----------TASK-01----------

```
'''
1. Write a Python program that includes three separate functions to:
 Calculate the area of a trapezoid
 Calculate the area of a parallelogram
 Calculate the surface area and volume of a cylinder
 Each function should take the necessary arguments and if user select Area for
trapezoid then It should be print.

'''
from math import pi
def area_of_trapezoid(b1,b2,h):
    return (0.5*(b1+b2)*h)


def area_of_parallelogram(b,h):
    return (b*h)


def surface_area_of_cylinder(r,h):

    return (2*pi*(r**2)+2*pi*r*h)

def volume_of_cylinder(r,h):
    return (pi*(r**2)*h)

def input_taking_func():
    while True:
        print("\nSelect an option:")
        print("1. Calculate the area of a trapezoid")
        print("2. Calculate the area of a parallelogram")
        print("3. Calculate the surface area of a cylinder")
        print("4. Calculate the volume of a cylinder")
        print("5. Exit")

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

        if option == 1:
            base1 = float(input("Enter the value of Base-01: "))
            base2 = float(input("Enter the value of Base-02: "))
```

----------OUTPUT-01----------

```
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS> python -u "c:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS\ASSIGMENT WEEK 02\TASK_01.
py"

Select an option:
1. Calculate the area of a trapezoid
2. Calculate the area of a parallelogram
3. Calculate the surface area of a cylinder
4. Calculate the volume of a cylinder
5. Exit
Enter your choice (1-5): 4
Enter the Radius value: 34
Enter the Height value: 4
Volume of Cylinder: 14526.72

Select an option:
1. Calculate the area of a trapezoid
2. Calculate the area of a parallelogram
3. Calculate the surface area of a cylinder
4. Calculate the volume of a cylinder
5. Exit
Enter your choice (1-5): 5
Exiting the program. Goodbye!
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS>
```

**----------TASK-02----------**

ASSIGMENT WEEK 02 > TASK_02.py > ...

```python
'''
Write a Python function to check if the last letter of user input string is a vowel or a
consonant.

'''
def check(s):
    return "(TRUE) EXISTS" if s[-1] in [ 'a', 'e', 'i', 'o', 'u'] else "(FALSE) DOESN'T EXIST"


user_string = input("ENTER THE STRING TO CHECK: ")

print(check(user_string))

```

**----------OUTPUT-02----------**

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS                                   Code

PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS> python -u "c:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS\ASSIGMENT WEEK 02\TASK_02.
py"
ENTER THE STRING TO CHECK: hellow world
(FALSE) DOESN'T EXIST
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS>
```

**----------TASK-03----------**

ASSIGMENT WEEK 02 > TASK_03.py > ...

```python
'''
3. Write a program to make function employee() meeting following requirements:
Employee name and monthly salary will be passed to this function. This function will cut
2 percent tax from salary and display salary after tax with name of employee. If the
salary is missing in the function call then assign default value 10,000 to salary.
'''


def employee(name,salary =10000):
    return f"Employee: {name}, Salary after 2% tax: {(salary - salary*0.02):.2f}"




name = input("ENTER THE EMPLOYEE'S NAME: ")

salary_input = input("ENTER THE EMPLOYEE'S SALARY (leave blank for default 10,000): ")
if salary_input.strip():
    salary = int(salary_input)
else:
    salary = 10000

print(employee(name,salary))
```

----------OUTPUT-03----------

```
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS> python -u "c:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS\ASSIGMENT WEEK 02\TASK_03.
py"
ENTER THE EMPLOYEE'S NAME: Mudasir Abrar Baig
ENTER THE EMPLOYEE'S SALARY (leave blank for default 10,000):
Employee: Mudasir Abrar Baig, Salary after 2% tax: 9800.00
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS>
```

----------TASK-04----------

ASSIGMENT WEEK 02 > TASK_04.py > ...

```python
'''
4. Write a function that will take a number 'n' input by user and will return its factorial.

'''

def factorial(n):
    if n == 0:
        return 1
    else:
        return n*factorial(n-1)

num = int(input("ENTER THE NUMBER: "))
print(factorial(num))
```

----------OUTPUT-04----------

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS Code

```
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS> python -u "c:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS\ASSIGMENT WEEK 02\TASK_04.py"
ENTER THE NUMBER: 6
720
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS>
```

----------TASK-05----------

ASSIGMENT WEEK 02 > TASK_05.py > ...

```python
'''
Write a Python function to multiply all the numbers in a list.
'''

def multiply(l):
    multiply = 1
    for x in l:
        multiply*=x
    return multiply



user_list = map(int,input("ENTER A COMMA SEPERATED LIST: ").split(","))

print(multiply(user_list))
```

----------OUTPUT-05----------

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS Code

```
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS> python -u "c:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS\ASSIGMENT WEEK 02\TASK_05.py"
ENTER A COMMA SEPERATED LIST: 1,2,3,4,5,6,7,8,9
362880
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS>
```

----------TASK-06----------

ASSIGMENT WEEK 02 > TASK_06.py > ...

```python

'''
temperature_list= [22, 24, 30, 35, 29, 22, 20, 19]
average = (sum(temperature_list))/(len(temperature_list))

highest_temp = max(temperature_list)
lowest_temp = min(temperature_list)

soreted_list = sorted(temperature_list)

try:
    index_to_remove = int(input("ENTER THE INDEX/DAY (0-7) YOU WANT TO REMOVE FROM THE LIST: "))
    if 0 <= index_to_remove < len(temperature_list):
        removed_temp = temperature_list[index_to_remove]
        del temperature_list[index_to_remove]
        print(f"Removed temperature: {removed_temp}°C")
        print(f"LIST AFTER REMOVING {temperature_list}")
    else:
        print("Invalid index. Please enter a number between 0 and 7.")
except ValueError:
    print("Invalid input. Please enter a valid integer.")

print(f"THE AVERAGE TEMERATURE IS {average:.2f}")
print(f"HIGHEST TEMPERATURE IS {highest_temp}")
print(f"LOWEST  TEMPERATURE IS {lowest_temp}")
print(f"SORTED LIS IN ASCENDING ORDER IS {soreted_list}")

```

----------OUTPUT-06----------

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS> python -u "c:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS\ASSIGMENT WEEK 02\TASK_06.py"
ENTER THE INDEX/DAY (0-7) YOU WANT TO REMOVE FROM THE LIST: 6
Removed temperature: 20°C
LIST AFTER REMOVING [22, 24, 30, 35, 29, 22, 19]
THE AVERAGE TEMERATURE IS 25.12
HIGHEST TEMPERATURE IS 35
LOWEST  TEMPERATURE IS 19
SORTED LIS IN ASCENDING ORDER IS [19, 20, 22, 22, 24, 29, 30, 35]
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS>
```

----------TASK-07----------

```
ASSIGMENT WEEK 02 > TASK_07.py > ...
  print the result.
  Do not use any library. Make logic yourself.

  '''
  try:
      list01 = input("ENTER THE FIRST LIST(comma-separated): ").split(",")

      list02 = input("ENTER THE SECOND LIST(comma-separated): ").split(",")

      if len(list01)!=len(list02):
          raise ValueError("LISTS AREN'T OF SAME LENGTH")


      print("List are of same length")
  except ValueError as e:
      print(f"Error: {e}")
  except Exception as e:
      print(f"An unexpected error occurred: {e}")



  my_dict = {}
  for i in range(len(list01)):
      my_dict[list01[i]] = list02[i]

  print(my_dict)
```

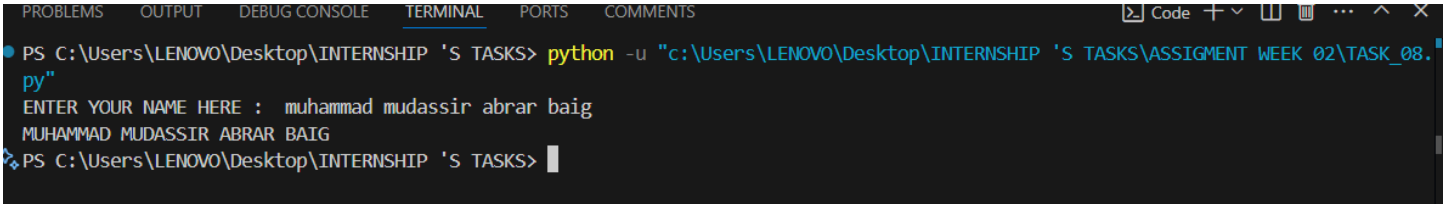
----------OUTPUT-07----------

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS                                Code

PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS> python -u "c:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS\ASSIGMENT WEEK 02\TASK_07.
py"
ENTER THE FIRST LIST(comma-separated): first,second,third
ENTER THE SECOND LIST(comma-separated): 1,2,3
List are of same length
{'first': '1', 'second': '2', 'third': '3'}
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS>
```

----------TASK-08----------

```
ASSIGMENT WEEK 02 > TASK_08.py > ...
  '''

  Write a function named Capital_Convertor() that get the name from user and return the
  name is in Capital/Block Letters.
  '''

  def Capital_Convertor(name):
      return name.upper()



  user_name = input("ENTER YOUR NAME HERE :  ")
  print(Capital_Convertor(user_name))
```

----------OUTPUT-08----------




----------TASK-09----------

```
'''

9. Why do we use functions in Python? Explain at least three advantages of using functions
in programming with examples. (Not required coding, theory question)

'''



#1st advantage: Functions increase the reusibilty of the program.
#2nd advantage: Functions increase the scablibility of the program.
#3rd advantage: Functions make the code consise , clear and easy to manage.
#4th advantage: Functions make the code modular.
```

----------OUTPUT-09----------

```
'''

9. Why do we use functions in Python? Explain at least three advantages of using functions
in programming with examples. (Not required coding, theory question)

'''



#1st advantage: Functions increase the reusibilty of the program.
#2nd advantage: Functions increase the scablibility of the program.
#3rd advantage: Functions make the code consise , clear and easy to manage.
#4th advantage: Functions make the code modular.
```

----------TASK-10----------

ASSIGMENT WEEK 02 > TASK_10.py > ...

```python
'''
10. An online shopping platform wants to calculate the total bill for a customer, including a
discount. The discount is applied based on the total purchase amount.
If the total amount is above $500, a 10% discount is applied.
If the total amount is above $1000, a 20% discount is applied.
Write a Python function that takes the total purchase amount as an argument and
returns the final bill after applying the discount. Explain why using a function makes the
code more efficient.
'''
def functions_role():
    print("Functions make the code more reuseable , modular , efficient , easy to handle and consise")
def bill_generator(amount):
    try:
        if amount < 0:
            raise ValueError("AMOUNT CAN'T BE NEGATIVE")
        elif amount <= 500:
            return f"Sorry! You can't get a discount at ${amount}. Shop more to qualify for discounts.
        elif 500 < amount <= 1000:
            return f"The final bill after a 10% discount is: ${amount - (amount * 0.1):.2f}"
        else:
            return f"The final bill after a 20% discount is: ${amount - (amount * 0.2):.2f}"


    except ValueError as e:
        print(f"ERROR: {e}")



total_amount = int(input("ENTER THE TOTAL PURCHASE AMOUNT:  "))


print(bill_generator(total_amount))
print("<------------------------------------------------------------------------>")
functions_role()
```

**----------OUTPUT-10----------**

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS                                   Code

PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS> python -u "c:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS\ASSIGMENT WEEK 02\TASK_10.py"
ENTER THE TOTAL PURCHASE AMOUNT:  345
Sorry! You can't get a discount at $345. Shop more to qualify for discounts. ;)
<------------------------------------------------------------------------>
Functions make the code more reuseable , modular , efficient , easy to handle and consise
PS C:\Users\LENOVO\Desktop\INTERNSHIP 'S TASKS>
```