



MERCER
COUNTY COMMUNITY COLLEGE

COURSE OUTLINE

Course Number	Course Title	Credits
IST 110	Introduction to Python	3
Hours: Lecture/Lab/Other	Pre-requisite	Implementation Semester & Year
2/2/0	IST 101 or IST 102	FALL 2022

Catalog description:

Designed for students majoring in Computer Information Systems or those with little or no programming background. Python is a widely used interpreted, object-oriented programming language focused on readability and code optimization with a simple, easy to learn syntax. This course is designed for students with basic programming experience in an object-oriented language

General Education Category: Not GenEd

Course coordinator:

Terry Voldase, Associate Professor of Computer Information Systems, 609-570-3481, voldaset@mccc.edu

Required texts & Other materials:

- Python – Electronic Textbook/ Revel Software – Starting Out with Python 4th edition by Tony Gaddis – Pearson Education publisher
- Free Open Source Python Installation (latest 3.x version) – www.python.org/downloads
 - Python IDLE (GUI included with the Python Installation)
- Microsoft Office 2019 – free software provided by MCCC
- PC and Mac computers with software downloads permissions

Course Student Learning Outcomes (SLO):

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

1. Define various concepts of computers such as computer terminology and nomenclature, with respect to computers and programming [**Supports ILGs 4, 11; PLOs 3, 4**]
2. Demonstrate the concepts and logics of Python software programming [**Supports ILGs 2, 4, 11; PLOs 1, 2, 7**]
3. Design algorithms and develop solutions using a formal programming language [**Supports ILGs 2, 4, 11; PLOs 1, 2, 7**]
4. Utilize critical thinking skills to analyze and create programs [**Supports ILGs 2, 4, 11; PLOs 4, 5**]
5. Define variables and constants, implement decisions using *if* statements, writing statements using Boolean expressions, and implement *while* and *for* loops and nested loops [**Supports ILGs 2, 4, 11; PLOs 4, 5, 6**]
6. Create turtle graphics programs using basic shapes and texts and write exception-handling code [**Supports ILGs 2, 4, 11; PLOs 6, 7**]

Course-specific Institutional Learning Goals (ILG):

Institutional Learning Goal 2. Mathematics. Students will use appropriate mathematical and statistical concepts and operations to interpret data and to solve problems.

Institutional Learning Goal 4. Technology. Students will use computer systems or other appropriate forms of technology to achieve educational and personal goals.

Institutional Learning Goal 11. Critical Thinking: Students will use critical thinking skills understand, analyze, or apply information or solve problems.

Program Learning Outcomes for: Computer Information Systems (A.S.) and Information Technology – Cybersecurity Concentration (A.A.S.)

1. Transfer to a four-year college as a junior;
2. Explain, interpret, and develop computer information policies and procedures;
3. Understand business organizations and practices, and the role of information technology in organizations;
4. Develop, describe, understand, and apply network protocols and technology;
5. Design, program, implement, and document a computer application or website to install and implement computer systems;
6. Work effectively individually and in workgroups to install and implement information systems;
7. Communicate in written documents and in oral presentations in technical or business settings.

Units of study in detail – Unit Student Learning Outcomes:

Unit I [Introduction to Computers and Python Programming] [Supports Course SLO #1, 2]

Learning Objectives

The student will be able to:

- Define course objectives and course requirements and collaborate effectively with other classmates and the professor.
- Demonstrate how computers work, how data is stored and manipulated
- Understand why programs are written in high-level languages
- Demonstrate knowledge of programming software.
- Manage Python through the interactive mode, script mode, and the IDLE environment.

Unit II [Unit II Input, Processing, and Output] [Supports Course SLOs #1, 2, 3, 4, 6]

Learning Objectives

The student will be able to:

- Understand the program develop cycle
- Design, write and compile simple programs that read input from the keyboard
- Understand the importance of comments in sections of a program
- Perform mathematical operations, and produce screen output
- Design programs through the introduction of pseudocode and flowcharts
- Design programs through the introduction to the turtle graphics library

Unit III [Unit III Decision Structures and Boolean Logic] [Supports Course SLOs #2, 3, 4, 5, 6]

Learning Objectives

The student will be able to:

- Learn relational operators and Boolean expressions
- Control the flow of a program with decision structures – if, if-else, and if-elif-else
- Implement nested decision structures and logical operators in programs
- Utilize decision structures to test the state of turtle graphics

Unit IV[Unit IV Repetition Structures] [Supports Course SLOs #2, 3, 4, 5, 6]

Learning Objectives

The student will be able to:

- Create repetition structures using the *while* loop and *for* loop
- Learn counters, accumulators, running totals, and sentinels
- Implement techniques for writing input validation loops
- Utilize loops to draw designs with the turtle graphics library

Unit V [Unit V Functions] [Supports Course SLOs #3, 4, 6]

Learning Objectives

The student will be able to:

- Demonstrate the benefits of using functions to modularize programs
- Discuss top-down design approach
- Pass arguments to functions
- Define and call library functions and use modules to organize functions
- Modularize turtle graphics code with functions

Unit VI [Unit VI Files and Exceptions] [Supports Course SLOs #3, 4, 6]

Learning Objectives

The student will be able to:

- Identify sequential file input and output
- Learn to read and write large sets of data
- Store data as fields and records
- Identify exceptions and write exception-handling code

Evaluation of student learning: [Evaluates SLOs #1, 2, 3, 4]

Students' achievement of the course objectives will be evaluated through the use of the following:

- Programming Labs assessing students' problem solving ability and programming skills.
- Quizzes and Exams assessing students' comprehension of programming environments and concepts.
- A term project to assess the students' ability to solve relatively complex problems using formal programming language.

Grade Criteria

Item	Percent
Midterm Exam	15%
Programming Labs	30%
Quizzes	15%
Final Project	15%
Final Exam	25%
Total	100%