Project #17

Name: Dipali Kunwar

Reg. No.: 12210828

Roll No.: B53

CA: 1

Your task is to create a functionality in which user will input a range of two dates. Then your module will find and print all years in the range of given dates those are leap years separatly and rest of the years those are non-leap separatly.

For Example:

Input date range in the format **dd/mm/yyyy**

(12/01/2000) to (13/12/2048)

Leaps years are:

2000 2004 2008 2012 2016 2020 2024 2028 2032 2036 2040 2044 2048

Non leap yars are:

001 2002 2003 2005 2006 2007 2009 2010 2011 2013 2014 2015 2017 2018 2019 2021 2022 2023 2025 2026 2027 2029 2030 2031 2033 2034 2035 2037 2038 2039 2041 2042 2043 2045 2046 2047026 2027 2029 2030 2031 2033 2034 2035 2037 2038 2039 2041 2042 2043 2045 2046 2047

__(Student is fre to decide the input and output layout for this mini project)__

Solution

Step - 1

Define a function with argument as year1, year2

```
def leapyear(year1, year1):
    pass
```

Step - 2

Initializing an empty list for **leap year** and **non leap year**

```
leap_year = []
    non_leap_year = []
```

Step - 3

Iterating through for loop from year1 to year2

checking the condtion for being a leap year

```
for i in range(n1, n2+1):
        if(i % 4 == 0 and i % 100 != 0 or i % 400 == 0):
            leap_year.append(i)      # appending in leap_year list
        else:
            non_leap_year.append(i)  # appending in non_leap_year list
```

Step - 4

To print leap year and non leapp year in seprate line

- In First iteration iterating through leap_year_list to print leap years
- In Second iterating through non_leap_year_list to print non leap years

```
print('leap years are: ')
    for i in leap_year:    # iterating through list directly
        print(i, end = ' ')
    print()
    print("non leap years are: ")
    for j in range(len(non_leap_year)):     # iterating through list using
range
        print(non_leap_year[j], end = ' ')
```

Step - 5

Take input from user to pass argument in the function

```
d1 = [int(x) for x in input("Enter date1 as dd/mm/yyyy: ").split('/')]
n1 = d1[2]    # taking year for argument1 in function
d2 = [int(x) for x in input("Enter date2 as dd/mm/yyyy: ").split('/')]
n2 = d2[2]    # taking year for argument2 in function
```

Here is some special functionalties to input from user:

- if year of date 1 greater than date 2 then:

Swap the values using swap two number

```
if n1 > n2:
    n1, n2 = n2, n1
```

Step 6

Calling the above function and pass given parameters

```
leapyear(n1, n2)
```

Full Code without using any in-built module

In [ ]:
```
# define a function with argument as year1, year2
def leapyear(n1,n2):

    leap_year = []
    non_leap_year = []
    for i in range(n1, n2+1):

        # condition for being a leap year
        if(i % 4 == 0 and i % 100 != 0 or i % 400 == 0):
            leap_year.append(i)
        else:
            non_leap_year.append(i)
```

```
    print('leap years are: ')
    for i in leap_year:    # iterating through list directly
        print(i, end = ' ')
    print()   # to print a new line between leap years and non leap years

    print("non leap years are: ")
    for j in range(len(non_leap_year)):     # iterating through list using range
        print(non_leap_year[j], end = ' ')

d1 = [int(x) for x in input("Enter date1 as dd/mm/yyyy: ").split('/')]
n1 = d1[2]    # taking year for argument1 in function
d2 = [int(x) for x in input("Enter date2 as dd/mm/yyyy: ").split('/')]
n2 = d2[2]    # taking year for argument2 in function
print(f'Start Date: {d1}')
print(f'End Date: {d2}')
if n1 > n2:
    n1, n2 = n2, n1
leapyear(n1, n2)   # calling function
```

```
Start Date: [1, 1, 2000]
End Date: [13, 12, 2048]
leap years are:
2000 2004 2008 2012 2016 2020 2024 2028 2032 2036 2040 2044 2048
non leap years are:
2001 2002 2003 2005 2006 2007 2009 2010 2011 2013 2014 2015 2017 2018 2019 2021 2022
2023 2025 2026 2027 2029 2030 2031 2033 2034 2035 2037 2038 2039 2041 2042 2043 2045
2046 2047
```

Full Code using the in-built module **datetimee**

In [ ]:

```
### import module
from datetime import datetime

# define a function with argument as year1, year2
def leapyear(n1,n2):

    leap_year = []       # initializing an empty list for leap year
    non_leap_year = []   # initializing an empty list for non leap year

    # iterating through for loop from year1 to year2
    for i in range(n1, n2+1):

        # condition for being a leap year
        if(i % 4 == 0 and i % 100 != 0 or i % 400 == 0):
            leap_year.append(i)      # appending in leap_year list
        else:
            non_leap_year.append(i)  # appending in non_leap_year list

    print('leap years are: ')
    # iterating through leap_year_list to print leap years
    for i in leap_year:    # iterating through list directly
        print(i, end = ' ')

    print()   # to print a new line between leap years and non leap years

    print("non leap years are: ")
    # iterating through leap_year_list to print non leap years
    for j in range(len(non_leap_year)):     # iterating through list using range
        print(non_leap_year[j], end = ' ')

d1 = input('Enter date1 as dd-mm-yyyy: ')
d2 = input('Enter date2 as dd-mm-yyyy: ')

date1 = datetime.strptime(d1, "%d-%m-%Y")
date2 = datetime.strptime(d2, "%d-%m-%Y")
print(f'Start Date: {date1}')
print(f'End Date: {date2}')
year1 = date1.year
year2 = date2.year

# if year of date 1 greater than date 2 then swap the year
if year1 > year2:
    year1, year2 = year2, year1
leapyear(year1, year2)   # calling function
```

```
Start Date: 2000-01-01 00:00:00
End Date: 2048-12-13 00:00:00
leap years are:
2000 2004 2008 2012 2016 2020 2024 2028 2032 2036 2040 2044 2048
non leap years are:
2001 2002 2003 2005 2006 2007 2009 2010 2011 2013 2014 2015 2017 2018 2019 2021 2022
2023 2025 2026 2027 2029 2030 2031 2033 2034 2035 2037 2038 2039 2041 2042 2043 2045
2046 2047
```