

INSY 336 - DataHandl&Coding for Analytics

EdStem & Datacamp Tutorials

Instructor: Prof. Kyunghee Lee

TA: Yanda Tao

Fall 2025 - Section 001 & 002



EdStem

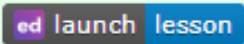
Introduction of EdStem

EdStem

EdStem will be our main platform for running Python code. It provides a Jupyter notebook environment directly in your browser with no setup required. EdStem also serves as our learning platform where you can:

- Submit assignments and receive feedback
- Ask questions and discuss course materials with classmates
- Access course resources

How-to

- Click on this badge  whenever you see it to open notebooks in EdStem
- You've already been added to the course on [EdStem](#) - just sign in with your McGill email
- The Ed course will be active on the first day of classes.
- See [EdStem tutorials](#) for more.

Click here to access the EdStem platform.

You can find this page in this [Link](#).

The Interface of EdStem

At the **top** of the page, you can select which section you want to access in the EdStem platform: Lessons or Discussion

Click here to access the [discussion](#) page

The discussion page is used to check announcements for this course, communicate with classmates, ask questions to the TA or professor, and answer questions from your peers.

Click here to access the [Lessons](#) page.

Lessons page contains:

- Assignments
- Exams
- Quiz

Discussion Page

The screenshot shows the McGill Ed Discussion interface. On the left is a sidebar with a 'New Thread' button and a list of categories: General, Lectures, Sections, Problem Sets, Assignments, and Social. The main area displays a list of discussion threads. On the right is a 'New Question' form with tabs for Question, Post, and Announcement. The form includes a title field, a category selector (with 'General' selected), a rich text editor, and checkboxes for 'Pinned' and 'Private'. A 'Post' button is at the bottom right of the form. A top navigation bar contains icons for chat, notifications, analytics, settings, home, and a user profile.

ed McGill INSY 336 001 – Ed Discussion

Cancel

New Thread

Search

Filter

COURSES

CATEGORIES

- General
- Lectures
- Sections
- Problem Sets
- Assignments
- Social

Select type of your thread “Question” or “Post”

Select category of your thread.

If you don’t want your thread to be seen by others (except the TA and the professor), you can mark it as **Private**.

Click this to create a new thread

New Question

Question Post Announcement

Title

Category

General Lectures Sections Problem Sets Assignments Social

Paragraph

B *I* U <> ↺ ↻

☐ Pinned
Keep at top of thread list

☐ Private
Visible to you and staff only

Post

To enter this page, you can click this button.

Include Your Code on the Discussion page

To include runnable Python code in your question, click the **code snippet** option and paste your code inside it.

New Post

? Question Post Announcement

Title

Category General Lectures Sections Problem Sets Assignments Social

Paragraph B I U <> 🔗 ☰ ☷ 🖼️ 📎 ▶️ 📄 Σ <> 🌐 ...

Code Snippet

You can copy your code directly into the code snippet box so that the TA and professor can better understand and address your questions. If you'd like to switch the programming language, click the **Python** button.

? Question Post Announcement

Title

Category General Lectures Sections Problem Sets Assignments Social

Paragraph B I U <> 🔗 ☰ ☷ 🖼️ 📎 ▶️ 📄 Σ <> 🌐 ...

Run Line Numbers Runnable Python

```
1 x = 1
2 y = 1
3 print(x+y)
4
```

2

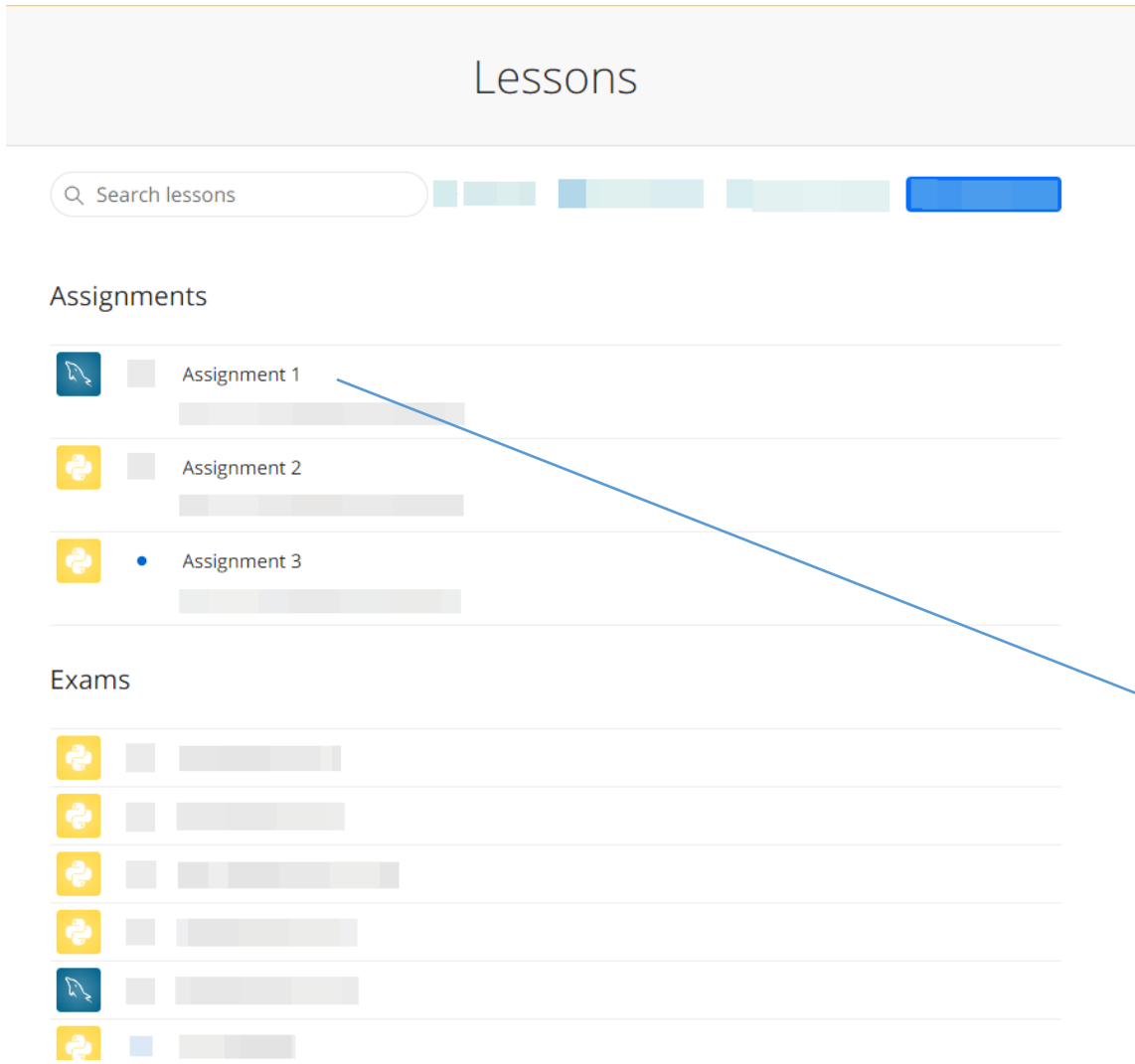
Usage of the Discussion Page

Note: In this course, important announcements will be posted on the discussion page of this platform (rather than on MyCourses announcements). Please check the discussion page frequently.

We also strongly recommend that you post your questions on the discussion page instead of sending emails, as the TA and professor can respond more quickly.

In addition, we encourage peer assistance: you are welcome to answer questions from your classmates. Your contributions on Ed Discussion will be considered when assigning participation grades.

The Interface of Lessons Page



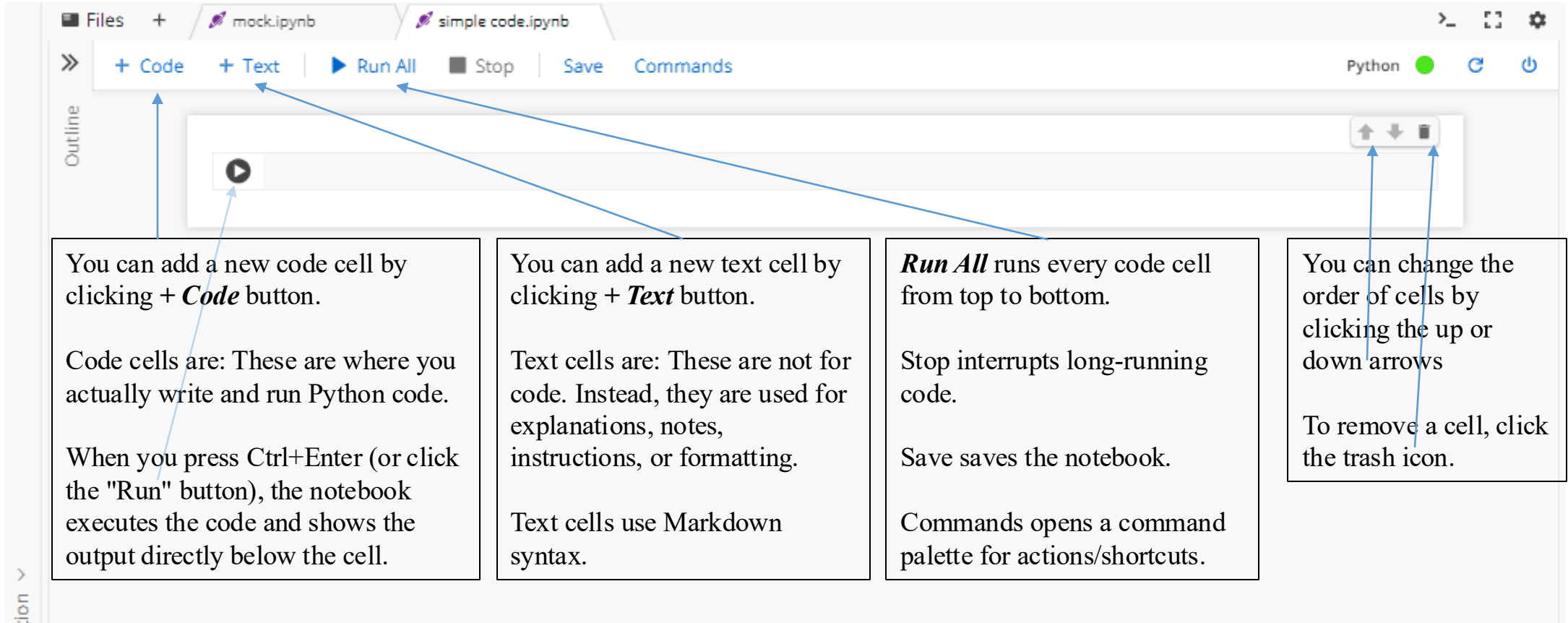
When you click on the Lessons page, you will see a list of **assignments, quizzes, and exams**.

Please note that it is normal for this page to appear empty at the beginning of the semester. Assignments and quizzes will become visible as the course progresses.

For example, you can click here to access into the Assignment 1.

Jupyter Notebook (1)

A Jupyter-style notebook allows you to run a Python kernel in your browser.
Jupyter Notebook will also be the main format used for **assignments in this course**.



The screenshot shows a Jupyter Notebook interface with two tabs: 'mock.ipynb' and 'simple code.ipynb'. The 'simple code.ipynb' tab is active, showing a single code cell. The interface includes a top toolbar with buttons for '+ Code', '+ Text', 'Run All', 'Stop', 'Save', and 'Commands'. On the right, there are status indicators for 'Python' (green dot), a refresh icon, and a power icon. Below the toolbar, a code cell is visible with a play button icon on the left and up/down arrows and a trash icon on the right. Blue arrows point from the text boxes to these specific UI elements.

You can add a new code cell by clicking + **Code** button.

Code cells are: These are where you actually write and run Python code.

When you press Ctrl+Enter (or click the "Run" button), the notebook executes the code and shows the output directly below the cell.

You can add a new text cell by clicking + **Text** button.

Text cells are: These are not for code. Instead, they are used for explanations, notes, instructions, or formatting.

Text cells use Markdown syntax.

Run All runs every code cell from top to bottom.

Stop interrupts long-running code.

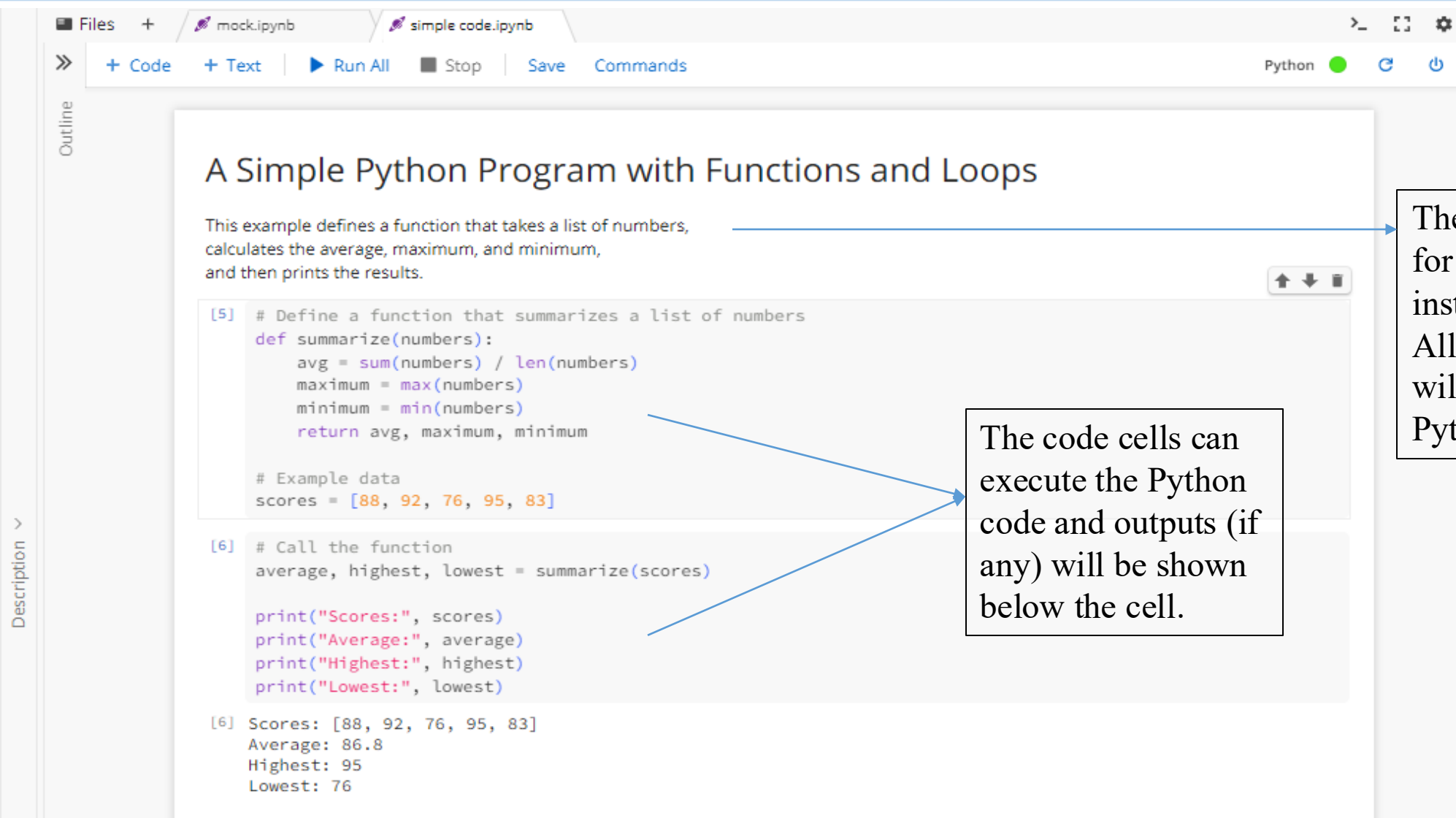
Save saves the notebook.

Commands opens a command palette for actions/shortcuts.

You can change the order of cells by clicking the up or down arrows

To remove a cell, click the trash icon.

Jupyter Notebook (2) – An example



The screenshot shows a Jupyter Notebook interface with three cells. The first cell is a text cell containing an introduction to a Python program. The second and third cells are code cells containing Python code. The output of the third cell is displayed below it. Arrows point from explanatory text boxes to each of the three cells.

Files + mock.ipynb simple code.ipynb

+ Code + Text Run All Stop Save Commands Python

A Simple Python Program with Functions and Loops

This example defines a function that takes a list of numbers, calculates the average, maximum, and minimum, and then prints the results.

```
[5] # Define a function that summarizes a list of numbers
def summarize(numbers):
    avg = sum(numbers) / len(numbers)
    maximum = max(numbers)
    minimum = min(numbers)
    return avg, maximum, minimum

# Example data
scores = [88, 92, 76, 95, 83]
```

```
[6] # Call the function
average, highest, lowest = summarize(scores)

print("Scores:", scores)
print("Average:", average)
print("Highest:", highest)
print("Lowest:", lowest)
```

```
[6] Scores: [88, 92, 76, 95, 83]
Average: 86.8
Highest: 95
Lowest: 76
```

The text cell, which is used for explanations, notes, instructions, or formatting. All content in the text cell will not be executed by Python.

The code cells can execute the Python code and outputs (if any) will be shown below the cell.

Assignment Page

ed

McGill

INSY 336 001 – Ed Lessons

< Lessons

Prev

Next

Test Assignment

★ Challenge

Checkpoints

✓ Solution (hidden)

Edit Slide

Submit

Mock Assignment

Test Assignment

Files + mock.ipynb

+ Code + Text ▶ Run All ■ Stop Save Commands

Python

Outline

Mock Assignment: Python Basics

This is a mock practice assignment.

Q1: Create a function that can get a list and print the sum of values.

Q2: Create a function that can check a number is even or odd

⚠ Comment lines:

Each cell starts with a comment line that indicates the type of the cell:

- `# DO NOT CHANGE THE CODE`: No need to change anything in this cell.
- `### TEST FUNCTION`: Each of these cells contains a function definition.
- `### SKIP`: These cells are for anything other than defining functions, such as testing the implementation, and plotting.

When you see the comment line `# YOUR CODE HERE`, that's where you are supposed to write code.

Do not remove or change the comment lines as it will affect the grading of your assignment.

[0] `### TEST FUNCTION: test_q1`

`# DO NOT REMOVE THE LINE ABOVE`

`# YOUR CODE HERE`

`def q1(numbers):`

[0] `### SKIP`

`q1([1,2,3])`

`q1([4,5])`

[0] `### TEST FUNCTION: test_q2`

`# DO NOT REMOVE THE LINE ABOVE`

`# YOUR CODE HERE`

`def q2(number):`

[0] `### SKIP`

`q2(3)`

`q2(4)`

You can execute your code unlimited times before submitting.

This section contains the background and requirements of the assignment. Please read it carefully before you begin working on it

The Python-related assignments are using Jupyter Notebook. For more information about Jupyter, please check slides before.

As indicated in the line above (`# DO NOT REMOVE THE LINE ABOVE`), please do not delete any line before that.


Please make sure you strictly follow the assignment requirements. Failure to do so may affect your grade.

Click the “Submit” button to submit your assignment.

As indicated in the line (`# YOUR CODE HERE`). You should add the code here.

The `###SKIP` commands usually indicate that the code below is used to help test your program. You do not need to modify it. If it raises errors when you run it, this usually means there is an issue with your own code earlier in the assignment.

You can also use it to check your code. For example, in this case, the expected output is 6 and 9.

 McGill

INO DO

Assignment Result Page

The screenshot displays the 'Assignment Result Page' for a submission titled 'Mock Assignment'. The page is divided into several sections:

- Submissions:** A list on the left shows three submissions. Submission #3 is marked 'FINAL' and '2 minutes ago'. Submission #2 is '6 minutes ago'. Submission #1 is '11 minutes ago'.
- Description:** The main section shows the assignment title 'Mock Assignment' and a description: 'This is a mock assignment (used to show how does it work)'. It indicates 'Automatic Feedback' is enabled and shows a 'Passed' status.
- TESTCASES:** A table showing the results of automated tests. Both 'test_q1' and 'test_q2' are marked as 'passed' with green checkmarks.
- Feedback:** A section for the final mark, showing '2 / 0'. It includes a 'Rubric' section with an 'Edit' button and a 'Comment' section for leaving a comment.
- Code Editor:** On the right, a code editor shows the Python code for the assignment. It includes comments for 'Q1: Create a function that can get a list and print the sum of values.' and 'Q2: Create a function that can check a number is even or odd'. The code defines a function 'q1' that prints the sum of a list.

Arrows from the text boxes point to the 'TESTCASES' and 'Feedback' sections.

After you submit the code, the platform will automatically test it and show whether it meets our expectations. The results are displayed in the TESTCASES area.

In most cases, the TESTCASES results accurately reflect whether your solution is correct. However, in rare cases, the system may incorrectly mark a correct submission as incorrect. The grader will fix this during the grading process, so you don't need to worry too much if you believe your submission is correct, but the system shows an error message.

In the feedback section, the TA will update your grade after a couple of weeks and may include comments on your submissions.

Datacamp

Introduction of Datacamp

datacamp Home Learn Certification Sandbox **Groups** DataLab

Data Mining for Bu...
Classroom

Dashboard
Members
Teams
Settings

LEARN

Assignments

Assignments / Everyone

ACTIVE PAST DUE ARCHIVED

Create Assignments


Challenge your members to earn XP or complete specific DataCamp content then track their progress.

Access denied

Click this to access the active assignments.

The active assignment will show here.

Assignment (Example)

Home**Learn**CertificationSandboxGroupsDataLab

Progress

My library

Leaderboard

Assignments

LEARN

Tracks

Courses

Practice

Assessments

Tutorials

APPLY

Real World Projects

Code Alongs

Competitions

Popular Topics NEW

Getting Started (1/4)

INTERACTIVE COURSE

Introduction to Python

StartBookmark...

Basic

4 hr

11 videos

46 Exercises

6,419,551 participants

3900 XP

Updated: Jul 2025

Description

Python is a general-purpose programming language that is becoming ever more popular for data science. Companies worldwide are using Python to harvest insights from their data and gain a competitive edge. Unlike other Python tutorials, this course focuses on Python specifically for data science. In our Introduction to Python course, you'll learn about powerful ways to store and manipulate data, and helpful data science tools to begin conducting your own analyses. Start DataCamp's online Python

[Read More](#)

1 Python Basics

An introduction to the basic concepts of Python. Learn how to use Python interactively and by using a script. Create your first variables and acquaint yourself with Python's basic data types.

[Hide Chapter Details](#)

Hello Python!

50 XP

Your first Python code

100 XP

Python as a calculator

100 XP

Variables and Types

50 XP

Variable Assignment

100 XP

Start Chapter

SHARE


[Link](#)[LinkedIn](#)[Facebook](#)[Twitter](#)

PREREQUISITES

There are no prerequisites

PART OF THESE TRACKS


[Associate Data Scientist](#)[Data Analyst](#)[Python Data Fundamentals](#)



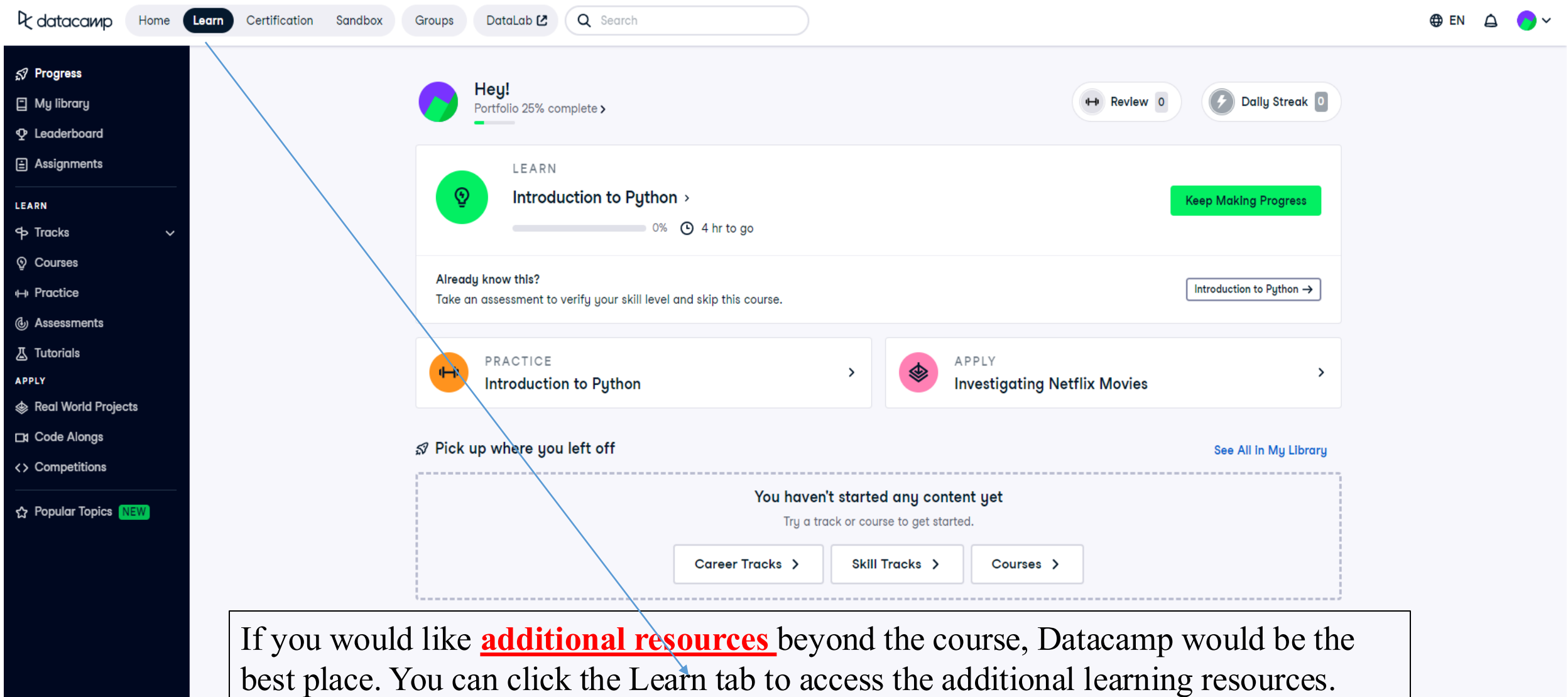
Hugo Bowne-Anderson

Data Scientist

Hugo is a data scientist,

 McGill

Additional Resources



The screenshot shows the DataCamp website's 'Learn' section. At the top, the navigation bar includes 'Home', 'Learn' (highlighted), 'Certification', 'Sandbox', 'Groups', and 'DataLab'. A search bar is also present. The left sidebar lists various navigation options: 'Progress', 'My library', 'Leaderboard', 'Assignments', 'LEARN' (with a dropdown arrow), 'Tracks', 'Courses', 'Practice', 'Assessments', 'Tutorials', 'APPLY', 'Real World Projects', 'Code Alongs', 'Competitions', and 'Popular Topics' (marked as 'NEW'). The main content area features a user profile 'Hey!' with a 'Portfolio 25% complete' indicator. It displays a 'LEARN' card for 'Introduction to Python' with a progress bar at 0% and a '4 hr to go' estimate. Below this is a 'PRACTICE' card for 'Introduction to Python' and an 'APPLY' card for 'Investigating Netflix Movies'. A section titled 'Pick up where you left off' contains a message: 'You haven't started any content yet. Try a track or course to get started.' with buttons for 'Career Tracks', 'Skill Tracks', and 'Courses'. A blue arrow originates from the 'Learn' tab in the top navigation bar and points to a text box at the bottom of the page.

If you would like **additional resources** beyond the course, Datacamp would be the best place. You can click the Learn tab to access the additional learning resources.


Additional Resources

The screenshot displays the DataCamp website's 'Learn' tab. The top navigation bar includes links for Home, Learn (active), Certification, Sandbox, Groups, and DataLab, along with a search bar. The left sidebar lists navigation options: Progress, My library, Leaderboard, Assignments, and a 'LEARN' section with sub-items: Tracks, Courses, Practice (highlighted with a blue arrow), Assessments, Tutorials, and an 'APPLY' section with Real World Projects, Code Alongs, and Competitions. The main content area shows a user profile 'Hey!' with a 25% complete portfolio. Below this, there are three main sections: 'LEARN' featuring 'Introduction to Python' (0% progress, 4 hr to go) with a 'Keep Making Progress' button; 'PRACTICE' featuring 'Introduction to Python'; and 'APPLY' featuring 'Investigating Netflix Movies'. A 'Pick up where you left off' section contains a message 'You haven't started any content yet' and buttons for 'Career Tracks', 'Skill Tracks', and 'Courses'. A blue arrow points from the 'Practice' link in the sidebar to the 'Practice' section. Another blue arrow points from the 'Assessments' link in the sidebar to the 'Assessments' section. A text box at the bottom states: 'Inside of the learn tab, you can find the learning resources in both “Practice” or “Assessments”’.

Additional Resources (Practice)







Practice ↔ Reinforce what you're learning

Keep your skills sharp with quick daily challenges on desktop or Mobile app. You earn XP for every practice round.



All **Python** SQL R Power BI Tableau Excel AWS Azure Docker Julia Theory

37 Practice sessions Filter Practice

<div>PRACTICE</div> <div>Data Manipulation with pandas</div> <div> Python Start</div>	<div>PRACTICE</div> <div>Introduction to Data Visualization with Seaborn</div> <div> Python Start</div>	<div>PRACTICE</div> <div>Introduction to Functions in Python</div> <div> Python Start</div>
<div>PRACTICE</div> <div>Python Toolbox</div> <div> Python Start</div>	<div>PRACTICE</div> <div>Introduction to NumPy</div> <div> Python Start</div>	<div>PRACTICE</div> <div>Introduction to Python</div> <div> Python Start</div>
<div>PRACTICE</div> <div>Intermediate Data Visualization with Seaborn</div> <div></div>	<div>PRACTICE</div> <div>Cleaning Data in Python</div> <div></div>	<div>PRACTICE</div> <div>Statistical Thinking in Python (Part 1)</div> <div></div>

In the Practice page, you can select the topic you are most interested in.

Please make sure you select the correct programming language: Python or SQL.

Thank you!
Any Questions?

