SAN DIEGO COMMUNITY COLLEGE DISTRICT
COLLEGE OF CONTINUING EDUCATION
COURSE OUTLINE

SECTION I

SUBJECT AREA AND COURSE NUMBER

COMP 660

COURSE TITLE

PROGRAMMING WITH PYTHON I

TYPE COURSE

NON-FEE   VOCATIONAL

CATALOG COURSE DESCRIPTION

This course is an introduction to the Python programming language for students without prior programming experience. The course will cover the spectrum of introductory Python programming topics ranging from the essentials of the language to control statements, functions and modules, working with strings, numbers, and dates and times. Examples and labs used in this course are drawn from diverse areas such as financial data processing, gaming applications and more. Students will be able to use this knowledge to pursue further course of study towards a career in the IT or business industry. (FT)

LECTURE/LABORATORY HOURS

126

ADVISORIES

NONE

RECOMMENDED SKILL LEVEL

- Possess a 12th grade reading level
- Ability to communicate effectively in the English language
- Knowledge of math concepts at the 8th grade level and computer literacy

INSTITUTIONAL STUDENT LEARNING OUTCOMES

1. Social Responsibility
   SDCCE students demonstrate interpersonal skills by learning and working cooperatively in a diverse environment.
2. Effective Communication
   SDCCE students demonstrate effective communication skills.

3. Critical Thinking
    SDCCE students critically process information, make decisions, and solve problems independently or cooperatively.
4. Personal and Professional Development
    SDCCE students pursue short term and life-long learning goals, mastering necessary skills and using resource management and self-advocacy skills to cope with changing situations in their lives.
5. Diversity, Equity, Inclusion, Anti-racism and Access
    SDCCE students critically and ethically engage with local and global issues using principles of equity, civility, and compassion as they apply their knowledge and skills: exhibiting awareness, appreciation, respect, and advocacy for diverse individuals, groups, and cultures.

COURSE GOALS

1. Introduce the Python programming language and the development environment.
2. Gain basic skills for working with Python.
3. Learn to identify the various control statements.
4. Learn to work with functions and modules.
5. Gain an understanding of the role of strings, numbers, dates and times.

COURSE OBJECTIVES

Upon successful completion of the course, the student will be able to:

1. Install and configure Python and its development environment.
2. Demonstrate a basic understanding of the essentials for working with Python.
3. Use control statements to handle iterative and conditional logic.
4. Componentize code by utilizing functions and modules.
5. Utilize common data types in various capacities.

SECTION II

COURSE CONTENT AND SCOPE

1. Getting Started with Python
    1.1. Definition
    1.2. Usage
    1.3. Language comparison
    1.4. Installation
    1.5. "Hello World" with IDLE (Integrated Development and Language Environment)
    1.6. Python interpreter
    1.7. Program execution
    1.8. Implementations
        1.8.1. IDE (Integrated Development Environment) options
        1.8.2. Installation
        1.8.3. Configuration
    1.9. Overview of the Hackathon

2.   Python Essentials
    2.1. Helpful functions
    2.2. Conceptual hierarchy
    2.3. Basic syntax
    2.4. Identifiers
    2.5. Operators
    2.6. Data types
    2.7. Data type conversions
    2.8. Documentation sources
3.   Control Statements
    3.1.   Conditionals
    3.2.   if/elif
    3.3.   Nested if
    3.4.   Iteration
    3.5.   while
    3.6.   for
    3.7.   break and continue
    3.8.   Nested loops
4.   Functions and Modules
    4.1. Working with Functions
        4.1.1. Defining and calling
        4.1.2. Arguments: by value vs. by reference
        4.1.3. Arguments: required, keyword, default, variable
        4.1.4. Defining and calling a main() function
        4.1.5. Scope: local vs. global variables
    4.2. Working with Modules
        4.2.1. Creating
        4.2.2. Importing
        4.2.3. Locating
        4.2.4. Documenting
        4.2.5. Standard modules
    4.3. Advanced Python Topics
        4.3.1. Recursion
        4.3.2. Generators
        4.3.3. Anonymous functions: lambda
5.   Strings
    5.1. Basic skills
    5.2. Search and loop
    5.3. Basic methods
    5.4. Methods for working with case
    5.5. Methods for working with spacing
    5.6. Finding and replacing parts of a string
    5.7. Splitting into lists
    5.8. Joining
6.   Numbers
    6.1. Floating point numbers
    6.2. Math module
    6.3. format() method
    6.4. Locale module

      6.5. Decimal module
7.   Dates and Times
      7.1. Usage
      7.2. Creating date, time, and datetime objects
      7.3. Creating datetime objects by parsing strings
      7.4. Formatting
      7.5. Spans of time
      7.6. Getting date and time parts
      7.7. Comparing date/time objects

APPROPRIATE READINGS

Reading topics are related to Python programming and include techniques for working with Python basics, control statements, function, modules, strings, numbers, and dates and times. These may include, but are not limited to, assigned readings from textbooks, supplemental reading assignments, industry-related periodicals or magazines, manuals, online help pages, articles posted on the Internet, and information from web sites, online libraries and databases.

WRITING ASSIGNMENTS

Writing assignments may include, but are not limited to, preparing text for an assigned project, documenting all laboratories and project work including the results of laboratories using essential Python concepts, and completing all written assigned reports, such as a case study that compares an application that utilizes the Python language versus one that uses a more traditional language such as C++ or Java.

OUTSIDE ASSIGNMENTS

Outside assignments may include, but are not limited to, appropriate internet research, reading from assigned textbooks and completing the assignments at the end of each chapter, and studying as needed to perform successfully in class. An assignment might include creating a financial loan application, a temperature conversion application, or a simple game that uses logic to compute a result.

APPROPRIATE ASSIGNMENTS THAT DEMONSTRATE CRITICAL THINKING

Assignments which demonstrate critical thinking may include, but are not limited to, outlining and building a functional Python-based application or game; separating code in a financial application into more manageable components using concepts in the Python language.

EVALUATION

Evaluation that a student has met the course competencies will include multiple measures of performance related to the course objectives. Evaluation methods may include, but are not limited to, performance in a variety of activities and assignments, such as completing a research project individually or in a group, hands-on projects, class participation, written and practical tests, attendance and punctuality.

Upon successful completion of the course, a Certificate of Course Completion will be issued.

Upon successful completion of all courses in the program, a Certificate of Program Completion will be issued.

METHOD OF INSTRUCTION

Methods of instruction may include, but are not limited to lectures, open discussion, hands-on demonstrations, computer-assisted instruction, field trips, and laboratory assignments. Students will also be expected to participate in online class discussion posts, in-class discussions and project reviews.

This course, or sections of this course, may be offered through distance education.

TEXTS AND SUPPLIES

Think Python: How to Think Like a Computer Scientist, Allen Downey, Green Tea Press, current edition
Exploring Data Using Python 3, Charles Severance, independent, current edition

Web Resources:

Udemy: Programming with Python, https://www.udemy.com/programming-with-python/learn/v4/overview
Python Org, https://www.python.org/
Wikibooks: Python Programming, https://en.wikibooks.org/wiki/Python_Programming

Supplies: Journal (composition book), USB Drive or other storage media

PREPARED BY   Zak Ruvalcaba   DATE   June 5, 2019

REVISED BY   Zak Ruvalcaba   DATE   May 3, 2023

Instructors must meet all requirements stated in Policy 5500 (Student Rights, Responsibilities and Administrative Due Process), and the Attendance Policy set forth in the Continuing Education Catalog.

REFERENCES:

San Diego Community College District Policy 5500
California Community Colleges, Title 5, Section 55002
College of Continuing Education Catalog