Bug 0

Describe the bug

Full retirement age calculates incorrectly when entering 0 for month. The program assumes the month is january. In MONTH\_NAMES we have 0 = lowercase january

def full\_retirement\_date(birth\_year, birth\_month):  
 birth\_month\_normalized = birth\_month  
 # if birth\_month == 0:  
 # birth\_month\_normalized = 1

Program

Full\_retirement\_age\_calculator\_app.py

Component

Full\_retiremnet\_calc.py

full\_retirement\_date() method

Error Type

Logic error.

Severity

Minor

Priority

Set by management

State

Open

To Reproduce

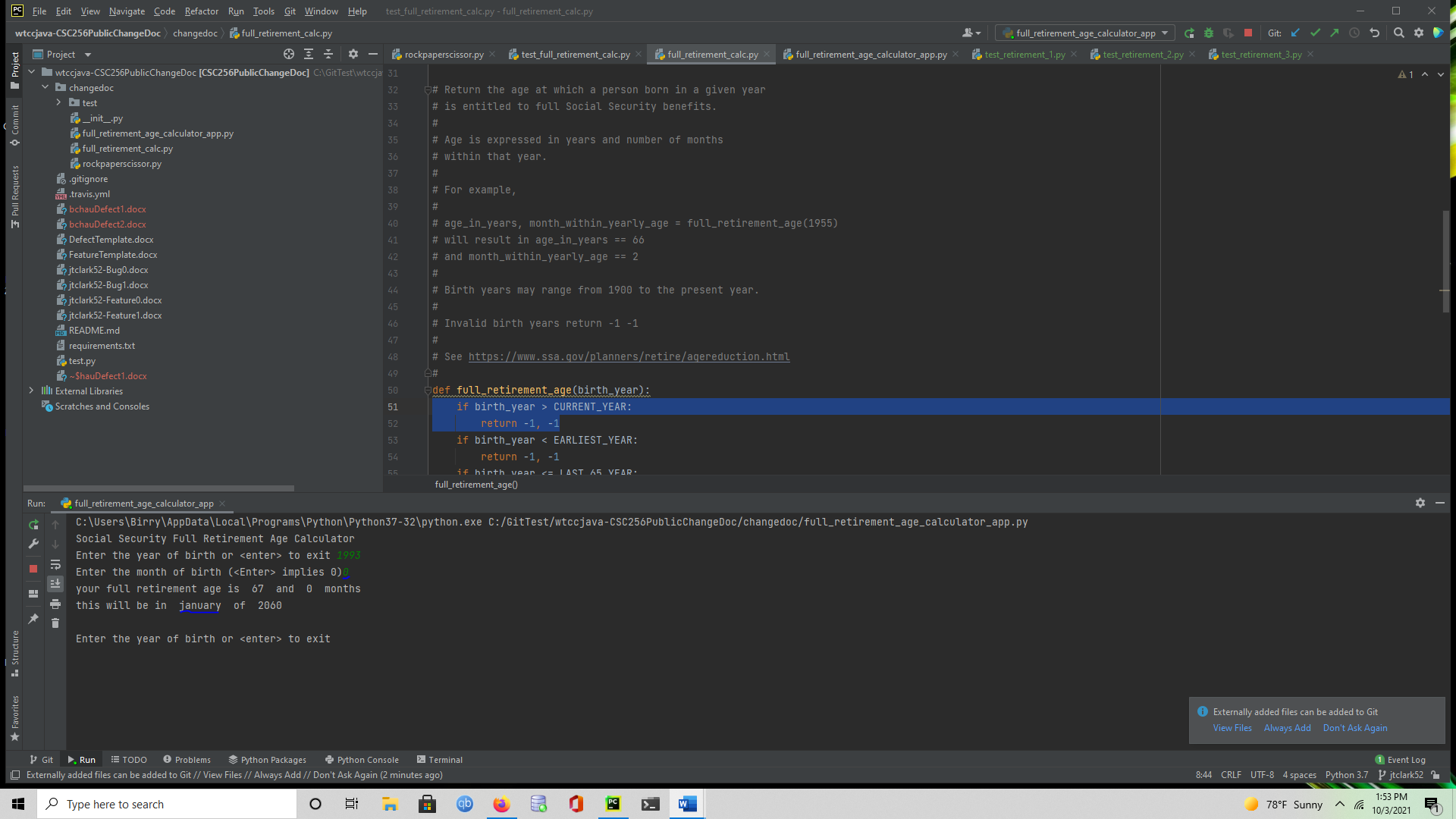
Steps to reproduce the behavior:

1. Enter any valid year between 1900 and 2021
2. Enter 0 for month
3. Calculation Error occurs

Expected behavior

I expected an exception/try catch/ if statement to catch invalid months. The birth month defaults to January if we enter 0.

Screenshots



MONTH\_NAMES = {0: 'january',  
 1: 'January',  
 2: 'February',  
 3: 'March',  
 4: 'April',  
 5: 'May',  
 6: 'June',  
 7: 'July',  
 8: 'August',  
 9: 'September',  
 10: 'October',  
 11: 'November',  
 12: 'December',  
 }

Desktop

Please complete the following information:

* OS: Windows
* Browser [Firefox]
* Python
* 3.6

Additional context

We should only run calculations when we get valid inputs. Since we can’t tell the date is incorrect just by looking at it.

Assigned

Retirement code writer.