

Course Syllabus Part I

DSC 510 Introduction to Programming

3 Credit Hours

Course Description

This course introduces the Python programming language as a tool to clean, slice, and build tools to analyze an existing dataset. Basic principles of programming are explored as well as techniques for configuring a computer for data science work.

Course Prerequisites:

Recommend DSC 500

Course Objectives

Students who successfully complete this course should be able to:

1. Demonstrate fundamental principles of programming, including data types, variables, loops, conditional statements, functions, and objects.
 2. Compose Python data wrangling and analysis programs by combining custom developed code with modules from Python's built-in libraries and from modules maintained by third parties.
 3. Create a user-defined, reusable Python module using classes, functions, and professional programming practices.
 4. Demonstrate basic Python and programming literacy by interpreting and commenting on Python code written by others.
 5. Make use of Python community programming best practices by following recommended naming conventions and style considerations.
-

Grading Scale

93 – 100% = A	87 – 89% = B+	77 – 79% = C+	67 – 69% = D+
90 – 92% = A-	83 – 86% = B	73 – 76% = C	63 – 66% = D
	80 – 82% = B-	70 – 72% = C-	60 – 62% = D-
			0 – 59% = F

Topic Outline

- I. Programming Fundamentals
 - A. Algorithmic Design and Complexity
 - B. Dynamic vs. Static Typing
 - C. Weak vs. Strong Typing
- II. Data Types
 - A. Numeric types
 - B. String types
 - C. Dictionaries, lists, tuples, and other collection types
- III. Statements and Syntax
 - A. For and while loops
 - B. If statements
 - C. Iterations and comprehensions
- IV. Functions
 - A. Variable scope
 - B. Passing arguments and options
- V. Modules
 - A. Importing and using external modules
 - B. Creating user-defined modules
 - C. Documenting and testing modules
- VI. Classes
 - A. Creating and using custom classes
 - B. Basics of object oriented design
- VII. Professional Programming
 - A. Use of Version Control
 - B. Code and project documentation
 - C. Checking code for style and other issues
 - D. Incorporating unit tests