**Logic for Survival Duration Calculation Project:**

1. From Datetime module , date time has been imported.
2. Regular expression as Regex has been imported to match a particular pattern of date.
3. Survival Calculator function has been defined.
4. User is prompted to make selection in order to know their age in the form of years, months, weeks, days, hours, minutes and seconds.
5. User is asked to make the selection in either of the three ways:

* By selecting A user can input their age to see their age details in the form of months, weeks, days, hours, minutes and seconds.
* User can also see their complete age in the form of years, months, weeks, days, hours, minutes and seconds by selecting B where in they need to input their date of birth.
* User can select C if they want to exit and close the loop.

1. A while loop is used so that the user is always prompted for a selection between A,B and C until the user decides to exit.

**If User Selects A**

* The user is prompted to give the unit in which they need to see their age, that is, months, weeks, days, hours, minutes and seconds.
* Here Try and Except condition is used to check whether an user inputs his/her age in intergers.
* If the user inputs the age in integer then Try block will be executed and age calculation takes place or else if the user enters any value other than the integer than the ValueError exception block will be executed and user will be asked to try again and input the age in number.
* If and elif statement is used to check multiple condition that the user inputs to check their age in different unit in months, weeks, days, hours, minutes and seconds.
* First the age will be printed in the unit asked by the user and the in the second line the entire age will be printed in the form of months, weeks, days, hours, minutes and seconds.

**Calculation** **Used**:

* Month is calculated by multiplying age \*12 ( 1 year=12 months)
* Weeks is calculated by multiplying age\*52 weeks. (1 year=52 weeks)
* Days is calculated by multiplying age\*365 days ( 1 year=365 days)
* Hours is calculated by multiplying days\*24 ( 1 day=24 hours)
* Minutes is calculated by multiplying hours\*60 ( 1 hour=60 mins)
* Seconds is calculated by multiplying minutes\*60 ( 1 min=60 secs)

**If User Selects B:**

* User is prompted to input their date of birth exactly in dd/mm/yy format.
* To ensure this, regular expression has been used. To enter the date in dd/mm/yy format the user has to first input a two digit number for date which can be denoted by using the regular expression \d{2} then a backword slash has to follow ‘/’ then again a two digit number for month \d{2} followed by a backword slash ‘/’ and finally a two digit number for year \d{2}. Hence the entire pattern has to be in “\d{2}/\d{2}/\d{2}.
* The match method is used where in the regular expression pattern will be checked.
* Therefore if the user inputs DOB that matches exactly in the pattern denoted by the regular expression only then the age calculation will be performed else the user will be prompted to select again due to incorrect format.
* To execute the above condition If and else condition is used

**Calculation Used:**

* To calculate user’s age from DOB we need to substract current date from DOB
* However the current date will be in integer format while the DOB entered by the user will be in string.
* Therefore we convert the DOB string into an integer by using datetime.strptime(DOB, '%d/%m/%y')
* We also had the current date by using the current date method from the Datetime module.
* Now if we subtract the current date from DOB we get the result in integer.
* However the above calculation will give the age in time delta object and will be in days, minutes, seconds and microsecond to avoid this .days calculation is used which will give the answer exactly in date
* Age= current date-DOB
* Age is then calculated in exactly in days by using .days calculation
* Years= age/365 ( 365/365 gives 1, hence a person is 1 year old)
* Month= age/30 ( 1 month 30 days)
* Weeks= age/7 ( 1 week=7 days)
* Days itself will be age.days function
* Hours is then calculated by converting the days into seconds by using total\_seconds function from datetime module, therefore 1 hour is 3600 seconds
* Minutes is seconds/divided by 60 ( 1 min 60 seconds)
* Seconds will be itself total\_seconds function which will convert the entire days in seconds

**If User Selects C:**

* Break statement is used as the while loop will be broken and the function will stop and user will be receiving a Thank You statement.