

# **San Luis Obispo Police Department Scheduling**

*Joshua R. Logier*

*Advisor: Bridget Benson, Electrical Engineering*

*Cal Poly State University, San Luis Obispo*

*Computer Engineering*

*June 10<sup>th</sup> 2015*

## **Abstract**

The San Luis Obispo Police Department (SLOPD) desired a new scheduling software that would enable their employees to view their work schedules online while maintaining the look and feel and ease of use of their current scheduling system. This project describes the design of such a new scheduling system that maintains the look and feel of the old scheduling system while providing a means to edit and view schedules online.

## **Acknowledgements**

I would like to thank my advisor, Bridget Benson for her supervision of the project, continued availability, and willingness to supervise my project despite already having way too many of them.

I would also like to thank Captain Keith Storton of the San Luis Obispo Police Department for pointing me in the right direction and getting me in contact with the right people as well as his effort working with me and others to find a mutually beneficial project.

A special thanks to Sergeant Janice Goodwin of the San Luis Obispo Police Department for her suggestions and guidance in deciding on a project and for taking the time to do a background investigation so I could work with the San Luis Obispo Police Department.

Lastly, a big thanks to Lieutenant Bill Proll of the San Luis Obispo Police Department for his availability, giving me access to the current software, and willingness to work directly with me on the many details of the project.

# Table of Contents

Abstract .....	2
Acknowledgements .....	3
Introduction .....	7
Requirements .....	8
Background.....	8
Design.....	9
Software .....	10
Problems Encountered.....	12
Testing .....	13
Testing Results and Conclusions .....	13
Future Work.....	14
References .....	14
Appendix A (User Manual) .....	15
Navigation.....	15
Officers and Shifts .....	15
Adding/Modifying an Employee.....	17
New Shift Rotation .....	19
Appendix B (VBA Code) .....	21
NewPayPeriod .....	21
AddAdministration .....	24
AddCommunications .....	29
AddDays .....	34
AddInvestigations.....	40
AddNights .....	45
AddRecords.....	51
AddSNAP .....	56
AddTrainee.....	61

## List of Tables and Figure

Figure 1: SpeedShift's basic design/layout .....	9
Figure 2: UserForm used to add an officer to the daywatch schedule .....	10
Figure 3: Software flow diagram .....	11
Figure 4: Tabs for separate sheets.....	15
Figure 5: Toolbar .....	15
Figure 6: Employee format in employee list .....	16
Figure 7: Shift format in shift list .....	16
Figure 8: Excel color options and their corresponding color code .....	16
Figure 9: Adding or modifying an employee .....	17
Figure 10: Assignment list in dropdown .....	18
Figure 11: Daywatch assignments .....	18
Figure 12: Viewing a comment .....	18
Figure 13: Calendar used to select a day .....	19
Figure 14: Button to create a new shift rotation .....	19
Figure 15: New shift rotation UserForm .....	20



# Introduction

Scheduling is a necessary part of all industries, especially public safety. With frequent overtime and shift changes, police departments need a reliable way to keep track of their employees. However, police departments do not need thousands of features with complex functionality. Scheduling for police departments is relatively straightforward with minimal needs beyond a basic spreadsheet style schedule.

There are many software suites that provide this functionality and much more typically at a very expensive price. There are currently no programs that provide only basic scheduling functionality for enterprise-level uses. The programs that are available can be convoluted and would require learning a new layout with similar functionality in different or inconvenient locations or even missing functions that were commonly used. This can lead to a lot of unnecessary time spent learning a new program. Also, most enterprise-level software is extremely expensive which may be justified for companies that use much of the built-in functionality and have the money to spend. However, much of the cost is wasted when only the bare essentials are used in practice as would be the case with the San Luis Obispo Police Department.

I have attempted to solve this problem by using Microsoft Excel to provide the functionality necessary to meet the San Luis Obispo Police Department's needs. Excel can be expanded beyond a simple spreadsheet with macros programmed in Visual Basic for Applications. This combination provides me with the tools needed to provide exactly what the San Luis Obispo Police Department wants. With this system, a supervisor will have editing permissions and sync the file with OneDrive online in order to create a view-only link that any other employee can see from the station or at home. OneDrive doubles as a redundant storage location so that all current and previous scheduling files can be accessed in the event of a local hardware failure.

SLOPD was unable to fully test the program as anticipated before the completion of this report. However, they provided invaluable feedback along the way and tried out little bits of the program and provided me with feedback that I could use to improve the program and test myself.

# Requirements

The following requirements were developed in accordance with the San Luis Obispo Police Department according to their specific needs.

- Maintain parity with SpeedShift (SLOPD's current scheduling software) design and functionality
- Have the ability to be posted and viewed online
- Ability to add/update an employee's schedule easily
- Provide additional functionality as needed

# Background

In order to best address the San Luis Obispo Police Department's needs I first familiarized myself with their current scheduling software, the reasons they were upgrading, and what functionality was most important to them.

There are many different types of software that provide scheduling. These software packages range in complexity and price depending on the needs of the target company. Most, if not all, of the software available on the market provides extensive feature sets and the ability to perform very complex tasks for thousands of employees. The San Luis Police department, however, needed something simpler to meet their needs.

SLOPD currently uses a program called SpeedShift as seen in Figure 1. This program works well for their needs, however, the company that made the software will be dropping support in favor of a newer program. The program also does not offer online accessibility. SpeedShift is essentially a scheduling program built around a spreadsheet. It has tools and features that make scheduling simpler and customizable.





*Figure 1: SpeedShift's basic design/layout*

I first worked with SLOPD to understand how SpeedShift works. I was given access to the software so that I could see and use the different features that SpeedShift provides. I gained a better understanding of the features that SLOPD uses the most and why they liked the features that they liked. I knew this project could be accomplished but would not be possible to build from scratch. I needed a program that I could build this functionality into.

Microsoft Excel has been used extensively to meet the needs of students and professionals alike. It is commonly used for maintaining data and performing complex calculations on the data. Businesses commonly use Excel for finances and keeping track of various types of data. Additionally, Excel provides the ability to code macros using Visual Basic for Applications (VBA). Macros allow Excel's uses to be practically endless.

After gaining a better understanding of the needs that needed to be met and seeing that the department was effectively using a spreadsheet built into a program, I decided to use Excel to develop the program. My knowledge of Excel was limited so I began familiarizing myself with the functionality and the possibilities.

Excel alone did not directly solve the problem of making the schedule online accessible. However, Microsoft does provide a cloud service called OneDrive where documents and files can be stored remotely to be accessed from anywhere. Additionally, Microsoft has built a basic Office suite online called Office Online. Through a OneDrive account, you can edit and view documents stored on OneDrive (including Excel documents) from anywhere. This solution is not elegant but meets the needs of the department. The department is also able to remotely backup their files using OneDrive, which provides additional data integrity.

## Design

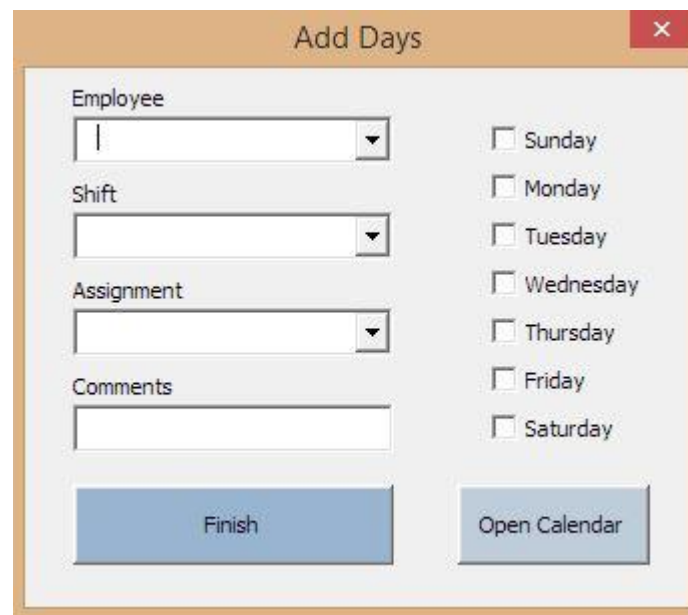
The following section will discuss the software design of the scheduling program. Several design iterations were made with each new design building on the previous. I began by designing the spreadsheet itself and maintaining familiarity with SpeedShift in order to make the transition easier. Following the layout and formatting design I began creating Visual Basic for Applications (VBA) modules.

I chose to go with a modular design that will allow for future changes. I created a separate worksheet in addition to the standard scheduling sheets that holds all the data used throughout the program. I have the officer list, shift list, shift abbreviations, and shift color all put into a list that can be easily updated as needed in the future. This allows the program to not be dependent on conditional formatting.

## *Software*

Much of the functionality of the project was done through macros and background code execution. These macros are typically tied to buttons that take user input and do something with it. All code was done in Visual Basic for Applications. Interacting with a button on the spreadsheet causes a UserForm to popup, allowing for user interaction.

A UserForm provides a way to create interactivity for the user. There are many different elements that can be added to a UserForm including drop-down menus, text fields, buttons, checkboxes, etc. These elements can be organized to create a custom popup that can be configured to do whatever needs to be done. The elements on the UserForm are connected to the spreadsheet and each other through VBA code.

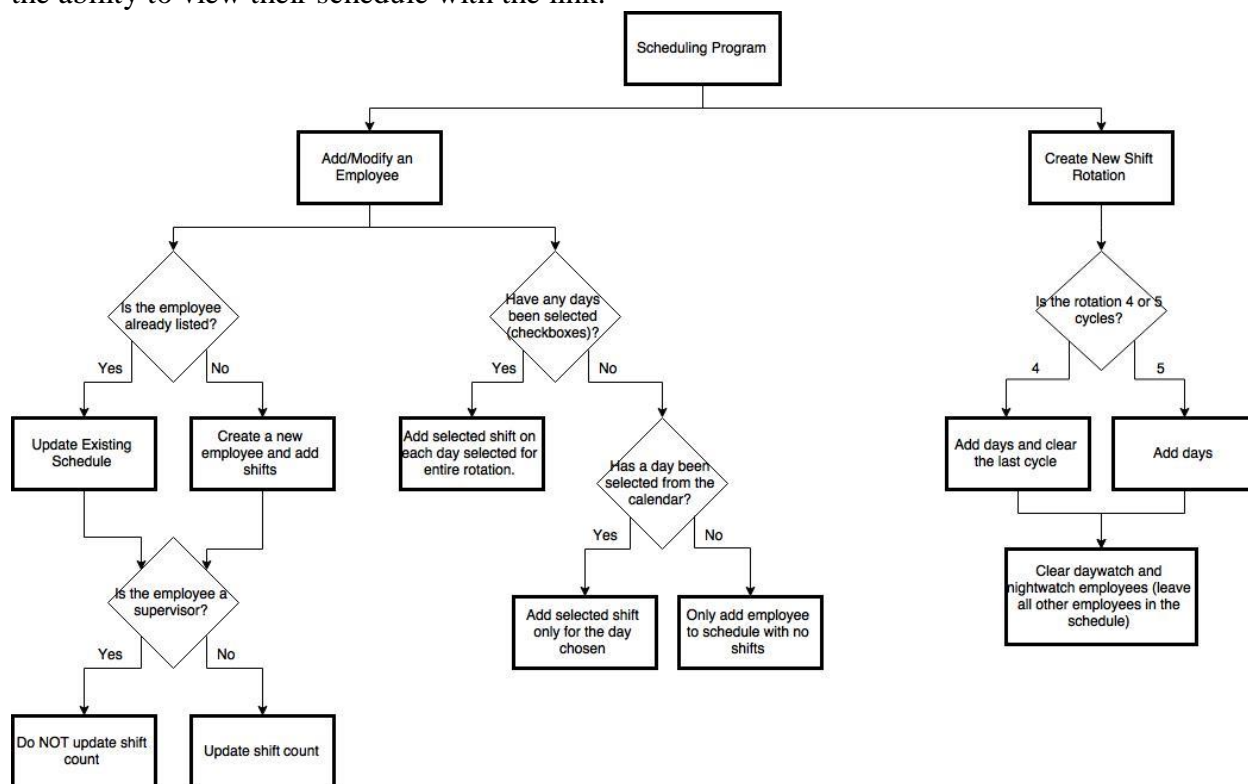


*Figure 2: UserForm used to add an officer to the daywatch schedule*

I rely heavily on UserForm's that take user input, such as Figure 2 shown above, in order to make intelligent decisions on what to do. I quickly discovered that my initial design relied too

much on background processing and guessing what the appropriate action was. I thought that would be helpful but it turned out it was slow and prone to bugs and errors. UserForm's allowed me to code a specific task to a specific input that could be easily tested and verified for correctness. This allows the user to be very specific about what he/she wants to input while not being cumbersome or time consuming to use. Appendix A details how to use the software and many of the design decisions can be seen.

OneDrive was chosen as an avenue to provide online access to the schedule. This was chosen due to the ease of integration between Excel and OneDrive as well as the backup capabilities that OneDrive provides for the files. You have the ability to upload the file to OneDrive and then create a link for sharing that can be view-only allowing the employees without edit permissions the ability to view their schedule with the link.



**Figure 3: Software flow diagram**

Figure 3 demonstrates the flow of logic based on the two main actions in the program which are adding/modifying and creating a new shift rotation. These are both buttons in the actual program and will behave as depicted in the diagram given various conditions.

When adding a new employee, care must be taken not to override the employee or duplicate the employee being added. Checks are made in the code to prevent either of those cases from happening.

When creating a new shift rotation, each worksheet must be properly prepared for the new time period. Police officers change shifts from daywatch to nightwatch and vice versa almost every rotation and that means that there is no easy way to keep the officers in their shift. Because of

this, when a new rotation is created, all the patrol officers are removed and the sheet is reset to allow the supervisors to rebuild the scheduling according to who is working.

## Problems Encountered

One of the biggest issues that I had throughout the development was a lack of robust testing. There is no easy way to test the code and macros that I wrote other than manual testing. I did the best I could to be thorough but there are inevitably going to be problems that aren't immediately caught. Working with SLOPD to test the functionality provided an added layer of testing but only tested a subset of the elements of the program.

There were a couple minor issues that could not be resolved due to the limitations of Excel. Specifically, I wasn't able to provide an interface for specific functionality such as double clicking a cell on the spreadsheet. There is no way to modify Excel's native behavior for basic operations. SLOPD was used to this functionality with SpeedShift when adding a shift for an officer. They would double click the cell they wanted to add a shift to and a shift window would popup and allow them to select the shift along with other details that would be placed into the spreadsheet according to the way they wanted it. The way I have attempted to solve this issue is through comments. On the "Add/Modify" UserForm the user has the option to add a comment along with the shift that will be shown when a user hovers their cursor over that cell. The comment can contain all the information necessary for the employees and supervisors to know everything about the particular shift.

There were a couple of Excel incompatibilities when the program was used on SLOPD's computers. Their computers have an older version of Excel than I was using and I did not have access to the older version for testing. I had to rely on SLOPD to try out the different functionality as it was added and see if there were any incompatibilities.

I initially wanted to use a built in calendar that could be used to pick a date from within the "Add/Modify" UserForm. Excel used to include the calendar widget in Excel but now requires a download to install it on newer versions. I was able to download it and integrate it into the project but quickly realized that would not be a lasting solution because SLOPD did not have the widget installed and would not be able to install it easily. I went back to the drawing board and ended up finding a custom calendar that Trevor Eyre wrote based off of a custom calendar window. This was a great solution as it didn't have any specific version requirements.

Working with this new calendar proved to be very convenient and added a nice touch to the project because anything about it could be adjusted including colors, font, and layout. However, there was still a minor type error when dealing with the *Date* type. Excel 2013 that I was using allowed me to define and use dates as the type *Date*. The version of Excel that SLOPD uses required an explicit *VBA.Date* type. Once I discovered this, I was able to identify the problem locations and fix the issue.

# Testing

Testing was done as elements of the software were developed. As I mentioned above, there were many iterations of the UI and code so there were different elements that needed to be tested at each stage.

The UI testing ultimately was a usability test. I had to ask myself “how easy is this to use and how intuitive is it?” This was tough to answer many times because easy for me many not be easy for someone else. That’s when I asked myself another question, “is this design similar enough to SpeedShift that the employees won’t notice much of a difference?” Creating a program that would provide a seamless transition for the department was my top priority so I made sure to test accordingly. If I felt it was easy to use but wasn’t similar to SpeedShift then I revisited my design. It was not my task to reimagine SpeedShift so I stuck with it to the best of my ability even if I didn’t agree with their current design decisions.

When it came to technical testing I tested the most common use cases first. I wanted to make sure that typical use would work as expected as that functionality is the most important. Once it appeared as though the common use cases were tested I tried to identify as many edge cases as possible and handle them in a logical way that wouldn’t break anything else. One of the main edge cases was if the employee field on my main form was left empty. If that happens then nothing should be added because an employee name is fundamental to the data on the spreadsheet.

As you can see there was no easy way to test the code I had written. Testing was by far the most laborious process because of the number of combinations of worksheets, employees, shifts, assignments, and other details. I went one by one through each combination of test cases until I discovered a case that would break or until I thought of a case that I didn’t know how the program would handle.

The testing process went well given the limitations I encountered. I was able to successfully find and fix many bugs throughout the development process so that all the commonly used functionality works as expected.

SLOPD minimally tested my program as I updated it. They held off on a full rollout test until all functionality and features that they need were added. As of this writing, they have not implemented the software for testing or actual use purposes.

## Testing Results and Conclusions

Every element of the software works as expected. Each VBA module and UserForm has been tested and produces expected behavior. I feel confident that SLOPD can use this software with little concern over functionality issues. As I described previously, not every feature was able to be added from SpeedShift but what was added should work well.

However, I am aware that unexpected bugs or use-cases may develop as the program is used frequently. These are the cases I hope SLOPD can catch as I am not able to consider every possible test case. My own testing proved to be invaluable to the success and quality of the program. I believe that can be continued through continued testing and timely fixes.

## Future Work

I will continue to develop this project as long as SLOPD has an interest in it. I will continue to provide the following support as needed:

- Add any additional features that were not brought to my attention
- Additional testing for any elements of the program including new ones
- Fix any bugs or errors that come up during any testing or real-world use
- Provide email support to answer any questions, solve any problems, and provide help regarding the use of the program

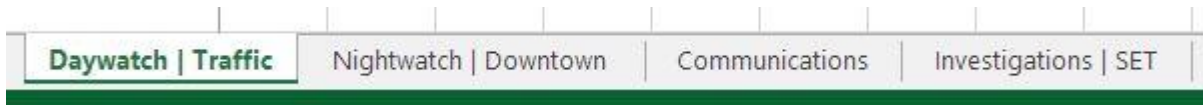
## References

- Eyre, Trevor. *CalendarForm*. Computer software. Vers. 1.4. N.p., 9 Sept. 2014. Web.  
<<https://www.dropbox.com/sh/0n9gyc32rlr31n2/AAAW7ADAWiLEZRHftERro9Tva?dl=0>>.
- InTime Solutions Inc. *SpeedShift User Guide*. N.p.: SoftScribes, 2003. Print.
- McRitchie, David. "Color Palette and the 56 Excel ColorIndex Colors." *Color Palette and the 56 Excel ColorIndex Colors*. N.p., 5 Apr. 2011. Web.  
<<http://dmcritchie.mvps.org/excel/colors.htm>>.
- "Office Support" *Microsoft Office Help and Support*. Microsoft, n.d. Web.  
<<https://support.office.com/en-GB>>.
- SpeedShift Software*. Digital image. *Officer.com*. N.p., n.d. Web.  
<[http://r2.officer.com/files/base/image/OFCR/2007/08/16x9/640x360/speedshiftsoftware\\_10043989.jpg](http://r2.officer.com/files/base/image/OFCR/2007/08/16x9/640x360/speedshiftsoftware_10043989.jpg)>.

## Appendix A (User Manual)

### *Navigation*

Navigation through the program is accessed through the sheet tabs located on the bottom of the Excel workbook as shown in Figure 4. Clicking on a tab will take you to that branch's schedule. There are currently nine tabs. The first eight tabs are for the separate departments within SLOPD and the last tab titled "Officers & Shifts" is used to maintain a list of officers and shifts that the program uses to populate menus.



**Figure 4: Tabs for separate sheets**

Other than the sheet tabs, most navigation will be contained within a worksheet and can be controlled using the scroll sliders on the right side and bottom of each worksheet. Each department's schedule contains all the days for the entire scheduling rotation (either four or five 28-day cycles).

Each worksheet has a mini-toolbar as seen in Figure 5. This toolbar is primarily where you will go to make changes to the schedule. There are four features



**Figure 5: Toolbar**

You can use the "Move Up" and "Move Down" buttons to reorder employees. To do this, simply select the officer that you would like to move then press "Move Up" or "Move Down" and that employee will be swapped with the employee above or below them.

You can clear a cell of all its contents (including colors and formatting) by using the "Clear Cell" button. To do this, select the cell that you would like to clear then press "Clear Cell".

### *Officers and Shifts*

Start by navigating to the last tab titled "Officers & Shifts". You should see a long list of employee names and shift titles. Here you can add or remove any employees and shifts according to your needs. The format for adding employees to the list is shown in Figure 6. Employees must be added in this fashion: Last name, First name, Call sign. This will ensure that all elements that use this list will work properly.



Doe	John	1
-----	------	---

**Figure 6: Employee format in employee list**

You can also add and remove any shifts associated with any department. Figure 7 shows how shifts are to be entered into the list.

Vacation Leave	VL	29	purple
----------------	----	----	--------

**Figure 7: Shift format in shift list**

The first column is the title of the shift. This is what you will see in any menu used throughout the program. This should typically be a common name for a shift such as “Vacation Leave”.

The second column is for the abbreviation. This is what will be displayed on the actual schedule itself if you select this shift for an employee. For vacation leave, “VL” has been selected. This abbreviation should be descriptive and easy to understand with just a quick glance without cluttering the schedule.

The third and fourth columns represent a cell color. The third column is the Excel color code as shown in Figure 8. The fourth column simply describes the color to make it easier for anyone looking at the shifts to know what color the number corresponds to. The number is what is used by the program NOT the name. Feel free to leave the name blank if you’d like.

You may want some shifts to be represented by a certain color so that they are more easily noticed when looking at the schedule. Many important shifts are colored so that you won’t easily miss them. Figure 8 shows a simple chart with 56 Excel colors that are available to be used.

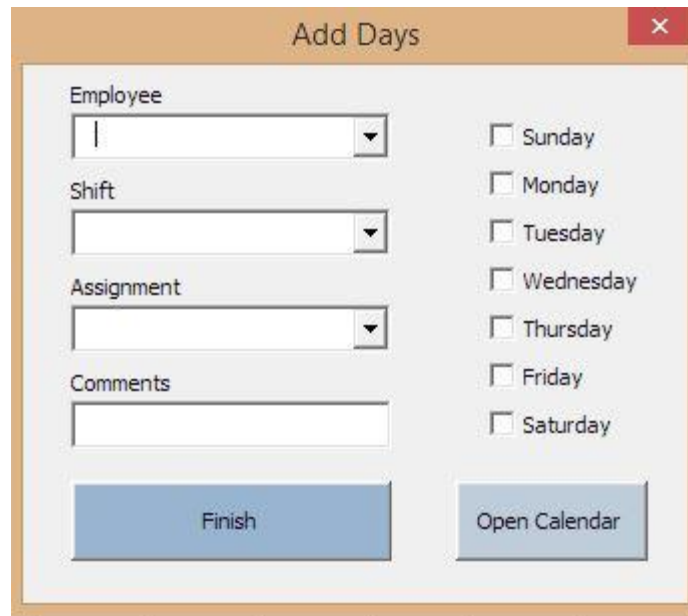
ColorIndex -- 56 Excel Colors [#colors56/#colorindex]							
[Color 0]	[Color 0]	[Color 15]	[Color 15]	[Color 30]	[Color 30]	[Color 45]	[Color 45]
	[Color 1]	[Color 16]	[Color 16]	[Color 31]	[Color 31]	[Color 46]	[Color 46]
[Color 2]		[Color 17]	[Color 17]	[Color 32]	[Color 32]	[Color 47]	[Color 47]
[Color 3]	[Color 3]	[Color 18]	[Color 18]	[Color 33]	[Color 33]	[Color 48]	[Color 48]
[Color 4]	[Color 4]	[Color 19]		[Color 34]	[Color 34]	[Color 49]	[Color 49]
[Color 5]	[Color 5]	[Color 20]		[Color 35]	[Color 35]	[Color 50]	[Color 50]
[Color 6]	[Color 6]	[Color 21]	[Color 21]	[Color 36]	[Color 36]	[Color 51]	[Color 51]
[Color 7]	[Color 7]	[Color 22]	[Color 22]	[Color 37]	[Color 37]	[Color 52]	[Color 52]
[Color 8]	[Color 8]	[Color 23]	[Color 23]	[Color 38]	[Color 38]	[Color 53]	[Color 53]
[Color 9]	[Color 9]	[Color 24]	[Color 24]	[Color 39]	[Color 39]	[Color 54]	[Color 54]
[Color 10]	[Color 10]		[Color 25]	[Color 40]	[Color 40]	[Color 55]	[Color 55]
	[Color 11]	[Color 26]	[Color 26]	[Color 41]	[Color 41]	[Color 56]	[Color 56]
[Color 12]	[Color 12]	[Color 27]	[Color 27]	[Color 42]	[Color 42]		
[Color 13]	[Color 13]	[Color 28]	[Color 28]	[Color 43]	[Color 43]		
[Color 14]	[Color 14]	[Color 29]	[Color 29]	[Color 44]	[Color 44]		

**Figure 8: Excel color options and their corresponding color code**



## ***Adding/Modifying an Employee***

Selecting the “Add/Modify” button on any worksheet to launch the UserForm used to add a new employee or modify an existing one. Figure 9 shows what the dialog will look like for the daywatch tab. Each sheet has a slightly different UserForm specific to that department’s needs but all of the basic functionality is the same.

The image shows a Windows-style dialog box titled "Add Days" with a red close button in the top right corner. The dialog is divided into two main sections. The left section contains four fields: "Employee" (a dropdown menu with a small arrow), "Shift" (a dropdown menu), "Assignment" (a dropdown menu), and "Comments" (a text input field). The right section contains seven checkboxes, one for each day of the week: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, and Saturday. At the bottom of the dialog, there are two buttons: "Finish" on the left and "Open Calendar" on the right.

**Figure 9: Adding or modifying an employee**

The first field titled “Employee” is a drop down list of employees from the “Officer & Shifts” sheet. You can start typing a last name and the program will suggest options. To modify this list, please read the above section entitled “Officers & Shifts”.

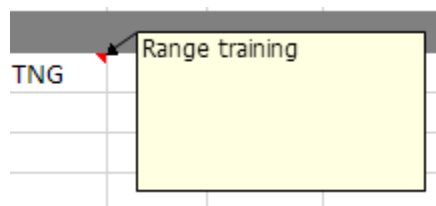
The second field titled “Shift” is a drop down list of shifts from the “Officers & Shifts” sheet. You can start typing a shift and it will suggest options that start with what you type. To modify this list, please read the above section entitled “Officers & Shifts”.

The third field titled “Assignment” lists the various assignment options for that specific sheet as seen in Figure 10. Each assignment is a dark gray header listed in the first column under the current date. As you can see in Figures 10 and 11, the assignment options are identical to the ones listed on the sheet. This is true for all worksheet. Some may have fewer assignments but to add to them you select the corresponding option in the “Assignment” drop down menu.

**Figure 10: Assignment list in dropdown**

**Figure 11: Daywatch assignments**

The fourth field titled “Comments” is used to make a comment to a shift or series of shifts. A comment is designated by a small red upper-right corner in the cell. To view the comment simply hover your mouse over the cell and it will popup. Figure 12 shows what a comment looks like on the schedule.



**Figure 12: Viewing a comment**

Comments are very useful when a specific shift needs a little more explanation. You can put the hours, location, or any other details into a comment so that others can see more about the shift.

On the right hand side of the UserForm is a list of checkboxes with days next to them. Use these to add shifts for the ENTIRE rotation. If you choose Sunday, Monday, and Tuesday the shift you select will be placed into every Sunday, Monday, and Tuesday for the whole rotation.

Below these checkboxes you will find a button that says “Open Calendar”. This button opens up a calendar like the one in Figure 13, where you can select an individual day. Many shifts such as vacation or sick leave don’t happen on a weekly basis. These are the shifts that are most suitable to use the calendar. To use the calendar, select a day and press “Okay” at the bottom. That will record the day you select to be used to fill in the shift.



**Figure 13: Calendar used to select a day**

## ***New Shift Rotation***

As you approach the end of a shift rotation it is time to create the next shift rotation. Underneath all of the assignments on the daywatch and nightwatch tabs is a button called “New Shift Rotation” shown in Figure 14.

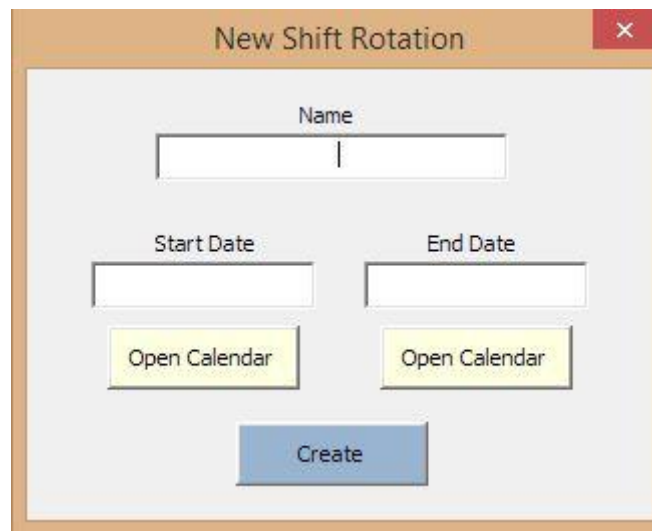


**Figure 14: Button to create a new shift rotation**

When you press this button you will be greeted with the UserForm displayed in Figure 15. This form allows you to enter the details of the new shift rotation so that the program knows what days to set it to. You will need to provide the name of the file to be created, the start date of the

rotation, and the end date of the rotation. You can enter the dates in manually in the format mm/dd/yyyy or you can use the “Open Calendar” to open the calendar (shown above in Figure 13) and select a date. Once you hit create you will need to wait a few seconds while the program processes all the information. This operation performs the following tasks:

- Creates a new file with the name you provided
- Copies over all the sheets from the current schedule
- Changes the dates to the dates you provided
- Deletes the patrol officers and supervisors listed in the “Daywatch | Traffic” and “Nightwatch | Downtown” tabs. All other employees remain on the schedule with their schedules removed.

The image shows a software window titled "New Shift Rotation" with a standard Windows-style title bar (orange) containing a close button (red 'X'). The main content area has a light gray background. At the top, there is a text label "Name" above a single-line text input field. Below this, there are two columns of controls. The left column has a text label "Start Date" above a text input field, which is followed by a yellow button labeled "Open Calendar". The right column has a text label "End Date" above a text input field, which is followed by a yellow button labeled "Open Calendar". At the bottom center of the form is a blue button labeled "Create".

**Figure 15: New shift rotation UserForm**

## Appendix B (VBA Code)

Much of the code is identical and should be refactored. As I got deep into development it was easier to manage multiple files making very small changes than it was to refactor all the code.

### *NewPayPeriod*

```
.....
'
'   NewPayPeriod
'   Developed by Josh Logier
'   jlogier@calpoly.edu
'
'   This UserForm is used to create a new workbook with new dates along the
'   top. It will also delete the employees in daywatch and nightwatch to be
'   able to move them around and start fresh.
'
'.....
```

```
Public Sub CommandButton1_Click()
    Dim startDay As Date

    startDay = CalendarForm.GetDate( _
        FirstDayOfWeek:=Monday, _
        DateFontSize:=12, _
        TodayButton:=True, _
        OkayButton:=True, _
        ShowWeekNumbers:=True, _
        BackgroundColor:=RGB(243, 249, 251), _
        HeaderColor:=RGB(147, 205, 2221), _
        HeaderFontColor:=RGB(255, 255, 255), _
        SubHeaderColor:=RGB(223, 240, 245), _
        SubHeaderFontColor:=RGB(31, 78, 120), _
        DateColor:=RGB(243, 249, 251), _
        DateFontColor:=RGB(31, 78, 120), _
        TrailingMonthFontColor:=RGB(155, 194, 230), _
        DateHoverColor:=RGB(223, 240, 245), _
        DateSelectedColor:=RGB(202, 223, 242), _
        SaturdayFontColor:=RGB(0, 176, 240), _
        SundayFontColor:=RGB(0, 176, 240), _
        TodayFontColor:=RGB(0, 176, 80))

    TextBox2.Value = Format(startDay, "mm/dd/yyyy")
End Sub
```

```
Public Sub CommandButton2_Click()
    Dim endDay As Date

    endDay = CalendarForm.GetDate( _
        FirstDayOfWeek:=Monday, _
        DateFontSize:=12, _
        TodayButton:=True, _
        OkayButton:=True, _
        ShowWeekNumbers:=True, _
        BackgroundColor:=RGB(243, 249, 251), _
        HeaderColor:=RGB(147, 205, 2221), _
        HeaderFontColor:=RGB(255, 255, 255), _
```

```

SubHeaderColor:=RGB(223, 240, 245), _
SubHeaderFontColor:=RGB(31, 78, 120), _
DateColor:=RGB(243, 249, 251), _
DateFontColor:=RGB(31, 78, 120), _
TrailingMonthFontColor:=RGB(155, 194, 230), _
DateHoverColor:=RGB(223, 240, 245), _
DateSelectedColor:=RGB(202, 223, 242), _
SaturdayFontColor:=RGB(0, 176, 240), _
SundayFontColor:=RGB(0, 176, 240), _
TodayFontColor:=RGB(0, 176, 80))

TextBox3.Value = Format(endDay, "mm/dd/yyyy")
End Sub

Private Sub CommandButton3_Click()
    Dim Wb1 As Workbook
    Dim Wb2 As Workbook
    Dim iRow As Integer
    Dim iCol As Integer
    Dim cycles5 As Integer
    Dim days As Integer
    Dim startDay As Date
    Dim endDay As Date
    Dim otDays As Integer
    Dim otNights As Integer
    Dim i As Integer

    iRow = 1
    iCol = 2
    cycles5 = 141 '140 days in a 5 28-day pay period + 1 offset

    startDay = DateValue(TextBox2.Value)
    endDay = DateValue(TextBox3.Value)
    days = DateDiff("d", startDay, endDay)

    With Application
        .ScreenUpdating = False
        .DisplayAlerts = False
        .EnableEvents = False
    End With

    Set Wb1 = ActiveWorkbook

    Set Wb2 = Application.Workbooks.Add(1)
    Wb1.Sheets(Array(Wb1.Sheets(1).Name, Wb1.Sheets(2).Name,
Wb1.Sheets(3).Name, Wb1.Sheets(4).Name, Wb1.Sheets(5).Name,
Wb1.Sheets(6).Name, Wb1.Sheets(7).Name, Wb1.Sheets(8).Name,
Wb1.Sheets(9).Name)).Copy Before:=Wb2.Sheets(1)
    Wb2.Sheets(Wb2.Sheets.Count).Delete

    ' Finds where "Overtime" (Days) is and deletes the officers before that
point
    Do
        Wb2.Sheets(1).Rows(4).EntireRow.Delete
    Loop While Not Wb2.Sheets(1).Cells(5, 1).Value = "Overtime"

```

```

' Finds where "Overtime" (Nights) is and deletes the officers before that
point
Do
'
    Wb2.Sheets(2).Rows(4).EntireRow.Delete
'
Loop While Not Wb2.Sheets(2).Cells(5, 1).Value = "Overtime"

iCol = 2
first = startDay
days = days + 1
Do
'
    If iCol <= days + 1 Then
        Wb2.Sheets(1).Cells(1, iCol) = Format(first, "dd-mmm")
        Wb2.Sheets(1).Cells(2, iCol) = Format(first, "DDD")
        Wb2.Sheets(2).Cells(1, iCol) = Format(first, "dd-mmm")
        Wb2.Sheets(2).Cells(2, iCol) = Format(first, "DDD")
        Wb2.Sheets(3).Cells(1, iCol) = Format(first, "dd-mmm")
        Wb2.Sheets(3).Cells(2, iCol) = Format(first, "DDD")
        Wb2.Sheets(4).Cells(1, iCol) = Format(first, "dd-mmm")
        Wb2.Sheets(4).Cells(2, iCol) = Format(first, "DDD")
        Wb2.Sheets(5).Cells(1, iCol) = Format(first, "dd-mmm")
        Wb2.Sheets(5).Cells(2, iCol) = Format(first, "DDD")
        Wb2.Sheets(6).Cells(1, iCol) = Format(first, "dd-mmm")
        Wb2.Sheets(6).Cells(2, iCol) = Format(first, "DDD")
        Wb2.Sheets(7).Cells(1, iCol) = Format(first, "dd-mmm")
        Wb2.Sheets(7).Cells(2, iCol) = Format(first, "DDD")
        Wb2.Sheets(8).Cells(1, iCol) = Format(first, "dd-mmm")
        Wb2.Sheets(8).Cells(2, iCol) = Format(first, "DDD")
        Wb2.Sheets(9).Cells(1, iCol) = Format(first, "dd-mmm")
        Wb2.Sheets(9).Cells(2, iCol) = Format(first, "DDD")
    ElseIf iCol > days + 1 Then
        Wb2.Sheets(1).Cells(1, iCol).ClearContents
        Wb2.Sheets(1).Cells(2, iCol).ClearContents
        Wb2.Sheets(2).Cells(1, iCol).ClearContents
        Wb2.Sheets(2).Cells(2, iCol).ClearContents
        Wb2.Sheets(3).Cells(1, iCol).ClearContents
        Wb2.Sheets(3).Cells(2, iCol).ClearContents
        Wb2.Sheets(4).Cells(1, iCol).ClearContents
        Wb2.Sheets(4).Cells(2, iCol).ClearContents
        Wb2.Sheets(5).Cells(1, iCol).ClearContents
        Wb2.Sheets(5).Cells(2, iCol).ClearContents
        Wb2.Sheets(6).Cells(1, iCol).ClearContents
        Wb2.Sheets(6).Cells(2, iCol).ClearContents
        Wb2.Sheets(7).Cells(1, iCol).ClearContents
        Wb2.Sheets(7).Cells(2, iCol).ClearContents
        Wb2.Sheets(8).Cells(1, iCol).ClearContents
        Wb2.Sheets(8).Cells(2, iCol).ClearContents
        Wb2.Sheets(9).Cells(1, iCol).ClearContents
        Wb2.Sheets(9).Cells(2, iCol).ClearContents
    End If

    first = first + 1
    iCol = iCol + 1

```

```

'
Loop While iCol <= cycles5

Wb2.SaveAs Filename:="c:\Users\Josh\Desktop\" & TextBox1.Value,
FileFormat:=52

Wb2.Close
With Application
    .ScreenUpdating = True
    .DisplayAlerts = True
    .EnableEvents = True
End With

Unload Me
End Sub

```

## ***AddAdministration***

```

'
'
' AddAdministration
' Developed by Josh Logier
' jlogier@calpoly.edu
'
' This UserForm takes an employee name, assignment, shift, comment,
' and date (or days) to create a new employee on the sheet.
'
'
'
Public pickDay As Date

Private Sub Userform_initialize()
    Dim ws As Worksheet
    Dim i As Integer
    Dim employee As String
    Dim shift As String

    Set ws = Worksheets("Officers & Shifts")

    i = 1

    ' Adds employees from the list of employees on the last worksheet
    With ComboBox2
        Do
            employee = ws.Cells(i, 1).Value & ", " & ws.Cells(i, 2).Value & "
" & ws.Cells(i, 3).Value
            .AddItem employee
            i = i + 1
        Loop While Not IsEmpty(ws.Cells(i, 1))
    End With

    i = 1

    ' Adds the shifts from the shift list on the last worksheet
    With ComboBox3
        Do
            shift = ws.Cells(i, 6).Value

```



```

        .AddItem shift
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 6))
End With

With ComboBox1
    .AddItem "Regular Duty"
    .AddItem "Light Duty"
End With

CheckBox1.Value = False
CheckBox2.Value = False
CheckBox3.Value = False
CheckBox4.Value = False
CheckBox5.Value = False
CheckBox6.Value = False
CheckBox7.Value = False

End Sub

Private Sub CommandButton1_Click()
    Dim ws As Worksheet
    Dim iRow As Integer
    Dim iCol As Integer
    Dim days(7) As String
    Dim matchRow As Integer
    Dim index As Integer
    Dim officer As String
    Dim abbrev As String
    Dim shift As String
    Dim i As Integer
    Dim colorIndex As Integer

    Set ws = Worksheets("Officers & Shifts")

    iCol = 1
    iRow = 2
    matchRow = 1
    index = 3
    colorIndex = 0

    If Not ComboBox2 = "" Then
        officer = ComboBox2
        Do
            '
            If officer = Cells(index, 1).Value Then
                matchRow = index
            End If
            index = index + 1
        Loop While index < 75

        ' Placing employee into shift section indicated on the UserForm
        If matchRow = 1 Then
            If ComboBox1 = "Normal Duty" Then
                iRow = iRow + 1
            ElseIf ComboBox1 = "Light Duty" Then

```

```

        Do
            '
            iRow = iRow + 1
            '
            Loop While Not Cells(iRow, 1).Value = "Light Duty"

            iRow = iRow + 1
        Else
            iRow = 4
        End If

        Rows(iRow).EntireRow.Insert
        Rows(iRow).EntireRow.ClearFormats
        Rows(iRow).EntireRow.Select
        Cells(iRow, 1).Value = officer
    Else
        iRow = matchRow
    End If

    ' Setting the days array to the day if checkbox is checked otherwise
null
    If CheckBox1 = True Then
        days(0) = "SUN"
    Else
        days(0) = "NULL"
    End If
    If CheckBox2 = True Then
        days(1) = "MON"
    Else
        days(1) = "NULL"
    End If
    If CheckBox3 = True Then
        days(2) = "TUE"
    Else
        days(2) = "NULL"
    End If
    If CheckBox4 = True Then
        days(3) = "WED"
    Else
        days(3) = "NULL"
    End If
    If CheckBox5 = True Then
        days(4) = "THU"
    Else
        days(4) = "NULL"
    End If
    If CheckBox6 = True Then
        days(5) = "FRI"
    Else
        days(5) = "NULL"
    End If
    If CheckBox7 = True Then
        days(6) = "SAT"
    Else
        days(6) = "NULL"
    End If

```

```

' Sets the shift value to its abbreviation so it can be placed in the
spreadsheet
shift = ComboBox3
i = 1
Do
    If shift = ws.Cells(i, 6) Then
        abbrev = ws.Cells(i, 7)
        colorIndex = ws.Cells(i, 8)
    End If
    i = i + 1
Loop While Not IsEmpty(ws.Cells(i, 6))

' If any days are checked, put shift value in those days for entire
trimester
If CheckBox1 = True Or CheckBox2 = True Or CheckBox3 = True Or
CheckBox4 = True Or CheckBox5 = True Or CheckBox6 = True Or CheckBox7 = True
Then
    Do
        iCol = iCol + 1
        If days(0) = "SUN" And Cells(2, iCol).Value = "SUN" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(1) = "MON" And Cells(2, iCol).Value = "MON" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(2) = "TUE" And Cells(2, iCol).Value = "TUE" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(3) = "WED" And Cells(2, iCol).Value = "WED" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then

```

```

        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(4) = "THU" And Cells(2, iCol).Value = "THU" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(5) = "FRI" And Cells(2, iCol).Value = "FRI" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(6) = "SAT" And Cells(2, iCol).Value = "SAT" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
End If
Loop Until IsEmpty(Cells(1, iCol))
Else
    ' No days were checked; get value from calendar to enter an
individual day
    Do
        iCol = iCol + 1
    Loop Until Cells(1, iCol).Value = pickDay
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
End If
End If

Unload Me
End Sub

Public Sub CommandButton2_Click()

```

```

pickDay = CalendarForm.GetDate( _
    FirstDayOfWeek:=Monday, _
    DateFontSize:=12, _
    TodayButton:=True, _
    OkayButton:=True, _
    ShowWeekNumbers:=True, _
    BackgroundColor:=RGB(243, 249, 251), _
    HeaderColor:=RGB(147, 205, 221), _
    HeaderFontColor:=RGB(255, 255, 255), _
    SubHeaderColor:=RGB(223, 240, 245), _
    SubHeaderFontColor:=RGB(31, 78, 120), _
    DateColor:=RGB(243, 249, 251), _
    DateFontColor:=RGB(31, 78, 120), _
    TrailingMonthFontColor:=RGB(155, 194, 230), _
    DateHoverColor:=RGB(223, 240, 245), _
    DateSelectedColor:=RGB(202, 223, 242), _
    SaturdayFontColor:=RGB(0, 176, 240), _
    SundayFontColor:=RGB(0, 176, 240), _
    TodayFontColor:=RGB(0, 176, 80))
End Sub

```

## ***AddCommunications***

```

'
' AddCommunications
' Developed by Josh Logier
' jlogier@calpoly.edu
'
' This UserForm takes an employee name, assignment, shift, comment,
' and date (or days) to create a new employee on the sheet.
'
'
Public pickDay As Date

Private Sub Userform_initialize()
    Dim ws As Worksheet
    Dim i As Integer
    Dim employee As String
    Dim shift As String

    Set ws = Worksheets("Officers & Shifts")

    i = 1

    ' Adds employees from the list of employees on the last worksheet
    With ComboBox2
        Do
            employee = ws.Cells(i, 1).Value & ", " & ws.Cells(i, 2).Value & "
" & ws.Cells(i, 3).Value
            .AddItem employee
            i = i + 1
        Loop While Not IsEmpty(ws.Cells(i, 1))
    End With

```

```

i = 1

' Adds the shifts from the shift list on the last worksheet
With ComboBox3
    Do
        shift = ws.Cells(i, 6).Value
        .AddItem shift
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 6))
End With

With ComboBox1
    .AddItem "Days"
    .AddItem "Nights"
End With

CheckBox1.Value = False
CheckBox2.Value = False
CheckBox3.Value = False
CheckBox4.Value = False
CheckBox5.Value = False
CheckBox6.Value = False
CheckBox7.Value = False

End Sub

Private Sub CommandButton1_Click()
    Dim ws As Worksheet
    Dim iRow As Integer
    Dim iCol As Integer
    Dim days(7) As String
    Dim matchRow As Integer
    Dim index As Integer
    Dim officer As String
    Dim abbrev As String
    Dim shift As String
    Dim i As Integer
    Dim colorIndex As Integer

    Set ws = Worksheets("Officers & Shifts")

    iCol = 1
    iRow = 4
    matchRow = 1
    index = 3
    colorIndex = 0

    If Not ComboBox2 = "" Then
        officer = ComboBox2
        Do
            '
            If officer = Cells(index, 1).Value Then
                matchRow = index
            End If
            index = index + 1
        Loop While index < 75
    
```

```

' Placing employee into shift section indicated on the UserForm
If matchRow = 1 Then
    If ComboBox1 = "Days" Then
        iRow = iRow + 1
    ElseIf ComboBox1 = "Nights" Then
        Do
            '
            iRow = iRow + 1
            '
        Loop While Not Cells(iRow, 1).Value = "Nights"

        iRow = iRow + 1
    Else
        iRow = 4
    End If

    Rows(iRow).EntireRow.Insert
    Rows(iRow).EntireRow.ClearFormats
    Rows(iRow).EntireRow.Select
    Cells(iRow, 1).Value = officer
Else
    iRow = matchRow
End If

' Setting the days array to the day if checkbox is checked otherwise
null
If CheckBox1 = True Then
    days(0) = "SUN"
Else
    days(0) = "NULL"
End If
If CheckBox2 = True Then
    days(1) = "MON"
Else
    days(1) = "NULL"
End If
If CheckBox3 = True Then
    days(2) = "TUE"
Else
    days(2) = "NULL"
End If
If CheckBox4 = True Then
    days(3) = "WED"
Else
    days(3) = "NULL"
End If
If CheckBox5 = True Then
    days(4) = "THU"
Else
    days(4) = "NULL"
End If
If CheckBox6 = True Then
    days(5) = "FRI"
Else
    days(5) = "NULL"
End If

```

```

If CheckBox7 = True Then
    days(6) = "SAT"
Else
    days(6) = "NULL"
End If

' Sets the shift value to its abbreviation so it can be placed in the
spreadsheet
shift = ComboBox3
i = 1
Do
    If shift = ws.Cells(i, 6) Then
        abbrev = ws.Cells(i, 7)
        colorIndex = ws.Cells(i, 8)
    End If
    i = i + 1
Loop While Not IsEmpty(ws.Cells(i, 6))

' If any days are checked, put shift value in those days for entire
trimester
If CheckBox1 = True Or CheckBox2 = True Or CheckBox3 = True Or
CheckBox4 = True Or CheckBox5 = True Or CheckBox6 = True Or CheckBox7 = True
Then
    Do
        iCol = iCol + 1
        If days(0) = "SUN" And Cells(2, iCol).Value = "SUN" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(1) = "MON" And Cells(2, iCol).Value = "MON" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(2) = "TUE" And Cells(2, iCol).Value = "TUE" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(3) = "WED" And Cells(2, iCol).Value = "WED" Then

```



```

Cells(iRow, iCol).Value = abbrev
Cells(iRow, iCol).Interior.colorIndex = colorIndex
If colorIndex = 1 Then
    Cells(iRow, iCol).Font.Color = vbWhite
End If
If Not TextBox4.Value = "" Then
    Cells(iRow, iCol).AddComment
    Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
End If
ElseIf days(4) = "THU" And Cells(2, iCol).Value = "THU" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(5) = "FRI" And Cells(2, iCol).Value = "FRI" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(6) = "SAT" And Cells(2, iCol).Value = "SAT" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
End If
Loop Until IsEmpty(Cells(1, iCol))
Else
    ' No days were checked; get value from calendar to enter an
individual day
    Do
        iCol = iCol + 1
    Loop Until Cells(1, iCol).Value = pickDay
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
End If

```

```

        End If

        Unload Me
    End Sub

Public Sub CommandButton2_Click()
    pickDay = CalendarForm.GetDate( _
        FirstDayOfWeek:=Monday, _
        DateFontSize:=12, _
        TodayButton:=True, _
        OkayButton:=True, _
        ShowWeekNumbers:=True, _
        BackgroundColor:=RGB(243, 249, 251), _
        HeaderColor:=RGB(147, 205, 221), _
        HeaderFontColor:=RGB(255, 255, 255), _
        SubHeaderColor:=RGB(223, 240, 245), _
        SubHeaderFontColor:=RGB(31, 78, 120), _
        DateColor:=RGB(243, 249, 251), _
        DateFontColor:=RGB(31, 78, 120), _
        TrailingMonthFontColor:=RGB(155, 194, 230), _
        DateHoverColor:=RGB(223, 240, 245), _
        DateSelectedColor:=RGB(202, 223, 242), _
        SaturdayFontColor:=RGB(0, 176, 240), _
        SundayFontColor:=RGB(0, 176, 240), _
        TodayFontColor:=RGB(0, 176, 80))
End Sub

```

## ***AddDays***

```

.....
'
'   AddDays
'   Developed by Josh Logier
'   jlogier@calpoly.edu
'
'   This UserForm takes an employee name, assignment, shift, comment,
'   and date (or days) to create a new employee on the sheet.
'
.....

```

```

Public pickDay As Date

Private Sub Userform_initialize()
    Dim ws As Worksheet
    Dim i As Integer
    Dim employee As String
    Dim shift As String

    Set ws = Worksheets("Officers & Shifts")

    i = 1

    ' Adds employees from the list of employees on the last worksheet
    With ComboBox2
        Do

```

```

        employee = ws.Cells(i, 1).Value & ", " & ws.Cells(i, 2).Value & "
" & ws.Cells(i, 3).Value
        .AddItem employee
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 1))
End With

i = 1

' Adds the shifts from the shift list on the last worksheet
With ComboBox3
    Do
        shift = ws.Cells(i, 6).Value
        .AddItem shift
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 6))
End With

With ComboBox1
    .AddItem "Supervisor"
    .AddItem "Officer"
    .AddItem "Daywatch OT"
    .AddItem "Downtown Days"
    .AddItem "CAT"
    .AddItem "Traffic"
    .AddItem "FST"
    .AddItem "Trainee"
End With

CheckBox1.Value = False
CheckBox2.Value = False
CheckBox3.Value = False
CheckBox4.Value = False
CheckBox5.Value = False
CheckBox6.Value = False
CheckBox7.Value = False

End Sub

Private Sub CommandButton1_Click()
    Dim ws As Worksheet
    Dim iRow As Integer
    Dim iCol As Integer
    Dim days(7) As String
    Dim matchRow As Integer
    Dim index As Integer
    Dim officer As String
    Dim abbrev As String
    Dim shift As String
    Dim i As Integer
    Dim colorIndex As Integer

    Set ws = Worksheets("Officers & Shifts")

    iCol = 1
    iRow = 4
    matchRow = 1

```

```

index = 3
colorIndex = 0

If Not ComboBox2 = "" Then
    officer = ComboBox2
    Do
        '
        If officer = Cells(index, 1).Value Then
            matchRow = index
        End If
        index = index + 1
    '
Loop While index < 75

' Placing employee into shift section indicated on the UserForm
If matchRow = 1 Then
    If ComboBox1 = "Supervisor" Then
        iRow = iRow + 1
    ElseIf ComboBox1 = "Officer" Then
        Do
            '
            iRow = iRow + 1
            '
            Loop While Not Cells(iRow, 1).Value = "Officers"

            iRow = iRow + 1
        ElseIf ComboBox1 = "Daywatch OT" Then
            Do
                '
                iRow = iRow + 1
                '
                Loop While Not Cells(iRow, 1).Value = "Overtime"

                iRow = iRow + 1
            ElseIf ComboBox1 = "Downtown Days" Then
                Do
                    '
                    iRow = iRow + 1
                    '
                    Loop While Not Cells(iRow, 1).Value = "Downtown Days"

                    iRow = iRow + 1
                ElseIf ComboBox1 = "CAT" Then
                    Do
                        '
                        iRow = iRow + 1
                        '
                        Loop While Not Cells(iRow, 1).Value = "CAT"

                        iRow = iRow + 1
                    ElseIf ComboBox1 = "Traffic" Then
                        Do
                            '
                            iRow = iRow + 1
                            '
                            Loop While Not Cells(iRow, 1).Value = "Traffic"

```

```

        iRow = iRow + 1
    ElseIf ComboBox1 = "FST" Then
        Do
            '
            iRow = iRow + 1
            '
            Loop While Not Cells(iRow, 1).Value = "FST"

            iRow = iRow + 1
        ElseIf ComboBox1 = "Trainee" Then
            Do
                '
                iRow = iRow + 1
                '
                Loop While Not Cells(iRow, 1).Value = "Trainee"

            iRow = iRow + 1
        Else
            iRow = 4
        End If

        Rows(iRow).EntireRow.Insert
        Rows(iRow).EntireRow.ClearFormats
        Rows(iRow).EntireRow.Select
        Cells(iRow, 1).Value = officer
    Else
        iRow = matchRow
    End If

    ' Setting the days array to the day if checkbox is checked otherwise
null
    If CheckBox1 = True Then
        days(0) = "SUN"
    Else
        days(0) = "NULL"
    End If
    If CheckBox2 = True Then
        days(1) = "MON"
    Else
        days(1) = "NULL"
    End If
    If CheckBox3 = True Then
        days(2) = "TUE"
    Else
        days(2) = "NULL"
    End If
    If CheckBox4 = True Then
        days(3) = "WED"
    Else
        days(3) = "NULL"
    End If
    If CheckBox5 = True Then
        days(4) = "THU"
    Else
        days(4) = "NULL"
    End If
    If CheckBox6 = True Then

```



```

End If
ElseIf days(2) = "TUE" And Cells(2, iCol).Value = "TUE" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(3) = "WED" And Cells(2, iCol).Value = "WED" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(4) = "THU" And Cells(2, iCol).Value = "THU" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(5) = "FRI" And Cells(2, iCol).Value = "FRI" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(6) = "SAT" And Cells(2, iCol).Value = "SAT" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
End If
Loop Until IsEmpty(Cells(1, iCol))
Else
    ' No days were checked; get value from calendar to enter an
individual day
Do

```





```

Dim i As Integer
Dim employee As String
Dim shift As String

Set ws = Worksheets("Officers & Shifts")

i = 1

' Adds employees from the list of employees on the last worksheet
With ComboBox2
    Do
        employee = ws.Cells(i, 1).Value & ", " & ws.Cells(i, 2).Value & "
" & ws.Cells(i, 3).Value
        .AddItem employee
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 1))
End With

i = 1

' Adds the shifts from the shift list on the last worksheet
With ComboBox3
    Do
        shift = ws.Cells(i, 6).Value
        .AddItem shift
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 6))
End With

With ComboBox1
    .AddItem "Daywatch"
    .AddItem "SET"
    .AddItem "SNU"
End With

CheckBox1.Value = False
CheckBox2.Value = False
CheckBox3.Value = False
CheckBox4.Value = False
CheckBox5.Value = False
CheckBox6.Value = False
CheckBox7.Value = False

End Sub

Private Sub CommandButton1_Click()
    Dim ws As Worksheet
    Dim iRow As Integer
    Dim iCol As Integer
    Dim days(7) As String
    Dim matchRow As Integer
    Dim index As Integer
    Dim officer As String
    Dim abbrev As String
    Dim shift As String
    Dim i As Integer

```

```

Set ws = Worksheets("Officers & Shifts")

iCol = 1
iRow = 3
matchRow = 1
index = 3
colorIndex = 0

If Not ComboBox2 = "" Then
    officer = ComboBox2
    Do
        '
        If officer = Cells(index, 1).Value Then
            matchRow = index
        End If
        index = index + 1
    Loop While index < 75

    ' Placing employee into shift section indicated on the UserForm
    If matchRow = 1 Then
        If ComboBox1 = "Daywatch" Then
            iRow = iRow + 1
        ElseIf ComboBox1 = "SET" Then
            Do
                '
                iRow = iRow + 1
            Loop While Not Cells(iRow, 1).Value = "SET"

            iRow = iRow + 1
        ElseIf ComboBox1 = "SNU" Then
            Do
                '
                iRow = iRow + 1
            Loop While Not Cells(iRow, 1).Value = "SNU"

            iRow = iRow + 1
        Else
            iRow = 4
        End If

        Rows(iRow).EntireRow.Insert
        Rows(iRow).EntireRow.ClearFormats
        Rows(iRow).EntireRow.Select
        Cells(iRow, 1).Value = officer
    Else
        iRow = matchRow
    End If

    ' Setting the days array to the day if checkbox is checked otherwise
null
    If CheckBox1 = True Then
        days(0) = "SUN"
    Else
        days(0) = "NULL"
    End If

```

```

End If
If CheckBox2 = True Then
    days(1) = "MON"
Else
    days(1) = "NULL"
End If
If CheckBox3 = True Then
    days(2) = "TUE"
Else
    days(2) = "NULL"
End If
If CheckBox4 = True Then
    days(3) = "WED"
Else
    days(3) = "NULL"
End If
If CheckBox5 = True Then
    days(4) = "THU"
Else
    days(4) = "NULL"
End If
If CheckBox6 = True Then
    days(5) = "FRI"
Else
    days(5) = "NULL"
End If
If CheckBox7 = True Then
    days(6) = "SAT"
Else
    days(6) = "NULL"
End If

' Sets the shift value to its abbreviation so it can be placed in the
spreadsheet
shift = ComboBox3
i = 1
Do
    If shift = ws.Cells(i, 6) Then
        abbrev = ws.Cells(i, 7)
    End If
    i = i + 1
Loop While Not IsEmpty(ws.Cells(i, 6))

' If any days are checked, put shift value in those days for entire
trimester
If CheckBox1 = True Or CheckBox2 = True Or CheckBox3 = True Or
CheckBox4 = True Or CheckBox5 = True Or CheckBox6 = True Or CheckBox7 = True
Then
    Do
        iCol = iCol + 1
        If days(0) = "SUN" And Cells(2, iCol).Value = "SUN" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.ColorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
        End If
    
```

```

        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    ElseIf days(1) = "MON" And Cells(2, iCol).Value = "MON" Then
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.colorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    ElseIf days(2) = "TUE" And Cells(2, iCol).Value = "TUE" Then
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.colorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    ElseIf days(3) = "WED" And Cells(2, iCol).Value = "WED" Then
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.colorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    ElseIf days(4) = "THU" And Cells(2, iCol).Value = "THU" Then
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.colorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    ElseIf days(5) = "FRI" And Cells(2, iCol).Value = "FRI" Then
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.colorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    ElseIf days(6) = "SAT" And Cells(2, iCol).Value = "SAT" Then
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.colorIndex = colorIndex

```

```

        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    End If
    Loop Until IsEmpty(Cells(1, iCol))
Else
    ' No days were checked; get value from calendar to enter an
individual day
    Do
        iCol = iCol + 1
    Loop Until Cells(1, iCol).Value = pickDay
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.ColorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
End If

Unload Me
End Sub

Public Sub CommandButton2_Click()
    pickDay = CalendarForm.GetDate( _
        FirstDayOfWeek:=Monday, _
        DateFontSize:=12, _
        TodayButton:=True, _
        OkayButton:=True, _
        ShowWeekNumbers:=True, _
        BackgroundColor:=RGB(243, 249, 251), _
        HeaderColor:=RGB(147, 205, 222), _
        HeaderFontColor:=RGB(255, 255, 255), _
        SubHeaderColor:=RGB(223, 240, 245), _
        SubHeaderFontColor:=RGB(31, 78, 120), _
        DateColor:=RGB(243, 249, 251), _
        DateFontColor:=RGB(31, 78, 120), _
        TrailingMonthFontColor:=RGB(155, 194, 230), _
        DateHoverColor:=RGB(223, 240, 245), _
        DateSelectedColor:=RGB(202, 223, 242), _
        SaturdayFontColor:=RGB(0, 176, 240), _
        SundayFontColor:=RGB(0, 176, 240), _
        TodayFontColor:=RGB(0, 176, 80))
End Sub

```

## ***AddNights***

```

.....

```

```

' AddNights
' Developed by Josh Logier
' jlogier@calpoly.edu
'
' This UserForm takes an employee name, assignment, shift, comment,
' and date (or days) to create a new employee on the sheet.
'
.....

Public pickDay As Date

Private Sub Userform_initialize()
    Dim ws As Worksheet
    Dim i As Integer
    Dim employee As String
    Dim shift As String

    Set ws = Worksheets("Officers & Shifts")

    i = 1

    ' Adds employees from the list of employees on the last worksheet
    With ComboBox2
        Do
            employee = ws.Cells(i, 1).Value & ", " & ws.Cells(i, 2).Value & "
" & ws.Cells(i, 3).Value
            .AddItem employee
            i = i + 1
        Loop While Not IsEmpty(ws.Cells(i, 1))
    End With

    i = 1

    ' Adds the shifts from the shift list on the last worksheet
    With ComboBox3
        Do
            shift = ws.Cells(i, 6).Value
            .AddItem shift
            i = i + 1
        Loop While Not IsEmpty(ws.Cells(i, 6))
    End With

    With ComboBox1
        .AddItem "Supervisor"
        .AddItem "Officer"
        .AddItem "Nightwatch OT"
        .AddItem "DUI Officer"
        .AddItem "Downtown Nights"
    End With

    CheckBox1.Value = False
    CheckBox2.Value = False
    CheckBox3.Value = False
    CheckBox4.Value = False
    CheckBox5.Value = False
    CheckBox6.Value = False
    CheckBox7.Value = False

```

End Sub

```
Private Sub CommandButton1_Click()  
    Dim ws As Worksheet  
    Dim iRow As Integer  
    Dim iCol As Integer  
    Dim days(7) As String  
    Dim matchRow As Integer  
    Dim index As Integer  
    Dim officer As String  
    Dim abbrev As String  
    Dim shift As String  
    Dim i As Integer  
    Dim colorIndex As Integer  
  
    Set ws = Worksheets("Officers & Shifts")  
  
    iCol = 1  
    iRow = 4  
    matchRow = 1  
    index = 3  
  
    If Not ComboBox2 = "" Then  
        officer = ComboBox2  
        Do  
            '   
            If officer = Cells(index, 1).Value Then  
                matchRow = index  
            End If  
            index = index + 1  
        Loop While index < 75  
  
        ' Placing employee into shift section indicated on the UserForm  
        If matchRow = 1 Then  
            If ComboBox1 = "Supervisor" Then  
                iRow = iRow + 1  
            ElseIf ComboBox1 = "Officer" Then  
                Do  
                    '   
                    iRow = iRow + 1  
                Loop While Not Cells(iRow, 1).Value = "Officers"  
  
                iRow = iRow + 1  
            ElseIf ComboBox1 = "Nightwatch OT" Then  
                Do  
                    '   
                    iRow = iRow + 1  
                Loop While Not Cells(iRow, 1).Value = "Overtime"  
  
                iRow = iRow + 1  
            ElseIf ComboBox1 = "DUI Officer" Then  
                Do  
                    '   

```

```

        iRow = iRow + 1
    '
    Loop While Not Cells(iRow, 1).Value = "DUI Officer"

    iRow = iRow + 1
ElseIf ComboBox1 = "Downtown Nights" Then
    Do
    '
        iRow = iRow + 1
    '
    Loop While Not Cells(iRow, 1).Value = "Downtown Nights"

    iRow = iRow + 1
Else
    iRow = 4
End If

Rows(iRow).EntireRow.Insert
Rows(iRow).EntireRow.ClearFormats
Rows(iRow).EntireRow.Select
Cells(iRow, 1).Value = officer
Else
    iRow = matchRow
End If

' Setting the days array to the day if checkbox is checked otherwise
null
If CheckBox1 = True Then
    days(0) = "SUN"
Else
    days(0) = "NULL"
End If
If CheckBox2 = True Then
    days(1) = "MON"
Else
    days(1) = "NULL"
End If
If CheckBox3 = True Then
    days(2) = "TUE"
Else
    days(2) = "NULL"
End If
If CheckBox4 = True Then
    days(3) = "WED"
Else
    days(3) = "NULL"
End If
If CheckBox5 = True Then
    days(4) = "THU"
Else
    days(4) = "NULL"
End If
If CheckBox6 = True Then
    days(5) = "FRI"
Else
    days(5) = "NULL"
End If

```



```

If CheckBox7 = True Then
    days(6) = "SAT"
Else
    days(6) = "NULL"
End If

' Sets the shift value to its abbreviation so it can be placed in the
spreadsheet
shift = ComboBox3
i = 1
Do
    If shift = ws.Cells(i, 6) Then
        abbrev = ws.Cells(i, 7)
        colorIndex = ws.Cells(i, 8)
    End If
    i = i + 1
Loop While Not IsEmpty(ws.Cells(i, 6))

' If any days are checked, put shift value in those days for entire
trimester
If CheckBox1 = True Or CheckBox2 = True Or CheckBox3 = True Or
CheckBox4 = True Or CheckBox5 = True Or CheckBox6 = True Or CheckBox7 = True
Then
    Do
        iCol = iCol + 1
        If days(0) = "SUN" And Cells(2, iCol).Value = "SUN" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(1) = "MON" And Cells(2, iCol).Value = "MON" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(2) = "TUE" And Cells(2, iCol).Value = "TUE" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(3) = "WED" And Cells(2, iCol).Value = "WED" Then

```

```

Cells(iRow, iCol).Value = abbrev
Cells(iRow, iCol).Interior.colorIndex = colorIndex
If colorIndex = 1 Then
    Cells(iRow, iCol).Font.Color = vbWhite
End If
If Not TextBox4.Value = "" Then
    Cells(iRow, iCol).AddComment
    Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
End If
ElseIf days(4) = "THU" And Cells(2, iCol).Value = "THU" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(5) = "FRI" And Cells(2, iCol).Value = "FRI" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(6) = "SAT" And Cells(2, iCol).Value = "SAT" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
End If
Loop Until IsEmpty(Cells(1, iCol))
Else
    ' No days were checked; get value from calendar to enter an
individual day
    Do
        iCol = iCol + 1
    Loop Until Cells(1, iCol).Value = pickDay
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
End If

```

```

        End If

        Unload Me
    End Sub

Public Sub CommandButton2_Click()
    pickDay = CalendarForm.GetDate( _
        FirstDayOfWeek:=Monday, _
        DateFontSize:=12, _
        TodayButton:=True, _
        OkayButton:=True, _
        ShowWeekNumbers:=True, _
        BackgroundColor:=RGB(243, 249, 251), _
        HeaderColor:=RGB(147, 205, 221), _
        HeaderFontColor:=RGB(255, 255, 255), _
        SubHeaderColor:=RGB(223, 240, 245), _
        SubHeaderFontColor:=RGB(31, 78, 120), _
        DateColor:=RGB(243, 249, 251), _
        DateFontColor:=RGB(31, 78, 120), _
        TrailingMonthFontColor:=RGB(155, 194, 230), _
        DateHoverColor:=RGB(223, 240, 245), _
        DateSelectedColor:=RGB(202, 223, 242), _
        SaturdayFontColor:=RGB(0, 176, 240), _
        SundayFontColor:=RGB(0, 176, 240), _
        TodayFontColor:=RGB(0, 176, 80))
End Sub

```

## ***AddRecords***

```

' .....
'
'   AddRecords
'   Developed by Josh Logier
'   jlogier@calpoly.edu
'
'   This UserForm takes an employee name, shift, comment,
'   and date (or days) to create a new employee on the sheet.
'
' .....

```

```

Public pickDay As Date

Private Sub Userform_initialize()
    Dim ws As Worksheet
    Dim i As Integer
    Dim employee As String
    Dim shift As String

    Set ws = Worksheets("Officers & Shifts")

    i = 1

    ' Adds employees from the list of employees on the last worksheet
    With ComboBox2
        Do

```

```

        employee = ws.Cells(i, 1).Value & ", " & ws.Cells(i, 2).Value & "
" & ws.Cells(i, 3).Value
        .AddItem employee
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 1))
End With

i = 1

' Adds the shifts from the shift list on the last worksheet
With ComboBox3
    Do
        shift = ws.Cells(i, 6).Value
        .AddItem shift
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 6))
End With

CheckBox1.Value = False
CheckBox2.Value = False
CheckBox3.Value = False
CheckBox4.Value = False
CheckBox5.Value = False
CheckBox6.Value = False
CheckBox7.Value = False

End Sub

Private Sub CommandButton1_Click()
    Dim ws As Worksheet
    Dim iRow As Integer
    Dim iCol As Integer
    Dim days(7) As String
    Dim matchRow As Integer
    Dim index As Integer
    Dim officer As String
    Dim abbrev As String
    Dim shift As String
    Dim i As Integer
    Dim colorIndex As Integer

    Set ws = Worksheets("Officers & Shifts")

    iCol = 1
    iRow = 3
    matchRow = 1
    index = 3
    colorIndex = 0

    If Not ComboBox2 = "" Then
        officer = ComboBox2
        Do
            '
            If officer = Cells(index, 1).Value Then
                matchRow = index
            End If
            index = index + 1
        Loop
    End If

```

```

'
Loop While index < 75

' Placing employee into shift section indicated on the UserForm
If matchRow = 1 Then
    Rows(iRow).EntireRow.Insert
    Rows(iRow).EntireRow.ClearFormats
    Rows(iRow).EntireRow.Select
    Cells(iRow, 1).Value = officer
Else
    iRow = matchRow
End If

' Setting the days array to the day if checkbox is checked otherwise
null
If CheckBox1 = True Then
    days(0) = "SUN"
Else
    days(0) = "NULL"
End If
If CheckBox2 = True Then
    days(1) = "MON"
Else
    days(1) = "NULL"
End If
If CheckBox3 = True Then
    days(2) = "TUE"
Else
    days(2) = "NULL"
End If
If CheckBox4 = True Then
    days(3) = "WED"
Else
    days(3) = "NULL"
End If
If CheckBox5 = True Then
    days(4) = "THU"
Else
    days(4) = "NULL"
End If
If CheckBox6 = True Then
    days(5) = "FRI"
Else
    days(5) = "NULL"
End If
If CheckBox7 = True Then
    days(6) = "SAT"
Else
    days(6) = "NULL"
End If

' Sets the shift value to its abbreviation so it can be placed in the
spreadsheet
shift = ComboBox3
i = 1
Do
    If shift = ws.Cells(i, 6) Then

```

```

        abbrev = ws.Cells(i, 7)
        colorIndex = ws.Cells(i, 8)
    End If
    i = i + 1
Loop While Not IsEmpty(ws.Cells(i, 6))

' If any days are checked, put shift value in those days for entire
trimester
If CheckBox1 = True Or CheckBox2 = True Or CheckBox3 = True Or
CheckBox4 = True Or CheckBox5 = True Or CheckBox6 = True Or CheckBox7 = True
Then
    Do
        iCol = iCol + 1
        If days(0) = "SUN" And Cells(2, iCol).Value = "SUN" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(1) = "MON" And Cells(2, iCol).Value = "MON" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(2) = "TUE" And Cells(2, iCol).Value = "TUE" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(3) = "WED" And Cells(2, iCol).Value = "WED" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex
            If colorIndex = 1 Then
                Cells(iRow, iCol).Font.Color = vbWhite
            End If
            If Not TextBox4.Value = "" Then
                Cells(iRow, iCol).AddComment
                Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
            End If
        ElseIf days(4) = "THU" And Cells(2, iCol).Value = "THU" Then
            Cells(iRow, iCol).Value = abbrev
            Cells(iRow, iCol).Interior.colorIndex = colorIndex

```

```

        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    ElseIf days(5) = "FRI" And Cells(2, iCol).Value = "FRI" Then
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.ColorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    ElseIf days(6) = "SAT" And Cells(2, iCol).Value = "SAT" Then
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.ColorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    End If
    Loop Until IsEmpty(Cells(1, iCol))
Else
    ' No days were checked; get value from calendar to enter an
individual day
    Do
        iCol = iCol + 1
        Loop Until Cells(1, iCol).Value = pickDay
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.ColorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    End If
End If

Unload Me
End Sub

Public Sub CommandButton2_Click()
    pickDay = CalendarForm.GetDate( _
        FirstDayOfWeek:=Monday, _
        DateFontSize:=12, _
        TodayButton:=True, _
        OkayButton:=True, _
        ShowWeekNumbers:=True, _

```

```

        BackgroundColor:=RGB(243, 249, 251), _
        HeaderColor:=RGB(147, 205, 2221), _
        HeaderFontColor:=RGB(255, 255, 255), _
        SubHeaderColor:=RGB(223, 240, 245), _
        SubHeaderFontColor:=RGB(31, 78, 120), _
        DateColor:=RGB(243, 249, 251), _
        DateFontColor:=RGB(31, 78, 120), _
        TrailingMonthFontColor:=RGB(155, 194, 230), _
        DateHoverColor:=RGB(223, 240, 245), _
        DateSelectedColor:=RGB(202, 223, 242), _
        SaturdayFontColor:=RGB(0, 176, 240), _
        SundayFontColor:=RGB(0, 176, 240), _
        TodayFontColor:=RGB(0, 176, 80))
End Sub

```

## AddSNAP

```

' .....
'
'   AddSNAP
'   Developed by Josh Logier
'   jlogier@calpoly.edu
'
'   This UserForm takes an employee name, shift, comment,
'   and date (or days) to create a new employee on the sheet.
'
' .....

Public pickDay As Date

Private Sub Userform_initialize()
    Dim ws As Worksheet
    Dim i As Integer
    Dim employee As String
    Dim shift As String

    Set ws = Worksheets("Officers & Shifts")

    i = 1

    ' Adds employees from the list of employees on the last worksheet
    With ComboBox2
        Do
            employee = ws.Cells(i, 1).Value & ", " & ws.Cells(i, 2).Value & "
" & ws.Cells(i, 3).Value
            .AddItem employee
            i = i + 1
        Loop While Not IsEmpty(ws.Cells(i, 1))
    End With

    i = 1

    ' Adds the shifts from the shift list on the last worksheet
    With ComboBox3
        Do
            shift = ws.Cells(i, 6).Value

```



```

        .AddItem shift
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 6))
End With

CheckBox1.Value = False
CheckBox2.Value = False
CheckBox3.Value = False
CheckBox4.Value = False
CheckBox5.Value = False
CheckBox6.Value = False
CheckBox7.Value = False

End Sub

Private Sub CommandButton1_Click()
    Dim ws As Worksheet
    Dim iRow As Integer
    Dim iCol As Integer
    Dim days(7) As String
    Dim matchRow As Integer
    Dim index As Integer
    Dim officer As String
    Dim abbrev As String
    Dim shift As String
    Dim i As Integer
    Dim colorIndex As Integer

    Set ws = Worksheets("Officers & Shifts")

    iCol = 1
    iRow = 3
    matchRow = 1
    index = 3
    colorIndex = 0

    If Not ComboBox2 = "" Then
        officer = ComboBox2
        Do
            '
            If officer = Cells(index, 1).Value Then
                matchRow = index
            End If
            index = index + 1
        Loop While index < 75

        ' Placing employee into shift section indicated on the UserForm
        If matchRow = 1 Then
            Rows(iRow).EntireRow.Insert
            Rows(iRow).EntireRow.ClearFormats
            Rows(iRow).EntireRow.Select
            Cells(iRow, 1).Value = officer
        Else
            iRow = matchRow
        End If
    End If

```

```

' Setting the days array to the day if checkbox is checked otherwise
null
If CheckBox1 = True Then
    days(0) = "SUN"
Else
    days(0) = "NULL"
End If
If CheckBox2 = True Then
    days(1) = "MON"
Else
    days(1) = "NULL"
End If
If CheckBox3 = True Then
    days(2) = "TUE"
Else
    days(2) = "NULL"
End If
If CheckBox4 = True Then
    days(3) = "WED"
Else
    days(3) = "NULL"
End If
If CheckBox5 = True Then
    days(4) = "THU"
Else
    days(4) = "NULL"
End If
If CheckBox6 = True Then
    days(5) = "FRI"
Else
    days(5) = "NULL"
End If
If CheckBox7 = True Then
    days(6) = "SAT"
Else
    days(6) = "NULL"
End If

' Sets the shift value to its abbreviation so it can be placed in the
spreadsheet
shift = ComboBox3
i = 1
Do
    If shift = ws.Cells(i, 6) Then
        abbrev = ws.Cells(i, 7)
        colorIndex = ws.Cells(i, 8)
    End If
    i = i + 1
Loop While Not IsEmpty(ws.Cells(i, 6))

' If any days are checked, put shift value in those days for entire
trimester
If CheckBox1 = True Or CheckBox2 = True Or CheckBox3 = True Or
CheckBox4 = True Or CheckBox5 = True Or CheckBox6 = True Or CheckBox7 = True
Then
    Do

```

```

iCol = iCol + 1
If days(0) = "SUN" And Cells(2, iCol).Value = "SUN" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(1) = "MON" And Cells(2, iCol).Value = "MON" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(2) = "TUE" And Cells(2, iCol).Value = "TUE" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(3) = "WED" And Cells(2, iCol).Value = "WED" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(4) = "THU" And Cells(2, iCol).Value = "THU" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(5) = "FRI" And Cells(2, iCol).Value = "FRI" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If

```

```

        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    ElseIf days(6) = "SAT" And Cells(2, iCol).Value = "SAT" Then
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.colorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    End If
    Loop Until IsEmpty(Cells(1, iCol))
Else
    ' No days were checked; get value from calendar to enter an
individual day
    Do
        iCol = iCol + 1
        Loop Until Cells(1, iCol).Value = pickDay
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.colorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    End If
End If

Unload Me
End Sub

Public Sub CommandButton2_Click()
    pickDay = CalendarForm.GetDate( _
        FirstDayOfWeek:=Monday, _
        DateFontSize:=12, _
        TodayButton:=True, _
        OkayButton:=True, _
        ShowWeekNumbers:=True, _
        BackgroundColor:=RGB(243, 249, 251), _
        HeaderColor:=RGB(147, 205, 221), _
        HeaderFontColor:=RGB(255, 255, 255), _
        SubHeaderColor:=RGB(223, 240, 245), _
        SubHeaderFontColor:=RGB(31, 78, 120), _
        DateColor:=RGB(243, 249, 251), _
        DateFontColor:=RGB(31, 78, 120), _
        TrailingMonthFontColor:=RGB(155, 194, 230), _
        DateHoverColor:=RGB(223, 240, 245), _
        DateSelectedColor:=RGB(202, 223, 242), _
        SaturdayFontColor:=RGB(0, 176, 240), _
        SundayFontColor:=RGB(0, 176, 240), _
        TodayFontColor:=RGB(0, 176, 80))

```

End Sub

## ***AddTrainee***

```
.....  
'  
' AddTrainee  
' Developed by Josh Logier  
' jlogier@calpoly.edu  
'  
' This UserForm takes an employee name, shift, comment,  
' and date (or days) to create a new employee on the sheet.  
'  
.....  
  
Public pickDay As Date  
  
Private Sub Userform_initialize()  
    Dim ws As Worksheet  
    Dim i As Integer  
    Dim employee As String  
    Dim shift As String  
  
    Set ws = Worksheets("Officers & Shifts")  
  
    i = 1  
  
    ' Adds employees from the list of employees on the last worksheet  
    With ComboBox2  
        Do  
            employee = ws.Cells(i, 1).Value & ", " & ws.Cells(i, 2).Value & "  
" & ws.Cells(i, 3).Value  
            .AddItem employee  
            i = i + 1  
        Loop While Not IsEmpty(ws.Cells(i, 1))  
    End With  
  
    i = 1  
  
    ' Adds the shifts from the shift list on the last worksheet  
    With ComboBox3  
        Do  
            shift = ws.Cells(i, 6).Value  
            .AddItem shift  
            i = i + 1  
        Loop While Not IsEmpty(ws.Cells(i, 6))  
    End With  
  
    CheckBox1.Value = False  
    CheckBox2.Value = False  
    CheckBox3.Value = False  
    CheckBox4.Value = False  
    CheckBox5.Value = False  
    CheckBox6.Value = False  
    CheckBox7.Value = False
```

```

End Sub

Private Sub CommandButton1_Click()
    Dim ws As Worksheet
    Dim iRow As Integer
    Dim iCol As Integer
    Dim days(7) As String
    Dim matchRow As Integer
    Dim index As Integer
    Dim officer As String
    Dim abbrev As String
    Dim shift As String
    Dim i As Integer
    Dim colorIndex As Integer

    Set ws = Worksheets("Officers & Shifts")

    iCol = 1
    iRow = 3
    matchRow = 1
    index = 3
    colorIndex = 0

    If Not ComboBox2 = "" Then
        officer = ComboBox2
        Do
            '
            If officer = Cells(index, 1).Value Then
                matchRow = index
            End If
            index = index + 1
        Loop While index < 75

        ' Placing employee into shift section indicated on the UserForm
        If matchRow = 1 Then
            Rows(iRow).EntireRow.Insert
            Rows(iRow).EntireRow.ClearFormats
            Rows(iRow).EntireRow.Select
            Cells(iRow, 1).Value = officer
        Else
            iRow = matchRow
        End If

        ' Setting the days array to the day if checkbox is checked otherwise
null    If CheckBox1 = True Then
        days(0) = "SUN"
    Else
        days(0) = "NULL"
    End If
    If CheckBox2 = True Then
        days(1) = "MON"
    Else
        days(1) = "NULL"
    End If
    If CheckBox3 = True Then

```

```

        days(2) = "TUE"
    Else
        days(2) = "NULL"
    End If
    If CheckBox4 = True Then
        days(3) = "WED"
    Else
        days(3) = "NULL"
    End If
    If CheckBox5 = True Then
        days(4) = "THU"
    Else
        days(4) = "NULL"
    End If
    If CheckBox6 = True Then
        days(5) = "FRI"
    Else
        days(5) = "NULL"
    End If
    If CheckBox7 = True Then
        days(6) = "SAT"
    Else
        days(6) = "NULL"
    End If

    ' Sets the shift value to its abbreviation so it can be placed in the
spreadsheet
    shift = ComboBox3
    i = 1
    Do
        If shift = ws.Cells(i, 6) Then
            ' Gets the abbreviation of the shift from the list
            abbrev = ws.Cells(i, 7)
            ' Gets the color of the cell based on the list
            colorIndex = ws.Cells(i, 8)
        End If
        i = i + 1
    Loop While Not IsEmpty(ws.Cells(i, 6))

    ' If any days are checked, put shift value in those days for entire
pay cycle
    ' Sets the text of the cell as the shift abbreviation
    ' Sets the color of the cell as the color index
    If CheckBox1 = True Or CheckBox2 = True Or CheckBox3 = True Or
CheckBox4 = True Or CheckBox5 = True Or CheckBox6 = True Or CheckBox7 = True
Then
        Do
            iCol = iCol + 1
            If days(0) = "SUN" And Cells(2, iCol).Value = "SUN" Then
                Cells(iRow, iCol).Value = abbrev
                Cells(iRow, iCol).Interior.colorIndex = colorIndex
                If colorIndex = 1 Then
                    Cells(iRow, iCol).Font.Color = vbWhite
                End If
                If Not TextBox4.Value = "" Then
                    Cells(iRow, iCol).AddComment

```

```

        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(1) = "MON" And Cells(2, iCol).Value = "MON" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(2) = "TUE" And Cells(2, iCol).Value = "TUE" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(3) = "WED" And Cells(2, iCol).Value = "WED" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(4) = "THU" And Cells(2, iCol).Value = "THU" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(5) = "FRI" And Cells(2, iCol).Value = "FRI" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite
    End If
    If Not TextBox4.Value = "" Then
        Cells(iRow, iCol).AddComment
        Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
    End If
ElseIf days(6) = "SAT" And Cells(2, iCol).Value = "SAT" Then
    Cells(iRow, iCol).Value = abbrev
    Cells(iRow, iCol).Interior.colorIndex = colorIndex
    If colorIndex = 1 Then
        Cells(iRow, iCol).Font.Color = vbWhite

```



```

        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    End If
    Loop Until IsEmpty(Cells(1, iCol))
Else
    ' No days were checked; get value from calendar to enter an
individual day
    Do
        iCol = iCol + 1
        Loop Until Cells(1, iCol).Value = pickDay
        Cells(iRow, iCol).Value = abbrev
        Cells(iRow, iCol).Interior.colorIndex = colorIndex
        If colorIndex = 1 Then
            Cells(iRow, iCol).Font.Color = vbWhite
        End If
        If Not TextBox4.Value = "" Then
            Cells(iRow, iCol).AddComment
            Cells(iRow, iCol).Comment.Text Text:=TextBox4.Value
        End If
    End If
End If

Unload Me
End Sub

Public Sub CommandButton2_Click()
    pickDay = CalendarForm.GetDate( _
        FirstDayOfWeek:=Monday, _
        DateFontSize:=12, _
        TodayButton:=True, _
        OkayButton:=True, _
        ShowWeekNumbers:=True, _
        BackgroundColor:=RGB(243, 249, 251), _
        HeaderColor:=RGB(147, 205, 222), _
        HeaderFontColor:=RGB(255, 255, 255), _
        SubHeaderColor:=RGB(223, 240, 245), _
        SubHeaderFontColor:=RGB(31, 78, 120), _
        DateColor:=RGB(243, 249, 251), _
        DateFontColor:=RGB(31, 78, 120), _
        TrailingMonthFontColor:=RGB(155, 194, 230), _
        DateHoverColor:=RGB(223, 240, 245), _
        DateSelectedColor:=RGB(202, 223, 242), _
        SaturdayFontColor:=RGB(0, 176, 240), _
        SundayFontColor:=RGB(0, 176, 240), _
        TodayFontColor:=RGB(0, 176, 80))
End Sub

```