PYTHON LAB CYCLE

Muhammad Anshad P A

MCA2336

1. Python Programming

1.1. Create a simple calculator in Python.

PROGRAM:

```
#Simple Calculator

num1 = float(input('Enter the first Number : '))
num2 = float(input('Enter the second Number : '))
op = input('Select the operator : \n+\n-\n*\n/\n')

if op == '+' :
    sum = num1 + num2
    print('Sum of two numbers is ',sum)
elif op == '-' :
    diff = num1 - num2
    print('Difference of two numbers is ',diff)
elif op == '*' :
    prod = num1 * num2
    print('Product of two numbers is : ',prod)
elif op == '/' :
    if num2 == 0 :
        print('Division by zero Not possible')
    div = num1 / num2
    print('Division : ',div)
else :
    print('Invalid input')
```

OUTPUT:

```
PS E:\MUHAMMAD_ANSHAD_P_A\SEM_3\Python\Python_lab> & C:/Users/ccl26/AppData/Local/Programs/Python
ycle/p1_1_calculator.py
Enter the first Number : 5
Enter the second Number : 3
Select the operator : ( + , - , * , /  )

*
Product of two numbers is :  15.0
PS E:\MUHAMMAD_ANSHAD_P_A\SEM_3\Python\Python_lab>
```

1.2. An electric power distribution company charges domestic customers as follows: Consumption unit Rate of charge:

1.2.1. 0-200 Rs. 0.50 per unit

1.2.2. 201-400 Rs. 0.65 per unit in excess of 200

1.2.3. 401-600 Rs 0.80 per unit excess of 400

1.2.4. 601 and above Rs 1.00per unit excess of 600

1.2.5. If the bill exceeds Rs. 400, then a surcharge of 15% will be charged,

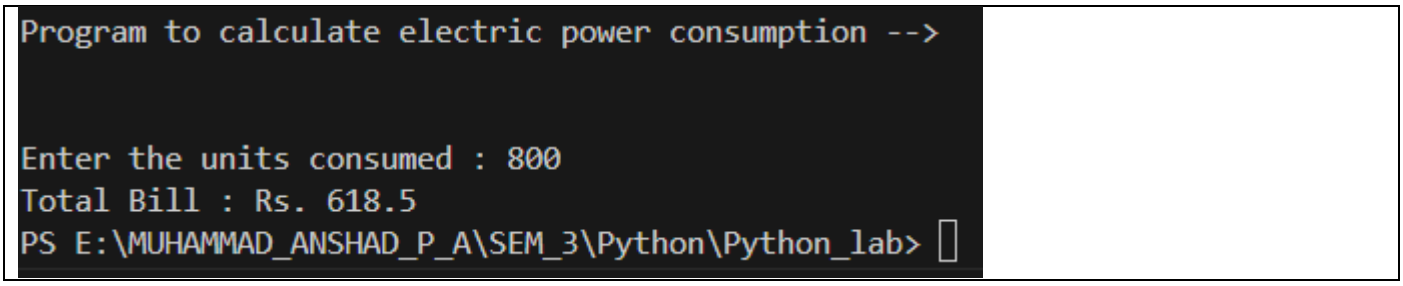
and the minimum bill should be Rs. 100/-

Create a Python program based on the scenario mentioned above.

PROGRAM:

```
#Function to calculate Bill
def billCalculator(cUnits):
    if cUnits <= 200:
        bill = cUnits * 0.50
    elif cUnits <= 400:
        bill = (200 * 0.50) + (cUnits -200) * 0.65
    elif cUnits <= 600:
        bill = (200 * 0.50) + (200 * 0.65) + (cUnits - 400) * 0.80
    else:
        bill = (200 * 0.50) + (200 * 0.65) + (200 * 0.80) + (cUnits - 600) * 1.00

    #Add surcharge to bill if bill amount > 400
    if bill > 400 :
        surCharge = (bill - 400) * 0.15
        bill += surCharge

    #If bill < 100 Make bill as 100
    if bill < 100 :
        bill = 100

    return bill
print("\nProgram to calculate electric power consumption -->\n")

consumedUnits = float(input("\nEnter the units consumed : "))

totalBill = billCalculator(consumedUnits)
print(f"Total Bill : Rs. {totalBill}")
```

OUTPUT:


Program to calculate electric power consumption -->
Enter the units consumed : 800
Total Bill : Rs. 618.5
PS E:\MUHAMMAD_ANSHAD_P_A\SEM_3\Python\Python_lab>

1.3. Print the pyramid of numbers using for loops.

PROGRAM:

```
#Program to display number Pyramid
def numPyramid(n):
    for i in range(1 , n + 1):
        for j in range( n - i):
            print(" ",end ="") #To print leading spaces

        for j in range(1 , i+1):
            print(j ,end="") #printing num in ascending order

        for j in range (i - 1 ,0 , -1):
            print(j , end="") #print in desc order

        print() # To move to next line


n = int(input("Enter how many no.of rows to display : "))
numPyramid(n)
```

OUTPUT:

```
Enter how many no.of rows to display : 5
    1
   121
  12321
 1234321
123454321
PS E:\MUHAMMAD_ANSHAD_P_A\SEM_3\Python\Python_lab>
```

1.4. Write a program to find the number and sum of all integers greater than 100 and less than 200 that are divisible by 7.

PROGRAM:

```
#Program to find the number and
#Sum of all num > 100 and < 200 that are divisible by 7
def findNumAndSum():
    count = 0
    total_sum = 0

    for num in range(101, 200):
        if num % 7 == 0:
            count += 1
            total_sum += num

    return count, total_sum

count, totalSum = findNumAndSum()
print(f"The number of integers greater than 100 and less than 200 that are divisible by 7 is: {count}")
print(f"The sum of all these integers is: {totalSum}")
```

OUTPUT:

```
The number of integers greater than 100 and less than 200 that are divisible by 7 is: 14
The sum of all these integers is: 2107
PS E:\MUHAMMAD_ANSHAD_P_A\SEM_3\Python\Python_lab>
```

1.5. Write a recursive function to calculate the sum of numbers from 0 to 10

PROGRAM :

```python
#Program to calculate the sum of numbers from 0 to 10 using recursion
def recursiveSum(n):
    # Base case
    if n == 0:
        return 0
    else:
        # Recursive case
        return n + recursiveSum(n - 1)


result = recursiveSum(10)
print(f"The sum of numbers from 0 to 10 is: {result}")
```

OUTPUT:

```
The sum of numbers from 0 to 10 is: 55
PS E:\MUHAMMAD_ANSHAD_P_A\SEM_3\Python\Python_lab>
```

1.6. Write a Python program to reverse the digits of a given number and add them to the original. If the sum is not a palindrome, repeat this procedure.

PROGRAM :

```
#Write a Python program to reverse the digits of a given number and add them
#to the original. If the sum is not a palindrome, repeat this procedure.

def isPalindrome(n):
    original = n
    reverse = 0
    while n > 0:
        digit = n % 10
        reverse = reverse * 10 + digit
        n = n // 10
    return original == reverse

def reverseNumber(n):
    reverse = 0
    while n > 0:
        digit = n % 10
        reverse = reverse * 10 + digit
        n = n // 10 # It returns the quotient which is rounded down to the nearest
integer.
    return reverse

def reverseAndAddUntilPalindrome(n):
    while not isPalindrome(n):
        reversed_n = reverseNumber(n)
        n = n + reversed_n
        print(f"Reversed: {reversed_n}, Sum: {n}")
    return n

number = int(input("Enter a number: "))
result = reverseAndAddUntilPalindrome(number)
print(f"The resulting palindrome is: {result}")
```

OUTPUT :

```
Enter a number: 597
Reversed: 795, Sum: 1392
Reversed: 2931, Sum: 4323
Reversed: 3234, Sum: 7557
The resulting palindrome is: 7557
PS E:\MUHAMMAD_ANSHAD_P_A\SEM_3\Python\Python_lab>
```

1.7. Write a menu-driven program that performs the following operations on strings

1.7.1. Check if the String is a Substring of Another String

1.7.2. Count Occurrences of Character

1.7.3. Replace a substring with another substring

1.7.4. Convert to Capital Letters

PROGRAM :

```
#Write a menu-driven program that performs the following operations on
# strings
# 1. Check if the String is a Substring of Another String
# 2. Count Occurrences of Character
# 3. Replace a substring with another substring
# 4. Convert to Capital Letters

def checkSubstring():
    string = input("Enter the main string: ")
    substring = input("Enter the substring to check: ")

    if substring in string:
        print(f"'{substring}' is a substring of '{string}'")
    else:
        print(f"'{substring}' is not a substring of '{string}'")

def countOccurrences():
    string = input("Enter the string: ")
    char = input("Enter the character to count: ")

    count = string.count(char)
    print(f"Number of occurrences of '{char}' in '{string}': {count}")

def replaceSubstring():
    string = input("Enter the main string: ")
    old_substring = input("Enter the substring to replace: ")
    new_substring = input("Enter the new substring: ")

    new_string = string.replace(old_substring, new_substring)
    print(f"Modified string: '{new_string}'")

def convertToUpper():
    string = input("Enter the string to convert to uppercase: ")
    uppercase_string = string.upper()
    print(f"Uppercase string: '{uppercase_string}'")

# Main program
while True:
    print("\nMenu:")
    print("1. Check if String is a Substring of Another String")
    print("2. Count Occurrences of Character")
    print("3. Replace a substring with another substring")
    print("4. Convert to Capital Letters")
```

```
    print("5. Exit")

    choice = input("Enter your choice (1-5): ")

    if choice == '1':
        checkSubstring()
    elif choice == '2':
        countOccurrences()
    elif choice == '3':
        replaceSubstring()
    elif choice == '4':
        convertToUpper()
    elif choice == '5':
        print("Exiting the program...")
        break
    else:
        print("Invalid choice! Please enter a number from 1 to 5.")
```

OUTPUT :

```
Menu:
1. Check if String is a Substring of Another String
2. Count Occurrences of Character
3. Replace a substring with another substring
4. Convert to Capital Letters
5. Exit
Enter your choice (1-5): 1
Enter the main string: Anshad Muhammad
Enter the substring to check: Anshad
'Anshad' is a substring of 'Anshad Muhammad'
```

```
Menu:
1. Check if String is a Substring of Another String
2. Count Occurrences of Character
3. Replace a substring with another substring
4. Convert to Capital Letters
5. Exit
Enter your choice (1-5): 2
Enter the string: Anshad Muhammad
Enter the character to count: a
Number of occurrences of 'a' in 'Anshad Muhammad': 3
```

```
Menu:
1. Check if String is a Substring of Another String
2. Count Occurrences of Character
3. Replace a substring with another substring
4. Convert to Capital Letters
5. Exit
Enter your choice (1-5): 3
Enter the main string: Anshad Muhammad
Enter the substring to replace: Anshad
Enter the new substring: Nihal
Modified string: 'Nihal Muhammad'
```

```
Menu:
1. Check if String is a Substring of Another String
2. Count Occurrences of Character
3. Replace a substring with another substring
4. Convert to Capital Letters
5. Exit
Enter your choice (1-5): 4
Enter the string to convert to uppercase: Anshad Muhammad
Uppercase string: 'ANSHAD MUHAMMAD'
```

```
Menu:
1. Check if String is a Substring of Another String
2. Count Occurrences of Character
3. Replace a substring with another substring
4. Convert to Capital Letters
5. Exit
Enter your choice (1-5): 5
Exiting the program...
PS E:\MUHAMMAD_ANSHAD_P_A\SEM_3\Python\Python_lab>
```

1.8. Write a function to find the factorial of a number but also store the factorials calculated in a dictionary.

PROGRAM:

OUTPUT: