**Analysis**

Steps:

1) I implanted date class. Specifically, addDay(day) method override in ExtDate class. This method update days, month, and year.

2) Created ExtDate class that extends Date class and override addDays(day). I called super.addDays() and set string months in the method. It will update string months when user add more days.

3) Created day class and implemented all methods.

4) Created a calendar class and it has two private object Class Day and Class ExtDate. I never used day object because it was not necessary.

5) test method has a method that write current day to txt file. It only write recent date to text file.

How it works after updating days:

1) If the initial date is 11/25/ 2019.

It will print November 25, 2019. Case 6: let you add days to date. If you add 60 days. The new date will be January 24, 2020. If you write the date to txt file, new date will be saved.

2) case 7: prints the calendar of the initial date. It doesn’t update the calendar date when add new days. If you want to print calendar of new date, you have terminate and give a different initial value.

Note: Calendar date is updated only one with initial date. It will print calendar for only initial date. You can do all task from 1 – 8 until user enter 9 to exit.

**UML Design**

|  |
| --- |
| Date class |

|  |
| --- |
| class ExtDate |

|  |
| --- |
| class Day |

|  |
| --- |
| class Calendar |

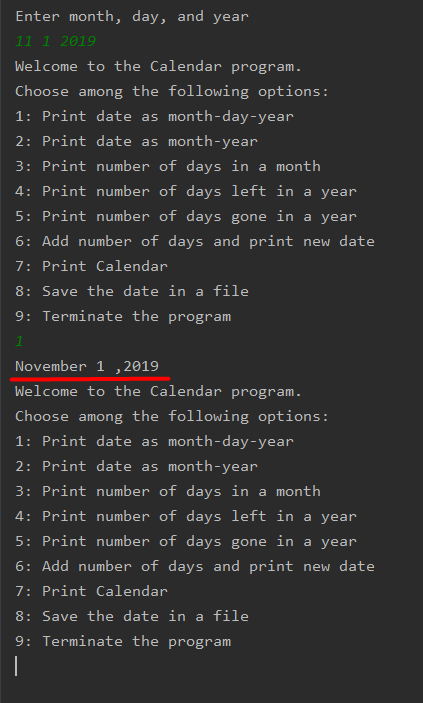
|  |
| --- |
| Date class |
| - day : int |
| - month : int |
| - year : int |
| + Date() |
| +Date(month:int, day:int, year:int) |
| +setDate(month:int, day:int, year:int):void |
| +setDay(day:int):void |
| +setMonth(month:int):void |
| +setYear(year:int):void |
| +getDay():int |
| +getMonth():int |
| +getYear():int |
| +isLeapYear():boolean |
| +getDaysInMOnth(month:int):int |
| +daysPassedInYear():int |
| +daysRemaningInTheYear():int |
| +addDays(day:int):void |
| CopyOfDate():date |
| +CopyOfAnotherDate(x: Date):void |
| +toString():String |

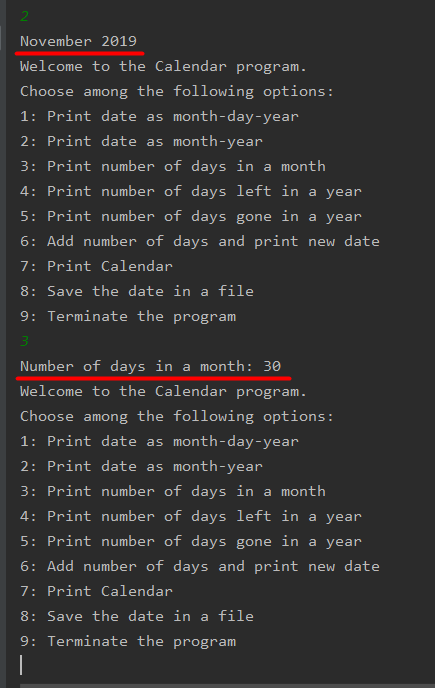
|  |
| --- |
| ExtDate class |
| - months : String |
| - name : String [] |
| ExtDate(month:int, day:int, year:int) |
| +setMonths():void |
| +getMonths():String |
| +printAsMonthDayYear():void |
| +printAsMonthYear():void |
| +addDays(day:int):void |

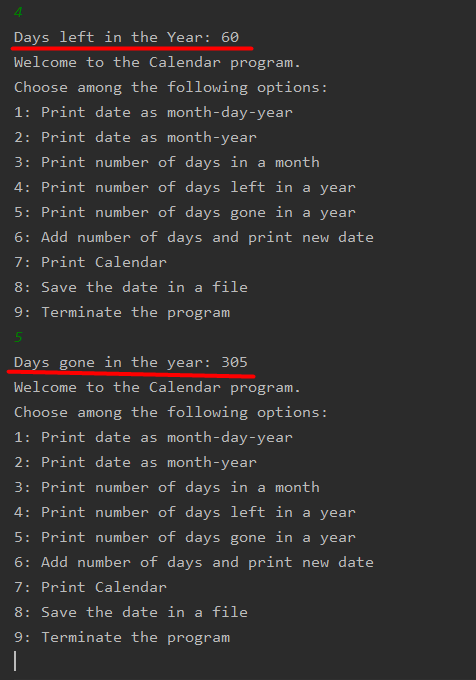
|  |
| --- |
| Day class |
| - days: String |
| - d: String [] |
| Day() |
| Day(d:String) |
| +setDays(d:String):void |
| +getDays():String |
| +printDay():void |
| +nextDay():String |
| +previousDay():String |
| +addDays(day:int):String |

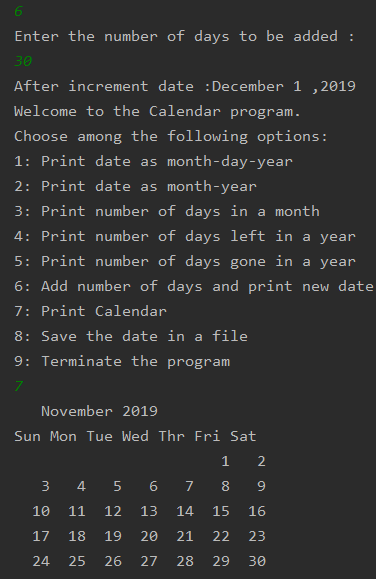
|  |
| --- |
| Calendar Class |
| - e : ExtDate |
| - d: Day |
| +firstDayOfMonth():int |
| +setMonth(m:int):void |
| +setDay(d:int):void |
| +setYear(y:int):void |
| +getMonth():int |
| +getDay():int |
| +getYear():int |
| +getStringMonth():String |
| +printCalendar():void |

Testing sample 1:









Case 8:

