

BIRCHWOOD UNIVERSITY

Instructor:**Course Name:** Python Programming**Course Number:** MDS500**Phone:****E-Mail:****Course Description**

In this course, students will develop a comprehensive understanding of Python programming language and its applications in artificial intelligence (AI). They will learn Python syntax, expressions, literals, variables, and comments, as well as data types, operators, decision flow control statements, and data structures. Students will also gain proficiency in file I/O handling, exception handling, and database connectivity using Python. They will explore Python libraries such as NumPy, Pandas, Matplotlib, and Seaborn for data analysis, visualization, and database connectivity tasks. Through hands-on projects and real-world examples, students will develop practical skills in Python programming and apply them to solve real-world problems.

Course Outcomes

1. Students will be able to demonstrate an understanding of Python programming concepts such as data types, operators, decision flow control statements, data structures, functions, file I/O handling, exception handling, and Python libraries such as NumPy, Pandas, Matplotlib, and Seaborn.
2. Students will be able to interpret and explain the features of Python, including its suitability for artificial intelligence (AI) applications, and analyze Python syntax, expressions, literals, variables, and comments.
3. Students will apply Python programming concepts to solve real-world problems by creating Python scripts and programs.
4. Students will analyze and evaluate different Python libraries, their functionalities, and their applications in data analysis, visualization, and database connectivity.
5. Students will be able to synthesize different Python programming concepts to create complex programs, design and implement user-defined functions, and utilize Python libraries such as NumPy, Pandas, Matplotlib, and Seaborn to perform advanced data operations, visualization, and database connectivity tasks.
6. Students will evaluate the effectiveness and efficiency of Python programming concepts and libraries in solving real-world problems and analyze the limitations and advantages of different Python features.

Course Prerequisites

- Basics of any programming language
- Mathematics or Statistics

Textbooks and Reference books

1. Title: Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython
Author: Wes McKinney
ISBN-13: 978-1491957660
Publisher: O'Reilly Media
2. Title: Python Data Science Handbook: Essential Tools for Working with Data
Author: Jake VanderPlas
ISBN-13: 978-1491912058
Publisher: O'Reilly Media

Other Learning Resources

Technical E-journals, E-magazines, web links and video links if any, shall be given in the class and uploaded on the LMS.

Grading Scheme

Assessment type	Weightage
Quizzes	10 %
Assignments	20 %
Class Discussions	5%
Midterm	30 %
Final exam	30 %
Attendance	5 %

Important Note: To pass this course, a student must achieve a minimum of 60% on the total of the grading components.

Course Schedule

Course duration - 10 weeks

Lecture & lab - 8 weeks

Mid-term - Week 6

Final exam (Project demonstration) - Week 10

Standards of Academic Conduct and Student Integrity

- **Attendance**

Students are expected to attend every lecture and lab session. Please refer the Birchwood University attendance policy for more details. Students who fail to meet the attendance criteria shall be withdrawn from the course.

- **Class Conduct**

During lectures and lab sessions, the student is expected to act in a professional and respectful manner towards other students and instructors; students not conducting themselves in such a manner may be asked to leave. In particular, private conversations during lectures or labs are NOT allowed. Anyone doing so will be asked to leave the lecture or lab immediately.

- **Late Homework Policy**

Any homework that is late shall receive a mark of zero (0).

- **Missed work**

All exams and quizzes must be written at the designated time. Missed exams, and assignments will count as zero unless medical reasons supported by a medical doctor's note are provided.

- **Academic Honesty Policy**

All work submitted must be your own. Birchwood University's academic honesty is followed in this course. Copied work will result in an automatic F grade for the course. The student is expected to be familiar with these policies.

- **Plagiarism**

Presenting someone else's work as though it is your own. In an academic community the use of words, ideas, or discoveries of another person without explicit, formal acknowledgement constitutes an act of theft or plagiarism. In order to avoid the charge of plagiarism, students must engage in standard academic practices such as putting quotation marks around words that are not their own, employing the appropriate documentation or citation, and including a formal acknowledgement of the source in the proper format.

Academic Plan

Week	Topics
Week 1	Introduction to Python <ul style="list-style-type: none">• Introduction• Features of Python• Python for AI• IDE, Anaconda distribution installation, Script mode programming
Week 2	Python Syntax <ul style="list-style-type: none">• Expressions, Print statement and Formats• Input Statements, Python Keywords• Literals• Variables and its types• Comments
Week 3	Data types and Operators <ul style="list-style-type: none">• Integers, floating-point numbers, and complex numbers

	<ul style="list-style-type: none"> • Strings • Boolean data type • Arithmetic, Comparison, Logical and Identity operators, Bitwise operators
Week 4	Decision Flow Control Statements <ul style="list-style-type: none"> • If and else statement, Nested loops • Loop Statement: While Loop, do and while loop, for loop statement • Continue, Break and pass Statement, Conditional Statements
Week 5	Python Data Structures <ul style="list-style-type: none"> • Lists and its operations • Tuples and its operations • Sets and its operations • Dictionaries and its operations • Arrays
Week 6	Functions <ul style="list-style-type: none"> • Built-in-functions, library functions • User defined functions • Lambda Functions • Recursive functions
Week 7	File I/O Handling and Exception Handling <ul style="list-style-type: none"> • Files I/O operations, Read / Write Operations, • File Opening modes with keywords • Manipulating files and directories • OS and SYS modules • Types of errors • Error processing • Exception Raising and Handling
Week 8	Python Libraries <ul style="list-style-type: none"> • Numpy: Introduction to NumPy • Creating and Printing Nddarray, Class and Attributes of Nddarray, Basic operatios • Mathematical Functions of NumPy • Pandas: Understanding Data frame, View and Select Data • Missing Values, Data Operations • File read and write operation • Matplotlib and Seaborn: Introduction to Matplotlib library • Line properties, Plots and subplots, Types of Plots • Introduction to Seaborn, Color palettes

Note: The above listing is a broad outline of the topics intended to be covered in the course. As the course progresses, each module will be outlined in detail.

Lab Topics

Programming exercises on:

1. Introduction to Python: Basic Syntax and Printing Output
2. Working with Variables and Data Types in Python
3. Decision Flow Control: Implementing If-Else Statements in Python
4. Looping: Building While and For Loops in Python
5. Python Data Structures: Lists and Tuples Operations
6. Python Data Structures: Sets and Dictionaries Operations
7. Functions in Python: Built-in, User-Defined, and Lambda Functions
8. File I/O Handling: Reading and Writing Files in Python
9. Exception Handling: Handling Errors and Raising Exceptions in Python
10. Python Library: NumPy Basics and Mathematical Functions

Lab Tools

- Python version 3.11