

COURSE TITLE	FINANCIAL ANALYTICS FUNDAMENTALS
COURSE CODE	05MF0101
COURSE CREDITS	4

Objective:

- 1 To understand about Big Data with its characteristic.
- 2 To impart a thorough understanding about Python Programming, Python Variable
- 3 To understand about how to analytics data with conditional statement
- 4 To learn about basic Financial Calculation
- 5 To learn techniques for representation of the data in the real world

Course Outcomes: After completion of this course, student will be able to:

- 1 Students will proficiently develop programming skills in Python and learn about various data structures relevant for storing and processing financial big data.
- 2 Students will be introduced to the fundamentals of Python programming language, including its syntax and basic structure.
- 3 Students will learn various operations and manipulations on NumPy arrays, including sorting and arithmetic operations.
- 4 Students will gain hands-on experience in working with financial datasets using Pandas and NumPy, applying their knowledge to real-world scenarios.
- 5 Students will proficiently develop programming skills in Python, focusing on various data structures essential for storing and processing financial big data. They will gain hands-on experience with NumPy and Pandas for real-world financial data analysis, mastering operations, manipulations, and visualizations to effectively communicate insights.

Pre-requisite of course:Basic Knowledge about Finance

Teaching and Examination Scheme

Theory Hours	Tutorial Hours	Practical Hours	ESE	IA	CSE	Viva	Term Work
2	0	4	50	30	20	0	50

Contents : Unit	Topics	Contact Hours
1	Financial Big Data & Fundamentals Of Python Introduction To the Big Data, Why and Where Characteristics of Big., Dimensions Of Big Data, Scalability, Explorations, Management-, Hadoop Ecosystem, IT Programming Python and Various Data Structure for Storage., Big Data system Programming, Explaining the Various Analytical Programming., Introduction To the Python Programming, Python Variable with examples., Data Types Introduction with examples, Python Operators Introduction, , Arithmetic operator With Example, Python Logical Operator With Examples, Python Some Extra Operator Discuss With Examples	15
2	Fundamentals of Python Analytics & Financial Data Calculations Projects Using Python Python Conditional Statement, Introduction, Python Conditional Statement Explain If, If Else, Elif., Introduction to The Python Loops For, While, Practical Examples on Loop, Introduction To the Function, Practice Loops & Function with practical example, Introduction To the NumPy Libraries, converting a list to the Numpy array with practical example, converting any data types in Numpy array, Numpy Array introduction, How to get Numpy fill with on specific value, Finding Number of Elements in NumPy, Sort NumPy Array with Example, Compare NumPy Array to The Value, Performing Some Arithmetic Operations., Introduction to the pandas, Basic Examples on Pandas, Various Financial Calculations Methods, Working with Financial Data with NumPy	15
Total Hours		30

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
1	Unit 1 Demonstrate the concept of processing financial big data., Write a Python script to calculate the area of a circle given its radius., Create a program that asks the user for their name and age, then prints out a message with their name and age., Write a Python script to convert temperatures from Celsius to Fahrenheit and vice versa., Create a program that takes two numbers as input and performs arithmetic operations (addition, subtraction, multiplication, division)., Write a Python script to check if a number is even or odd., Create a program that prints the Fibonacci sequence up to a specified number., Write a Python function to calculate the factorial of a number., Create a program that defines a function to check if a given year is a leap year., Write Python code to create NumPy arrays from lists and perform basic operations (e.g., addition, subtraction) on them.	30

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
2	Unit 2 Create a program that generates random numbers and stores them in a NumPy array., Build a predictive model using machine learning algorithms (e.g., linear regression, decision trees) to forecast future stock prices based on historical data., Create a program that fetches crypto currency prices from an API (e.g., CoinGecko), displays real-time price changes, and generates visualizations to analyze trends over time., Build a Python application that retrieves real-time stock data using an API (e.g., Alpha Vantage), Performs analysis on the user's stock portfolio., Develop a Python program that allows users to input their monthly expenses and income, then calculates and displays their total expenditure and savings., Write a program that asks the user for their name and prints it in uppercase., Write a program that calculates the area of a rectangle given its length and width., Create a function that takes a string as input and returns the number of vowels it contains., Create a function that takes two numbers as input and returns the result of raising the first number to the power of the second number., Develop a Python program that allows users to input their monthly expenses and income, then calculates and displays their total expenditure and savings.	30
Total Hours		60

Textbook :

- 1 Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, Wes McKinney, O'Reilly Media, 2017

References:

- 1 Python for Finance: Analyze Big Financial Data, Python for Finance: Analyze Big Financial Data, Yves Hilpisch, O'Reilly Media, 2015
- 2 Big Data in Finance: From Data to Decisions, Big Data in Finance: From Data to Decisions, Thomas W. Miller III, FT Press, 2019

Suggested Theory Distribution:

The suggested theory distribution as per Bloom's taxonomy is as follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process

Distribution of Theory for course delivery					
Remember / Knowledge	Understand	Apply	Analyze	Evaluate	Higher order Thinking / Creative
10.00	20.00	25.00	25.00	10.00	10.00

Instructional Method:

- 1 Board Work

Instructional Method:

- 2 PPT
- 3 Demo

Supplementary Resources:

- 1 <https://www.datacamp.com/tracks/finance-fundamentals-in-python>
- 2 <https://trenton3983.github.io/posts/python-finance/>