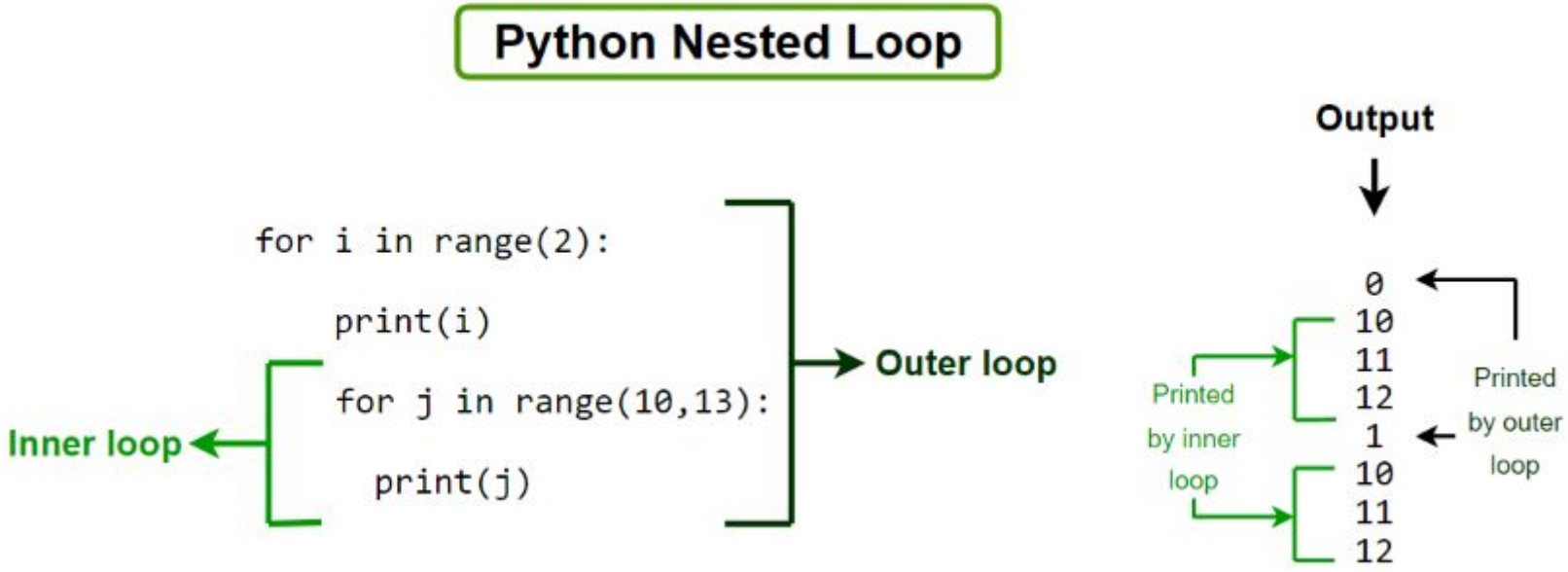


Psych 138: Section 3

Ryan Pili
Fall 2023
Office Hours: Tues., 2:30 - 4 PM
rpili@ucsc.edu

Agenda

1. Section Assignments Submission Instructions
2. Work on Section Assignment 3
    a. Importing Modules
    b. Misc. Style and Grading Expectations
        i. No extra parentheses
        ii. No camel case, underscores only
        iii. No error handling

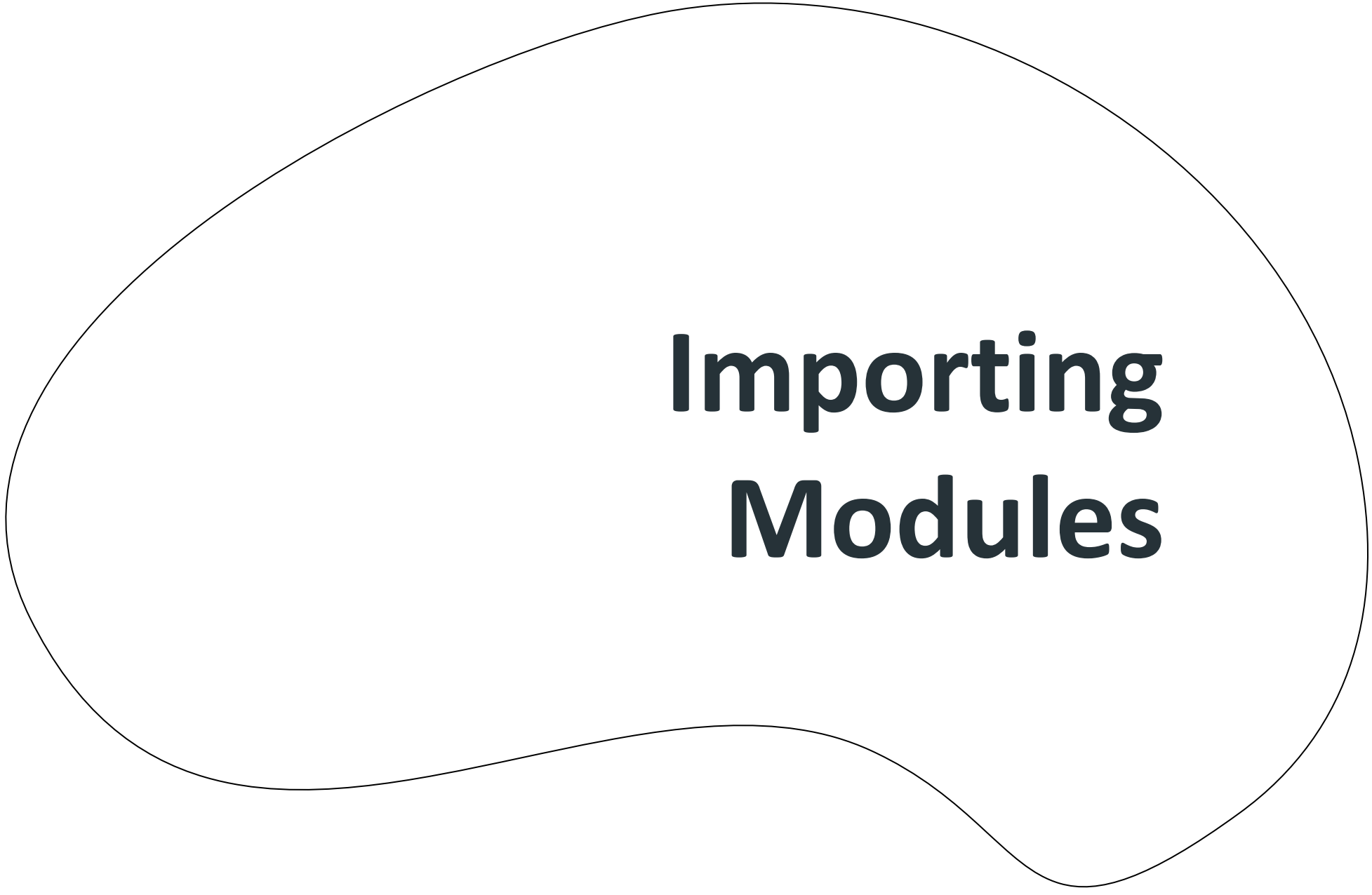
Goals

1. Clarify how to submit Section Assignments.
2. Complete Section 3 Assignment by the end of section.
3. Clarify importing modules

Section Assignment Submission Instructions

Section Assignment Submission Instructions

- New expectations for submitting section assignments to clarify “submissions” of final versions.
- Steps:
  - 1. Name the Google Colab Notebook: “Section2_Assignment_NameOne_NameTwo_Final.ipynb”
    - Replace the Section2 to the correct section.
    - Put all names in pair into the title using FirstnameLastname format.
    - Do not keep the “Copy of” in the title.
  - 2. Once you have a version that you want to “submit”, pin and name the current revision, “Final_Submission”
- For example, “Section3_Assignment_RyanPili_TravisSeymour_Final.ipynb”
- I will only grade the most recent revision/version that is named “Final_Submission”, and that was saved before the assignment deadline.
- If you do not follow these steps, your assignment is not submitted and will receive a 0.


Importing
Modules

Importing Modules

- Python modules contain functions, classes, variables, etc.
- Importing a module allows you to access those contained functions and use them later on in your code.
- Typically, modules are imported at the top of your code.
- Emulate this by creating a code cell at the top of your notebooks that only contains imports.
  - This is preferred over adding import in the middle of your code.
- Once you run the code cell with all your imports, you can use functions from those modules in other code cells.

```
1 import math  # put imports of modules at the very top.
2 import random
3
4 number = 123456
5 print(math.sqrt(number))  # can then use functions from imported modules later on
```

Misc. Style and Grading Expectations

Misc. Style Expectations

- 1. Do not use unnecessary parentheses.
  - Do use parentheses when they are necessary for the desired functionality, or when they improve readability, or as needed when writing equations.
- 2. Do not use camelcase when naming variables.
  - Do use underscores, “_” as needed when naming variables.
- Use of these could be reflected in your grades.

```
1 x = "something"
2
3 # the parentheses around the expression do not change how the code runs, its just harder to read.
4 is_x_a_digit = (x.isdigit())
5 length_of_x = (len(x))
6
7 # naming variables
8 DontUseCamelCase = "yes"  # don't use camelcase for naming variables in this class.
9 dont_use_camel_case = "yes"  # instead use underscores "_" as needed.
```

Error Handling

- Do not use exception handling (try-except-else) at this time.
  - Use of this could be reflected in your grades.

Section 3 Assignment

Due by the end of Section!

Make sure to follow submission rules!

Closing

- Email me if you have any questions.
- Have a great weekend!

3	5	12-Oct	Thur	Descriptive statistics using the While Loop	Ch2: Vid6
				Section Assignment; Weekly Assignment due Tuesday	
4	6	17-Oct	Tues	Coding a 2-Back Task using For Loops	Ch2: Vid7
4		19-Oct	Thur	Managing data using Lists.	Ch4: Vid13
				Section Assignment; ~~Weekly Assignment due Tuesday~~	
5	7	24-Oct	Tues	MIDTERM EXAM 1	
5	8	26-Oct	Thur	Coding a Categorization Task Experiment	Ch4: Vid14