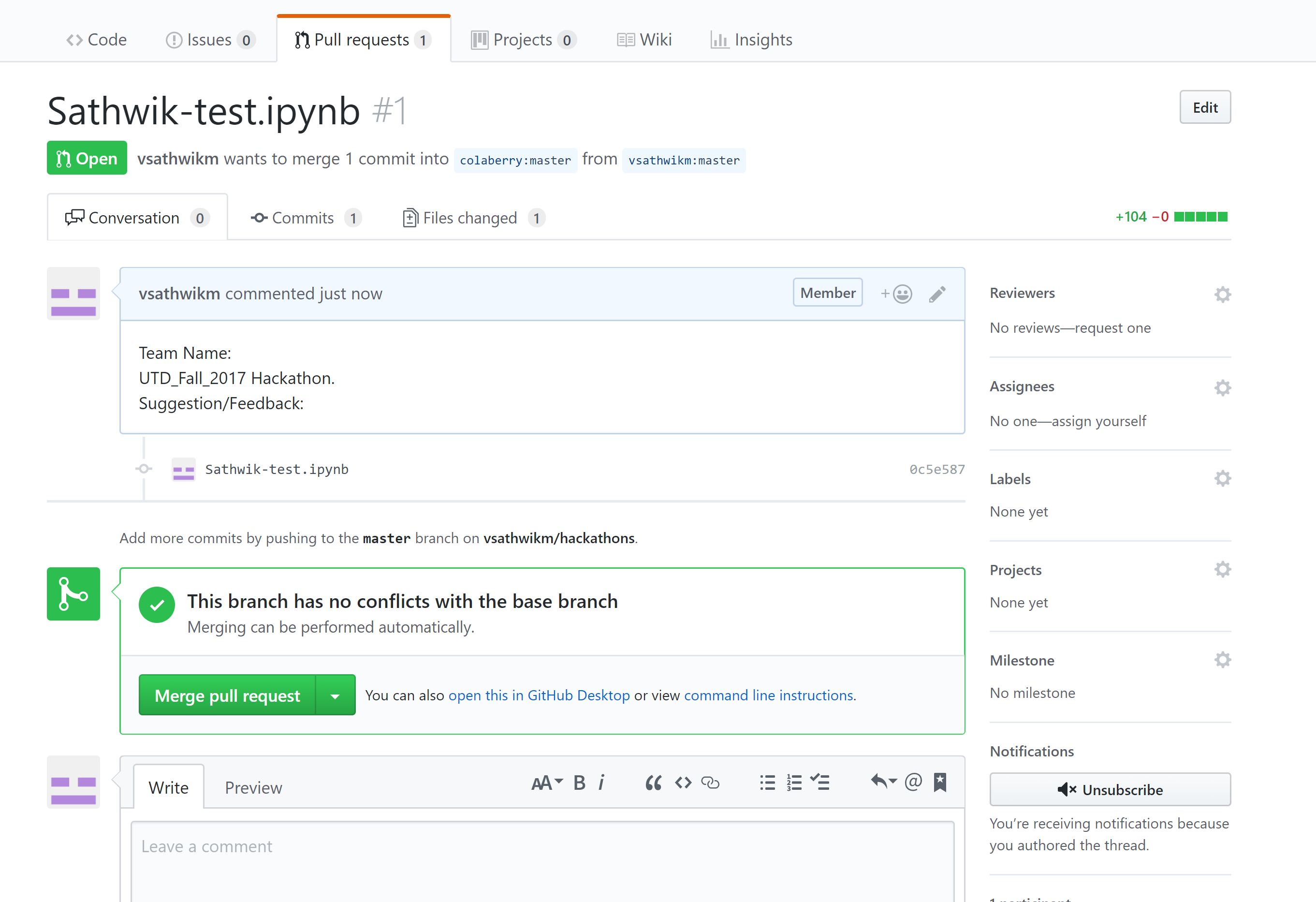
* In the next screen, a request to pull code from your local repository onto Colaberry’s organization repository would be created. Ensure that “base fork” would be Colaberry/Hackathon and “head fork” would be <yourgithub>/hackathon. **Do not change anything here.** Click on the “Create Pull Request” button.



* In the next, “Open Pull Request” screen, Add your Team Name, UTD\_Fall\_2017 and any suggestions/feedback you have and click on “Create Pull Request” button.



* Once the request is created, you are all set from your side. Congratulations on completing the tasks!



**GITHUB Basics:**

Kindly have an account ready on GITHUB before you present your findings. We will provide you with the Repository where you will need to push you work. **Only 1 push per team. Multiple push from same team will lead to disqualification. The push should be named under the team name**.

GITHUB workflow is provided here as follow:

* Get access to the Github repository (provided by us as a link on GITHUB)
* Install Git bash on desktop

**Windows:** https://git-scm.com/download/gui/win

**Mac:** https://git-scm.com/download/mac

**Linux:** [https://git-scm.com/download/gui/linux](https://www.google.com/url?q=https%3A%2F%2Fgit-scm.com%2Fdownload%2Fgui%2Flinux&sa=D&sntz=1&usg=AFQjCNHAE_YehDUIsmZdRrwMaW_7XsZ1vg)

* Open bash terminal

Change directory to documents

* **Command:** cd \documents

Initiate git repository (if using for the first time)

* **Command:** git init
* This command creates a git repository in the current folder. However, by default this folder is considered as the master.

Generating a SSH key

* **Command:** ssh-keygen -t rsa -b 4096 -C "email-registered-on-github"
* While creating SSH
* Do not give any name to the file. Just keep hitting enter.
* You may or may not assign a passphrase. If you do assign remember it.

Copying the RSA key

* **Command:** clip < ~/.ssh/id\_rsa.pub
* Enter this key in online github portal SSH key tab under account settings. Provide PC name and add the SSH key.

Cloning the repository to remote location/desktop

* **Command:** git clone [git@github.com:xyz/abc.ai](mailto:git@github.com:xyz/abc.ai) (Just an example)

Change directory to the cloned repository

* **Command:** cd abc.ai

Pull the latest version from the master source on github

* **Command:** git pull

Create a branch within the master git repository

* **Command:** git checkout -b Team1

\*\*\* Perform required Operations/edits to the code or documents \*\*\*

After making changes you can check status of the repository to show and review all the changes made

* **Command:** git status

Add the file/files that has/have been changed

* **Command:** git add <filename> (Eg. git add README.md)

Commit the changes in remote source to online github repository

* **Command:** git commit –m “Updates Readme\*\*”

Push the changes to the master repository to which you are a collaborator

**Command:** git push --set-upstream origin Team1

Kindly note, that it is expected from each student to have basic knowledge about GITHUB, of how to push a notebook using remote terminal.

**FEEDBACK**

**After you have completed your challenge, kindly fill the feedback form which will be provided to you at the time of Hackathon. Each member from the team is required to fill this form as it will be cross checked with the submission done on GITHUB and any member of the team not filling the survey will lead to negative marking on their final submission.**

**ALL THE BEST.**

**Refactored.ai Team**