**Program 4: DateBirth Error Log**

|  |  |
| --- | --- |
| Error Number | Description |
|  | ***At line:***  daysBetweenDates = DateDiff("y", dayOfBirth & monthOfBirth & adjustedYearOfBirth, Now)  ***Error:***    ***Explanation:***  This appeared because I tested “Aprol” instead of the correct month “April” and found the function DateDiff did not recognize the right month.  ***Solution:***  I will use exception handling to give the user no more than three tries to fix their error. |
|  | ***Error:***    ***Explanation***:  A variable can’t have the same name as a function.  ***Solution***:  Change the variable name or function name. |
|  | Logic Error:    ***Explanation***:  The “actualDays” variable declared in Sub Main() has no relation to the “actualDays” variable from the findInterval() function.  ***Solution***:  Separate the functionality of findInterval() so a function is getUserInputAsString() and returns the user input as a string in the variable birthDate. The other function called calculateAgeFromBirthDate() finds the age with the DateDiff function with the birthDate variable as its argument. The result of the calculate function is equal to the age.  Obscurity over what problem I faced here and how to solve it led me to this mistake.  To write the solution:  age = calculateAgeFromBirthDate( getUserInputAsString () ) |
|  | ***Error:***    ***At line:***  While (Integer.TryParse(Console.ReadLine(), monthOfBirth) = False And numberOfTries < 3)  ***Explanation:***  The variable monthOfBirth is not an integer, and therefore using TryParse to convert from a string to an integer doesn’t make sense.  ***Solution***:  Use another method to figure out if the month is misspelled. |
|  | ***Error:***    ***At Line:***  givenYearOfBirth = Console.ReadLine()  ***Explanation:***  The variable givenYearOfBirth which is of type “long” can’t be converted to a “string”  ***Solution:***  Change the “long” to an “integer” so it can become a “string” |
|  | ***Error:***    ***At Line:***  daysBetweenDates = DateDiff("y", birthDate, Now)  ***Explanation:***  At this point in the program, the month came as a number instead of a word so the DateDiff function didn’t recognize it.  ***Solution:***  Make sure the DateDiff function gets the month in a string format. More specifically, make the getUserInputAsString() function return monthOfBirthName (which is a string) instead of monthOfBirthInt (which is an integer). |
|  | ***Error:***    ***At Line:***  age = calculateAgeFromBirthDate(getUserBirthDateAsString())  ***Explanation:***  This program generates too many errors because it is too complicated to take in the day, month, and year as separate inputs.  ***Solution:***  I did more research and found that I can input the birthDate all at once in any format if I declare it as a Date object. First I will input a birthDateAsString from the console, then parse it into the Date object birthDate.  Console.WriteLine("Input your date of birth ")  birthDateString = Console.ReadLine()  birthDate = Date.Parse(birthDateString) |
|  | ***Error:***    ***At line:***  If ((Date.Parse(birthDateString)).Equals(False)) Then  ***Explanation:***  The Parse method throws the Format Exception when it can’t parse the birthDateString correctly into a Date object.  ***Solution:***  Use the TryParse method so no exception is thrown and so the user can get several tries. |
|  | ***Solution:***  Make sure I declare the array as a Date object array, correctly. |