Programming in Python LAB

(CIE - 332P)

Faculty Name : Mr. Anupam Kumar

Name : Amit Singhal

Enrollment No. : 11614802722

Semester : 6

Group : AIML-II-B

Maharaja Agrasen Institute of Technology, PSP Area,

Sector – 22, Rohini, New Delhi – 110085

MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY

VISION

"To attain global excellence through **education, innovation, research,** and **work ethics** with the commitment to **serve humanity."**

MISSION

M1. To promote diversification by adopting advancement in science, technology, management, and allied discipline through continuous learning

M2. To foster **moral values** in students and equip them for developing sustainable solutions to serve both national and global needs in society and industry.

M3. To **digitize educational resources and process** for enhanced teaching and effective learning.

M4. To cultivate an **environment** supporting **incubation, product development, technology transfer, capacity building and entrepreneurship**.

M5. To encourage **faculty-student networking with alumni, industry, institutions,** and other **stakeholders** for collective engagement.

Department of Computer Science and Engineering

MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY

VISION

"To attain global excellence through education, innovation, research, and work ethics in the field of Computer Science and engineering with the commitment to serve humanity."

MISSION

M1. To lead in the advancement of computer science and engineering through internationally recognized research and education.

M2. To prepare students for full and ethical participation in a diverse society and encourage lifelong learning.

M3. To foster development of problem solving and communication skills as an integral component of the profession.

M4. To impart knowledge, skills and cultivate an environment supporting incubation, product development, technology transfer, capacity building and entrepreneurship in the field of computer science and engineering.

M5. To encourage faculty, student's networking with alumni, industry, institutions, and other stakeholders for collective engagement.

LAB INDEX

S.No	Experiment	R1	R2	R3	R4	R5	Total Marks (10)	Remarks	Faculty Signature
		Is able to identify and define the objective of the given problem?	Is proposed design /procedure /algorithm solves the problem?	Has the understanding of the tool/programming language to implement the proposed solution?	Are the result(s) verified using sufficient test data to support the conclusions?	Individuality of submission?			
		2 Marks	2 Marks	2 Marks	2 Marks	2 Marks			

Experiment – 1.1

AIM :: Create a program that prompts the user for the length and width of a rectangle and then calculates and prints the area.

Code ::

Output ::

```
Amit - 116$ python3 rectangle.py
Length of Rectangle: 105
Width of Rectangle: 116
Area of Rectangle is: 12180
```

Experiment – 1.2

AIM :: Create a program that prompts the user for the price of an item and the sales tax rate (as a percentage). Then, calculate and print the final price including tax.

Code ::

Output ::

```
Amit - 116$ python3 price.py
Enter the price of the item: 100.25
Enter the tax rate of the item: 67
Final price including tax is:  167.41750000000002
```

Experiment – 1.3

AIM :: Create a program that asks the user for two numbers and then prints out which number is larger (or if they are equal).

Code ::

Output ::

```
Amit - 116$ python3 larger.py
Enter num1: 78
Enter num2: 66
Greater of the 2 number is:  78
```

Experiment – 1.4

AIM :: Create a program that prompts the user for their age and favorite number. Then, add their age to their favorite number and print the result.

Code ::

Output ::

```
Amit - 116$ python3 aging.py
Enter age: 21
Enter your favourite number: 712
Congrats new number is:  733
```

Experiment – 1.5

AIM :: Create a program that prompts the user for their name and age and prints a personalized message

Code ::

Output ::

```
Amit - 116$ python3 basic.py
Enter age: 21
Enter name: Amit
Hello Amit Congrats on turning  21  this year.
```

WEEK - 2

Experiment - 1

AIM :: Create a program that prompts the user to enter a number and determines whether the number is even or odd. The program should then print the result.

Code ::

Output ::

```
Week - 2$ python3 oddeven.py
Week 2 - Experiment 1
Amit Singhal | 11614802722
Enter a number: 23
The number 23 is Odd.
Week - 2$ python3 oddeven.py
Week 2 - Experiment 1
Amit Singhal | 11614802722
Enter a number: 100
The number 100 is Even.
```

Experiment - 2

AIM :: Create a program that prompts the user to enter their marks and classifies them into different grades (A+, A, B+, B, C+, C, F) based on predefined thresholds. The program should print the corresponding grade.

Code ::

Output ::

```
Week - 2$ python3 grade.py
Week 2 - Experiment 2
Amit Singhal | 11614802722
Enter student's marks (0-100): 89
Assigned Grade: B
```

Experiment - 3

AIM :: Create a program that prompts the user to enter a number and checks if it is divisible by 3 or 5. The program should print whether the number is divisible by 3, 5, or both, or neither.

Code ::

Output ::

```
Week - 2$ python3 divisibility.py
Week 2 - Experiment 3
Amit Singhal | 11614802722
Enter the number: 15
The number 15 is divisible by both 3 and 5.
```

Experiment - 4

AIM :: Create a program that prompts the user to enter the lengths of three sides of a triangle. The program should then determine the type of triangle – Equilateral, Isosceles & Scalene.

Code ::

Output ::

```
Week - 2$ python3 triangle.py
Week 2 - Experiment 4
Amit Singhal | 11614802722
Enter the sides of the triangle:
Side 'a': 3
Side 'b': 6
Side 'c': 3
The triangle is Isosceles.
```

Experiment - 5

AIM :: Create a program that prompts the user to enter an amount and calculates the final amount after an 18% increase. The program should print the final amount after applying the increase.

Code ::

Output ::

```
Week - 2$ python3 billcalculator.py
Week 2 - Experiment 5
Amit Singhal | 11614802722
Enter the amount: 999
Final amount after an 18% increase: 1178.82
```

Experiment - 6

AIM :: Create a program that prompts the user to choose between Rock, Paper, or Scissors. The program should then randomly choose one of these options for the opponent and determine the winner based on the rules of the game.

Code ::

Output ::

```
Week - 2$ python3 game.py
Week 2 - Experiment 6
Amit Singhal | 11614802722
Choose one of the following:
1 - Rock
2 - Paper
3 - Scissors
Enter your choice (1, 2, 3): 2
You chose: Paper
Opponent chose: Rock
You Win!
```

Experiment - 7

AIM :: Create a program that prompts the user to enter their age and determines if they are eligible to vote. The program should print whether they are eligible or not based on the voting age requirement.

Code ::

Output ::

```
Week - 2$ python3 vote.py
Week 2 - Experiment 7
Amit Singhal | 11614802722
Enter your age: 21
You are eligible to vote.
```

Experiment - 8

AIM :: Create a program that prompts the user to enter a number and then prints all the even numbers from 0 to that number.

Code ::

Output ::

```
Week - 2$ python3 printeven.py
Week 2 - Experiment 8
Amit Singhal | 11614802722
Enter a natural number: 35
Even numbers up to 35:
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34
```

Experiment - 9

AIM :: Create a program that prompts the user to enter a number and calculates the sum of all natural numbers from 1 up to that number. The program should then print the sum.

Code ::

Output ::

```
Week - 2$ python3 sum\ of\ n.py
Week 2 - Experiment 9
Amit Singhal | 11614802722
Enter a natural number: 15
The sum of the first 15 natural numbers is 120
```

Experiment - 10

AIM :: Create a program that prompts the user to enter a number and calculates its factorial. The program should print the factorial of the number.

Code ::

Output ::

```
Week - 2$ python3 factorial.py
Week 2 - Experiment 10
Amit Singhal | 11614802722
Enter a number to calculate its factorial: 7
The factorial of 7 is 5040
```

Experiment - 11

AIM :: Create a program that prompts the user to enter a number and checks if the number is prime. The program should print whether the number is prime or not.

Code ::

Output ::

```
Week - 2$ python3 checkPrime.py
Week 2 - Experiment 11
Amit Singhal | 11614802722
Enter a number to check if it's prime: 11
11 is a prime number.
```

Experiment - 12

AIM :: Create a program that prompts the user to enter the number of rows and generates a pyramid pattern with increasing and decreasing numbers for each row.

Code ::

Output ::

```
Week - 2$ python3 pattern.py
Week 2 - Experiment 12
Amit Singhal | 11614802722
Enter the number of rows: 6
          1
        1 2 1
      1 2 3 2 1
    1 2 3 4 3 2 1
  1 2 3 4 5 4 3 2 1
1 2 3 4 5 6 5 4 3 2 1
```

WEEK - 3

Experiment - 1

AIM :: Write a Python program to find the maximum and minimum values in a list.

Code ::

Output ::

```
Week - 3$ python3 exp1.py
Week 3 - Experiment 1
Amit Singhal | 11614802722
The list of numbers is: [1, 2, 3, 4, 5, 6, 7, 8, 9]
The maximum value in the list is: 9
The minimum value in the list is: 1
```

Experiment - 2

AIM :: Write a program to calculate the sum and average of all elements in a list.

Code ::

Output ::

```
Week - 3$ python3 exp2.py
Week 3 - Experiment 2
Amit Singhal | 11614802722
The list of numbers is: [1, 2, 3, 4, 5, 6, 7, 8, 9]
The total sum of the list is: 45
The average value of the list is: 5.0
```

Experiment - 3

AIM :: Write a Python program to count the occurrences of a specific element in a list.

Code ::

Output ::

```
Week - 3$ python3 exp3.py
Week 3 - Experiment 3
Amit Singhal | 11614802722
The list of numbers is: [12, 15, 22, 18, 25, 30, 15, 40, 50, 15, 60, 15, 75, 80, 90]
Enter the number to count: 15
The number 15 appears 4 times in the list.
```

Experiment - 4

AIM :: Write a program to reverse a list.

Code ::

Output ::

```
Week - 3$ python3 exp4.py
Week 3 - Experiment 4
Amit Singhal | 11614802722
The original list is: [10, 25, 30, 45, 50, 60, 75, 80, 90, 100, 110, 120, 130]
The reversed list is: [130, 120, 110, 100, 90, 80, 75, 60, 50, 45, 30, 25, 10]
```

Experiment - 5

AIM :: Write a Python program to check if a list is empty.

Code ::

Output ::

```
Week - 3$ python3 exp5.py
Week 3 - Experiment 5
Amit Singhal | 11614802722
The list is not empty.
The list is: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
```

Experiment - 6

AIM :: Write a program to remove duplicates from a list and display the unique elements.

Code ::

Output ::

```
Week - 3$ python3 exp6.py
Week 3 - Experiment 6
Amit Singhal | 11614802722
Enter numbers separated by spaces: 1 2 3 3 4 5 2 4 5 6 5 1 2 3 5
The list with unique elements is: [1, 2, 3, 4, 5, 6]
```

Experiment - 7

AIM :: Write a Python program to merge two lists and sort the resulting list in ascending order.

Code ::

Output ::

```
Week - 3$ python3 exp7.py
Week 3 - Experiment 7
Amit Singhal | 11614802722
Enter the first list of numbers separated by spaces: 5 15 25 35 45
Enter the second list of numbers separated by spaces: 10 20 30 40 50
The first list is: [5, 15, 25, 35, 45]
The second list is: [10, 20, 30, 40, 50]
The merged and sorted list is: [5, 10, 15, 20, 25, 30, 35, 40, 45, 50]
```

Experiment - 8

AIM :: Write a program to find the second largest number in a list.

Code ::

Output ::

```
Week - 3$ python3 exp8.py
Week 3 - Experiment 8
Amit Singhal | 11614802722
Enter a list of numbers separated by spaces: 5 25 15 35 10 15 45 55 50 35 55 50
The second largest number in the list is: 50
```

Experiment - 9

AIM :: Write a program to check if two lists have any common elements.

Code ::

Output ::

```
Week - 3$ python3 exp9.py
Week 3 - Experiment 9
Amit Singhal | 11614802722
Enter the first list of numbers separated by spaces: 5 10 15 20 25 30 35 40
Enter the second list of numbers separated by spaces: 15 25 35 45 55 65
The common elements between both lists are: [15, 25, 35]
```

Experiment - 10

AIM :: Write a Python program to rotate a list to the right by a given number of positions.

Code ::

Output ::

```
Week - 3$ python3 exp10.py
Week 3 - Experiment 10
Amit Singhal | 11614802722
Enter a list of numbers separated by spaces: 5 10 15 20 25 30 35 40
Enter the number of positions to rotate by: 4
The rotated list is: [25, 30, 35, 40, 5, 10, 15, 20]
```

Experiment - 11

AIM :: Write a program to flatten a nested list (e.g., [[1, 2], [3, [4, 5]]] → [1, 2, 3, 4, 5]).

Code ::

Output ::

```
Week - 3$ python3 exp11.py
Week 3 - Experiment 11
Amit Singhal | 11614802722
Enter a nested list: [[10, 20], [30, [40, 50]], 60, [70, [80, 90]]]
The flattened list is: [10, 20, 30, 40, 50, 60, 70, 80, 90]
```

Experiment - 12

AIM :: Write a Python program to find all subsets of a given list.

Code ::

Output ::

```
Week - 3$ python3 exp12.py
Week 3 - Experiment 12
Amit Singhal | 11614802722
Enter a list of numbers separated by spaces: 1 2 3
All subsets of the given list are: [[1, 2, 3], [1, 2], [1, 3], [1], [2, 3], [2], [3], []]
```

Experiment - 13

AIM :: Write a program to generate all permutations of a list.

Code ::

Output ::

```
Week - 3$ python3 exp13.py
Week 3 - Experiment 13
Amit Singhal | 11614802722
Enter a list of numbers separated by spaces: 7 1 2
All permutations of the given list are:
[7, 1, 2]
[7, 2, 1]
[1, 7, 2]
[1, 2, 7]
[2, 1, 7]
[2, 7, 1]
```

Experiment - 14

AIM :: Write a Python program to find the longest increasing subsequence in a list of numbers.

Code ::

Output ::

```
Week - 3$ python3 exp14.py
Week 3 - Experiment 14
Amit Singhal | 11614802722
Enter a list of numbers separated by spaces: 10 22 9 33 21 50 41 60 80
The longest increasing subsequence is: [10, 22, 33, 50, 60, 80]
```

Experiment - 15

AIM :: Write a program to partition a list into sublists of specified length.

Code ::

Output ::

```
Week - 3$ python3 exp15.py
Week 3 - Experiment 15
Amit Singhal | 11614802722
Enter a list of numbers separated by spaces: 1 2 3 4 5 6 7 8 9
Enter the size of each sublist: 3
The partitioned list is: [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
```

WEEK - 4

Experiment - 1

AIM :: Write a Python program to create a dictionary with three key-value pairs and print it.

Code ::

Output ::

```
Week - 4$ python3 01_Input-Dictionary.py
Week 4 - Experiment 1
Amit Singhal | 11614802722
Enter key : amit
Enter value For Key 'amit' : 116
Enter key : shaswat
Enter value For Key 'shaswat' : 105
Enter key : yash
Enter value For Key 'yash' : 122
{'amit': 116, 'shaswat': 105, 'yash': 122}
```

Experiment - 2

AIM :: Write a Python program to access the value of a given key in a dictionary.

Code ::

Output ::

```
Week - 4$ python3 02_Dictionary-Keys.py
Week 4 - Experiment 2
Amit Singhal | 11614802722
The Value At Key 'name' Is : Amit Singhal
The Value At Key 'age' Is : 21
The Value At Key 'city' Is : New Delhi
```

Experiment - 3

AIM :: Write a Python program to add a new key-value pair to an existing dictionary.

Code ::

Output ::

```
Week - 4$ python3 03_Add-New-Item.py
Week 4 - Experiment 3
Amit Singhal | 11614802722
{'name': 'Amit', 'age': 21, 'city': 'New Delhi', 'CGPA': 9.03}
```

Experiment - 4

AIM :: Write a Python program to remove a key from a dictionary using the pop() method.

Code ::

Output ::

```
Week - 4$ python3 04_Remove-Item.py
Week 4 - Experiment 4
Amit Singhal | 11614802722
{'age': 21, 'city': 'New Delhi'}
```

Experiment - 5

AIM :: Write a Python program to check if a key exists in a dictionary.

Code ::

Output ::

```
Week - 4$ python3 05_Search-Item.py
Week 4 - Experiment 5
Amit Singhal | 11614802722
Enter The Key To Be Searched : name
The Value At Key 'name' Is : Amit
```

Experiment - 6

AIM :: Write a Python program to merge two dictionaries into one.

Code ::

Output ::

```
Week - 4$ python3 06_Merge-Dictionary.py
Week 4 - Experiment 6
Amit Singhal | 11614802722
{'name': 'Amit', 'age': 21, 'city': 'New Delhi', 'CGPA': 9.03, 'Branch': 'CSE', 'University': 'MAIT'}
```

Experiment - 7

AIM :: Write a Python program to iterate through a dictionary and print each key-value pair.

Code ::

Output ::

```
Week - 4$ python3 07_Iterate-Dictionary.py
Week 4 - Experiment 7
Amit Singhal | 11614802722
The Value At Key 'name' Is : Amit
The Value At Key 'age' Is : 21
The Value At Key 'city' Is : New Delhi
```

Experiment - 8

AIM :: Write a Python program to find the key with the maximum value in a dictionary.

Code ::

Output ::

```
Week - 4$ python3 08_Min-Value-Key.py
Week 4 - Experiment 8
Amit Singhal | 11614802722
The Key With Max. value Is book With Value 20
```

Experiment - 9

AIM :: Write a Python program to count the occurrences of each character in a string using a dictionary.

Code ::

Output ::

```
Week - 4$ python3 09_Count-Characters.py
Week 4 - Experiment 9
Amit Singhal | 11614802722
Enter The String : hello how are u
The Dictionary Is :
{'a': 1, 'e': 2, 'h': 2, 'l': 2, 'o': 2, 'r': 1, 'u': 1, 'w': 1}
```

Experiment - 10

AIM :: Write a Python program to invert a dictionary (swap keys and values).

Code ::

Output ::

```
Week - 4$ python3 10_Invert-Dictionary.py
Week 4 - Experiment 10
Amit Singhal | 11614802722
{'India': 'country', 'Uttar Pradesh': 'state', 'Lucknow': 'city'}
```

Experiment - 11

AIM :: Write a Python program to sort a dictionary by its values in ascending and descending order.

Code ::

Output ::

```
Week - 4$ python3 11_Sort-Dictionary.py
Week 4 - Experiment 11
Amit Singhal | 11614802722
Ascending order: {'b': 1, 'd': 2, 'a': 3, 'c': 5}
Descending order: {'c': 5, 'a': 3, 'd': 2, 'b': 1}
```

Experiment - 12

AIM :: Write a Python program to find the intersection of two dictionaries (common keys with the same values).

Code ::

Output ::

```
Week - 4$ python3 12_Intersection-Of-Dict.py
Week 4 - Experiment 12
Amit Singhal | 11614802722
{'b': 7, 'd': 116}
```

Experiment - 13

AIM :: Write a Python program to convert two lists into a dictionary (one as keys, the other as values).

Code ::

Output ::

```
Week - 4$ python3 13_Lists-To-Dict.py
Week 4 - Experiment 13
Amit Singhal | 11614802722
{'name': 'amit', 'age': '21', 'city': 'hello_world'}
```

Experiment - 14

AIM :: Write a Python program to group a list of dictionaries by a common key.

Code ::

Output ::

```
Week - 4$ python3 14_Group-Dict.py
Week 4 - Experiment 14
Amit Singhal | 11614802722

Grouped by name:
{'amit': [{'age': 21, 'name': 'amit'}, {'age': 25, 'name': 'amit'}],
 'lakshya': [{'age': 21, 'name': 'lakshya'}],
 'shaswat': [{'age': 18, 'name': 'shaswat'}]}

Grouped by age:
{18: [{'age': 18, 'name': 'shaswat'}],
 21: [{'age': 21, 'name': 'amit'}, {'age': 21, 'name': 'lakshya'}],
 25: [{'age': 25, 'name': 'amit'}]}
```

Experiment - 15

AIM :: Write a Python program to implement a nested dictionary where each key contains another dictionary, and retrieve specific values dynamically.

Code ::

Output ::

```
Week - 4$ python3 15_Nested-Dict.py
Week 4 - Experiment 15
Amit Singhal | 11614802722
amit's age: 21
shaswat's city: bihari
yash's city: delhite
```

WEEK - 5

Experiment - 1

AIM :: Write a Python program to pretty-print a list of dictionaries containing employee details (name, age, department).

Code ::

Output ::

```
Week - 5$ python3 01_pprint_employee.py
Week 5 - Experiment 1
Amit Singhal | 11614802722
[{'age': 30, 'department': 'HR', 'name': 'Alice'},
 {'age': 25, 'department': 'IT', 'name': 'Bob'},
 {'age': 28, 'department': 'Finance', 'name': 'Charlie'}]
```

Experiment - 2

AIM :: Format and pretty-print a table displaying student names and marks using string formatting.

Code ::

Output ::

```
Week - 5$ python3 02_pprint_student.py
Week 5 - Experiment 2
Amit Singhal | 11614802722
Name      Marks
--------------------
Alice     85
Bob       78
Charlie   92
```

Experiment - 3

AIM :: Create a Car class with attributes brand, model, and year. Implement a method to display car details.

Code ::

Output ::

```
Week - 5$ python3 03_car_DS.py
Week 5 - Experiment 3
Amit Singhal | 11614802722
Car: Toyota Corolla, Year: 2020
```

Experiment - 4

AIM :: Design an Employee class with attributes name, ID, and salary, and a method to give a percentage-based salary increment.

Code ::

Output ::

```
Week - 5$ python3 04_employee_DS.py
Week 5 - Experiment 4
Amit Singhal | 11614802722
Employee: Alice, ID: E123, Salary: 55000.0
```

Experiment - 5

AIM :: Write a Python program to reverse each word in a given string while maintaining the word order.

Code ::

Output ::

```
Week - 5$ python3 05_reverse_string.py
Week 5 - Experiment 5
Amit Singhal | 11614802722
olleH dlroW nohtyP
```

Experiment - 6

AIM :: Implement a function to check whether a given string is a valid password (at least 8 characters, contains digits, uppercase, lowercase, and a special character).

Code ::

Output ::

```
Week - 5$ python3 06_valid_password.py
Week 5 - Experiment 6
Amit Singhal | 11614802722
Is 'Strong@123' a valid password? True
```

Experiment - 7

AIM :: Write a Python program that handles ZeroDivisionError and prompts the user to enter a valid denominator.

Code ::

Output ::

```
Week - 5$ python3 07_zeroDiv_error.py
Week 5 - Experiment 7
Amit Singhal | 11614802722
Enter numerator: 10
Enter denominator: 0
Error: Denominator cannot be zero. Please enter a valid denominator.
Enter numerator: 25
Enter denominator: 5
Result: 5.0
```

Experiment - 8

AIM :: Implement a function that opens a file and handles FileNotFoundError if the file does not exist.

Code ::

Output ::

```
Week - 5$ python3 08_fileNotFound_error.py
Week 5 - Experiment 8
Amit Singhal | 11614802722
Error: The file 'sample.txt' was not found.
```

Experiment - 9

AIM :: Create a Product class with attributes name, price, and quantity. Handle exceptions when setting price (must be positive) and use pretty printing to display multiple products.

Code ::

Output ::

```
Week - 5$ python3 09_product_DS.py
Week 5 - Experiment 9
Amit Singhal | 11614802722
[{'Name': 'Laptop', 'Price': 50000, 'Quantity': 2},
 {'Name': 'Mouse', 'Price': 1500, 'Quantity': 5},
 {'Name': 'Keyboard', 'Price': 2500, 'Quantity': 3}]
```

Experiment - 10

AIM :: Implement a function that takes a string as input, stores each unique character as a key in a dictionary with its frequency as the value, and pretty-prints the dictionary. Handle TypeError if input is not a string.

Code ::

Output ::

```
Week - 5$ python3 10_stringToDict.py
Week 5 - Experiment 10
Amit Singhal | 11614802722
{' ': 1, 'd': 1, 'e': 1, 'h': 1, 'l': 3, 'o': 2, 'r': 1, 'w': 1}
```

Experiment - 11

AIM :: Develop a Student class with attributes name and marks (list). Implement methods to calculate the average marks and format the output using pretty printing. Handle ValueError if marks contain invalid data.

Code ::

Output ::

```
Week - 5$ python3 11_student_DS.py
Week 5 - Experiment 11
Amit Singhal | 11614802722
[{'Average': 87.66666666666667, 'Marks': [85, 90, 88], 'Name': 'Alice'},
 {'Average': 80.0, 'Marks': [78, 80, 82], 'Name': 'Bob'},
 {'Average': 95.0, 'Marks': [92, 95, 98], 'Name': 'Charlie'}]
```

WEEK - 6

Experiment - 1

AIM :: Create a program that prompts the user for their name and age and prints a personalized message.

```
Week - 6$ python3 01_basic_prg.py
Week 6 - Experiment 1
Amit Singhal | 11614802722
Enter your name: Amit
Enter your age: 21
Hello Amit, you are 21 years old!
```

Experiment - 2

AIM :: Create a program that prompts the user for their age and tells them if they can vote in the next election.

```
Week - 6$ python3 02_voting.py
Week 6 - Experiment 2
Amit Singhal | 11614802722
Enter your age: 21
You are eligible to vote in the next election.
```

Experiment - 3

AIM :: Create a program that calculates the factorial of a number entered by the user using a loop.

```
Week - 6$ python3 03_factorial.py
Week 6 - Experiment 3
Amit Singhal | 11614802722
Enter a number: 7
The factorial of 7 is 5040
```

Experiment - 4

AIM :: Create a program that prompts the user for a list of numbers and then sorts them in ascending order.

```
Week - 6$ python3 04_sort_list.py
Week 6 - Experiment 4
Amit Singhal | 11614802722
Enter numbers separated by spaces: 87 56 93 12 76 90 56 34 79 99
Sorted numbers: [12, 34, 56, 56, 76, 79, 87, 90, 93, 99]
```

Experiment - 5

AIM :: Create a program that prompts the user for a string and then prints out the string reversed.

```
Week - 6$ python3 05_reverse_string.py
Week 6 - Experiment 5
Amit Singhal | 11614802722
Enter a string: helloworld
Reversed string: dlrowolleh
```

Experiment - 6

AIM :: Create a program that defines a function to calculate the area of a circle based on the radius entered by the user.

```
Week - 6$ python3 06_area_of_circle.py
Week 6 - Experiment 6
Amit Singhal | 11614802722
Enter the radius of the circle: 10
The area of the circle is 314.16
```

Experiment - 7

AIM :: Create a program that defines a class to represent a car and then creates an object of that class with specific attributes.

```
Week - 6$ python3 07_car_class.py
Week 6 - Experiment 7
Amit Singhal | 11614802722
Car: 2020 Toyota Camry
```

Experiment - 8

AIM :: Create a program that reads data from a file and writes it to another file in a different format.

```
Week - 6$ python3 08_read_write.py
Week 6 - Experiment 8
Amit Singhal | 11614802722
Data has been read from input.txt and written to output.txt in uppercase.
```

input.txt

```
hello world!
nice to meet you.
```

output.txt

```
HELLO WORLD!
NICE TO MEET YOU.
```

Experiment - 9

AIM :: Create a program that uses regular expressions to find all instances of a specific pattern in a text file.

sample.txt

```
cello
hello
fellow
mellow
pillow
jello
```

```
Week - 6$ python3 09_find_patterns.py
Week 6 - Experiment 9
Amit Singhal | 11614802722
Enter the pattern to search for: llow
Found 3 words containing the pattern: {'mellow', 'pillow', 'fellow'}
```

Experiment - 10

AIM :: Create a program that prompts the user for two numbers and then divides them, handling any exceptions that may arise.

```
Week - 6$ python3 10_exception_handling.py
Week 6 - Experiment 10
Amit Singhal | 11614802722
Enter first number: 72
Enter second number: 4
Result: 18.0
Enter first number: 67
Enter second number: 0
Error: Cannot divide by zero.
```

WEEK - 7

Experiment - 1

AIM :: Write a program to read a file and display its content.

sample.txt

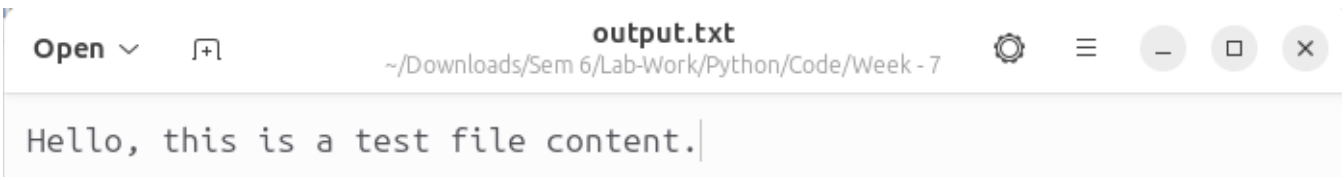
```
hello world!
nice to meet you.
```

```
Week - 7$ python3 01_reading_from_file.py
Week 7 - Experiment 1
Amit Singhal | 11614802722
File Content:
 hello world!
nice to meet you.
```

Experiment - 2

AIM :: Write a program to write content to a file.

```
Week - 7$ python3 02_writing_to_file.py
Week 7 - Experiment 2
Amit Singhal | 11614802722
Content written to output.txt
```



Experiment - 3

AIM :: Write a program to check if a file exists.

```
Week - 7$ python3 03_check_file_exists.py
Week 7 - Experiment 3
Amit Singhal | 11614802722
sample.txt exists.
```

Experiment - 4

AIM :: Write a program to save a list to a file using the pprint module.

```
Week - 7$ python3 04_saving_list.py
Week 7 - Experiment 4
Amit Singhal | 11614802722
List saved to output.txt
```

output.txt

```
['apple', 'banana', 'cherry', 'date']
```

Experiment - 5

AIM :: Write a program to read a file line by line and print each line.

sample.txt

```
hello world!
nice to meet you.
```

```
Week - 7$ python3 05_reading_line_by_line.py
Week 7 - Experiment 5
Amit Singhal | 11614802722
hello world!
nice to meet you.
```

Experiment - 6

AIM :: Write a program to combine the contents of two text files into one.

sample.txt

```
hello world!
nice to meet you.
```

output.txt

```
['apple', 'banana', 'cherry', 'date']
```

combined.txt

```
hello world!
nice to meet you.

['apple', 'banana', 'cherry', 'date']
```

```
Week - 7$ python3 06_combining_multiple_files.py
Week 7 - Experiment 6
Amit Singhal | 11614802722
Contents of sample.txt and output.txt combined into combined.txt
```

Experiment - 7

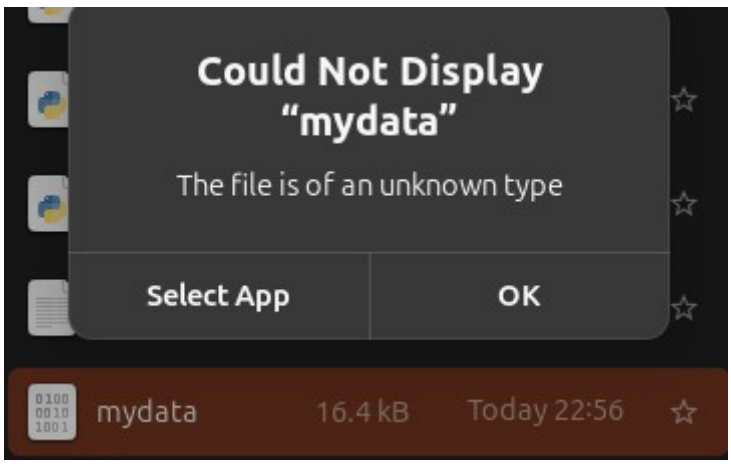
AIM :: Write a program that finds the absolute file path using os.path.

```
Week - 7$ python3 07_finding_file_path.py
Week 7 - Experiment 7
Amit Singhal | 11614802722
Absolute path of sample.txt: /home/singhal-amit/Downloads/Sem 6/Lab-Work/Python/Code/Week - 7/sample.txt
```

Experiment - 8

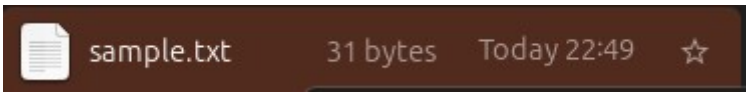
AIM :: Write a program to save a dictionary using the shelve module and then read it back.

```
Week - 7$ python3 08_saving_dictionary.py
Week 7 - Experiment 8
Amit Singhal | 11614802722
Stored Data: {'name': 'Alice', 'age': 30, 'city': 'New York'}
```

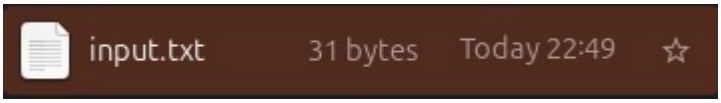


Experiment - 9

AIM :: Write a program that renames a file using the os module.




```
Week - 7$ python3 09_renaming_file.py
Week 7 - Experiment 9
Amit Singhal | 11614802722
Renamed sample.txt to input.txt
```



Experiment - 10

AIM :: Write a program to read from and write to a CSV file using Python's csv module.

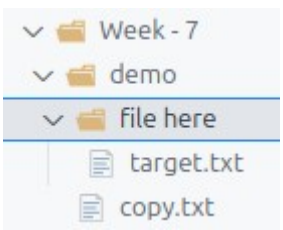
```
Week - 7$ python3 10_read_write_csv.py
Week 7 - Experiment 10
Amit Singhal | 11614802722
['Name', 'Age', 'City']
['Alice', '30', 'New York']
['Bob', '25', 'Los Angeles']
```

data.csv

```
1  Name,Age,City
2  Alice,30,New York
3  Bob,25,Los Angeles
```

Experiment - 11

AIM :: Write a program to search for a file in a directory and all its subdirectories using recursion.



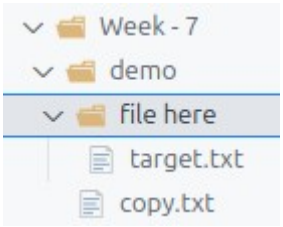

```
Week - 7$ python3 11_recursive_file_search.py
Week 7 - Experiment 11
Amit Singhal | 11614802722
./demo/file here/target.txt
```

Experiment - 12

AIM :: Write a program that copies a file from one location to another, including directories.

sample.txt

```
hello world!
nice to meet you.
```




```
Week - 7$ python3 12_copying_files.py
Week 7 - Experiment 12
Amit Singhal | 11614802722
Copied sample.txt to demo/copy.txt
```

Experiment - 13

AIM :: Write a program that appends data to a file and checks if the file exists using os.path.

sample.txt

```
hello world!
nice to meet you.
```

sample.txt

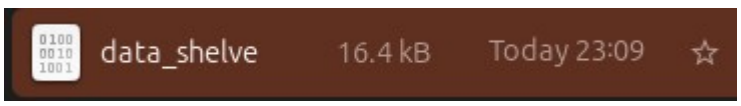
```
hello world!
nice to meet you.

Appended text.
```

```
Week - 7$ python3 13_appending_data.py
Week 7 - Experiment 13
Amit Singhal | 11614802722
Data appended to sample.txt
```

Experiment - 14

AIM :: Write a program to serialize and deserialize objects using the shelve module.

```
Week - 7$ python3 14_serialization_deserialization.py
Week 7 - Experiment 14
Amit Singhal | 11614802722
Stored Fruits: ['apple', 'banana', 'cherry']
```



Experiment - 15

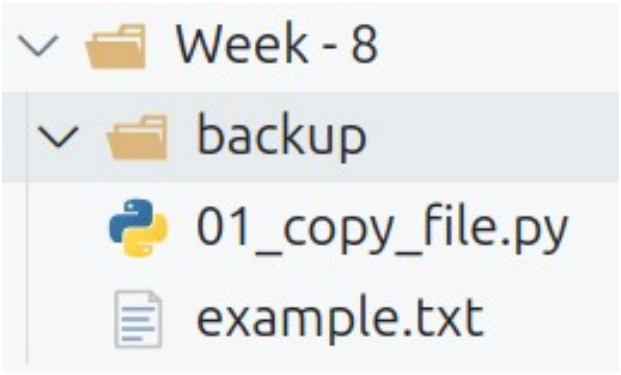
AIM :: Write a program that calculates and prints the size of a file using os.path and os module.

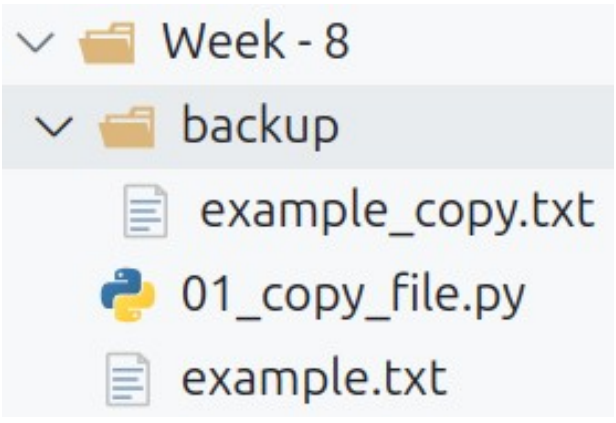
```
Week - 7$ python3 15_calculate_file_size.py
Week 7 - Experiment 15
Amit Singhal | 11614802722
Size of sample.txt: 46 bytes
```

WEEK - 8

Experiment - 1

AIM :: Write a Python program that copies a file from one directory to another using the shutil module.

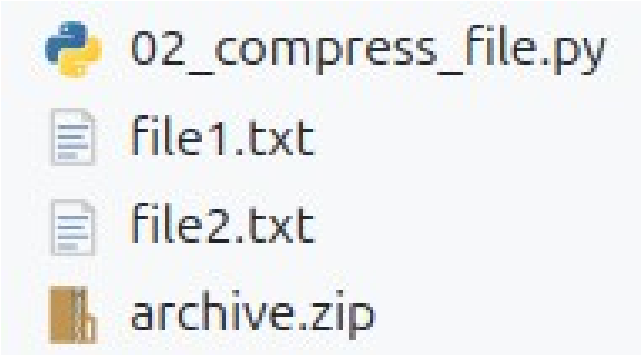





```
Week - 8$ python3 01_copy_file.py
Week 8 - Experiment 1
Amit Singhal | 11614802722
Copied example.txt to backup/example_copy.txt
```

Experiment - 2

AIM :: Write a Python script that compresses multiple files into a .zip archive using the zipfile module.




```
Week - 8$ python3 02_compress_file.py
Week 8 - Experiment 2
Amit Singhal | 11614802722
Files compressed into archive.zip
```

Experiment - 3

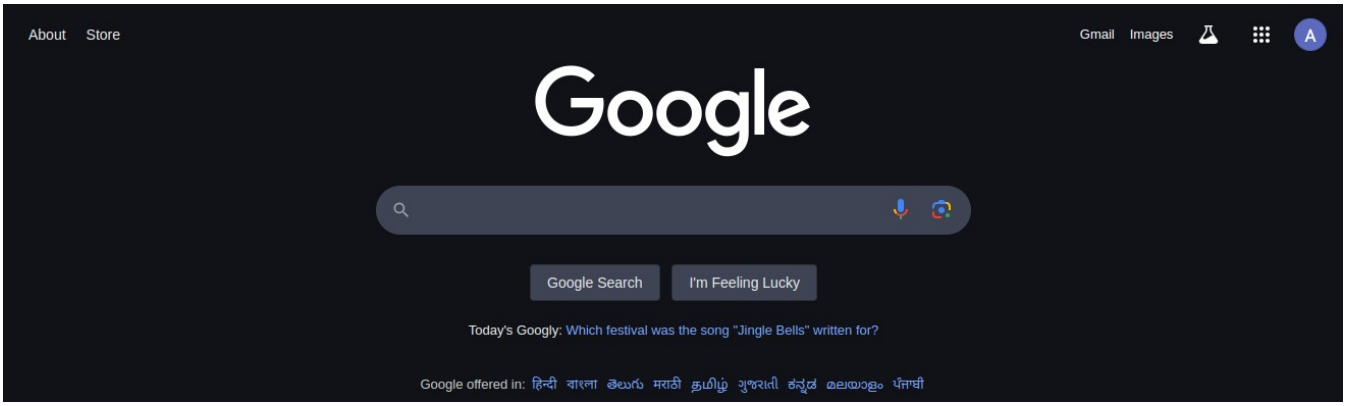
AIM :: Write a program to display the directory structure using os.walk().

```
Week - 8$ python3 03_directory.py
Week 8 - Experiment 3
Amit Singhal | 11614802722
Directory: .
  File: file2.txt
  File: 02_compress_file.py
  File: 03_directory.py
  File: file1.txt
  File: archive.zip
  File: 01_copy_file.py
Directory: ./backup
  File: example_copy.txt
```

Experiment - 4

AIM :: Use the webbrowser module to open a specific webpage in the default web browser.

```
Week - 8$ python3 04_openURL.py
Week 8 - Experiment 4
Amit Singhal | 11614802722
Week - 8$ Opening in existing browser session.
```



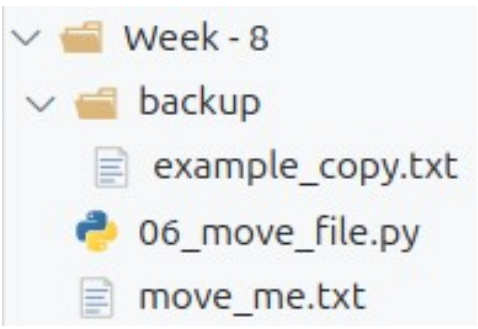
Experiment - 5

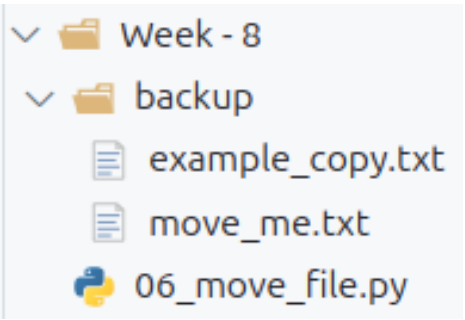
AIM :: Walk through a directory and list all .txt files using os.walk().

```
Week - 8$ python3 05_list_txt_files.py
Week 8 - Experiment 5
Amit Singhal | 11614802722
Text file found: ./file2.txt
Text file found: ./file1.txt
Text file found: ./backup/example_copy.txt
```

Experiment - 6

AIM :: Write a Python program that moves files from one folder to another using shutil.move().

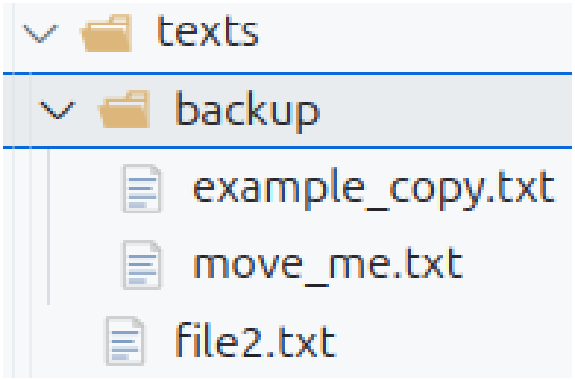





```
Week - 8$ python3 06_move_file.py
Week 8 - Experiment 6
Amit Singhal | 11614802722
Moved move_me.txt to backup/move_me.txt
```

Experiment - 7

AIM :: Compress all .txt files in a directory into one .zip archive using the zipfile module.




```
Week - 8$ python3 07_compress_files.py
Week 8 - Experiment 7
Amit Singhal | 11614802722
All .txt files compressed into texts.zip
```

Experiment - 8

AIM :: Write a Python program that creates a directory structure and writes a few files into the directories using os and shutil.


```
Week - 8$ python3 08_create_files.py
Week 8 - Experiment 8
Amit Singhal | 11614802722
Directory structure and file created.
```

Experiment - 9

AIM :: Use requests and BeautifulSoup to scrape data from a webpage and save the content to a file.

```
Week - 8$ python3 09_saving_webdata.py
Week 8 - Experiment 9
Amit Singhal | 11614802722
Webpage content saved to scraped.txt
```

scraped.txt

```
<!DOCTYPE html>
<html>
 <head>
  <title>
   Example Domain
  </title>
  <meta charset="utf-8"/>
  <meta content="text/html; charset=utf-8" http-equiv="Content-type"/>
  <meta content="width=device-width, initial-scale=1" name="viewport"/>
  <style type="text/css">
```

Experiment - 10

AIM :: Write a Python script to extract all hyperlinks from a webpage and save them to a text file.


```
Week - 8$ python3 10_extract_links.py
Week 8 - Experiment 10
Amit Singhal | 11614802722
Hyperlinks saved to links.txt
```

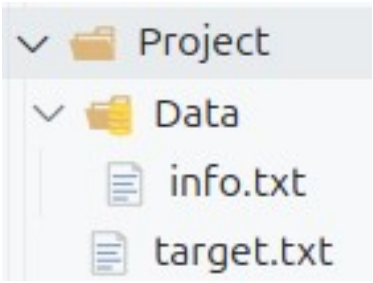
Experiment - 11

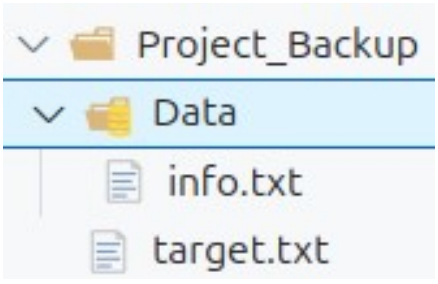
AIM :: Write a program to recursively search for a file in a directory and all its subdirectories using recursion.

```
Week - 8$ python3 11_recursive_file_search.py
Week 8 - Experiment 11
Amit Singhal | 11614802722
./Project/Data/target.txt
```

Experiment - 12

AIM :: Write a Python program to copy an entire directory along with its files to another location using shutil.

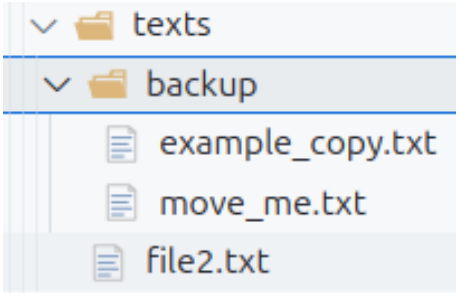


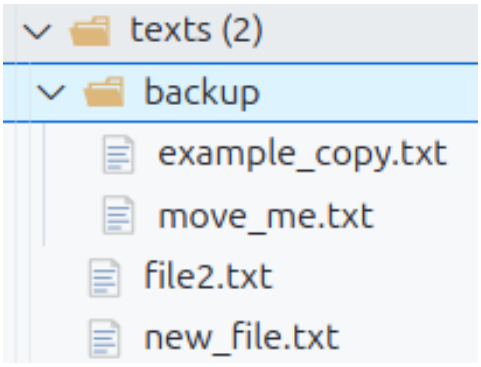



```
Week - 8$ python3 12_copy_directory.py
Week 8 - Experiment 12
Amit Singhal | 11614802722
Copied entire directory Project to Project_Backup
```

Experiment - 13

AIM :: Write a Python script to append new files to an existing .zip archive using the zipfile module.






```
Week - 8$ python3 13_append_data.py
Week 8 - Experiment 13
Amit Singhal | 11614802722
Appended new_file.txt to texts.zip
```

Experiment - 14

AIM :: Write a program to scrape data from a website with multiple pages and save all the scraped data to a file.

paged_data.txt

1408
1409
1410

```
Week - 8$ python3 14_web_scraping.py
Week 8 - Experiment 14
Amit Singhal | 11614802722
Scraped paginated data saved to paged_data.txt
```

Add topic

Experiment - 15

AIM :: Write a Python program to compress all .txt files from multiple directories into separate .zip archives.

```
Week - 8$ python3 15_compress_txt_files.py
Week 8 - Experiment 15
Amit Singhal | 11614802722
Created archive: ._archive.zip
Created archive: texts (2)_archive.zip
Created archive: backup_archive.zip
Created archive: texts_archive.zip
Created archive: backup_archive.zip
Created archive: backup_archive.zip
Created archive: Project_archive.zip
Created archive: Data_archive.zip
Created archive: Project_Backup_archive.zip
Created archive: Data_archive.zip
```