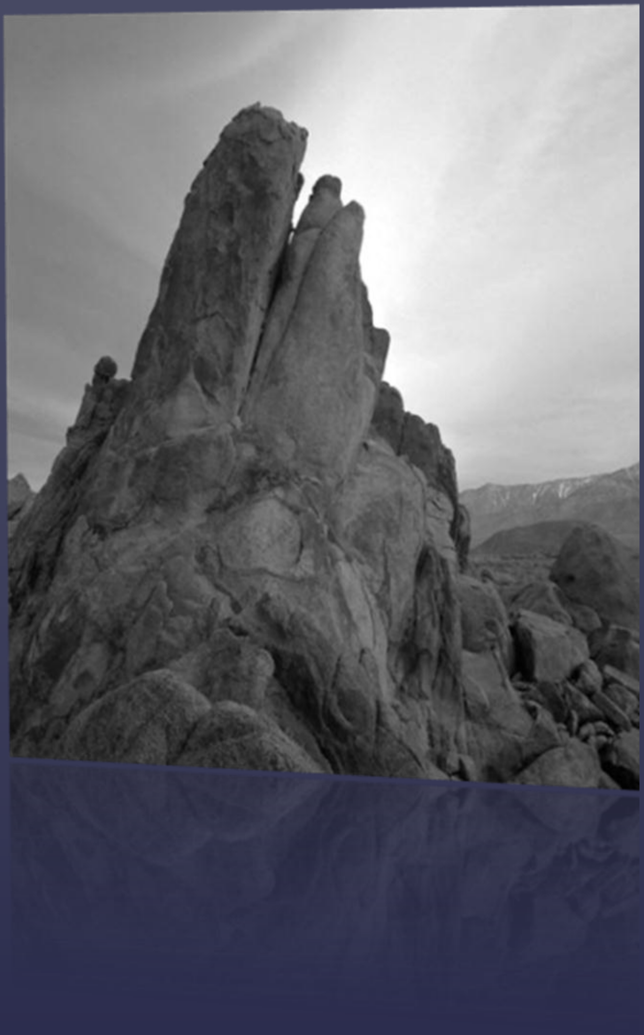


SHERPA PROGRAM

FINAL VBA PROJECT

Darrin Craner

Executive Summary

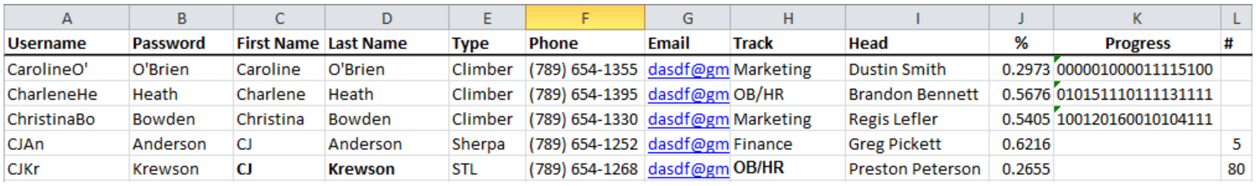
Sherpa Program

All First Year MBA students are assigned mentors for the internship search. These mentors are Second Year students and are referred to as Sherpas. The Sherpa program is structured to help develop resumes, hone interview skills, and facilitate networking opportunities.

A Better Way?

Currently the Sherpa program uses a couple of two by two table on Google Spreadsheets to track the information and progress of each climber (1st year). There have been issues however in the data mining process. Toward the end of the year much of the information needed to be accessed which proved to be an ominous task and very inefficient. The following Sherpa Program is the Beta version of what I refer to as the Sherpa Dashboard. Here any member of the program, at any level, can log in and see the progress of each climber, Sherpa, track lead or entire track. Over the summer I will continue to make additions to get it fully functional for SQL Database, but this semester I was mainly focused on creating the foundation for the program which I am glad to say that, over 1300 lines of code and over 60 hours later (no exaggeration) I have successfully written a program that can easily be linked to just one SQL table and serve as a user friendly access point. I will not be able to discuss all aspects of the program, so I will stick to the interesting stuff.

Guts of the Program



A	B	C	D	E	F	G	H	I	J	K	L
Username	**Password**	**First Name**	**Last Name**	**Type**	**Phone**	**Email**	**Track**	**Head**	**%**	**Progress**	**#**
CarolineO'	O'Brien	Caroline	O'Brien	Climber	(789) 654-1355	dasdf@gm	Marketing	Dustin Smith	0.2973	000001000011115100	
CharleneHe	Heath	Charlene	Heath	Climber	(789) 654-1395	dasdf@gm	OB/HR	Brandon Bennett	0.5676	010151110111131111	
ChristinaBo	Bowden	Christina	Bowden	Climber	(789) 654-1330	dasdf@gm	Marketing	Regis Lefler	0.5405	100120160010104111	
CJAn	Anderson	CJ	Anderson	Sherpa	(789) 654-1252	dasdf@gm	Finance	Greg Pickett	0.6216		5
CJKr	Krewson	**CJ**	**Krewson**	STL	(789) 654-1268	dasdf@gm	OB/HR	Preston Peterson	0.2655		80

Database

Currently this document only has one sheet to store data. As mentioned above, this summer I will find a place to host and SQL Table to retrieve and upload the table. This will not change the code however, it will merely add a step to pull data and upload. The table is set up on a sheet called "Users" and contains all necessary information for each Track Lead, Sherpa, and Climber. The column headings are as shown in Figure 1. This database has been filled with random information to test the code. True, it is extremely ugly, but the user will NEVER see this sheet.

Username/Password

The Username and password are set when the user is added and can only be edited through a form which requires the original password. At Login the username and password is screened against these fields. This sheet is xlveryhidden and the VBA code will be password protected to prevent "prying eyes".

There is a skeleton key that unlocks all which will be given to the head Sherpa in case of emergency.

Type/Head/#

Given the iterations of the application, these columns are not necessarily in order, but the beauty is in the fact that order does not matter if the code is written correctly.

The "Type" column is treated as a level of clearance for the application. Depending on the level of clearance the user will be able to perform certain tasks that others cannot. The Hierarchy of Types is as follows:

King Sherpa ➥ Track Leads ➥ Sherpas ➥ Climber

Each user is granted access to all users lower in the hierarchy. Track leads can see all Sherpas and climbers in their track.

At each level the user is given a person they report to. The code references this as the "Head". The importance this field will be made evident when we look at the user interface.

The "#" column is used to determine the number of underlings (for lack of a better word) each user has. This is used to create weighted progress percentages later on.

Progress/%

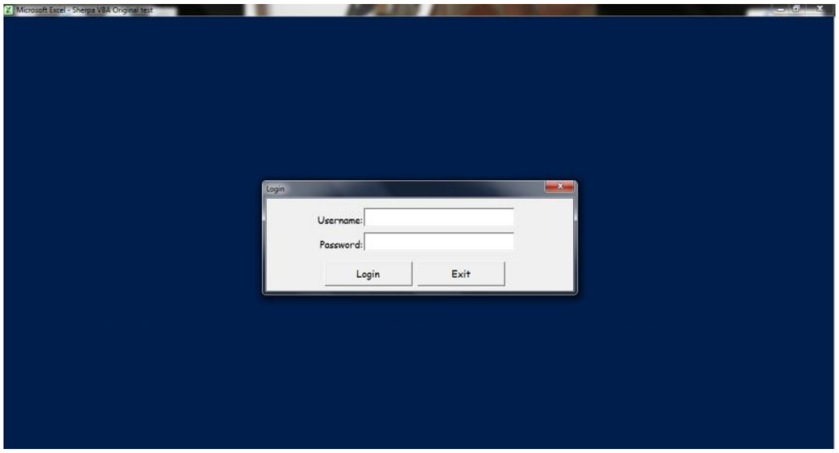
The "Progress" column is the heart of the application. I did not want to have an endless number of columns for each user to determine where they are in the program. Each skill would need its own column and is not dynamic. Instead, I created an 18 digit code where each digit represents a different skill and the value of each digit determine how far along they are in each skill. We will analyze the following code for example:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	**0**	**0**	**1**	**5**	**0**	**0**	**4**	**0**	**0**	**1**	**1**	**0**	**9**	**0**	**0**	**0**	**1**

Each "skill" the climber is asked to master will have a theoretic number which will coincide with the digit placement. Skill #14 (Practice 10 STAR stories) has the value of 9 which means the climber has practiced 9 of the 10 recommended STAR stories. We will discuss the updating and modification in the UI portion of the paper.

USER INTERFACE

Login

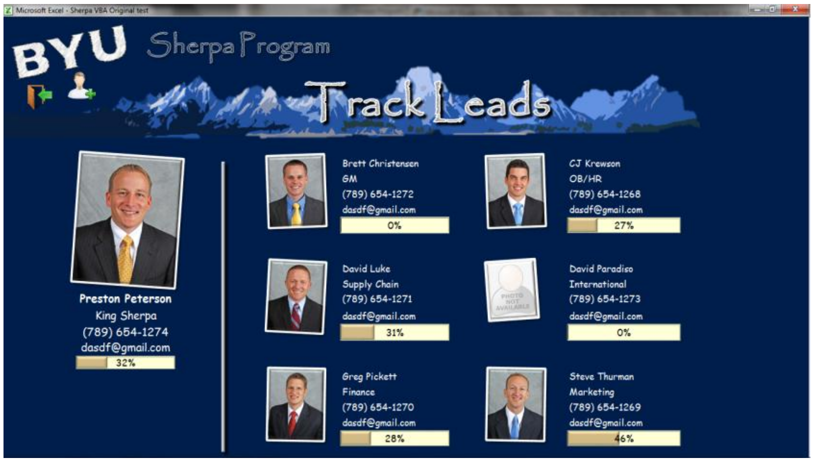


At the Login a number of things become apparent. At the close of the program I deleted all dynamic shapes on the dashboard by running a for each shape loop. With an if-then statement I saved the static images, but set their visible property to false. Program will always run in full screen mode. I disabled the esc key while THIS file is open with (.OnKey "{ESC}", "") to prevent users from accessing excel tools that could fowl up the application. However, when the user switches to another excel file open on their computer, the esc key is reinstated and full screen disabled. And when they come back to the application full screen is back and esc disabled. Once the user logs in, they are presented with their custom dashboard including permissions.

Dashboard

The Dashboard is where all the magic happens. Here the user has a quick view of how his area is doing. Here I have logged in as Preston Peterson, the Head

Sherpa. Beneath him is a progress bar of how the entire program is doing. And next to his direct reports (Track Leads) are progress bars of each track in the program. The following code drills through the 18-digit number for each climber and returns a percentage of how far they are:



Microsoft Excel - Sherpa VBA Original test

BYU Sherpa Program

Track Leads

Preston Peterson
King Sherpa
(789) 654-1274
dasdf@gmail.com
32%

Brett Christensen
GM
(789) 654-1272
dasdf@gmail.com
0%

CJ Krewson
OB/HR
(789) 654-1268
dasdf@gmail.com
27%

David Luke
Supply Chain
(789) 654-1271
dasdf@gmail.com
31%

David Paradiso
International
(789) 654-1273
dasdf@gmail.com
0%

Greg Pickett
Finance
(789) 654-1270
dasdf@gmail.com
28%

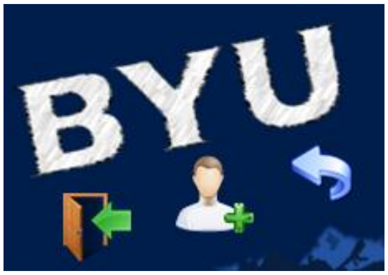
Steve Thurman
Marketing
(789) 654-1269
dasdf@gmail.com
46%

Each picture on the dashboard is actionable. When direct reports are clicked the dashboard is refreshed to show the Track Lead on the left and each of his Sherpas on the right with their respective progress bars. The title above the direct reports then dynamically changes to "Sherpas". Drilling down further, the Sherpa's dashboard shows each of his climbers. Once the climber is clicked his profile page is shown. Likewise, at any time, the Image of the "Superior" on the dashboard can be clicked to show their profile. Any user that is not a "Climber" will have a unique profile page, but we are still determining what that is going to show, this will be added during the summer.

```
Sub Choice()

Dim Temp As String
Temp = Application.Caller

If DBType.Value = "Sherpa" Or DBUser.Value = Temp Then
    UserProfName = Temp
    User_Profile.Show
Else
    LoadDashVariables (Temp)
End If

End Sub
```

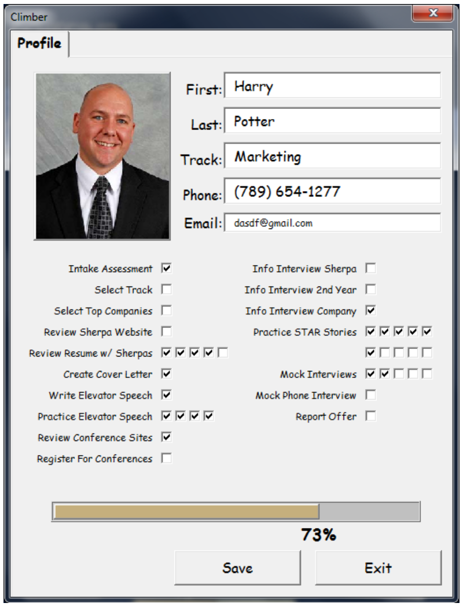
These actionable Photos are given a unique ID when placed on the page. The ID they are given is the user name of the person they represent. Using the ".OnAction " I call a sub that determines the permission of the user and acts accordingly. Within the sub I refer back to the photo's ID using the "application.caller" string to determine which button was clicked.

Icon Buttons



The icon buttons on the top left of the dashboard are for the only buttons necessary to run the application. There is LogOut (The Door) AddUser (The Guy) and Back (The Arrow). The Back icon only appears if you have clicked on one of your direct reports. This allows you to go back on level. Permissions only allow you to return as far as your own level in the hierarchy. The AddUser icon only appears if the person who logged in is the Head Sherpa or a Track Lead.

Profile Page



First: Harry
Last: Potter
Track: Marketing
Phone: (789) 654-1277
Email: dasdf@gmail.com

Intake Assessment ☑
Select Track □
Select Top Companies □
Review Sherpa Website □
Review Resume w/ Sherpas ☑ ☑ ☑ ☑ □
Create Cover Letter ☑
Write Elevator Speech ☑
Practice Elevator Speech ☑ ☑ ☑ ☑
Review Conference Sites ☑
Register For Conferences □

Info Interview Sherpa □
Info Interview 2nd Year □
Info Interview Company ☑
Practice STAR Stories ☑ ☑ ☑ ☑ ☑
☑ □ □ □ □
Mock Interviews ☑ ☑ □ □ □
Mock Phone Interview □
Report Offer □

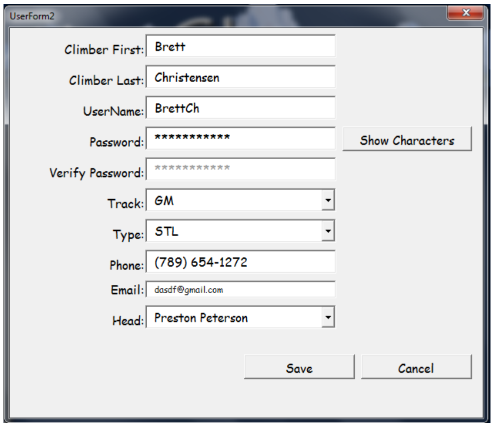
73%

The Profile Form has more loops than the L.A. Interstate. I labeled each CheckBox with a 3 digit number at the end. The first two numbers represent the digit in the 18-digit code for the user. The last number represents the value of each CheckBox, the fifth resume reviewed will place a five for the fifth digit in the progress code. (See Code Below)

```
For Each progbox In ProgBoxes
    y = Mid(progbox.Name, 9, 2)
    z = Mid(progbox.Name, 11, 1)
    If Mid(Sheets("Users").Range("k" & UserRow).Value, y, 1) >= z _
    Then progbox.Value = True
Next
```

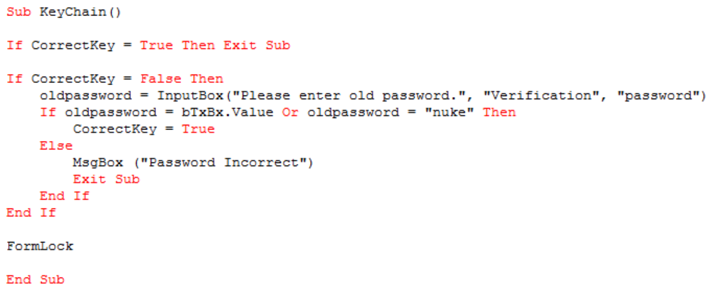
Later I am going to include a code that does not allow the user to select a check box to the right of an empty check box, but for now they just need to me smart☺. When the form is unloaded it zeros out the code on the database and replaces it with the updated version with another mean loop. The TextBoxes are locked, but the person's information can be updated by clicking on their photo in the form (The picture won't pop up for you if you open this on a different computer, I am working on how to load an image into a form from a worksheet.).

Add/Edit Users

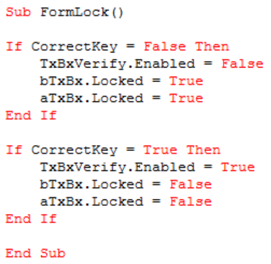


Climber First: Brett
Climber Last: Christensen
UserName: BrettCh
Password: ***********
Verify Password: ***********
Track: GM
Type: STL
Phone: (789) 654-1272
Email: dasdf@gmail.com
Head: Preston Peterson

This was a fun form. I had to determine if the user was entering the form to create a new user or to update a current user. If the user was updating a user then I needed to load all the fields and lock the password fields, username fields, and show characters button. To do this I created a KeyChain Sub that will pop an input window to enter the current password for that user if the "CorrectKey" Boolean is set to False. It will then run the LockForm Sub to either unlock the fields if the CorrectKey is True and Lock the fields if it is False. This allows me to set the CorrectKey to True if we are entering the form from the New User Button on the Dashboard and to False if entering the form from the profile form. When exiting the form all fields are updated on the Database. When adding a user the combo boxes dynamically change with the "exit" field event to narrow down the options. If the Marketing Track is selected with the type set to climber then the Head ComboBox is loaded with only the Marketing Sherpas.



```
Sub KeyChain()

If CorrectKey = True Then Exit Sub

If CorrectKey = False Then
    oldpassword = InputBox("Please enter old password.", "Verification", "password")
    If oldpassword = bTxBx.Value Or oldpassword = "nuke" Then
        CorrectKey = True
    Else
        MsgBox ("Password Incorrect")
        Exit Sub
    End If
End If

FormLock

End Sub
```



```
Sub FormLock()

If CorrectKey = False Then
    TxBxVerify.Enabled = False
    bTxBx.Locked = True
    aTxBx.Locked = True
End If

If CorrectKey = True Then
    TxBxVerify.Enabled = True
    bTxBx.Locked = False
    aTxBx.Locked = False
End If

End Sub
```

The Little Things

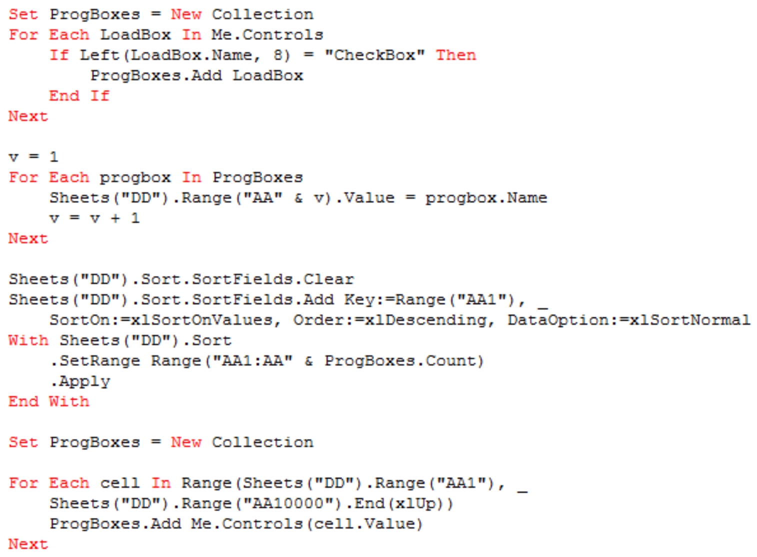
For the sake of length I will only mention some of the little codes to make the application look friendly and perform smoothly. Each time a picture is place on the dashboard the code runs a randbetween(-4,4) which I set as the image rotation. This gives the pictures a random appearance which is more comfortable to look at. I also run a custom photo sub on each image to give it texture and depth.

Every progress bar is weighted to reflect the true percentage of their direct reports. A track lead with 30 Sherpas and a Track Lead with 1 Sherpa, when combined their total is divided by 31, not 2.

Application and Learning

Hang-ups

I had a number of Hang-ups. The greatest of which was the order of that the CheckBoxes loaded in the collection. If the collection was out of numerical order then the 18-digit code would get all messed up. To fix this I dumped the names of every CheckBox onto a sheet, ordered them, and reloaded the collection. To ensure this does not happen again, it is perform each time the profile form is opened.

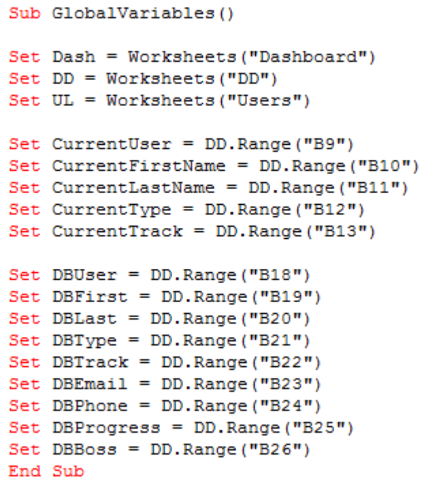


```
Set ProgBoxes = New Collection
For Each LoadBox In Me.Controls
    If Left(LoadBox.Name, 8) = "CheckBox" Then
        ProgBoxes.Add LoadBox
    End If
Next

v = 1
For Each progbox In ProgBoxes
    Sheets("DD").Range("AA" & v).Value = progbox.Name
    v = v + 1
Next

Sheets("DD").Sort.SortFields.Clear
Sheets("DD").Sort.SortFields.Add Key:=Range("AA1"), _
    SortOn:=xlSortOnValues, Order:=xlDescending, DataOption:=xlSortNormal
With Sheets("DD").Sort
    .SetRange Range("AA1:AA" & ProgBoxes.Count)
    .Apply
End With

Set ProgBoxes = New Collection

For Each cell In Range(Sheets("DD").Range("AA1"), _
    Sheets("DD").Range("AA10000").End(xlUp))
    ProgBoxes.Add Me.Controls(cell.Value)
Next
```

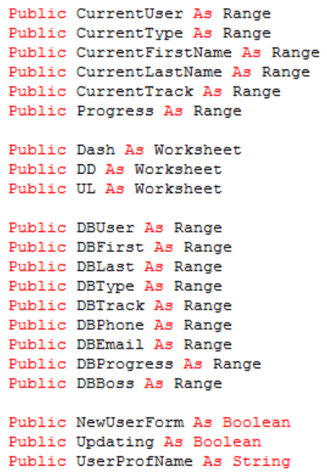
Another hang up was determining which button I clicked after I clicked it. I spent hours and hours looking for a solution. When I load the pictures onto the dashboard the loop requires that I send all .onaction to the same sub unless I want to create 20+ subs. I finally found the solution which came in the form of one statement, “applicaton.caller”. This simply returns the name of the button that called that sub. With this I was able to dynamically change the behavior of this Sub.

Public Variables are My Friends

With this much Code I got lost in variable land. I spent a couple of hours cleaning up my code and declaring and loading over a dozen Public Variable which I could use throughout the project. This made things incredibly simple and helped me focus on the building of the application. I was also able to store the current user (The person logged in) and the DashBoard User (The person in the picture on the right) as different people. This facilitated the code tremendously.



```
Sub GlobalVariables()

Set Dash = Worksheets("Dashboard")
Set DD = Worksheets("DD")
Set UL = Worksheets("Users")

Set CurrentUser = DD.Range("B9")
Set CurrentFirstName = DD.Range("B10")
Set CurrentLastName = DD.Range("B11")
Set CurrentType = DD.Range("B12")
Set CurrentTrack = DD.Range("B13")

Set DBUser = DD.Range("B18")
Set DBFirst = DD.Range("B19")
Set DBLast = DD.Range("B20")
Set DBType = DD.Range("B21")
Set DBTrack = DD.Range("B22")
Set DBEmail = DD.Range("B23")
Set DBPhone = DD.Range("B24")
Set DBProgress = DD.Range("B25")
Set DBBoss = DD.Range("B26")
End Sub
```



```
Public CurrentUser As Range
Public CurrentType As Range
Public CurrentFirstName As Range
Public CurrentLastName As Range
Public CurrentTrack As Range
Public Progress As Range

Public Dash As Worksheet
Public DD As Worksheet
Public UL As Worksheet

Public DBUser As Range
Public DBFirst As Range
Public DBLast As Range
Public DBType As Range
Public DBTrack As Range
Public DBPhone As Range
Public DBEmail As Range
Public DBProgress As Range
Public DBBoss As Range

Public NewUserForm As Boolean
Public Updating As Boolean
Public UserProfName As String
```

Key Chain

As I mentioned before, the KeyChain was a great discovery for me. This allows me to use the same form for multiple uses and keeps things neat and clean. I will use this in the future.

ADDITIONAL HELP / CAVEATS

Internet

Google was a very good friend to me over the last couple of months. However, I did not receive help from any one person.

Changed Final Project

My original Final Project was a personal budget program, but after the Sherpas were chosen for next year I decided to develop a program to help track climber progress and ultimately streamline the commutation within the program. My original project was almost finished, but I saw more potential in the implementation of this program.

Invitation

I feel that it is impossible to illustrate everything this application can do in a small report. I invite you to open the application and give it a try. I have spent a lot of time on this and learned more than I could have hoped for. I especially invite you to take a look at the DashBoardSetup Module. There are a lot of loops and statement that need to be seen to be understood. I knew NO VBA before the class, and now feel confident to utilize these skills in the work force.