

IT 0075 - PYTHON FOR MANY USES

Catalog Description

Advisory: Completion of IT 105 with grade of "C" or better
Hours: 72 (54 lecture, 18 laboratory)

Description: Learn to use Python to manipulate data for analytics, to manage IoT devices and/or for creating scripts to improve network security. This course will start at a very basic level and work up to applied solutions for real world work environments. (CSU)

Course Student Learning Outcomes

- CSLO #1: Solve business problems using Python programming concepts and libraries.
- CSLO #2: Design and produce Python programming solutions incorporating current trends, security, and best practices.
- CSLO #3: Employ Python programming concepts and terminology.
- CSLO #4: Demonstrate marketable Python programming career skills.

Effective Term

Fall 2023

Course Type

Credit - Degree-applicable

Contact Hours

72

Outside of Class Hours

90

Total Student Learning Hours

162

Course Objectives

Lecture Objectives:

1. Explain why Python is a useful scripting language.
2. Describe the basics of Python programming.
3. Explore the use of variable declarations, control structures, loop constructs and modules in Python.
4. Explain how to use lists, tuples, and dictionaries in Python programs.
5. Learn how to use indexing and slicing to access data in Python programs.
6. Define the structure and components of a Python program.
7. Explore writing loops and decision statements in Python.
8. Provide examples of functions and practice writing functions and pass arguments in Python.
9. Demonstrate how to build and package Python modules for reusability.
10. Demonstrate how to read and write files in Python.
11. Learn how to read data from a file.
12. Gain knowledge of Python visualization libraries.

Lab Objectives:

1. Automate tasks on the computer by writing simple Python programs.

2. Write programs that can do text pattern recognition with "regular expressions."
3. Programmatically generate and update Excel spreadsheets.
4. Parse PDFs and Word documents.
5. Crawl websites and pull information from online sources and create plot of retrieved data.
6. Write programs that send out email notifications.
7. Use Python's debugging tools to quickly figure out bugs in your code.
8. Programmatically control the mouse and keyboard to click and type for you.
9. Write simple TCP/IP networking client and server scripts and programs.
10. Write scripts to simplify system administration tasks.
11. Write scripts for cybersecurity and penetration testing.

General Education Information

- Approved College Associate Degree GE Applicability
- CSU GE Applicability (Recommended-requires CSU approval)
- Cal-GETC Applicability (Recommended - Requires External Approval)
- IGETC Applicability (Recommended-requires CSU/UC approval)

Articulation Information

- CSU Transferable

Methods of Evaluation

- Objective Examinations
 - Example: Instructor will prepare multiple choice questions, true/false and fill-in question sets that will be used to determine the level of basic understanding of concepts covered. Example: Given the following functions, determine which statements are valid (true) and invalid (false).
`def square_root(x): return math.sqrt(x)`
`def print_val(x): print(x)`
 1. `y = square_root(49.0)` True or False?
 2. `y = square_root(49.0) = z` True or False?
 3. `y = 1.0 + square_root(144.0)` True or False?
 4. `y = square_root(square_root(16.0))` True or False?
- Problem Solving Examinations
 - Example: Students will be given scenario based question describing a problem and the parameters involved and asked to determine the proper course of action that needs to be taken to correct the problem. Example: Assign `max_sum` with the greater of `num_a` and `num_b`, PLUS the greater of `num_y` and `num_z`. Use just one statement. Hint: Call `find_max()` twice in an expression. Sample output with inputs: 5.0 10.0 3.0 7.0 `max_sum` is: 17.0
- Projects
 - Example: Student shall provide an in-class or video presentation, explaining a key feature of Python programming and an example of code they wrote to solve a business problem. Instructor shall provide students with a rubric identifying grading criteria for the presentation.
- Skill Demonstrations
 - Example: Students are required to complete a variety of hands-on labs relevant to the topics being covered and submit for grading. Example: Define a function `calc_pyramid_volume()` with parameters `base_length`, `base_width`, and `pyramid_height`, that returns the volume of a pyramid with a rectangular base. `calc_pyramid_volume()` calls the given `calc_base_area()` function in the calculation. Relevant geometry equations: Volume = base

area x height x 1/3 (Watch out for integer division). Sample output with inputs: 4.5 2.1 3.0 Volume for 4.5, 2.1, 3.0 is: 9.45

Repeatable

No

Methods of Instruction

- Laboratory
- Lecture/Discussion
- Distance Learning

Lab:

1. Instructor will guide students through writing a program that parses a paragraph into individual strings and stores them in a list (Lab Objective 2).
2. Instructor will guide students through developing code that sends out an email based on a particular network activity (Lab Objective 6).

Lecture:

1. After receiving instruction on introductory Python syntax, students will enter a variety of expressions into the shell (Lecture Objective 2).
2. The instructor will discuss the importance of functions and the students will create a small program that is compartmentalized into logical groups (Lecture Objective 6).

Distance Learning

1. The LMS can be used to initiate discussion between the instructor and students, as well as, student to student similar to those that would take place in an on-ground course.
2. Students will share individual examples of Python code and the instructor and other students will comment on one another's work, with a focus on programming for reusability (Lecture Objective 9).
3. After the instructor provides instruction on GitHub, students will collaborate through an online repository, such as GitHub, and work on a collective program to parse PDFs and Word documents (Lab Objective 4).

Typical Out of Class Assignments

Reading Assignments

1. Read the chapter on loops from the text and be prepared for classroom/online discussion.
2. Students will be required to do web research on course related topics and report their findings.

Writing, Problem Solving or Performance

1. Following the reading and discussion of the chapter, students will do a written compare/contrast loops and decision statements in Python to show their level of comprehension.
2. Instructor will post a discussion topic on regular expressions and have students explain it in their own words, then also review and reply to posts from fellow students.

Other (Term projects, research papers, portfolios, etc.)

1. Students will write a Python program used for Data Analysis, IoT or Cyber Security and submit for grading.

Required Materials

- Programming in Python 3
 - Author: Bailey Miller
 - Publisher: zyBooks
 - Publication Date: 2020
 - Text Edition: 1st
 - Classic Textbook?: No
 - OER Link:
 - OER:
- Automate the Boring Stuff with Python
 - Author: Al Sweigart
 - Publisher: No Starch Press
 - Publication Date: 2015
 - Text Edition: 1st
 - Classic Textbook?: No
 - OER Link:
 - OER:

Other materials and-or supplies required of students that contribute to the cost of the course.