**PYTHON PROGRAM OUTPUTS**

* 1. Create a simple calculator in Python.

a = int(input('Enter the first number: '))

b = int(input('Enter the second number: '))

op = input('Select the operation [+, -, \*, / ] :')

if op == '+':

res = a + b

elif op == '-':

res = a - b

elif op == '\*':

res = a \* b

elif op == '/':

if b == 0:

print('Division by zero is not possible')

res = None

else:

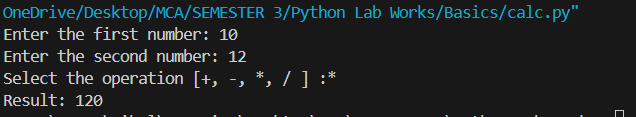
res = a / b

else:

print('Invalid operation')

res = None

if res is not None:

 print('Result:', res)

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.

def calc\_ebill(units):

if units <= 200:

bill = units \* 0.50

elif units <= 400:

bill = 200 \* 0.50 + (units - 200) \* 0.65

elif units <= 600:

bill = 200 \* 0.50 + 200 \* 0.65 + (units - 400) \* 0.80

else:

bill = 200 \* 0.50 + 200 \* 0.65 + 200 \* 0.80 + (units - 600) \* 1.00

if bill > 400:

bill += bill \* 0.15 #surcharge

if bill < 100:

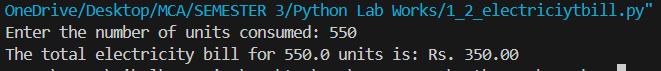
bill = 100 #min bill amt

return bill

units\_consumed = float(input("Enter the number of units consumed: "))

bill\_amount = calc\_ebill(units\_consumed)

print(f"The total electricity bill for {units\_consumed} units is: Rs. {bill\_amount:.2f}")



1.3. Print the pyramid of numbers using for loops.

def number\_pyramid(n):

for i in range(1, n + 1):

for j in range(n - i): # Print leading spaces

print(" ", end="")

for j in range(1, i + 1): # Print increasing numbers

print(j, end="")

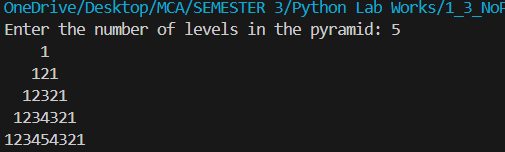
for j in range(i - 1, 0, -1): # Print decreasing numbers

print(j, end="")

print() # Move to the next line

levels = int(input("Enter the number of levels in the pyramid: "))

number\_pyramid(levels)



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.

count = 0

sum = 0

for i in range(101, 200):

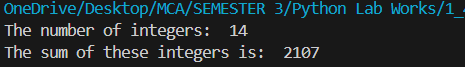
if i % 7 == 0:

count += 1

sum += i

print("The number of integers: ",count)

print("The sum of these integers is: ",sum)



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

def rec\_sum(n):

if n == 0:

return 0

else:

return n + rec\_sum(n - 1)

result = rec\_sum(10)

print("The sum of numbers from 0 to 10 is: ",result)



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.

def palindrome(n):

original = n

reversed\_number = 0

while n > 0:

digit = n % 10

reversed\_number = reversed\_number \* 10 + digit

n = n // 10 # // is used to round the result to the nearest

return original == reversed\_number

def rev\_add\_until\_palindrome(n):

while not palindrome(n):

reversed\_n = reverse\_number(n)

n = n + reversed\_n

return n

def reverse\_number(n):

reversed\_number = 0

while n > 0:

digit = n % 10

reversed\_number = reversed\_number \* 10 + digit

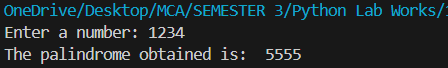
n = n // 10

return reversed\_number

number = int(input("Enter a number: "))

result = rev\_add\_until\_palindrome(number)

print("The palindrome obtained is: ",result)



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

def check\_substring(str1, str2):

if str1 in str2:

return True

else:

return False

def count\_occurrences(string, char):

count = 0

for c in string:

if c == char:

count += 1

return count

def replace\_substring(string, old\_substring, new\_substring):

return string.replace(old\_substring, new\_substring)

def convert\_to\_capital(string):

return string.upper()

def menu():

print("Menu:")

print("1. Check if String is 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): ")

return choice

while True:

choice = menu()

if choice == '1':

str1 = input("Enter the first string: ")

str2 = input("Enter the second string: ")

if check\_substring(str1, str2):

print(f"{str1} is a substring of {str2}")

else:

print(f"{str1} is not a substring of {str2}")

elif choice == '2':

string = input("Enter the string: ")

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

count = count\_occurrences(string, char)

print(f"The character '{char}' appears {count} times in '{string}'")

elif choice == '3':

string = input("Enter the string: ")

old\_substring = input("Enter the substring to replace: ")

new\_substring = input("Enter the new substring: ")

new\_string = replace\_substring(string, old\_substring, new\_substring)

print(f"Original string: '{string}'")

print(f"Modified string: '{new\_string}'")

elif choice == '4':

string = input("Enter the string: ")

capitalized\_string = convert\_to\_capital(string)

print(f"The string in capital letters: {capitalized\_string}")

elif choice == '5':

print("Exiting the program...")

break

else:

print("Invalid choice! Please enter a number from 1 to 5.")

