BASAVESHWAR ENGINEERING COLLEGE (Autonomous),BAGALKOTE- 587102



Department of electronics and communications

Certificate

This is to certify that project entitled “**AGE CALCULATOR USING PYTHON**” a bonafied work of**, Miss.Rakshita Mundinmani, Miss.Vijayalaxmi Bagewadi**. The report satisfies the academic requirements with respect to project work prescribed for 3th semester during the academic year 2022-2023. It is certified that all corrections/suggestions indicated assessement of the project have been satisfied.

PROJECT GUIDE : HEAD OF THE DEPARTMENT :

PROF.M.C.ARALIMARAD DR.SHRIDHAR KUNTOJI

SIGNATURE WITH DATE:

**INTRODUCTION**

An age calculator is a tool that allows users to calculate a person's age on any specific day based on their given date of birth. There are various methods of calculating a person's age depending upon the culture they belong to.

It is an application where a user enters his date of birth as input,and the application gives his age as output.

**Steps for Building Age Calculator:**These are the following steps to understand the code:

1. We will start by importing the required libraries and modules.
2. We Will define the necessary functions for the execution of the program.
3. We Will do nececssary substractions after getting required arguments from the user.
4. Then we are using try/except block, the code is excuted only when there are no exceptions.

Library functions used in the code:

Try: This function block lets you test a block of code for errors.

Except: This block lets you handle the error.

Since the try block raises an error,the except block will be executed. Without the try block,the program will crash and raise an error.

Here we are importing date from datetime, this module provides several functions and classes for handling dates and times.

from datetime import date

We will now create a function for calculating the age; we will call it calculate\_age(). Just below the import, paste these lines of code:

# defining the for calculating the age, the function takes day def calculate\_age(day, month, year):

# we are getting the current date using the today()

today = date.today()

# converting year, month and day into birthdate

birthdate = date(year, month, day)

# calculating the age

age = today.year - birthdate.year - ((today.month, today.day) < (birthdate.month, birthdate.day))

# return the age value

return age

In the above function, we are passing three arguments to the function, date, month, and year, we will get these arguments from the user. Inside the function, we are getting the current date from datetime module using the today() function. After that, we create a birthdate variable using the date() function; it takes a year, month, and day as arguments. From there, we are now calculating the age by doing some subtractions, the code:

((today.month, today.day) < (birthdate.month, birthdate.day))

Checks if the day or month of the variable today precedes the day or month of the birthdate variable, the return value is a boolean value, and finally, we return the calculated age.

Now we need to get the day, month, and year from the user, just after the function definition, paste these lines of code:

# the try/except block

# the try will execute if there are no exceptions

try:

# we are getting day, month, and year using input() function

day = input('Enter day:')

month = input('Enter month:')

year = input('Enter year:')

# creating a variable called age\_result and we are also calling the calculate\_age function

age\_result = calculate\_age(int(day), int(month), int(year))

print(f'You are {age\_result} years old')

# the except will catch all errors

except:

print(f'Failed to calculate age, either day or month or year is invalid')

Here we have a try/except block, inside the try statement, we have code for getting data from the user; this is done using the input() function, and with that user data, we calculate the age via the function call. If the code execution encounters any errors, the except statement will come to the rescue.

**Program:**

from datetime import date

# defining the for calculating the age, the function takes day

def calculate\_age(day, month, year):

# we are getting the current date using the today()

today = date.today()

# converting year, month and day into birthdate birthdate = date (year, month, day)

# calculating the age

age = today.year - birthdate.year - ((today.month, today.day) < (birthdate.month, birthdate.day))

# return the age value

return age

# the try/except block # the try will execute if there are no exceptions

try:

# we are getting day, month, and year using input() function

day = input('Enter day:')

month = input('Enter month:')

year = input('Enter year:')

# creating a variable called age\_result and we are also calling the claculate\_age function

age\_result = calculate\_age(int(day), int(month), int(year))

print(f'You are {age\_result} years old')

# the except will catch all errors

except: print(f'Failed to calculate age, either day or month or year is invalid')

**Output:**

Enter day:11

Enter month:9

Enter year:2000

You are 21 years old

Conclusion

At the end, age is required to be calculated in various sector for different purposes. With the development of technology, use of the software is preferred to manual hand calculation of age.We also learnt about different library functions in Python which are very useful.