

PROFESSIONAL TRAINING REPORT

at

**Satyabhama Institute of Science and Technology (Deemed
to be University)**

Submitted in partial fulfillment of the requirements for the award of Bachelor of
Technology Degree in Information Technology

By

JOHN WILIFRED DAVID J

(Reg. No:- 38120034)



**DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU**

OCTOBER 2020



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)



Accredited with Grade “A” by NAAC
(Established under Section 3 of UGC Act, 1956)
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI– 600119
www.sathyabamauniversity.ac.in

SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **JOHN WILIFRED DAVID J** who carried out the project entitled “**DIGITAL CALENDER USING PYTHON**” under my supervision from August 2020 to January 2021.

Internal Guide

Dr.Mary Gladence.L M.E., Ph.D.,

Head of the Department

Dr. R.Subhashini M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **JOHN WILIFRED DAVID J** hereby declare that the Project Report entitled **DIGITAL CALENDER** done by me under the guidance **Dr. L. MARY GLADENCE M.E., PH.D.**, at Sathyabama institute of science and technology is submitted in partial fulfillment of the requirements for the award of Bachelor of technology degree in information technology.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D.**, Dean School of Computing and **Dr.R.SUBHASHINI M.E., Ph.D.**, Head of the Department of information technology for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.MARY GLADENCE.L M.E., PH.D.**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of INFORMATION TECHNOLOGY** who were helpful in many ways for the completion of the project.

COURSE CERTIFICATE



INTERNSHIP CERTIFICATE

**RETECH**
SOLUTIONS PVT. LTD.
As ISO 9001:2015 certified company

INT: RE20S030

TO WHOMSOEVER IT MAY CONCERN

This is to certify that **JOHN WILFRED DAVID J** student of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, BTECH/IT** has undergone the Internship Training through online in our concern entitled **PYTHON** from **12th Aug 2020 to 26th Aug 2020** in relevant departments related to their academic studies.

During the above period, the performance was good & we wish great success in all your future endeavours.

J. more

Technology drives the world. We are the people who are behind the technology.

Authorised Signature



No-31, 1st Floor, Alagesan Street, Tamboram West, Chennai-45.
Office : 044-42077204 | +91-8122266653 | www.retechsolutions.in

ABSTRACT

It may seem like a step back to add a paper planner to your daily routine, especially when so many of us have digital calendars that we can share with others, handle scheduling, and prod us into action with handy reminders. Our lives are busy and complicated, and online or digital platforms give us the kind of connectivity that other mediums simply can't.

- The previous projects are build using inbuilt modules in python to print a particular month of a calendar
- Here in this project we have used some code to print the months in a calendar without using the inbuilt modules of a python.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No
	ABSTRACT	i
	LIST OF FIGURES	ii
1	INTRODUCTION	1
	1.1 INTRODUCTION TO PYTHON	1
	1.2 LITERATURE REVIEW	2
2.	AIM AND SCOPE OF THE PRESENT INVESTIGATION	3
	2.1 OUTLINE OF THE PROJECT	3
	2.2 PROBLEM STATEMENT	3
3.	EXPERIMENTAL OR MATERIALS AND METHODS; ALGORITHMS USED	5
	3.1 GENERAL	5
	3.2 OVERVIEW	5
	3.3 MODULE DESIGN	7
4.	RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS	10
	4.1 RESULTS	10
5	SUMMARY AND CONCLUSIONS	11
	REFERENCES	12
	APPENDIX	13
	A.SCREENSHOTS	13
	B. SOURCE CODE	15

LIST OF FIGURES

FIGURE N.O	FIGURE NAME	PAGE N.O
3.1	IMPORTING MODULES	7
3.2	CAL() METHOD	8
3.3	YEARS() METHOD	8
3.4	MONTHS() METHOD	8
3.5	ANOTHER() METHOD	9
3.6	DANDT() METHOD	9
3.7	LEAP() METHOD	9
4.1	WHOLE YEAR CALENDER	10
4.2	PARTICULAR MONTH CALENDER	10
1	ENTERING THE OPTION	13
2	WHOLE YEAR CALENDER	13
3	CHOOSING THE OPTION	14
4	PARTICULAR MONTH CALENDER	14

CHAPTER1

INTRODUCTION

1.1 INTRODUCTION TO PYTHON

Calendar module in Python has the calendar class that allows the calculations for various tasks based on date, month, and year. This module allows you to output calendars like the Unix cal program, and provides additional useful functions related to the calendar. By default, these calendars have Monday as the first day of the week, and Sunday as the last .

In this project we have also used another two modules date time module and module in order to print the digital calendar that is present system date and time. These are the modules that we have used in this project.

The time module exposes C library functions for manipulating dates and times. Since it is tied to the underlying C implementation, some details (such as the start of the epoch and maximum date value supported) are platform specific.

GENERAL

An "implementation" of Python should be taken to mean a program or environment which provides support for the execution of programs written in the Python language, as represented by the [Python](#) reference implementation.

There have been and are several distinct software packages providing of what we all recognize as Python, although some of those are more like distributions or variants of some existing implementation than a completely new implementation of the language.

SOFTWARE IMPLEMENTATION

We have used python 3.5.2 where there is some difference from python 2.7. X and python 3.5.2 is the latest version of python programming and we have used

Python compiler to execute this program which is different from python interpreter.

HARDWARE IMPLEMENTATION

Memory required to save and execute the program is 4kb which has been saved with .py extension since it is a python program. RAM :1 GB

ROM :1 GB

1.2 LITERATURE REVIEW

The previous projects are build using inbuilt modules in python to print a particular month of the calendar and in this project we have used some code to print the months in a calendar without using the inbuilt modules of a python.

Calendar module in Python has the calendar class that allows the calculations for various tasks based on date, month, and year. This module allows you to output calendars like the Unix cal program, and provides additional useful functions related to the calendar. By default, these calendars have Monday as the first day of the week, and Sunday as the last .

The authors present the rationale, design, implementation and evaluation of a new digital composite calendaring service which is deployed in two empirical settings, namely group vacation planning and collective information management. Each case features different re-configurations of calendaring to serve human intentions. In vacation planning, the digital composite of the calendar operates as a mashup allowing peers to negotiate, schedule and track vacation options and archive, recall or retrieve digital memories of

vacations. In the case of collective information management, the digital composite is further augmented so as to re-align performative and ostensive aspects of routines in a regional organic farming partnership.

CHAPTER 2

AIM AND SCOPE OF PRESENT INVESTIGATION

2.1 OUTLINE OF THE PROJECT

We can use this program to print the calendar for the any year between 1-9999 years and also when we run the program it will first print the digital calendar that is present date and time of the system and it will ask for the options that whether we want to get the whole year calendar or any month calendar .After choosing an option based on the option it will ask the question.

For example after printing the digital calendar it will ask for the options from the below:

Do you want to get the whole year calendar or one month calendar in that year

1.enter 1 IF YOU WANT TO GET WHOLE YEAR CALENDAR

2.enter 2 IF YOU WANT TO GET ONE MONTH OF THAT CALENDAR YEAR

Then based on choosing the option it will connect the input from the user.

If we choose option 1 then it will ask to enter the year of the calendar which we want On entering the year it will print the whole year calendar of the corresponding input.

If we choose option 2 then it will ask to enter the year first and on entering it ,it will ask for the month in the number form for example if we want for February then we are going to enter it as 2. Then we will get the calendar of that month for the corresponding year chosen.

2.2 PROBLEM STATEMENT

Write a Python script to display the a)

Digital Calendar

b) To print the calendar for a particular month in a year

- c) To print the calendar for a particular year in between 1-9999 years
- d) To know whether the year is a leap year or not e)

To print the current date and time

The main aim of this project is to print the digital calendar that is current date and time and to print the calendar of any year for all months or for a particular month of any year between 1-9999 years without using some inbuilt modules. This is the main intention of the project .

In vacation planning, the digital composite of the calendar operates as a mashup allowing peers to negotiate, schedule and track vacation options and archive, recall or retrieve digital memories of vacations. In the case of collective information management, the digital composite is further augmented so as to re-align performative and ostensive aspects of routines in a regional organic farming partnership.

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently whereas other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted:** Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive:** You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented:** Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language:** Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

CHAPTER 3

ALGORITHMS AND METHODS

3.1 GENERAL

The methods we have used in this are **cal()** , **years()** , **months()** , **another()**, **dandt()** and **leap()** .

- i. The **cal()** method is used to select an option by the user that is whether the user wants to get the whole calender of the year or a particular month of the year ,based upon the option the years() and months() methods are called.
- ii. The **years()** method is used to print the calendar of the whole year (in between years 1-9999)which the user wants .If the user input is less than 1 or >9999 then it will display a message that the “please choose another year”.
- iii. The **months()** method is used to print the calendar of the particular month of the year which the user wants. If the user input is less than 1 or greater than 12 then it will ask to choose the correct month .
- iv. The **another()** method is used to ask the user whether they want to get another calendar if the answer is yes it will follow the same procedure again by calling the cal() method and if the answer is no then it will print Thank you for this calendar.
- v. The **dandt()** method is used to print the digital calendar .
- vi. The **leap()** method is used to print whether the user entered year is a leap year or not.

3.2 OVERVIEW

The project main aim is to print the calendar of any year or any month of any particular year of a calendar .

Algorithm

Step 1: Import the calendar module and then import the time module . Step

2: Then define all the methods

Step 2.1: Defining the **cal()** method:

Step 2.1.1: Take the input either 1 or 2 from the user where 1 or 2 option given user. Based upon the option call the years() or months() methods.

Step 2.2: Defining the **years ()** method:

Step 2.2.1: Take the input from the user that is the year for which the user wants calender .

Step 2.2.2: Check the user input lying between 1 and 9999 if it is true it will print the calendar for the whole year by using the function `calendar.month(y,i+1)` .

Step 2.2.3: If the user input not lies between 1 and 9999 it warns the user to enter the new input since the input given is not valid.

Step 2.3:Defining the **months()** method:

Step 2.3.1:Take the input from the user that is the year and month for which the user wants the calendar.

Step 2.3.2:Check the user input lying between 1 and 9999 if it is true then it will check another input that is month is lying between 1

and12 if it is also true it will print the calendar of that month.

Step 2.4:Defining the **another()** method:

Step 2.4.1:If the user input is “y” call the cal() method again otherwise stop the execution of a program.

Step 2.5: Defining **dandt()** method:

Step 2.5.1: localtime=time.asctime(time.localtime(time.time())) which is used to print the digital calendar.

Step 2.6:Defining **leap()** method:

Step 2.6.1:The input year is checked whether it is a leap year or not by that doing modulo operation with 4 and if it is equal to zero then we can say the year is a leap year

3.3 MODULE DESIGN

import ()

```
import calendar
from datetime import datetime
import time
```

FIG 3.1 IMPORTING MODULES

The import statement is used to import the packages or modules that are required For that program and we can import in two different types implicit import and explicit Import.Implicit import means importing all the functions and variables present in that Package whereas explicit import means importing only the specific functions required.

b) cal()

```
def cal():
    #choose year between 1 to 9999
    print("
    print("
    print("
    p=input()
    if p=="1":
        years()
    if p=="2":
        months()
```

Do you want to get whole year calendar or one month calendar in that year
enter 1 if you want to get whole year calendar
enter 2 if you want to get a particular month of the year

FIG 3.2 CAL() METHOD

The cal method is used to take the input either 1 or 2 from the user where 1 or 2 option given to the user based upon the option it will call the years() , months() Methods.

c) years()

```
def years():
    print("ENTER THE YEAR OF THE CALENDER WHICH DO YOU WANT")
    try:
        y=int(input())
        if 1>y or y>9999:
            print("PLEASE CHOOSE ANOTHER YEAR")
        else:
            print("
            for i in range (12):
                print(calendar.month(y,i+1))
            another()
    except:
        print("PLESE CHOOSE A YEAR OF CALENDER")
```

" + leap(y) + "

FIG 3.3 YEARS() METHOD

The years method takes the input from the user that is the year of the calendar for which the user wants the calendar If the present year is in the range 1-9999 it will print the whole calendar it will say to enter another year.

d) months()

```
def months():
    print("
    try:
        y=int(input("ENTER THE YEAR: "))
        m=int(input("ENTER THE MONTH IN NUMBER FORM: "))
        if 1>y or y>9999:
            print("PLEASE CHOOSE ANOTHER YEAR")
        if m<1 or m>12:
            print("PLEASE CHOOSE ANOTHER MONTH")
        else:
            for z in range (13):
                if (m==z):
                    print(calendar.month(y,z))
            another()
    except:
        print("PLEASE CHOOSE THE YEAR AND MONTH OF THE CALENDER CORRECTLY")
```

ENTER THE YEAR AND MONTH OF WHICH DO YOU WANT

FIG 3.4 MONTHS () METHOD

Check the user input lying between 1 and 12 if it is also true it will print the calendar of that month in that year.

e) another()

```
def another():
    print("DO YOU WANT TO GET ANOTHER CALENDAR ??? ENTER (y/n)"+
    ch=input()
    if ch=="y" or ch=="Y":
        cal()
    else :
        print("THANK YOU FOR USING THIS CALENDAR")
```

FIG 3.5 ANOTHER() METHOD

If the user input is “y” call the cal() method again otherwise stop the execution of a program the input y represents that the user wants another calendar for another year.

f) dandt()

```
#date and time function which is used to print the digital calendar
def dandt():
    #do you want to know the at present date and time
    localtime=time.asctime(time.localtime(time.time()))
    print("-----DIGITAL CALENDAR IS-----")
    print(" "+localtime+"")
    print("-----")
```

FIG 3.6 DANDT() METHOD

The dandt method is used to print the digital calendar that is current system's date and time.

g) leap()

```
def leap(y):
    year=y
    if (year % 4) == 0:
        if (year % 100) == 0:
            if (year % 400) == 0:
                return("{0} IS A LEAP YEAR".format(year))
            else:
                return("{0} IS NOT A LEAP YEAR".format(year))
        else:
            return("{0} IS A LEAP YEAR".format(year))
    else:
        return("{0} IS NOT A LEAP YEAR".format(year))
```

FIG 3.7 LEAP() METHOD

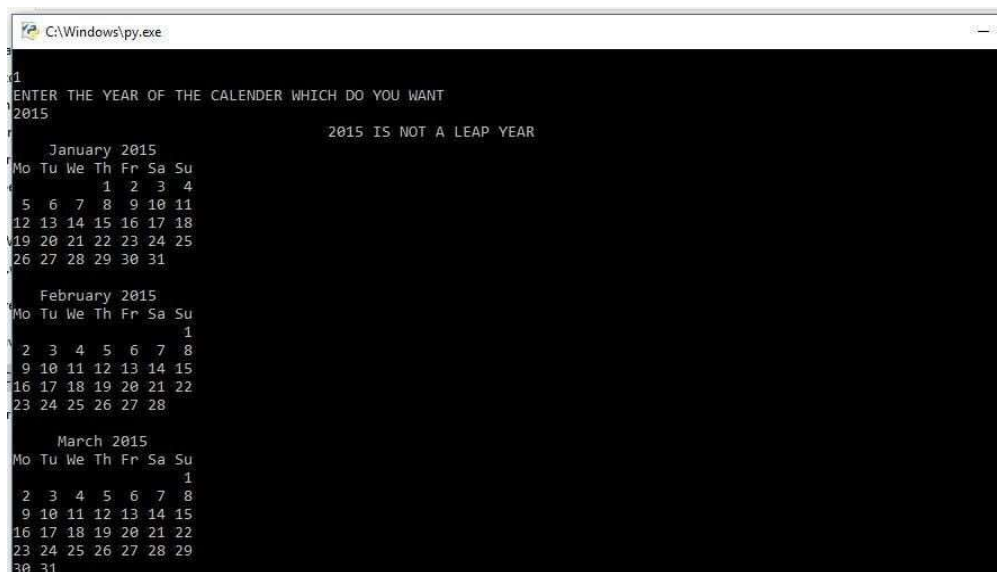
The leap method is used to print whether the entered year is a leap year or not

CHAPTER 4

RESULT AND DISCUSSION

4.1 RESULTS

The result of this program is going to be a calendar for the years in between 1-9999 years and it is also used to get the one month calendar of a particular year and we can also get a digital calendar that is the current system date and time along with the calendar it is also possible to know whether the user entered year is a leap year or not.



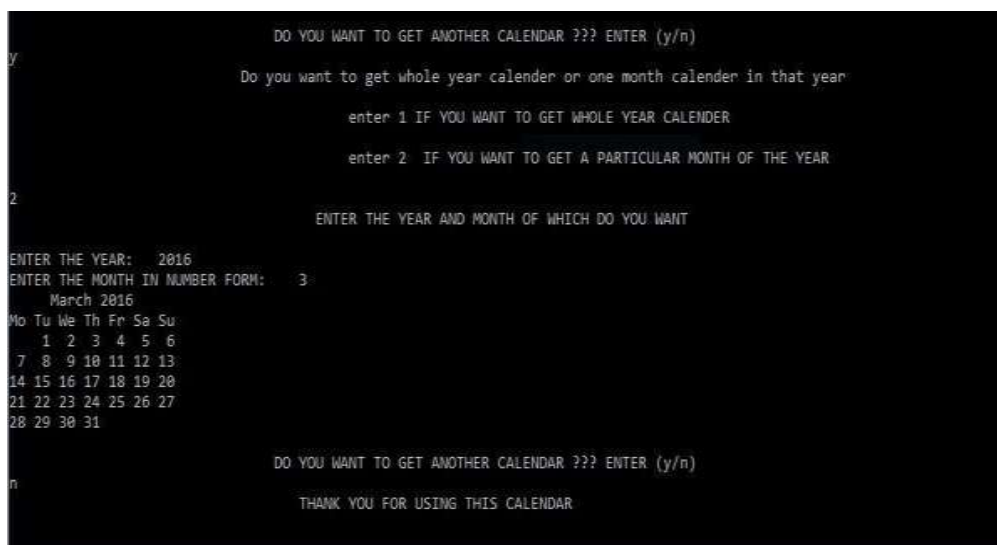
```
C:\Windows\py.exe
1
ENTER THE YEAR OF THE CALENDER WHICH DO YOU WANT
2015
2015 IS NOT A LEAP YEAR

January 2015
Mo Tu We Th Fr Sa Su
    1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

February 2015
Mo Tu We Th Fr Sa Su
    1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28

March 2015
Mo Tu We Th Fr Sa Su
    1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31
```

FIG 4.1 WHOLE YEAR CALENDAR



```
DO YOU WANT TO GET ANOTHER CALENDAR ??? ENTER (y/n)
y
Do you want to get whole year calendar or one month calendar in that year
enter 1 IF YOU WANT TO GET WHOLE YEAR CALENDAR
enter 2 IF YOU WANT TO GET A PARTICULAR MONTH OF THE YEAR
2
ENTER THE YEAR AND MONTH OF WHICH DO YOU WANT
ENTER THE YEAR: 2016
ENTER THE MONTH IN NUMBER FORM: 3
March 2016
Mo Tu We Th Fr Sa Su
 1  2  3  4  5  6
 7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31
DO YOU WANT TO GET ANOTHER CALENDAR ??? ENTER (y/n)
n
THANK YOU FOR USING THIS CALENDAR
```

FIG 4.2 PARTIULAR MONTH CALENDER

CHAPTER 5

SUMMARY AND CONCLUSION

5.1 SUMMARY

The trial has shown conclusively that it is both possible and desirable to use Python as the principal teaching language:

- 1.it is Free (as in both cost and source code).
- 2.it is trivial to install on a Windows PC allowing students to take their interest further. For many the hurdle of installing a Pascal or C compiler on a Windows machine is either too expensive or too complicated;
- 3.it is a flexible tool that allows both the teaching of traditional procedural programming and modern OOP; It can be used to teach a large number of transferable skills;
- 4.it is a real-world programming language that can be *and is* used in academia and the commercial world;
- 5.it appears to be quicker to learn and, in combination with its many libraries, this offers the possibility of more rapid student development allowing the course to be made more challenging and varied;
- 6.and most importantly, its clean syntax offers increased understanding and enjoyment for students
- 7.In conclusion, Python offers the optimum compromise of teachability and applicability.

5.2 FUTURE WORK

The future work for this project can be making it more useful by using html ,css and creating like a webpage or an app which is linked with a database so that we can add any things which we want to remember like birthdays or any events etc which can be more useful for the future references and also to get the calendar from particular date to any other particular date which is more useful to know the day of a week.

REFERENCES

- [1]. python, <https://www.python.org/> , Jan18,2013.
- [2]. Datetime, <https://stackoverflow.com/get-week-number-using-date-in-python>, Oct31,2017.
- [3]. Genrate digital calendar python, <https://stackoverflow.com/generate-calendar-python-webpage> , _sep13,2011.
- [4]. Yearly calendar, <https://stackoverflow.com/make-a-yearly-calendar-without-importing-a-calendar>, sep15,2015
- [5]. Dive into Python 3, Mark Pilgrim, <http://www.diveintopython3.net/>.
- [6]. Think python, 2nd Edition, Allen B.Downey, Algorithm Design, Jon Kleinb erg And Eva tardos, Pearson(2013).
- [7]. Calander module, <https://www.geeksforgeeks.org/python-calendar-module/amp/> .
- [8]. Quora, <https://www.quora.com>.
- [9]. Python html file, <https://docs.python.org/2/reference/index.html>.

APPENDIX

SCREENSHOTS

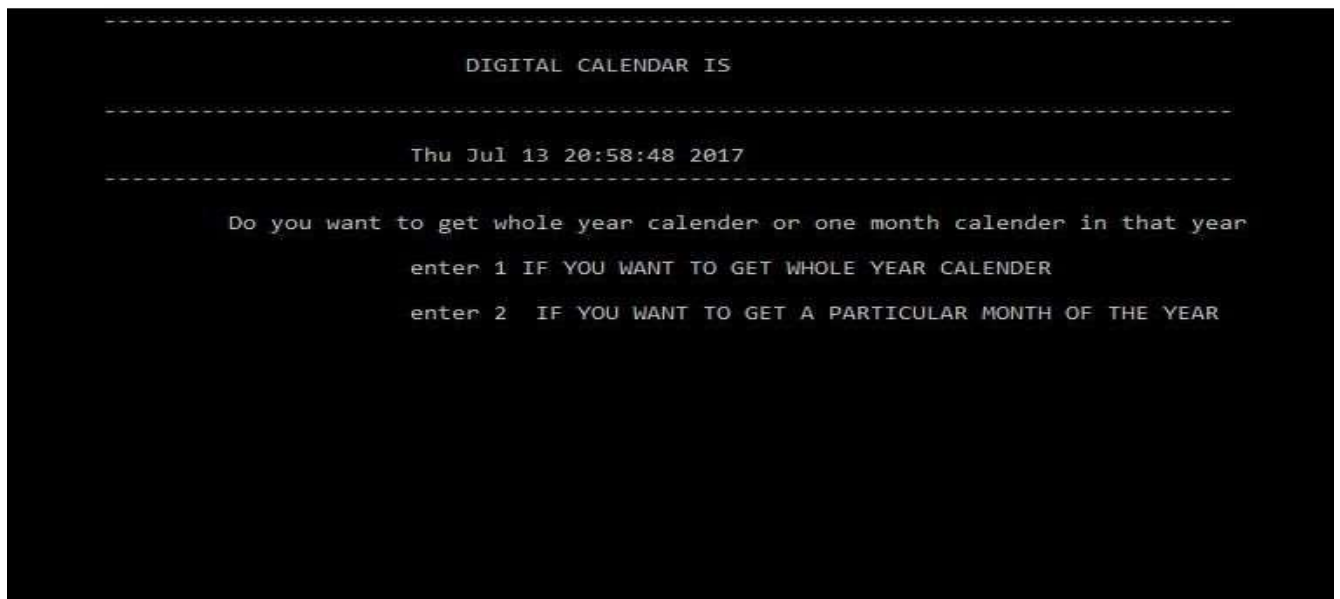


Figure1. Entering The Option

The output screen which on execution of a program will print the digital calendar that is current date and time and also the options from which the user can select his choice.

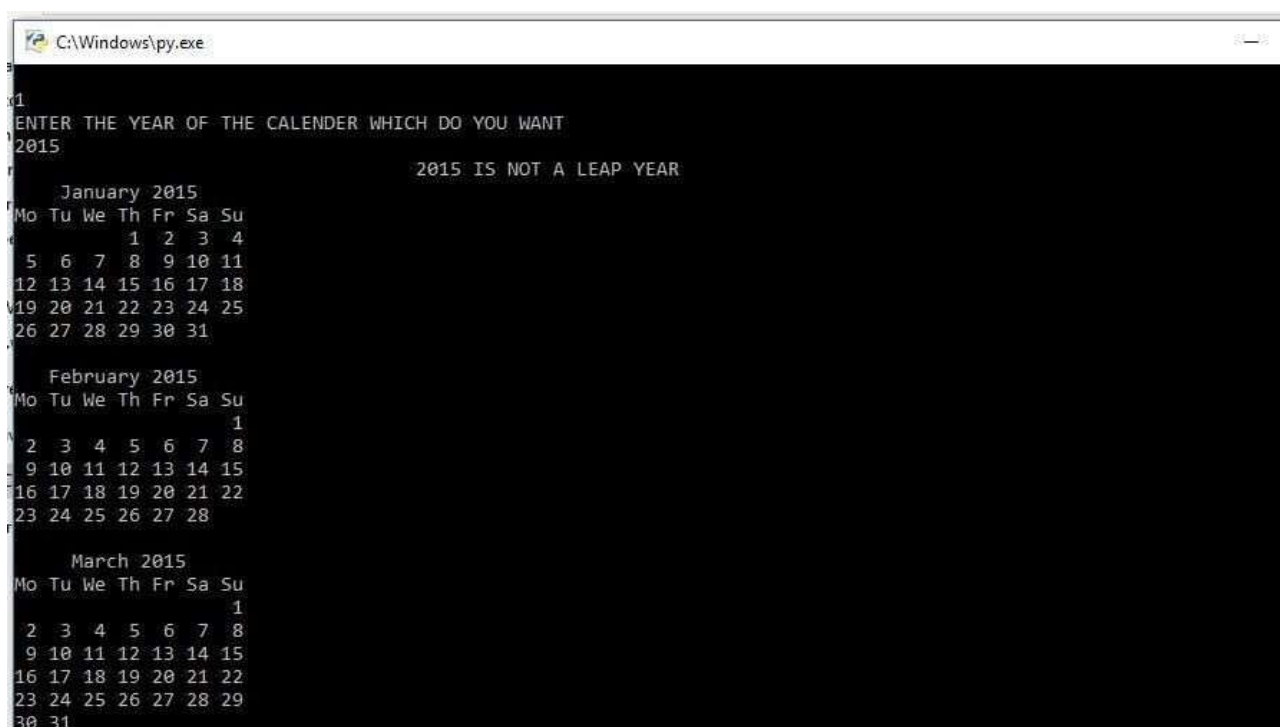


Figure 2.Wyhole year Calendar

The output displayed will be for the option 1 which prints the whole year calendar.



```
C:\Windows\py.exe
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31

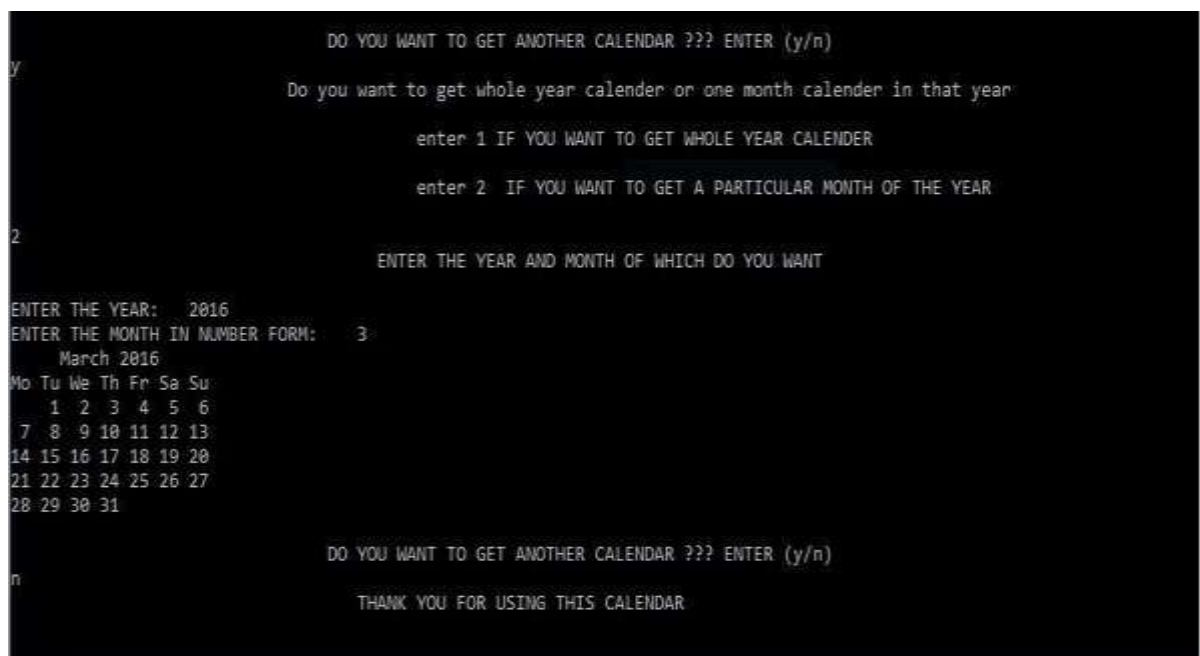
November 2015
Mo Tu We Th Fr Sa Su
      1
  2  3  4  5  6  7  8
  9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30

December 2015
Mo Tu We Th Fr Sa Su
      1  2  3  4  5  6
  7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

DO YOU WANT TO GET ANOTHER CALENDAR ??? ENTER (y/n)
y
Do you want to get whole year calender or one month calender in that year
    enter 1 IF YOU WANT TO GET WHOLE YEAR CALENDER
    enter 2 IF YOU WANT TO GET A PARTICULAR MONTH OF THE YEAR
```

Figure 3.Choosing The Option

After printing the output corresponding to the option entered first it is going to ask the user for another option.



```
DO YOU WANT TO GET ANOTHER CALENDAR ??? ENTER (y/n)
y
Do you want to get whole year calender or one month calender in that year
    enter 1 IF YOU WANT TO GET WHOLE YEAR CALENDER
    enter 2 IF YOU WANT TO GET A PARTICULAR MONTH OF THE YEAR
2
ENTER THE YEAR AND MONTH OF WHICH DO YOU WANT
ENTER THE YEAR: 2016
ENTER THE MONTH IN NUMBER FORM: 3
    March 2016
Mo Tu We Th Fr Sa Su
  1  2  3  4  5  6
  7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31

DO YOU WANT TO GET ANOTHER CALENDAR ??? ENTER (y/n)
n
THANK YOU FOR USING THIS CALENDAR
```

Figure 4.Particular Month Calendar

SOURCE CODE

```
import calendar

import datetime

import time

import csv


def new(y):
    dayofweek = datetime.date(y, 1, 1).strftime("%A")
    print(" 1st January is new year ",dayofweek)

def rep(y):
    dayofweek = datetime.date(y, 1, 26).strftime("%A")
    print(" 26th January Republic day is ",dayofweek)

def amdday(year):
    dayofweek = datetime.date(year, 4, 14).strftime("%A")
    print(" 14th April is Ambedkar Jayanti",dayofweek)

def independ(year):
    dayofweek = datetime.date(year, 8, 15).strftime("%A")
    print(" 15th August Independence day is ",dayofweek)

def teacher(year):
    dayofweek = datetime.date(year, 9, 5).strftime("%A")
    print(" 5th september is teachers day ",dayofweek)

def gandhi(year):
    dayofweek = datetime.date(year, 10, 5).strftime("%A")
    print(" 2nd october is Gandhi jayanthi ",dayofweek)

def child(year):
    dayofweek = datetime.date(year, 11, 14).strftime("%A")
```

```

    print(" 14th November Childrens day is ",dayofweek)

def mas(year):

    dayofweek = datetime.date(year, 12, 25).strftime("%A")

    print(" 25th December is Christmas ",dayofweek)


def cal():

    #select year between 1 to 9999

    print(" Do you want to get whole year calender or one month calender in that
year ")

    print(" Enter 1 IF YOU WANT TO GET WHOLE YEAR CALENDER
")

    print(" Enter 2 IF YOU WANT TO GET A PARTICULAR MONTH OF THE
YEAR ")

    p=input()

    if p=="1":

        years()

    if p=="2":

        months()


def years():

    print("ENTER THE YEAR OF THE CALENDER WHICH DO YOU WANT")

    try:

        y=int(input())

        if 1>y or y>9999:

            print("PLEASE CHOOSE ANOTHER YEAR")


    else:

        print(" "+leap(y)+"

```

```

")
    for i in range (12):
        print(calendar.month(y,i+1))
        print("-----National
Holidays-----")
        new(y)
        rep(y)
        amdday(y)
        independ(y)
        teacher(y)
        gandhi(y)
        child(y)
        mas(y)

```

```

k=input("Want to Set Remainder to any event Enter (y/n)")
if k == "y" or k == "Y":

```

```

    text1=input("Set remainder")
    newFile1=open("upload2.txt","w")
    newFile1.write(text1)
    newFile1.close()

```

```

    another()
except:

```

```

    another()

def months():
    print(" ENTER THE YEAR AND MONTH OF WHICH DO YOU WANT
")
    try:
        y=int(input("ENTER THE YEAR:  "))
        m=int(input("ENTER THE MONTH IN NUMBER FORM:  "))
        if 1>y or y>9999:
            print ("PLEASE CHOOSE ANOTHER YEAR")
            if m<1 or m>12:
                print("PLEASE CHOOSE ANOTHER MONTH")
            else:
                for z in range (13):
                    if (m==z):
                        print(calendar.month(y,z))

        print("-----
Holidays-----")
        if(m==1):
            rep(y)
        elif(m==2):
            print("No Holidayas")

```

```
elif(m==3):
```

```
    print("No Holidays")
```

```
elif(m==4):
```

```
    amdday(y)
```

```
elif(m==5):
```

```
    print("1st May May Day",y )
```

```
elif(m==6):
```

```
    print("No Holidays")
```

```
elif(m==7):
```

```
    print("No Holidays")
```

```
elif(m==8):
```

```
    independ(y)
```

```
elif(m==9):
```

```
    print("No Holidays")
```

```
elif(m==10):
```

```
    print("2nd October Gandhi Jayanti",y)
```

```
    print("19th October Dussehra",y)
```

```
elif(m==11):
```

```
    print("6th November Deepavali",y)
```

```
    print("23th November Guru Nanak's Birthday",y)
```

```
else:
```

```
    child(y)
```

```
w=input("Want to Set any Remainders Enter (y/n)")
```

```
if w=="y" or w=="Y":
```

```
    text=input("Set Remainder")
```

```
    saveFile=open('set.txt','w')
```

```
saveFile.write(text)
```

```
saveFile.close()
```

```
another()
```

```
except:
```

```
print("PLEASE CHOOSE THE YEAR AND MONTH OF THE CALENDER  
CORRECTLY")
```

```
#used if we want to print another calender month or year
```

```
def another():
```

```
    print("
")
```

```
    print("    "+"DO YOU WANT TO GET ANOTHER CALENDAR  
??? ENTER (y/n)"+ "    ")
```

```
    ch=input()
```

```
    if ch=="y" or ch=="Y":
```

```
        cal()
```

```
    else :
```

```
        print(" THANK YOU FOR USING THIS CALENDAR ")
```

```
#date and time function which is used to print the digital calendar
```

```
def dandt():
```

```
    #do you want to know the at present date and time
```

```
    localtime=time.asctime(time.localtime(time.time()))
```

```
    print("    -----
")
```

```

print("    DIGITAL CALENDAR IS ")
print("    ----- ")
print("    "+    localtime    +" ")
print("    ----- ")

```

```

def leap(y):
    year=y
    if (year % 4) == 0:
        if (year % 100) == 0:
            if (year % 400) == 0:
                return("{0} IS A LEAP YEAR".format(year))
            else:
                return("{0} IS NOT A LEAP YEAR".format(year))
        else:
            return("{0} IS A LEAP YEAR".format(year))
    else:
        return("{0} IS NOT A LEAP YEAR".format(year))

```

dandt()

cal()

input()

