

Introduction to Programming with Python

Module 05 - Advanced Collections and Error Handling

Overview

In this assignment, you learn about additional programming tools and techniques. Course assignments help you learn through reading, watching demonstrations, performing programming in Python, and reflecting on what you learned through writing.

This assignment includes the following tasks:

1. Read module text.
2. Watch the module videos.
3. Create a program
4. Document your knowledge.
5. Submit your work.

Tip: Consider the following questions while you work through the module to help you focus:

- What is the difference between a List and a Dictionary?
- What is the between an Index and a Key?
- How do you write data from a file into a Dictionary?
- How do you read data from a file into a Dictionary?
- What is a JavaScript Object Notation (JSON) file?
- What does Python's json module do?
- What is Structured Error Handling?
- Why is error handling using Try-Except recommend?
- What are two common locations for storing and sharing code files?
- What is GitHub, and why is it used?
- How does PyCharm work with GitHub?

Task 1: Read and Watch the Module Text and Videos.

Start the assignment by reading the module's text and watching the module's demonstration video. You will find both the text in the Mod-05 document and the link to the demo and lab videos on the Canvas Module page.

Tip: You do not need to watch the demonstration video's if their content was shown in the live sessions

Task 2: Watch the assignment videos.

Please watch the following video, in addition to the videos and demonstrations you watched in the module.

- [Python Tutorial for Beginners 5: Dictionaries - Working with Key-Value Pairs](#) (external site)
- [What Is JSON | Explained](#) (external site)
- [Exceptions in Python - Python Tutorial](#) (external site)
- [GitHub Tutorial - Beginner's Training Guide](#) (external site)

Task 3: Read about module topics

Please read the following articles, in addition to the text you read in the module.

1. [What Is A Dictionary In Python?](#) (external site)
2. [Python Exceptions: An Introduction](#) (external site)
3. [On-premise vs Cloud-based File Sharing: Pros and Cons](#) (external site)

Task 4: Create a program

Create a Python program that demonstrates using constants, variables, and print statements to display a message about a student's registration for a Python course. This program is very similar to Assignment04, but **It adds the use of data processing using dictionaries and exception handling.**

Note: Start by opening and reviewing the starter file `Assignment05-Starter.py`!

Acceptance Criteria

Your program must include the following features and code to be accepted as complete:

File Name:

- The file is named `Assignment05.py`

Script Header:

- The script header includes this text and has been updated with your name and the current date.

Constants:

- The constant **MENU: str** is set to the value:

```
---- Course Registration Program ----
Select from the following menu:
1. Register a Student for a Course
2. Show current data
3. Save data to a file
4. Exit the program
-----
```

- The constant **FILE_NAME: str** is set to the value "Enrollments.json"
- Constants values do not change throughout the program.

Variables:

- **student_first_name: str** is set to empty string.
- **student_last_name: str** is set to empty string.
- **course_name: str** is set to empty string.
- **json_data: str** is set to empty string.
- **file** is set to None.
- **menu_choice: str** is set to empty string.
- **student_data: dict** is set to and empty list
- **students: list** : list is set to and empty list

Input / Output:

- On menu choice 1, the program prompts the user to enter the student's first name and last name, followed by the course name, using the input() function and stores the inputs in the respective variables.
- On menu choice 2, the presents a string by formatting the collected data using the print() function.
- Data collected for menu choice 1 is added to a two-dimensional list table (list of dictionaries).
- All data in the list is displayed when menu choice 2 is used.

Processing

- When the program starts, the contents of the "Enrollments.json" are automatically read into a two-dimensional list table (a list of dictionary rows). (**Tip:** Make sure to put some starting data into the file or you will get an error!)
- On menu choice 3, the program opens a file named "Enrollments.json" in write mode using the open() function. It writes the content of the students variable to the file using the dump() function, then file is closed using the close() method. Then displays what was stored in the file.
- On menu choice 4, the program ends.

Error Handling

- The program provides structured error handling when the file is read into the list of dictionary rows.
- The program provides structured error handling when the user enters a first name.
- The program provides structured error handling when the user enters a last name.
- The program provides structured error handling when the dictionary rows are written to the file.

Test:

- The program takes the user's input for a student's first, last name, and course name.
- The program displays the user's input for a student's first, last name, and course name.
- The program saves the user's input for a student's first, last name, and course name to a Json file. (check this in a simple text editor like notepad.)
- The program allows users to enter multiple registrations (first name, last name, course name).
- The program allows users to display multiple registrations (first name, last name, course name).
- The program allows users to save multiple registrations to a file (first name, last name, course name).
- The program runs correctly in both **PyCharm** and from the **console or terminal**.

Source Control:

- The script file and the knowledge document are hosted on a GitHub repository.
- A link to the repository is included in the knowledge document.
- A link to the repository is included in the GitHub links forum.

Task 5: Document your knowledge

After you have created and tested your Python program, create a document **describing the steps you took in performing this assignment**.

- All resources for this assignment are found in the lectures, recommended reading, or recordings specified in the class syllabus. You do not need to locate additional resources outside of these.
- **Your document must conform to my professional document template to get full points!** Use my example template and video, in the General Files and Topics module, as a guide for what I expect a professional paper to look like. Make sure you format it like a college paper instead of a text message. Things like your name, date, class, citations, introductory and summary paragraphs are always expected! So, **not putting these in the document will cost you!** ([See this video for help!](#))
- Please save your files as a PDF file called **Assignment05_YourNameHere.pdf**. Here is an example from Google Docs, but it is similar in most text editors!

Important: Make sure to include an introduction and summary, as well as a heading. It does not have to be perfect, but you won't get credit for it if you turn in a simple blob of text! **watch this video** to help you understand what I am looking for: [Creating Professional Documents](#) and **see the examples** from other students posted in the **"General Information and Helpful Tips"** module in Canvas.

Task 6: Post your Files to GitHub

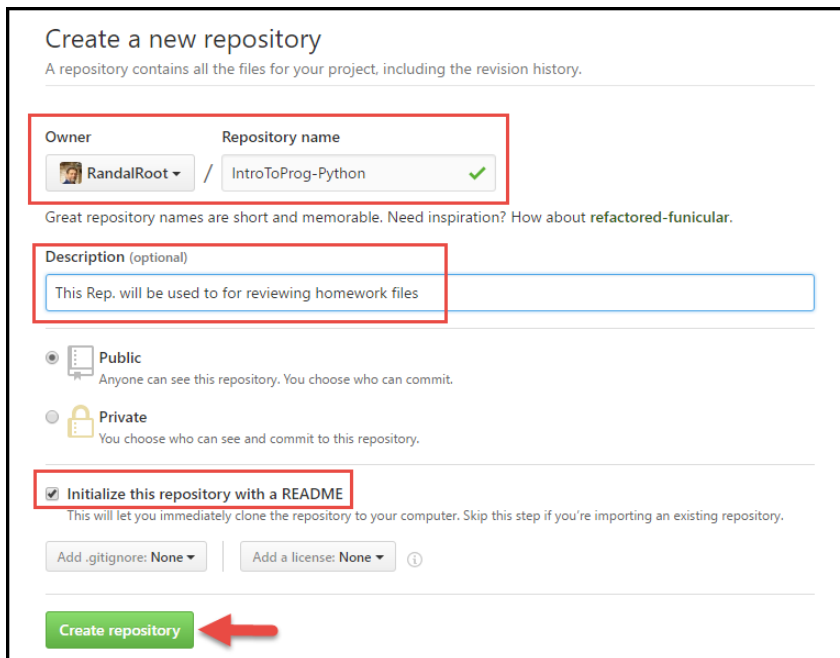
In this module, you need to **post** your files on a public **GitHub repository** so that others may review it. Please post **both your knowledge document and your Python script file**.

Once, you understand how it works, perform the following to **create** a repository for your code:

a. **Login** to <https://github.com> (Make a new account if needed!)

Important: GitHub requires an email account for your login. You may use a **new made-up email account** with a made-up name **if you are concerned about security**. If you need more safeguards than that, please talk to your instructor.

b. **Create** a repository called "**IntroToProg-Python-Mod05**" under your account. **Figure 1 shows the steps.**



Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: RandalRoot / Repository name: IntroToProg-Python ✓

Great repository names are short and memorable. Need inspiration? How about **refactored-funicular**.

Description (optional): This Rep. will be used to for reviewing homework files

☒ Public
Anyone can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

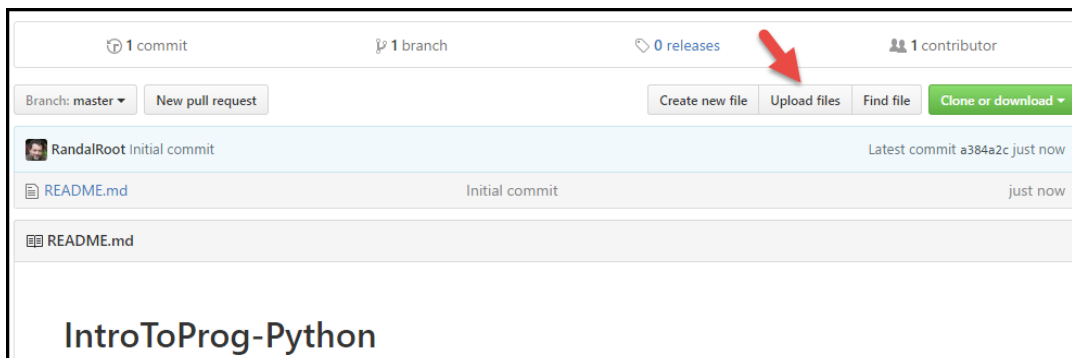
☒ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None ⓘ

Create repository

Figure 1. Creating a GitHub repository

c. Upload both of your files to the repository.



1 commit | 1 branch | 0 releases | 1 contributor

Branch: master | New pull request

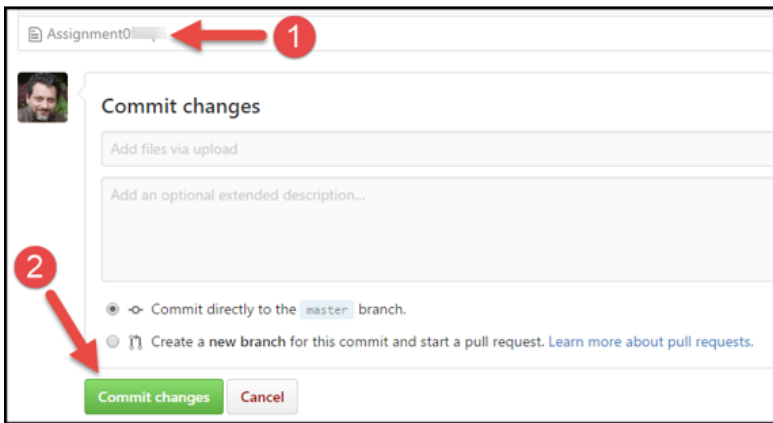
Create new file | Upload files | Find file | Clone or download

Author	Commit	Message	Time
RandalRoot	Initial commit	Initial commit	just now

README.md

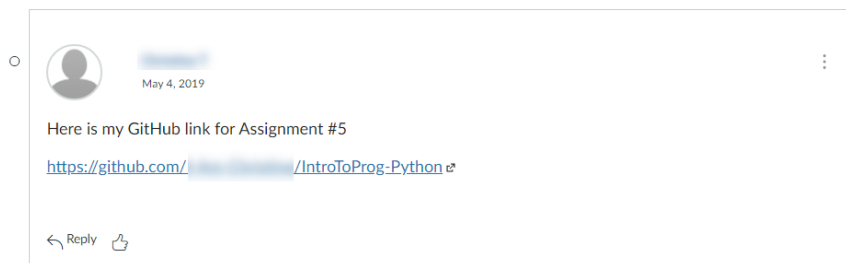
IntroToProg-Python

d. Commit the changes to save your work.



Task 7: Post a Link to GitHub

You will share your work using the Canvas discussion board. To do so, you must create a post with a link to your GitHub site. Other students will use this link to perform a peer review.



Important:

- Post only on the special discussion board called "Assignment 05 Documents for Review!"
- Please copy and paste the URL for your new GitHub site into your MS Word knowledge document (Figure 2). This makes grading a lot easier and is a big help! Thanks!

<Name>
<Date>
<Course>
<Assignment>
<GitHubURL>

Title

Introduction

Lorem ipsum is a made-up language. It is used as a placeholder for a

Task 8: Submit your work

Now place your document with the Python script into a folder named A05, then compress the folder into a ".zip" file, before finally uploading the file to the class assignment page on Canvas.



Notes:

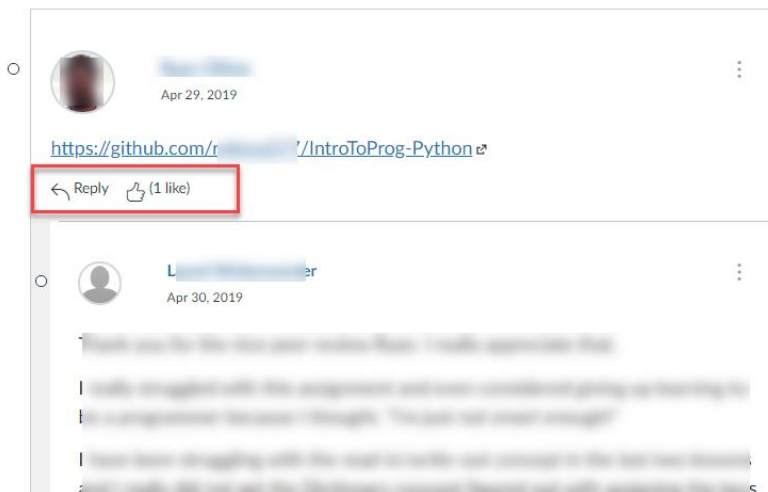
- Use the discussion board to request help on the assignment.
- The assignment can be completed using the lectures, assignment videos and reading, and module labs. You do not need to locate additional resources outside of the course material to complete it.
- If you are unfamiliar with Lorem Ipsum please see this link [What is Lorem Ipsum](#) (external link)
- Please read this article if you are unsure how to zip a folder [How to Make a Zip File](#) (external link)
- See the "General Information and Helpful Tips" module in Canvas for more help!

Step 9 - Perform a Peer Review (Not Graded!)

After you have posted your link to GitHub and submitted your assignment, go to the "Assignment 05 Documents for Review!" discussion board and **select another student's post and review**. Follow the link they posted and review their files on GitHub. **This is an informal review that does not affect either your or their grade. Try to pick someone's link that has NOT been reviewed yet, even if you have to wait a few days for one to appear!**

NOTES

- **Post** your comments as a reply to their posting so the review will be nested under the other student's posting.
- **Make sure** to say two things that you liked about their work
- **Make sure** to say one thing that could make the work better



Congratulations! You are done!