

**SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES,
CHITTOOR
(Autonomous)**

Approved by AICTE, New Delhi, Affiliated to JNTUA, Ananthapuramu.



DEPARTMENT OF MANAGEMENT STUDIES

LAB MANUAL

SUBJECT NAME: Introduction to Python

YEAR/BRANCH: II Year - III semester /MBA

REGULATION: R22

PREPARED BY :Dr R Priyadharsini, Associate Professor



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

INSTITUTE VISION AND MISSION

VISION:

To emerge as a Center of Excellence for Learning and Research in the domains of engineering, computing and management.

MISSION:

IM1: Provide congenial academic ambience with state-of-art resources for learning and research.

IM2: Ignite the students to acquire self-reliance in the latest technologies.

IM3: Unleash and encourage the innate potential and creativity of students.

IM4: Inculcate confidence to face and experience new challenges.

IM5: Foster enterprising spirit among students.

IM6: Work collaboratively with Technical Institutes / Universities / Industries of National,
International repute

DEPARTMENT OF MANAGEMENT STUDIES VISION AND MISSION

VISION

Become Center of Excellence for Educating Management Students as Leaders of Tomorrow.

MISSION

M1: Provide congenial academic ambience with necessary infrastructure and learning resources

M2: Inculcate confidence to face and experience new challenges from industry and society

M3: Ignite the students to acquire self-reliance in State-of-the-Art Technologies

M4: Foster Enterprising spirit among students



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Program Educational Objectives (PEOs)

- PEO1:** Have in-depth knowledge through life-long learning to conceptualize, critically analyze and add value in the areas of business management.
- PEO2:** Have lateral thinking enabling simple solutions for complex managerial problems.
- PEO3:** Ignite the passion for entrepreneurship.
- PEO4:** Inculcate a spirit of ethical and social commitment in the personal and professional life and to add value to the society.

PROGRAM OUTCOMES

- PO1:** Apply Knowledge of management theories and practices to solve business problems.
- PO2:** Foster analytical and critical thinking abilities for data - based decision making
- PO3:** Ability to develop value based leadership ability.
- PO4:** Ability to understand, analyze and communicate global, economic, legal and ethical aspects of business.
- PO5:** Ability to lead themselves and others in the achievement of organizational goals, contributing effectively to a team environment.
- PO6:** Apply specialized tools and techniques of human resource management in real life business practices.
- PO7:** Apply the Ever evolving marketing techniques to encounter the challenges and leverage opportunities.
- PO8:** Apply sophisticated financial tools and techniques for smooth management of organization.

.

PROGRAM SPECIFIC OUTCOMES (PSOs):

On successful completion of the program, the post graduates will be able to

- PSO1:** Apply core and functionary management skills for professional growth and business evaluation.
- PSO2:** Adapt to dynamic changes in an environment relevant to professional managerial practice and entrepreneurship as emerging leaders.



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES
[AUTONOMOUS], CHITTOOR - 517 127
DEPARTMENT OF MANAGEMENT STUDIES

II MBA – Semester - III								
Course Code		INTRODUCTION TO PYTHON			L	T	P	C
22MBA238					0	0	2	1
Course Educational Objectives (CEO): CEO1: Enhance the knowledge on basic principles of python. CEO2: Enhance the knowledge on functions and strings in python. CEO3: Acquire the knowledge on data structures in python. CEO4: Enable students to write simple object oriented programming in python. CEO5: Understand the exception handling and modules.								
UNIT - I		Introduction to Python					Lecture Hrs: 6	
Introduction to Python, Python Features, Operators, Variables, Control Statements (conditional, looping, transfer)								
UNIT - II		Functions and Strings					Lecture Hrs 6	
Functions: Function Definition, Function call, Types of Arguments, Lambda Function. Strings: String Handling Functions								
UNIT - III		Data Structures					Lecture Hrs:6	
Lists, Tuples, Sets and Dictionaries								
UNIT - IV		Object Oriented Programming					Lecture Hrs:9	
Object, Define Class, Constructor, Methods in Python, Inheritance, Abstraction, Polymorphism.								
UNIT - V		Exception Handling and Modules					Lecture Hrs:9	
Exception, Syntax errors, Runtime Errors, Module - Math Module, Creating Modules								
Course Outcomes:								
On successful completion of the course the student will be able to,						POs & PSOs related to COs		
CO1	Understanding the knowledge on basic principles of python.					PO2, PSO1		
CO2	Apply the functions and strings in python.					PO2, PSO1		
CO3	Analyze the data structures in python.					PO2, PSO1		
CO4	Apply simple object oriented programming in python.					PO2, PSO1		
CO5	Analyze the date handling and modules.					PO2, PSO1		
Text Books:								
1. Python for Programmers, Paul Deitel and Harvey Deitel, Pearson Education, 1st Edition, 2021.								
Reference Book:								
1. Computational Thinking: A Primer for Programmers and Data Scientists, 1/e, G Venkatesh and Madhavan Mukund, Notion Press, 2021. 2. Introduction to Computation and Programming Using Python: With Applications to Computational Modeling and Understanding Data, 3/e, John V Guttag, & Quot, MIT Press 2021.								



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES
[AUTONOMOUS], CHITTOOR - 517 127
DEPARTMENT OF MANAGEMENT STUDIES

Online Learning Resources:

<https://www.programiz.com/python-programming>
<https://www.youtube.com/watch?v=adNgan70iyU>
<https://www.youtube.com/watch?v=c235EsGFcZs>

COURSE OUTCOMES VS POs MAPPING (DETAILED; HIGH:3; MEDIUM:2; LOW:1):

Course	PO	PO	PO	PO	PO	PO	PO	PO	PO	PSO1	PSO2
	CO	1	2	3	4	5	6	7	8		
C2308 : INTRODUCTION TO PYTHON	C2308.1	-	3	-	-	-	-	-	-	3	-
	C2308.2	-	3	-	-	-	-	-	-	3	-
	C2308.3	-	3	-	-	-	-	-	-	3	-
	C2308.4	-	3	-	-	-	-	-	-	3	-
	C2308.5	-	3	-	-	-	-	-	-	3	-
	C2308	-	3	-	-	-	-	-	-	3	-



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

List of experiments

S.No	Experiment No	Experiment details
1	Experiment 01	Arithmetic Operators Using Direct Input Values
2	Experiment 02	Arithmetic Operators Using User Input
3	Experiment 03	Grade Classification using conditional statements
4	Experiment 04	Highest Sales Value using conditional statements
5	Experiment 05	Odd or Even Number
6	Experiment 06	While Loop - Print Odd Numbers
7	Experiment 07	Area of Rectangle using Functions
8	Experiment 08	Discount Calculation using Functions
9	Experiment 09	Total Cost Calculation using Functions
10	Experiment 10	Lists and Tuples
11	Experiment 11	String Handling
12	Experiment 12	Dictionary
13	Experiment 13	Class and Object Creation
14	Experiment 14	Inheritance and Polymorphism
15	Experiment 15	Exception Handling



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

TABLE 1: RUBRICS FOR PYTHON LAB

	Excellent(3)	Good(2)	Fair(1)
Assemble (CO1)	Student effectively understands the core features, applications, and how to effectively learn and utilize the language.	Student reasonably understands the core features, applications, and how to effectively learn and utilize the language.	Student does not understand the core features, applications, and how to effectively learn and utilize the language.
Exhibit (CO2)	Thorough execution of programs and getting outputs	Reasonable execution of programs and getting outputs	Improper execution of programs and getting outputs
Apply (CO3)	Student able to apply programming concepts in business scenario	Student somewhat able to apply programming concepts in business scenario	Student unable to apply programming concepts in business scenario
Develop (CO4)	Excellent knowledge in developing programs	Reasonable knowledge in developing programs	Improper knowledge in developing programs
Derive (CO5)	Excellent knowledge in extending the programming in dynamic situations.	Reasonable knowledge in extending the programming in dynamic situations.	Minimal knowledge in extending the programming in dynamic situations.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Introduction to Python

Python is a high-level, interpreted, general-purpose programming language developed by Guido van Rossum. It is known for its simple syntax, readability, and extensive libraries. Python is widely used in business analytics, finance, web development, artificial intelligence, machine learning, automation, and scientific computing. For MBA students, Python provides a practical platform for solving business problems such as sales analysis, budgeting, forecasting, customer analytics, and financial modelling.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Experiment 1: Arithmetic Operators Using Direct Input Values

Aim

To write a Python program that performs arithmetic operations using values assigned directly within the program.

Theory

Arithmetic operators are used to perform mathematical calculations on numeric data. Python provides seven arithmetic operators: addition (+), subtraction (-), multiplication (*), division (/), modulus (%), floor division (//), and exponentiation (**). In this program, the operands are predefined by the programmer, making it suitable for understanding the working of arithmetic operators without user interaction.

Algorithm

1. Start the program.
2. Assign values to two variables.
3. Perform addition, subtraction, multiplication, division, modulus, floor division, and exponentiation.
4. Display the result of each operation.
5. Stop the program.

Procedure

1. Open the Python IDE
2. Create a new Python file.
3. Assign two integer values to variables.
4. Apply all arithmetic operators.
5. Display the results using the print() function.
6. Execute the program.
7. Observe the output.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Program

```
a = 20
```

```
b = 6
```

```
print ("Addition =", a + b)
```

```
print("Subtraction =", a - b)
```

```
print("Multiplication =", a * b)
```

```
print("Division =", a / b)
```

```
print("Modulus =", a % b)
```

```
print("Floor Division =", a // b)
```

```
print("Exponentiation =", a ** b)
```

Result

Thus, the arithmetic operators using direct input values were successfully implemented and their outputs were verified.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Experiment 2: Arithmetic Operators Using User Input

Aim

To write a Python program that performs arithmetic operations using numbers entered by the user.

Theory

Python provides the `input()` function to accept data from the user during program execution. Since the `input()` function returns data as a string, the input must be converted into an integer or float using functions such as `int()` or `float()`. This enables users to perform arithmetic operations dynamically with different input values.

Algorithm

1. Start the program.
2. Read two numbers from the user.
3. Convert the inputs into integers.
4. Perform all arithmetic operations.
5. Display the results.
6. Stop the program.

Procedure

1. Open the Python IDE.
2. Create a new Python file.
3. Use the `input()` function to accept two numbers.
4. Convert the input values using `int()`.
5. Apply all arithmetic operators.
6. Display the results using the `print()` function.
7. Run the program and enter different values to verify the output.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Program

```
a = int(input("Enter First Number: "))  
b = int(input("Enter Second Number: "))
```

```
print("Addition =", a + b)  
print("Subtraction =", a - b)  
print("Multiplication =", a * b)  
print("Division =", a / b)  
print("Modulus =", a % b)  
print("Floor Division =", a // b)  
print("Exponentiation =", a ** b)
```

Result

Thus, the arithmetic operators using user-provided input were successfully implemented, and the results for all arithmetic operations were verified.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Experiment 3: Grade Classification using conditional statements

Aim

To classify student grades using conditional statements.

Theory

Conditional statements allow a program to make decisions based on logical conditions.

Conditional statements enable a program to make decisions based on specified conditions.

Python uses **if**, **elif**, and **else** statements to execute different blocks of code according to logical expressions. These statements help automate decision-making processes in business and education. They are widely used for grading systems, loan approvals, employee appraisal, and customer classification. Decision-making is one of the fundamental concepts in programming.

Algorithm

1. Start.
2. Read two subject marks.
3. Calculate average.
4. Display class.
5. Stop.

Procedure

1. Open Python IDLE
2. Create a new Python file.
3. Type the program.
4. Save the file with .py extension.
5. Run the program and verify the output.

Program

```
subject1=int(input("Enter Subject 1 Marks: "))
subject2=int(input("Enter Subject 2 Marks: "))
average=(subject1+subject2)/2
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

```
if average >= 75:  
    print("First Class")  
elif average >= 60:  
    print("Second Class")  
else:  
    print("Fail")
```

Result

Thus, the program was executed successfully and the expected result was obtained.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Experiment 4: Highest Sales Value using conditional statements

Aim

To identify the territory with the highest sales

Theory

Decision-making statements help compare values and support business analysis.

Decision-making statements are used to compare multiple values and identify the required result.

In this program, sales values of different territories are compared to determine the highest sales.

Such comparisons are useful in business organizations for evaluating regional performance.

Managers use similar techniques to identify the best-performing branch or salesperson. Python

simplifies these comparisons using conditional statements.

Algorithm

1. Start.
2. Read sales of three territories.
3. Compare values.
4. Display highest sales territory.
5. Stop.

Procedure

1. Open Python IDLE or Visual Studio Code.
2. Create a new Python file.
3. Type the program.
4. Save the file with .py extension.
5. Run the program and verify the output.

Program

```
t1=int(input("Enter Territory 1 Sales: "))
t2=int(input("Enter Territory 2 Sales: "))
t3=int(input("Enter Territory 3 Sales: "))
if t1>t2 and t1>t3:
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

```
print("Territory 1 has highest sales")  
elif t2>t1 and t2>t3:  
    print("Territory 2 has highest sales")  
else:  
    print("Territory 3 has highest sales")
```

Sample Output

Output varies depending on the input provided.

Result

Thus, the program was executed successfully and the expected result was obtained.

Learning Outcomes

- Understand the concept.
- Develop Python programs.
- Interpret the output.
- Apply the concept to business problems.



Experiment 5: Odd or Even Number

Aim

To determine whether a number is odd or even

Theory

The modulus (%) operator is used to determine whether a number is divisible by another number. If a number leaves a remainder of zero when divided by two, it is an even number; otherwise, it is odd. This concept is useful in data validation and mathematical computations. The program demonstrates the use of conditional statements with arithmetic operators. It is one of the simplest examples of logical decision-making in Python.

Algorithm

1. Start.
2. Read a number.
3. Check remainder on division by 2.
4. Display result.
5. Stop.

Procedure

1. Open Python IDLE or Visual Studio Code.
2. Create a new Python file.
3. Type the program.
4. Save the file with .py extension.
5. Run the program and verify the output.

Program

```
a=int(input("Enter a whole number: "))
if a%2==0:
    print("Even Number")
else:
    print("Odd Number")
```

Sample Output

Output varies depending on the input provided.

Result

Thus, the program was executed successfully and the expected result was obtained.



Experiment 6: While Loop - Print Odd Numbers

Aim

To print odd numbers using a while loop

Theory

A while loop repeats execution until the specified condition becomes false.

A **while** loop repeatedly executes a block of code as long as a specified condition remains true. It is useful when the number of iterations is not fixed in advance. Loops eliminate repetitive coding and improve program efficiency. They are commonly used in report generation, transaction processing, and data analysis. Proper loop control prevents infinite execution of programs.

Algorithm

1. Start.
2. Read limit.
3. Initialize $i=1$.
4. Repeat while $i \leq \text{limit}$.
5. Increment by 2.
6. Stop.

Procedure

1. Open Python IDLE or Visual Studio Code.
2. Create a new Python file.
3. Type the program.
4. Save the file with .py extension.
5. Run the program and verify the output.

Program

```
n=int(input("Enter the limit: "))
```

```
i=1
```

```
while i<=n:
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

```
print(i)
```

```
i+=2
```

Sample Output

Output varies depending on the input provided.

Result

Thus, the program was executed successfully and the expected result was obtained.

Learning Outcomes

- Understand the concept.
- Develop Python programs.
- Interpret the output.
- Apply the concept to business problems.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Experiment 7: Area of Rectangle using Functions

Aim

To calculate the area of a rectangle using a user-defined function

Theory

Functions improve code reusability and modularity. A function performs a specific task and can be called whenever required. In business applications, functions are useful for repeated calculations such as tax, salary, commission, and area computations.

A function is a reusable block of code designed to perform a specific task. Functions improve program readability, modularity, and code reusability. In this program, a function is created to calculate the area of a rectangle using length and breadth. Functions reduce repetition and simplify maintenance of programs. They are extensively used in software development and business applications.

Algorithm

1. Start the program.
2. Read required input.
3. Process using Python statements/functions.
4. Display the output.
5. Stop the program.

Procedure

1. Open Python IDLE or VS Code.
2. Create a new Python file.
3. Type the program with proper indentation.
4. Save and execute the program.
5. Verify the output.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Program

```
def area(length, breadth):  
    return length * breadth  
  
length = int(input("Enter Length: "))  
breadth = int(input("Enter Breadth: "))  
totalarea = area(length, breadth)  
print("Area of Rectangle =", totalarea)
```

Sample Output

The output depends on the values entered by the user.

Result

Thus, the program was executed successfully and the desired output was obtained.

Learning Outcomes

- Understand the programming concept.
- Write Python programs independently.
- Analyze business-oriented problems.
- Interpret the program output.



Experiment 8: Discount Calculation using Functions

Aim

To calculate the discounted price of a product using a function

Theory

Discount calculation is a common business application. The program accepts the original price and discount percentage and computes the final selling price.

Discount calculation is a common business application used in retail and e-commerce. This program uses a user-defined function to calculate the final price after applying a discount percentage. Functions simplify repetitive business calculations and improve program organization. They also enhance code reusability and maintainability. Python functions are widely used in financial and commercial software.

Algorithm

1. Start the program.
2. Read required input.
3. Process using Python statements/functions.
4. Display the output.
5. Stop the program.

Procedure

1. Open Python IDLE or VS Code.
2. Create a new Python file.
3. Type the program with proper indentation.
4. Save and execute the program.
5. Verify the output.

Program

```
def discprice(price, discount):  
    return price - (discount/100) * price
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

```
price = float(input("Enter Original Price: "))
```

```
discount = float(input("Enter Discount Percentage: "))
```

```
newprice = discprice(price, discount)
```

```
print("Discounted Price =", newprice)
```

Sample Output

The output depends on the values entered by the user.

Result

Thus, the program was executed successfully and the desired output was obtained.

Learning Outcomes

- Understand the programming concept.
- Write Python programs independently.
- Analyze business-oriented problems.
- Interpret the program output.



INTRODUCTION TO PYTHON - 22MBA238

Experiment 9: Total Cost Calculation using Functions

Aim

To calculate the total cost using price and quantity.

Theory

Functions can be used to perform repeated business calculations. This example calculates the total purchase cost using unit price and quantity.

Cost calculation is an essential business operation used in accounting and inventory management. This program calculates the total cost by multiplying the unit price by the quantity purchased. A user-defined function performs the calculation, making the program modular and reusable. Such calculations are frequently used in billing, invoicing, and sales management systems. Python functions make business computations efficient and reliable.

Algorithm

1. Start the program.
2. Read required input.
3. Process using Python statements/functions.
4. Display the output.
5. Stop the program.

Procedure

1. Open Python IDLE or VS Code.
2. Create a new Python file.
3. Type the program with proper indentation.
4. Save and execute the program.
5. Verify the output.

Program

```
def cost(price, quantity):  
    return price * quantity  
price = float(input("Enter Unit Price: "))  
quantity = float(input("Enter Quantity: "))
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

```
totalcost = cost(price, quantity)
```

```
print("Total Cost =", totalcost)
```

Sample Output

The output depends on the values entered by the user.

Result

Thus, the program was executed successfully and the desired output was obtained.

Learning Outcomes

- Understand the programming concept.
- Write Python programs independently.
- Analyze business-oriented problems.
- Interpret the program output.



INTRODUCTION TO PYTHON - 22MBA238

Experiment 10: Lists and Tuples

Aim

To perform basic operations on lists and tuples

Theory

Lists are mutable collections whereas tuples are immutable collections. Both are widely used for storing and processing multiple values in Python.

Lists and tuples are Python data structures used to store collections of items. Lists are mutable, allowing elements to be added, removed, or modified, whereas tuples are immutable after creation. These data structures help organize and manage large amounts of information efficiently. They are widely used in customer databases, inventory records, and business reports. Python provides several built-in methods to manipulate lists and tuples.

Algorithm

1. Start the program.
2. Read required input.
3. Process using Python statements/functions.
4. Display the output.
5. Stop the program.

Procedure

1. Open Python IDLE or VS Code.
2. Create a new Python file.
3. Type the program with proper indentation.
4. Save and execute the program.
5. Verify the output.

Program

List

```
fruits = ["Orange", "Papaya"]
```

```
fruits.append("Mango")
```

```
fruits.remove("Orange")
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

```
print("List:", fruits)
```

```
# Tuple
```

```
items = ("Orange", "Papaya")
```

```
print("Index of Orange:", items.index("Orange"))
```

```
print("Count of Papaya:", items.count("Papaya"))
```

```
print("Tuple:", items)
```

Sample Output

The output depends on the values entered by the user.

Result

Thus, the program was executed successfully and the desired output was obtained.

Learning Outcomes

- Understand the programming concept.
- Write Python programs independently.
- Analyze business-oriented problems.
- Interpret the program output.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Experiment 11: String Handling

Aim

To perform common string operations in Python

Theory

Python provides several built-in string methods for processing textual data such as converting case, removing spaces, replacing text, and searching within strings.

Strings represent textual information in Python and are widely used for storing names, addresses, product descriptions, and reports. Python provides numerous built-in methods to manipulate strings efficiently. Operations such as converting case, removing spaces, replacing text, and searching for words simplify text processing. String handling plays an important role in business communication and data management. It is one of the most frequently used features in Python programming.

Algorithm

1. Start the program.
2. Read required input.
3. Process using Python statements/functions.
4. Display the output.
5. Stop the program.

Procedure

1. Open Python IDLE or VS Code.
2. Create a new Python file.
3. Type the program with proper indentation.
4. Save and execute the program.
5. Verify the output.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Program

```
company = " Global Furnishings Pvt Ltd "  
print(company.strip())  
print(company.upper())  
print(company.lower())  
print(company.replace("Pvt", "Private"))  
print("Global" in company)
```

Sample Output

The output depends on the values entered by the user.

Result

Thus, the program was executed successfully and the desired output was obtained.

Learning Outcomes

- Understand the programming concept.
- Write Python programs independently.
- Analyze business-oriented problems.
- Interpret the program output.



Experiment 12: Dictionary

Aim

To perform basic operations on dictionaries in Python

Theory

Dictionary is a mutable collection that stores data as key-value pairs. It is widely used to represent structured business information such as customer records, employee details, and product databases.

A dictionary is a Python collection that stores data as key-value pairs. It allows quick retrieval and updating of information using unique keys. Dictionaries are widely used to represent structured data such as employee records, customer information, and product details. They are mutable and support efficient data management. Dictionaries are extensively used in business applications and database-related programs.

Algorithm

1. Start the program.
2. Accept/create required data.
3. Process the data.
4. Display the output.
5. Stop the program.

Procedure

1. Open Python IDLE/VS Code.
2. Create a Python file.
3. Type the program with proper indentation.
4. Save and execute.
5. Verify the output.

Program

```
customer = {  
    "Name": "Azeez",
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

"Age": 18

}

print("Customer Details:", customer)

customer["City"] = "Chittoor"

print("Updated Dictionary:", customer)

print("Customer Name:", customer["Name"])

Sample Output

Output depends on the input or program execution.

Result

Thus, the program was executed successfully.

Learning Outcomes

- Understand the concept.
- Develop Python programming skills.
- Apply concepts to business applications.
- Interpret program results.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Experiment 13: Class and Object Creation

Aim

To create a class and object in Python

Theory

Object-Oriented Programming (OOP) organizes programs using classes and objects. A class is a blueprint, while an object is an instance of the class.

Python supports object-oriented programming through classes and objects. A class serves as a blueprint for creating objects, while an object represents an instance of a class. This approach helps organize programs into reusable components. Object-oriented programming improves code maintainability, modularity, and scalability. It is widely used in developing enterprise-level business software.

Algorithm

1. Start the program.
2. Accept/create required data.
3. Process the data.
4. Display the output.
5. Stop the program.

Procedure

1. Open Python IDLE/VS Code.
2. Create a Python file.
3. Type the program with proper indentation.
4. Save and execute.
5. Verify the output.

Program

```
class Person:  
    def say_hello(self):  
        print("Hello!")
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

```
p1 = Person()  
p1.say_hello()
```

Sample Output

Output depends on the input or program execution.

Result

Thus, the program was executed successfully.

Learning Outcomes

- Understand the concept.
- Develop Python programming skills.
- Apply concepts to business applications.
- Interpret program results.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Experiment 14: Inheritance and Polymorphism

Aim

To demonstrate inheritance and polymorphism in Python

Theory

Inheritance allows one class to acquire the properties of another class. Polymorphism enables the same method name to perform different actions depending on the object.

Inheritance enables one class to acquire the properties and methods of another class, promoting code reuse. Polymorphism allows the same method to perform different actions depending on the object invoking it. These concepts reduce code duplication and improve software flexibility. They are fundamental principles of object-oriented programming. Business applications frequently use inheritance and polymorphism to develop scalable software systems.

Algorithm

1. Start the program.
2. Accept/create required data.
3. Process the data.
4. Display the output.
5. Stop the program.

Procedure

1. Open Python IDLE/VS Code.
2. Create a Python file.
3. Type the program with proper indentation.
4. Save and execute.
5. Verify the output.

Program

```
class Employee:  
    def work(self):  
        print("Working...")
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

```
class Manager(Employee):  
    def work(self):  
        print("Managing the team")  
class Developer(Employee):  
    def work(self):  
        print("Writing code")  
employees = [Manager(), Developer()]  
for emp in employees:  
    emp.work()
```

Sample Output

Output depends on the input or program execution.

Result

Thus, the program was executed successfully.

Learning Outcomes

- Understand the concept.
- Develop Python programming skills.
- Apply concepts to business applications.
- Interpret program results.



Experiment 15: Exception Handling

Aim

To handle runtime errors using exception handling

Theory

Exception handling prevents abnormal program termination by catching runtime errors and allowing the program to continue gracefully.

Exception handling is a mechanism used to detect and handle runtime errors without terminating the program unexpectedly. Python uses **try**, **except**, **else**, and **finally** blocks to manage exceptions effectively. It improves program reliability by allowing graceful recovery from errors. Exception handling is particularly useful in financial, banking, and database applications where uninterrupted execution is important. It helps developers build robust and user-friendly software.

Algorithm

1. Start the program.
2. Accept/create required data.
3. Process the data.
4. Display the output.
5. Stop the program.

Procedure

1. Open Python IDLE/VS Code.
2. Create a Python file.
3. Type the program with proper indentation.
4. Save and execute.
5. Verify the output.

Program

```
total_sales = 10000
num_employees = int(input("Enter number of employees: "))
sales_per_employee = total_sales / num_employees
print("Sales per Employee =", sales_per_employee)
```



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

except ZeroDivisionError:

```
print("Error: Number of employees cannot be zero.")
```

except ValueError:

```
print("Error: Please enter a valid integer.")
```

finally:

```
print("Program Execution Completed.")
```

Sample Output

Output depends on the input or program execution.

Result

Thus, the program was executed successfully.

Learning Outcomes

- Understand the concept.
- Develop Python programming skills.
- Apply concepts to business applications.
- Interpret program results.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Experiment

Bloom's Taxonomy

- | | |
|---|---|
| 1. Arithmetic Operators | Design a Python program to perform all arithmetic operations on two user-entered numbers and display the results. |
| 2. Grade Classification | Develop a Python program to calculate a student's average marks and assign the appropriate grade based on predefined conditions. |
| 3. Highest Sales Value | Construct a Python program to compare the sales of three territories and identify the territory with the highest sales. |
| 4. Odd or Even Number | Create a Python program to determine whether a given number is odd or even using conditional statements. |
| 5. While Loop | Build a Python program using a while loop to generate and display all odd numbers up to a user-specified limit. |
| 6. Area of Rectangle using Functions | Develop a Python program using a user-defined function to calculate the area of a rectangle for user-specified dimensions. |
| 7. Discount Calculation | Design a Python program using a function to calculate the discounted price of a product based on the original price and discount percentage. |
| 8. Total Cost Calculation | Construct a Python program using a function to calculate the total purchase cost based on unit price and quantity. |
| 9. Lists and Tuples | Create a Python program to perform insertion, deletion, and retrieval operations on lists and demonstrate tuple operations. |
| 10. String Handling | Develop a Python program to perform string manipulation operations such as case conversion, replacement, trimming, and searching. |
| 11. Dictionary | Design a Python program to create and manage a dictionary for storing and retrieving customer information. |
| 12. Class and Object Creation | Construct a Python program to create a class, instantiate an object, and invoke its methods to demonstrate object-oriented programming. |
| 13. Inheritance and Polymorphism | Develop a Python program to demonstrate inheritance and polymorphism using employee-related classes. |
| 14. Exception Handling | Create a Python program that handles runtime exceptions using appropriate exception handling constructs to ensure uninterrupted program execution. |



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

Python Programming Laboratory Manual (MBA)

Viva Voce Questions with Answers

S.No.	Viva Question	One-Line Answer
1	What is Python?	Python is a high-level, interpreted programming language.
2	Who developed Python?	Guido van Rossum developed Python in 1991.
3	Why is Python called an interpreted language?	Because it executes code line by line without compilation.
4	What is an operator?	An operator is a symbol used to perform operations on operands.
5	Name any four arithmetic operators.	Addition (+), Subtraction (-), Multiplication (*), and Division (/).
6	What is the purpose of the modulus (%) operator?	It returns the remainder after division.
7	What is floor division (//)?	It returns the integer quotient without the decimal part.
8	Which operator is used for exponentiation?	The ** operator is used for exponentiation.
9	What is a variable?	A variable is a named memory location used to store data.
10	Which function is used to accept input from the user?	The input() function is used to accept user input.
11	Why is int() used with input()?	It converts the input string into an integer.
12	What is a conditional statement?	It is a statement used to make decisions based on conditions.
13	Which keyword is used for multiple conditions?	The elif keyword is used for multiple conditions.
14	What is the purpose of the else statement?	It executes when all conditions are false.
15	Which operator compares two values for equality?	The == operator compares two values.
16	What is a loop?	A loop repeatedly executes a block of code.
17	Name the looping statements in Python.	for and while loops.
18	When is a while loop used?	It is used when the number of iterations is unknown.
19	What causes an infinite loop?	A loop condition that never becomes false.
20	What is the purpose of break?	It immediately terminates the loop.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES
(Autonomous)
CHITTOOR-517 127
DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

S.No.	Viva Question	One-Line Answer
21	What is the purpose of continue?	It skips the current iteration and moves to the next one.
22	What is a function?	A function is a reusable block of code that performs a specific task.
23	Which keyword is used to define a function?	The def keyword is used to define a function.
24	What is the purpose of the return statement?	It returns a value from a function.
25	What is a parameter?	A parameter is a variable used in a function definition.
26	What is a list?	A list is an ordered, mutable collection of elements.
27	Which method adds an element to a list?	The append() method adds an element.
28	Which method removes an element from a list?	The remove() method removes an element.
29	What is a tuple?	A tuple is an ordered, immutable collection of elements.
30	Can a tuple be modified after creation?	No, tuples are immutable.
31	What is a string?	A string is a sequence of characters enclosed in quotes.
32	Which method converts a string to uppercase?	The upper() method.
33	Which method removes leading and trailing spaces?	The strip() method.
34	Which method replaces a word in a string?	The replace() method.
35	What is a dictionary?	A dictionary stores data as key-value pairs.
36	Are dictionary keys unique?	Yes, dictionary keys must be unique.
37	What is a class?	A class is a blueprint for creating objects.
38	What is an object?	An object is an instance of a class.
39	What is Object-Oriented Programming (OOP)?	OOP is a programming paradigm based on classes and objects.
40	What is inheritance?	Inheritance allows one class to acquire properties of another class.
41	What is polymorphism?	Polymorphism allows the same method to behave differently for different objects.
42	What is method overriding?	It is redefining a parent class method in the child class.
43	What is an exception?	An exception is a runtime error that interrupts normal program execution.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

S.No.	Viva Question	One-Line Answer
44	Which block is used to handle exceptions?	The except block handles exceptions.
45	Why is the try block used?	It encloses code that may generate exceptions.
46	What is the purpose of the finally block?	It executes regardless of whether an exception occurs.
47	Which exception occurs when dividing by zero?	Zero Division Error.
48	What is indentation in Python?	Indentation defines the structure of code blocks.
49	Why is Python popular in business analytics?	Because it provides powerful libraries for data analysis and automation.
50	Mention one application of Python in MBA.	Python is used for financial analysis, sales forecasting, and business analytics.



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

DAY TO DAY EVALUATION

NAME:

ROLL NO:

S.No	Experiment Name	Assemble (CO1)	Exhibit (CO2)	Apply (CO3)	Develop (CO4)	Derive (CO5)
1	Arithmetic Operators using defined inputs					
2	Arithmetic operators using user inputs					
3	Grade Classification					
4	Highest Sales Value					
5	Odd or Even Number					
6	While Loop - Print Odd Numbers					
7	Area of Rectangle using Functions					
8	Discount Calculation using Functions					
9	Total Cost Calculation using Functions					
10	Lists and Tuples					
11	String Handling					
12	Dictionary					
13	Class and Object Creation					
14	Inheritance and Polymorphism					
15	Exception Handling					
	AVERAGE					

COURSE COORDINATOR



SREENIVASA INSTITUTE of TECHNOLOGY AND MANAGEMENT STUDIES

(Autonomous)

CHITTOOR-517 127

DEPARTMENT OF MANAGEMENT STUDIES

INTRODUCTION TO PYTHON - 22MBA238

INTERNAL EXAM EVALUATION

NAME:

ROLL NO:

S.No	Experiment Name	Written Test (5)	Knowledge Demonstrated (4)	Communication & Presentation skills (4)	Behavioural Skills (3)	Viva Voce (4)
1	Arithmetic Operators using defined inputs					
2	Arithmetic operators using user inputs					
3	Grade Classification					
4	Highest Sales Value					
5	Odd or Even Number					
6	While Loop - Print Odd Numbers					
7	Area of Rectangle using Functions					
8	Discount Calculation using Functions					
9	Total Cost Calculation using Functions					
10	Lists and Tuples					
11	String Handling					
12	Dictionary					
13	Class and Object Creation					
14	Inheritance and Polymorphism					
15	Exception Handling					

COURSE COORDINATOR