CS 140
Calendar Assignment
Part 1

Key topics:

Printing, data types, methods, operators, expressions, variables, for loops, class constants, parameters, returns, String objects, Scanner objects

Learning Outcomes:

- Begin to become familiar with setup, design, execution and testing of basic Java programs
- Design and develop a multi-method program in good style
- Demonstrate the ability to decompose a problem and reduce redundancy using methods
- Apply tools and techniques introduced in class to form a working program

By the end of this assignment, you should be familiar with the material covered in class so far and be able to demonstrate the ability to use it properly in practice.

Task:

Your task is to create a basic calendar using Java. Over the course of the quarter, we will be adding to this very basic calendar to make it more functional and robust. For this part, you will be drawing the calendar to the console.

The calendar's size should be scalable using class constants. You have some creative freedom with designing the calendar. What characters you use for the outline of the calendar and where you place the date in each calendar cell is up to you. An example of what a calendar might look like with a size of 10 after completing this assignment is on the next page.

Functional Rundown:

Upon executing your program, the console should ask what date you would like to look at, as seen in the example below. The user should be able to input a date. The month of that date should be displayed at the top of the calendar. The month and day input by the user should be broken up and displayed at the bottom of the calendar as shown below.

After asking the user for a date and printing the calendar for that date, your program should print a calendar for today's date followed by the month and day broken up.

Ultimately, after execution, your calendar should display a prompt to get a date from the user, a calendar for that month from the user input, the month and day input by the user broken up, a second calendar for the current month, and finally the month and day for the current day broken up. For now, you may have every month that the calendar displays have 35 days instead of the actual number. We will change that in a later part of the assignment.

Example Run:

```
What date would you like to look at?(mm/dd)
04/05
                              4
=================================================
|1       |2       |3       |4       |5       |6       |7       |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
|8       |9       |10      |11      |12      |13      |14      |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
|15      |16      |17      |18      |19      |20      |21      |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
|22      |23      |24      |25      |26      |27      |28      |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
|29      |30      |31      |32      |33      |34      |35      |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
Month: 4
Day: 5

This month:
                             10
=================================================
|1       |2       |3       |4       |5       |6       |7       |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
|8       |9       |10      |11      |12      |13      |14      |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
|15      |16      |17      |18      |19      |20      |21      |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
|22      |23      |24      |25      |26      |27      |28      |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
|29      |30      |31      |32      |33      |34      |35      |
|        |        |        |        |        |        |        |
|        |        |        |        |        |        |        |
=================================================
Month: 10
Day: 1
```

Scalability:

```
What date would you like to look at? 9/20
                                    9
===============================================================
| 1         | 2        | 3        | 4        | 5        | 6        | 7         |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
===============================================================
| 8         | 9        | 10       | 11       | 12       | 13       | 14        |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
===============================================================
| 15        | 16       | 17       | 18       | 19       | 20       | 21        |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
===============================================================
| 22        | 23       | 24       | 25       | 26       | 27       | 28        |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
===============================================================
| 29        | 30       | 31       | 32       | 33       | 34       | 35        |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
|           |          |          |          |          |          |           |
===============================================================
Month: 9
Day: 20
```

Your program should produce output like this if the size is 10. However, as stated above, you have the freedom to choose what characters you use for the calendar (the "|" and "=" in the example above) and the exact location of the number in each cell. Below is an example of a calendar produced with a size of 5:

```
What date would you like to look at? 9/20
|                      9
=================================
| 1  | 2  | 3  | 4  | 5  | 6  | 7  |
|    |    |    |    |    |    |    |
=================================
| 8  | 9  | 10 | 11 | 12 | 13 | 14 |
|    |    |    |    |    |    |    |
=================================
| 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|    |    |    |    |    |    |    |
=================================
| 22 | 23 | 24 | 25 | 26 | 27 | 28 |
|    |    |    |    |    |    |    |
=================================
| 29 | 30 | 31 | 32 | 33 | 34 | 35 |
|    |    |    |    |    |    |    |
=================================
Month: 9
Day: 20
```

Creative Portion:

In addition to the functionality of the calendar, your calendar should display some text art (ACSII art) above the calendar, like what you would see with a typical wall calendar. You have the freedom to choose what you display with these restrictions:

The ACSII art should be your own creation, not something found on the internet or elsewhere
You should make use of for loops to create your design and reduce redundancy within its code, but make sure there are no infinite loops
The picture should be appropriate and not include hateful, offensive, or inappropriate images
You should not use advanced material beyond what we have learned in class so far

For this portion of the assignment, you will get credit for completing something that demonstrates you have put some effort into it. If you follow the guidelines above and have demonstrated in your code that you have put a decent amount of effort into it, you will receive full points for the creative portion of the assignment.

Additional information about ASCII art can be found online by googling "ASCII art."

Implementation Details:

You program must include the following methods:

public static void drawMonth(int month)

This method takes in an integer representing the month and displays the month and a graphical representation of the calendar as seen in the examples above.

public static void drawRow(int row)

This method should be called in your drawMonth method. It should display one week on the calendar (one row). This method is passed an integer representing which row it is displaying.

public static void displayDate(int month, int day)

This method is passed the month and the day as integer values and displays the date information as seen in the above example underneath the graphical representation of the calendar.

public static int monthFromDate(String date)

This method should extract an integer value for the month and return it when passed a given date as a String. Using the indexOf and substring methods may be helpful with this.

public static int dayFromDate(String date)

This method should extract an integer value for the day and return it when passed a given date as a String. Using the indexOf and substring methods may be helpful with this.

Helpful Information:

For this assignment, you may want to use a few tools that we have not gone over in class.

To get today's date, you will likely want to make use of a Calendar object that is already implemented by Java. This object stores basic information that we can access. To use this object, you will want to create a new Calendar object (be careful that your class is also not named Calendar or Java will get confused). The code for that would look like Calendar **name** = Calendar.getInstance(). This gets the current date and stores it in this Calendar object.

To access the information we want, we need to use the Calendar's get method. For instance, if we wanted the month we would use **name**.get(Calendar.MONTH), or if we wanted the day we would use **name**.get(Calendar.DATE). Calendar.MONTH and Calendar.DATE are just ways to refer to the locations where those values are stored.

You also may make use of several String methods. The indexOf and substring methods could be of use. The indexOf method finds the index of the first occurrence of the given character in the String you call the method on. For example, if we had "Hello" store in a variable x, calling x.indexOf("l") would return a value of 2 since the first "l" is located at index 2 in the String "Hello".

The substring method creates a new string from the starting index given to the ending index given (it does not include the character in the ending index). If not ending index is given, then a substring from the beginning index to the end of the String is created. If we wanted a substring of "Hello" that just captured "ell" and if "Hello" was stored in variable x, we could call x.substring(1, 4). If we wanted just "llo", we could call x.substring(2).

Finally, you may find it useful to convert a number as a String into an integer value. To do this, we can call Integer.parseInt(**String**), where **String** is the String value of the integer we want.

I strongly encourage you to tackle this assignment in parts and practice iterative design. Work on a few things and then test them to make sure they work. Then, add some more and test those. Make sure that you come up with a plan before you start coding. Writing pseudocode can also be very helpful. I encourage you to speak with your classmates about your development plan for this assignment. I am also available to discuss how you might approach this assignment. Feel free to reach out to me with any questions or concerns.

Style:

It is important that you get used to writing code in good style. What is demonstrated in examples in class is considered good style. Additionally, you should look at the style guide located on Canvas. Code written not in good style will lose points.

Extra Credit Opportunities (Optional):

You can choose to do as many or as few of these as you would like. Each one completed successfully and in good style will earn some extra credit points towards this assignment.

1. As stated above in the specification, you can allow your calendar run to 35 days. For an added challenge, you can limit the number of days to 31. Completing this task successfully and in good style will result in extra credit.

2. If your accii art is truly remarkable, Additional extra credit may be awarded. The complexity, uniqueness, and apparent time commitment poured into the art will be taken into consideration. In CAS123A there are a few examples hanging on the wall.

Grading:

Category	Pts.
Style	10
Creativity	10
drawMonth method	15
drawRow method	15
displayDate method	15
monthFromDate method	7.5
dayFromDate method	7.5
Program functionality	20
Total:	**100**