**CYCLE 5**

**PROGRAM 1**

**Aim** : Write a program to determine whether a given year is a leap year [ Use Calendar Module]

**Source code :**

import calendar

year=int(input("Enter the Year:"))

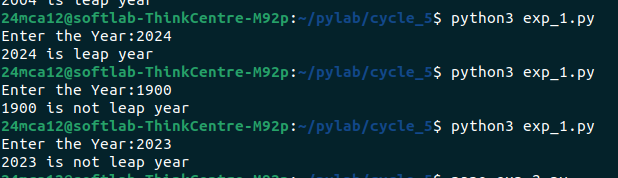
if(calendar.isleap(year)):

print(f"{year} is leap year")

else:

print(f"{year} is not leap year")

**Output :**

****

**PROGRAM 2**

**Aim :** Write a python script to display

a) Current date and time

b) Current Year

c) Month of the year

d) Week number of the year

e) Weekday of the week

f) Day of year

g) Day of the month

h) Day of week

[ Use time and datetime Module]

**Source code :**

from datetime import datetime,date

import calendar

current\_time=datetime.now()

today=datetime.now()

print(f"Current date and time : ",datetime.now())

print(f"Current year : ",current\_time.strftime("%Y"))

yr=date.today()

print(f"Month of the Year :")

print(calendar.month(yr.year,yr.month))

print(f"Week Number of the year :",current\_time.strftime("%W"))

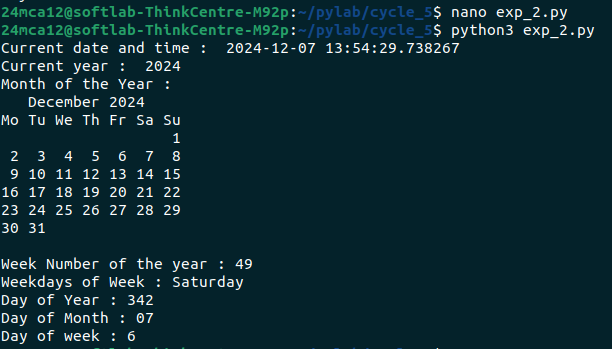
print(f"Weekdays of Week :",current\_time.strftime("%A"))

print(f"Day of Year :",current\_time.strftime("%j"))

print(f"Day of Month :",current\_time.strftime("%d"))

print(f"Day of week :",current\_time.strftime("%w"))

**Output**



**PROGRAM 3**

**Aim :** Write a python program to print yesterday, today and tomorrow.

**Source code :**

import calendar

from datetime import date,timedelta

today=date.today()

yesterday=today-timedelta(days=1)

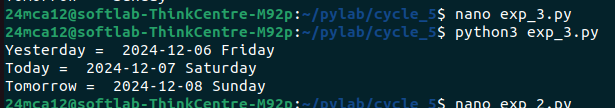
tomorrow=today+timedelta(days=1)

print("Yesterday=",yesterday.strftime('%Y-%d'),calendar.day\_name[yesterday.weekday()])

print("Today=",today.strftime('%Y-%m-%d'),calendar.day\_name[today.weekday()])

print("Tomorrow=",tomorrow.strftime('%Y-%d'),calendar.day\_name[tomorrow.weekday()])

**Output :**

****

**PROGRAM 4**

**Aim :** Write a function in file palindrome.py to check whether a string is Palindrome or not. Import the module to find the longest palindromic substring in a given string by checking every possible substring and verifying if it is a palindrome

**Source code :**

import palindrome

def longest\_palindromic\_substring(s: str) -> str:

longest = ""

for i in range(len(s)):

for j in range(i + 1, len(s) + 1):

substring = s[i:j]

if palindrome.is\_palindrome(substring) and len(substring) > len(longest):

longest = substring

return longest

input\_string = input("Enter a string: ")

result = longest\_palindromic\_substring(input\_string)

print(f"The longest palindromic substring is: {result}")

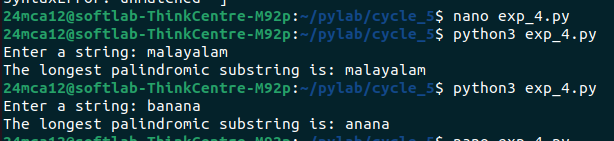
**palindrome.py**

def is\_palindrome(s: str) -> bool:

s = s.replace(" ", "").lower()

return s == s[::-1]

**Output :**

****

**PROGRAM 5**

**Aim :** Create a package graphics with modules rectangle, circle and sub-package 3D- graphics with modules cuboid and sphere. Include methods to find area and perimeter of respective figures in each module. Write programs that find the area and perimeter of figures by different importing statements. (Include selective import of modules and import \* statements)

**Source code :**

import graphics

import graphics.rectangle

import graphics.circle

from graphics.three\_d\_graphics.cuboid import cuboid\_area,cuboid\_volume

from graphics.three\_d\_graphics.sphere import \*

length=float(input("Enter Length of Rectangle:"))

width=float(input("Enter Width of Rectangle:"))

print("Rectangle Area :",graphics.rectangle.area(length,width))

print("Rectangle Perimeter :",graphics.rectangle.perimeter(length,width))

print()

radius=float(input("Enter Radius of Circle:"))

print("Circle Area :",graphics.circle.area(radius))

print("Circle Perimeter :",graphics.circle.perimeter(radius))

print()

length=float(input("Enter Length of Cuboid:"))

width=float(input("Enter Width of Cuboid:"))

height=float(input("Enter Height of Cuboid:"))

print("Cuboid Area :",cuboid\_area(length,width,height))

print("Cuboid Volume :",cuboid\_volume(length,width,height))

print()

radius=float(input("Enter Radius of Sphere:"))

print("Sphere Area :",area(radius))

print("Sphere Perimeter :",volume(radius))

**circle.py**

import math

def area(radius):

"""Calculate the area of a circle."""

return math.pi \* radius \*\* 2

def perimeter(radius):

"""Calculate the perimeter (circumference) of a circle."""

return 2 \* math.pi \* radius

**rectangle.py**

def area(length, width):

"""Calculate the area of a rectangle."""

return length \* width

def perimeter(length, width):

"""Calculate the perimeter of a rectangle."""

return 2 \* (length + width)

**sphere.py**

import math

def area(radius):

"""Calculate the surface area of a sphere."""

return 4 \* math.pi \* radius \*\* 2

def volume(radius):

"""Calculate the volume of a sphere."""

return (4/3) \* math.pi \* radius \*\* 3

**cuboid.py**

def cuboid\_area(length, width, height):

"""Calculate the surface area of a cuboid."""

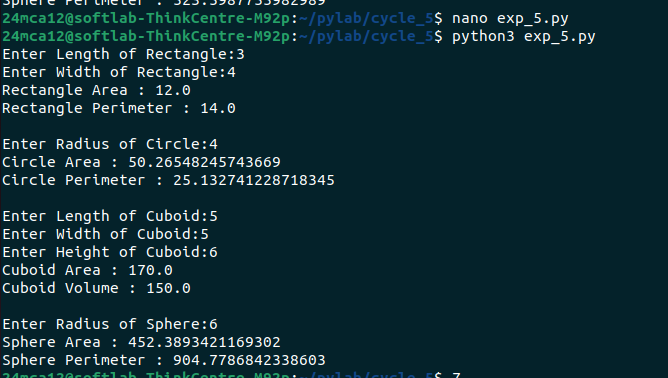
return 2 \* (length \* width + width \* height + height \* length)

def cuboid\_volume(length, width, height):

"""Calculate the volume of a cuboid."""

return length \* width \* height

**Output :**

****