Final Project Write-Up

Jesse Carman

2.1 Executive summary of the project. This should contain a description of the business along with an overview of the system you built

Carman Violin Studio is in need of a student database management system to enable the violin instructor (my wife) to easily create invoices and receipts for her students, to automatically generate personalized emails to her students, and to allow her to view students' profiles in an organized manner (student information is spread across 30+ columns of data). The management system also needs to keep a concise historical record of each student's charges, fees, credits and payments. It also requires the ability to easily add new students to the database and change their activity status (Active/Inactive) as students have a tendency to quit and then start-up lessons again from time to time. Because the instructor teaches both private and group lessons, the system needs the ability to differentiate between the two and enable the user to input, and make changes to, both types of lessons. Reschedules, cancellations, and no-shows are also a regular occurrence among students, thus the system will need to be able to account for such activities. Finally, the system will need to be able to track two periods of time for each student. First, how many reschedules individual students have requested over a rolling 3-month period. And second, what 2-month period each individual student is currently in (students must sign 2-month contracts which automatically renew until the student quits).

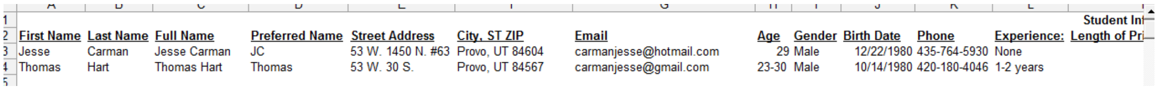
To address this situation, I created a Microsoft Excel workbook that includes an "Invoice-Receipt" sheet, a database sheet listing all students in alphabetical order, and a separate sheet for each student which tracks their respective transaction history. I created a user form that enables the instructor to view any student's information in a concise manner. I created a "Go To" feature that enables to instructor to quickly move from one student's file to another. I also created sub procedures enabling the instructor to perform the following tasks: add new students to the database; schedule private and/or group lessons; record 48-hour notices of cancellation, reschedules, no-shows, late payment fees, returned check fees, and payments; change student activity status; and generate emails to be sent from her violin studio email account (a record of the day and time emails are sent is also kept). The workbook includes a hidden template worksheet of how a student's file should look which is copied when adding students to the database. Each sub procedure opens a user form to enable the instructor to input the required information for each student and then takes that information and records it accordingly. Finally, I created a separate macro toolbar, "Violin Tools," that has each of these commands listed on it for accessibility on any worksheet at any time. The toolbar is automatically opened (and nested below the regular toolbars) upon open of the workbook and automatically hidden upon workbook close.

As time went on I added to my project a "Puzzle" game which I created in Excel and can be played during boring lectures in class. It is the file titled "Puzzle." This game is a 4 x 4 block of 16 cells containing the numbers 1-15 and a blank cell. The numbers 1-15 are randomly placed within the 4 x 4 block and the user is supposed to rearrange them in the correct order, reading left to right starting in the top left cell, using the blank cell. By simply clicking on any cell in the 4 x 4 block the numbers will shift as allowed.

2.2 Implementation documentation. Provide a concise, well-organized documentation of what you actually did for your solution. You may want to use tables or bulleted lists to describe the components of your solution and their role in the overall task. In any case, you should provide a textual description of the elements so it is clear what you have done, why it was included, how it is intended to be used in the task. Screen captures may be helpful in illustrating what you have done.

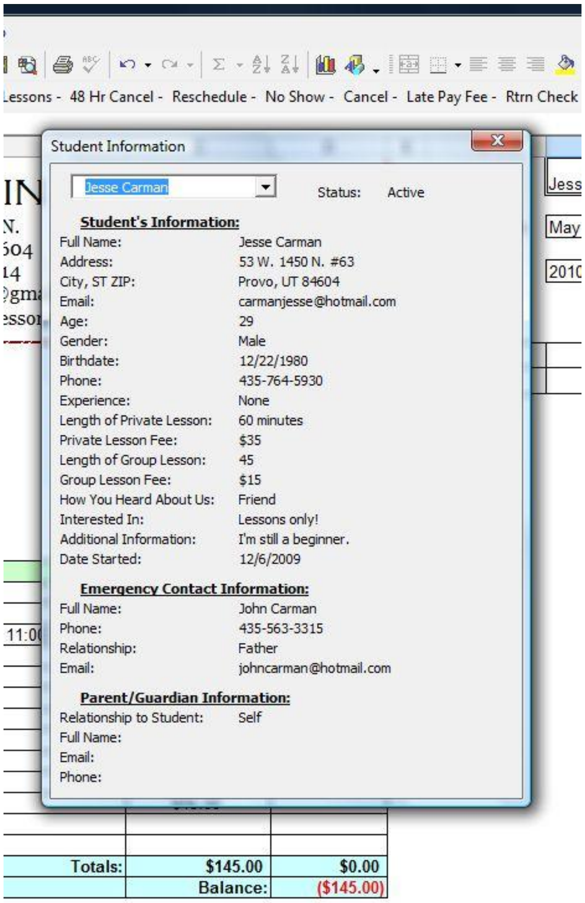
Violin Studio Workbook

To begin with, all the instructor had was a list of students she's currently teaching compiled in an Excel spreadsheet along with some scattered information such as: phone number, age, past experience, email and address. What I first put together was a more organized database spreadsheet with a column for every piece of data she would like to have in the future (for privacy reasons I am not including actual students but rather fictitious students I made up). This can be seen in the "Data" worksheet of the workbook and a snip-it is shown below. From the snip-it it is clear how much of a pain it would be to try to view all the data for any student. It would require scrolling horizontally back and forth across some 30 columns.



												Student Inf
First Name	Last Name	Full Name	Preferred Name	Street Address	City, ST ZIP	Email	Age	Gender	Birth Date	Phone	Experience:	Length of Pri
Jesse	Carman	Jesse Carman	JC	53 W. 1450 N. #63	Provo, UT 84604	carmanjesse@hotmail.com	29	Male	12/22/1980	435-764-5930	None	
Thomas	Hart	Thomas Hart	Thomas	53 W. 30 S.	Provo, UT 84567	carmanjesse@gmail.com	23-30	Male	10/14/1980	420-180-4046	1-2 years	

To solve this problem I created a user form that allows all the data in columns A through AD to be viewed at once. The user form is shown below and is activated by pressing the "View Profiles" button on the "Violin Tools" toolbar, or calling the "info" sub procedure. The form shown below, frmStudentInformation, will then appear with a drop-down box containing all the students. Simply changing the drop-down box will change the data shown to the respective student. This user form can be shown on any worksheet in the workbook. The sub procedure loads the user form which pulls all the current students from the "Data" sheet to load into the drop-down box. Then upon click of the student in the box the user form pulls the respective data from the "Data" sheet and displays it in the user form. This was one of three main functions the instructor wanted the workbook to be able to do.


Lessons - 48 Hr Cancel - Reschedule - No Show - Cancel - Late Pay Fee - Rtrn Check
Student Information
X
Jess
May
2010
11:00
Totals: $145.00 $0.00
Balance: ($145.00)

Jesse Carman Status: Active

Student's Information:

Full Name:	Jesse Carman
Address:	53 W. 1450 N. #63
City, ST ZIP:	Provo, UT 84604
Email:	carmanjesse@hotmail.com
Age:	29
Gender:	Male
Birthdate:	12/22/1980
Phone:	435-764-5930
Experience:	None
Length of Private Lesson:	60 minutes
Private Lesson Fee:	$35
Length of Group Lesson:	45
Group Lesson Fee:	$15
How You Heard About Us:	Friend
Interested In:	Lessons only!
Additional Information:	I'm still a beginner.
Date Started:	12/6/2009

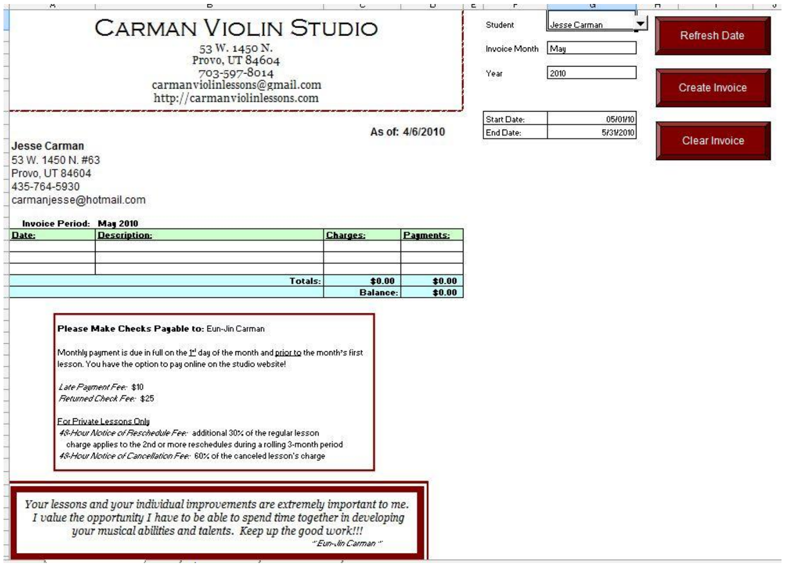
Emergency Contact Information:

Full Name:	John Carman
Phone:	435-563-3315
Relationship:	Father
Email:	johncarman@hotmail.com

Parent/Guardian Information:

Relationship to Student:	Self
Full Name:	
Email:	
Phone:	

The next thing I did was create the "Invoice-Receipt" worksheet that enables the instructor to create monthly invoices to be sent to students. The instructor wanted it to be limited to one month at a time. Because of the numerous formulas embedded into the worksheet I locked all cells on this page except for three: the student, invoice month, and year cells. These three cells contain built-in drop-down boxes containing, respectively, all the students, the months January through December, and the years 2010 through 2020 (this can easily be increased if there is need). Once the user selects the student, his/her name, address, phone number, and email are pulled, using vlookup formulas, from the "Data" sheet and listed (as shown in the image below). When the month and year are chosen they are concatenated into cell B16 showing the invoice month and year. Cell D9 contains the "=today()" formula for the invoice date. There are three buttons actually on this sheet (the only buttons not in the "Violin Tools" toolbar). One button, "Refresh Date," refreshes the dates shown in cells G8 and G9 (indicating how many days are in the month); the next button, "Create Invoice," pulls all transactions for that student relating to the chosen month and year from the students profile sheet and loads them into the invoice range; and the "Clear Invoice" button that clears the invoice. When this page is printed off the *print range* is set to only include the invoice range and not include the buttons or other parts of the worksheet used for creating the invoice (as shown in the second image below). This was the second capability the instructor wanted the workbook to have.



CARMAN VIOLIN STUDIO
53 W. 1450 N.
Provo, UT 84604
703-597-8014
carmanviolinlessons@gmail.com
http://carmanviolinlessons.com

Student Jesse Carman
Invoice Month May
Year 2010

Start Date:	05/01/10
End Date:	5/31/2010

As of: 4/6/2010

Jesse Carman
53 W. 1450 N. #63
Provo, UT 84604
435-764-5930
carmanjesse@hotmail.com

Invoice Period: May 2010

Date:	Description:	Charges:	Payments:
	Totals:	$0.00	$0.00
		Balance:	$0.00

Please Make Checks Payable to: Eun-Jin Carman

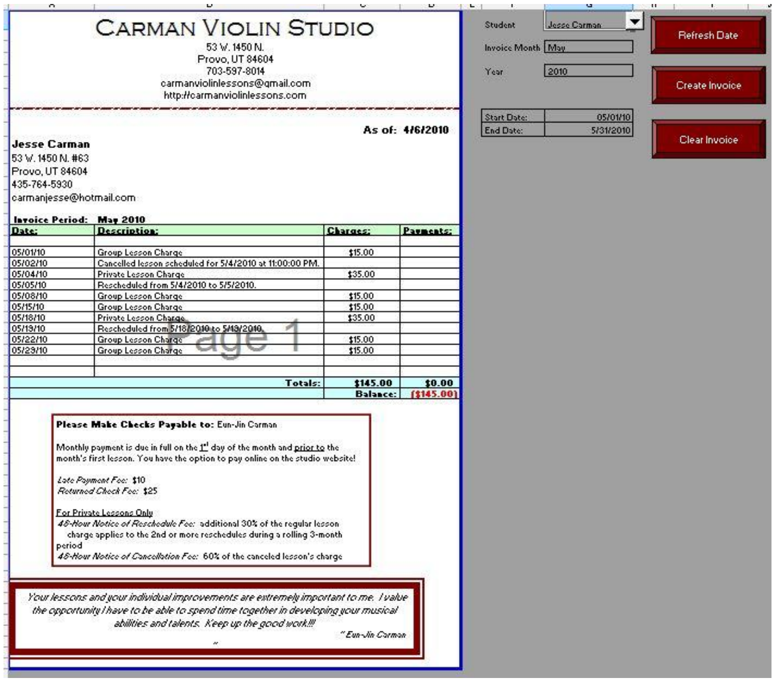
Monthly payment is due in full on the 1st day of the month and prior to the month's first lesson. You have the option to pay online on the studio website!

Late Payment Fee: $10
Returned Check Fee: $25

For Private Lessons Only
48-Hour Notice of Reschedule Fee: additional 30% of the regular lesson charge applies to the 2nd or more reschedules during a rolling 3-month period
48-Hour Notice of Cancellation Fee: 60% of the canceled lesson's charge

Your lessons and your individual improvements are extremely important to me. I value the opportunity I have to be able to spend time together in developing your musical abilities and talents. Keep up the good work!!!
"Eun-Jin Carman"

Before creating the "Invoice-Receipt" worksheet, I created a template worksheet that is hidden at all times. This template is used when a new student is added to the database. The template contains all the cell formatting that will be needed for normal use of the workbook. When the user presses the "Add Student" button in the toolbar, the "addNewStudent" sub



CARMAN VIOLIN STUDIO
53 W. 1450 N.
Provo, UT 84604
703-597-8014
carmanviolinlessons@gmail.com
http://carmanviolinlessons.com

Student Jesse Carman
Invoice Month May
Year 2010

Start Date:	05/01/10
End Date:	5/31/2010

As of: 4/6/2010

Jesse Carman
53 W. 1450 N. #63
Provo, UT 84604
435-764-5930
carmanjesse@hotmail.com

Invoice Period: May 2010

Date:	Description:	Charges:	Payments:
05/01/10	Group Lesson Charge	$15.00	
05/02/10	Cancelled lesson scheduled for 5/4/2010 at 11:00:00 PM.		
05/04/10	Private Lesson Charge	$35.00	
05/05/10	Rescheduled from 5/4/2010 to 5/5/2010.		
05/08/10	Group Lesson Charge	$15.00	
05/15/10	Group Lesson Charge	$15.00	
05/18/10	Private Lesson Charge	$35.00	
05/19/10	Rescheduled from 5/18/2010 to 5/19/2010.		
05/22/10	Group Lesson Charge	$15.00	
05/29/10	Group Lesson Charge	$15.00	
	Totals:	$145.00	$0.00
		Balance:	($145.00)

Please Make Checks Payable to: Eun-Jin Carman

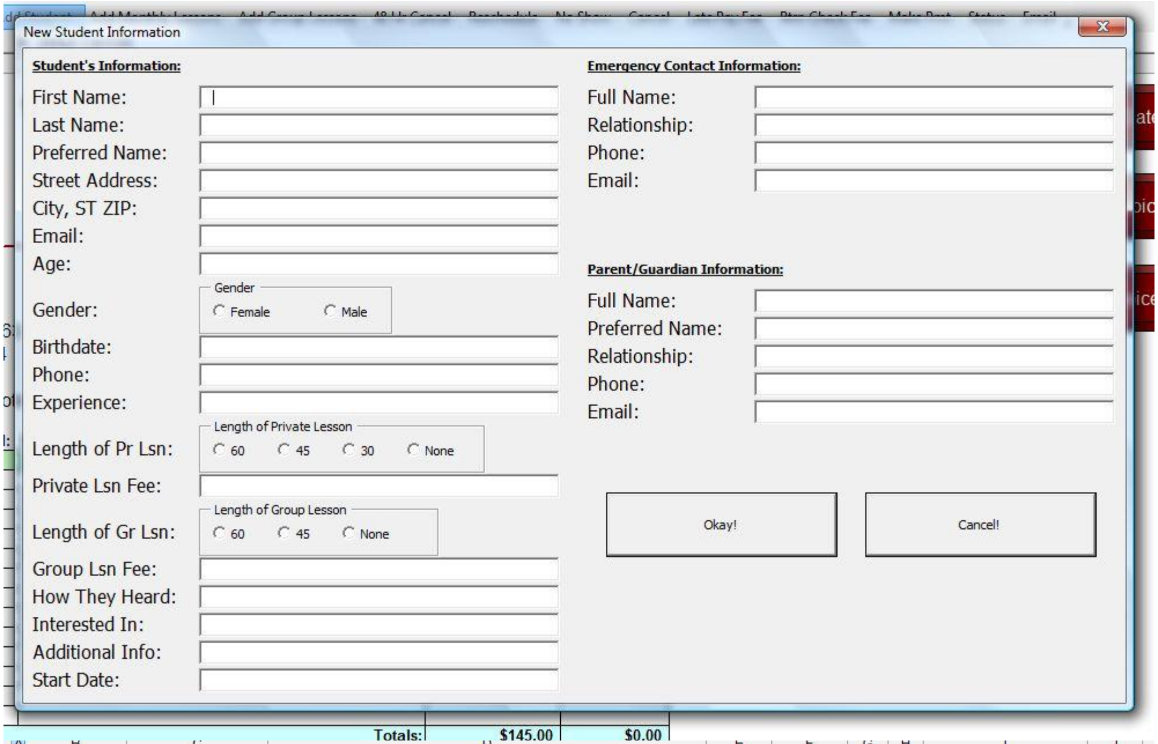
Monthly payment is due in full on the 1st day of the month and prior to the month's first lesson. You have the option to pay online on the studio website!

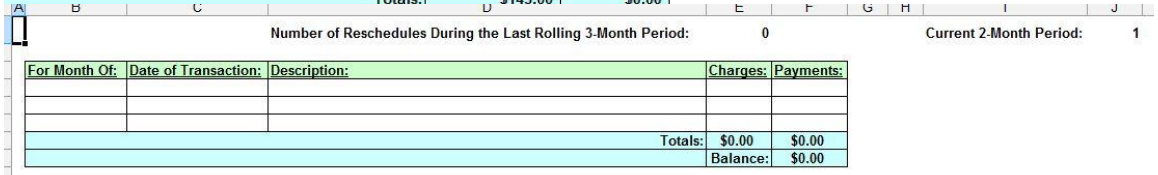
Late Payment Fee: $10
Returned Check Fee: $25

For Private Lessons Only
48-Hour Notice of Reschedule Fee: additional 30% of the regular lesson charge applies to the 2nd or more reschedules during a rolling 3-month period
48-Hour Notice of Cancellation Fee: 60% of the canceled lesson's charge

Your lessons and your individual improvements are extremely important to me. I value the opportunity I have to be able to spend time together in developing your musical abilities and talents. Keep up the good work!!!
"Eun-Jin Carman"

procedure is called which opens up the frmnewStudent user form. This user form allows the instructor to input all necessary information for the new student and then takes that information and adds the student to the database, then re-alphabetizes the database by the student's first name. It then creates a new worksheet titled as the student's full name which is simply a copy of the template. The user form, student file template, and an example of a filled out student's file are shown below.

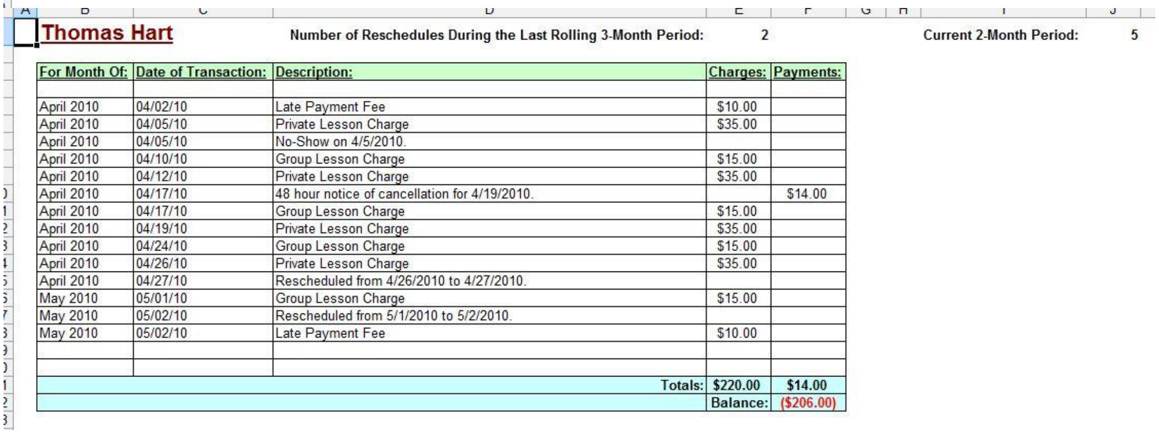




Number of Reschedules During the Last Rolling 3-Month Period: 0

Current 2-Month Period: 1

For Month Of:	Date of Transaction:	Description:	Charges:	Payments:
		Totals:	$0.00	$0.00
			Balance:	$0.00



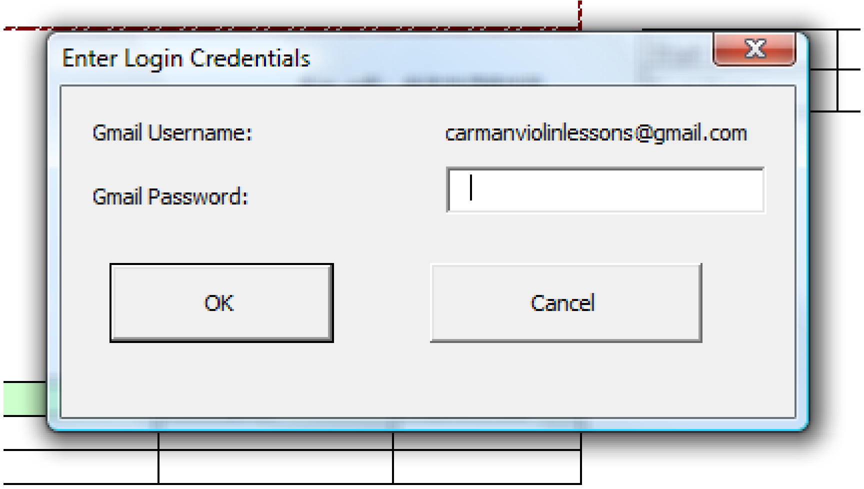
Thomas Hart

Number of Reschedules During the Last Rolling 3-Month Period: 2

Current 2-Month Period: 5

For Month Of:	Date of Transaction:	Description:	Charges:	Payments:
April 2010	04/02/10	Late Payment Fee	$10.00	
April 2010	04/05/10	Private Lesson Charge	$35.00	
April 2010	04/05/10	No-Show on 4/5/2010.		
April 2010	04/10/10	Group Lesson Charge	$15.00	
April 2010	04/12/10	Private Lesson Charge	$35.00	
April 2010	04/17/10	48 hour notice of cancellation for 4/19/2010.		$14.00
April 2010	04/17/10	Group Lesson Charge	$15.00	
April 2010	04/19/10	Private Lesson Charge	$35.00	
April 2010	04/24/10	Group Lesson Charge	$15.00	
April 2010	04/26/10	Private Lesson Charge	$35.00	
April 2010	04/27/10	Rescheduled from 4/26/2010 to 4/27/2010.		
May 2010	05/01/10	Group Lesson Charge	$15.00	
May 2010	05/02/10	Rescheduled from 5/1/2010 to 5/2/2010.		
May 2010	05/02/10	Late Payment Fee	$10.00	
		Totals:	$220.00	$14.00
			Balance:	($206.00)

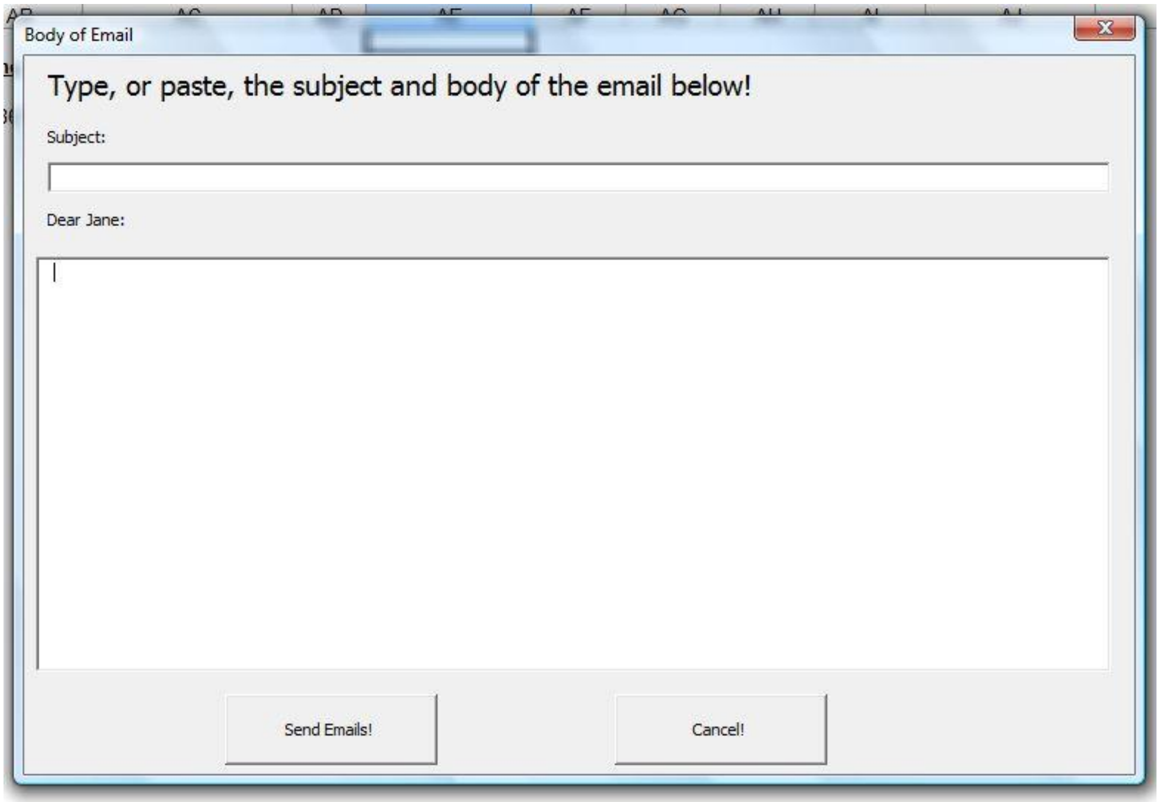
The third and final capability the instructor wanted the workbook to have is the ability to generate mass and individualized emails. To do this the instructor simply needs to press the “Email” button in the toolbar and the “sendmessages” sub procedure is called. First, the *password* user form pops up (as shown below) requesting the user to enter her password (the studio email address is automatically shown as the username without having to type it). This window actually verifies the password typed in before proceeding because I have it coded into the sub procedure. Though potentially dangerous if someone were to look into the code, the instructor liked this idea better because she will know that she entered the password correctly before she types the entire email and hits send. The workbook itself will be password protected so that only the instructor can open it anyway. The added email password is one final precaution for sending emails. Once the password is verified, the user form shown below opens up and allows the instructor to type the subject and body of the email. The sub procedure then personalizes each email based on the information in the “Data” sheet (their first name, and email address) and then sends the email with her personalized salutation at the end (built-in to the sub procedure so she does not have to type it each time). The last email sent is one to her own studio so she can always have a copy of the emails to see how they look. When the procedure is done sending all the emails (only to “Active” students) then a message box appears stating that such is the case. The user simply presses enter or the okay button and then the procedure records the date and time each email was sent in a log for each student. The log is stored on the “Data” page simply for a reference. It is stored in the columns AJ and up.



Enter Login Credentials

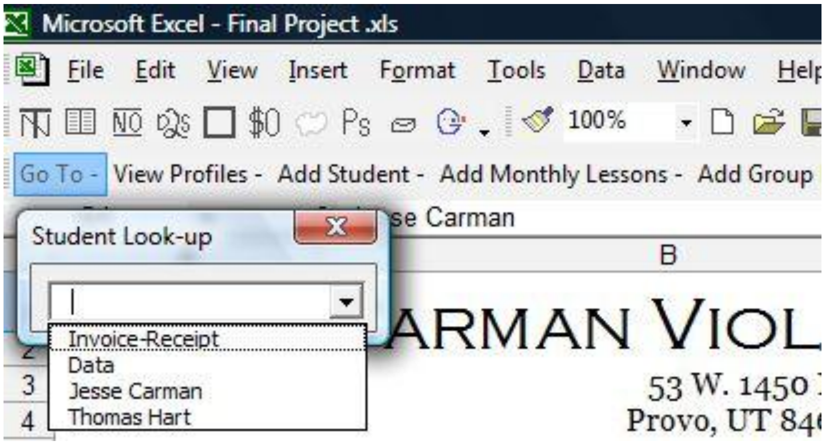
Gmail Username: carmanviolinlessons@gmail.com

Gmail Password:



Type, or paste, the subject and body of the email below!

Since sending email was the third and final function of the workbook, the remainder of this section of the paper will discuss the various sub procedures used for maintaining the student profiles. This can become exceedingly boring, but I felt it was necessary because it was the majority of my work in the workbook. I won't be offended if you skim the rest (of this section). The first is the "Go To" button that calls the "goToStudent" sub procedure which opens up the "frmStudentLookUp" user form. This user form contains only a drop-down box containing: "Invoice-Receipt," "Data," and each students name. As the student, or other page, is selected the workbook goes directly to the corresponding worksheet in the workbook. This will come in handy when the instructor has many students. The form is shown below.



The next button, and each button discussed hereafter, first calls the "goToStudent" sub to get the user on the right profile, then executes its respective function. The first button not yet discussed is the “Add Monthly Lessons” button that calls the “addNewPrivate” procedure which opens the “frmAddMonth” user form. This user form enables the user to choose what days the lessons will be on and in what month, assuming the lessons are on the same day each week. After the form is filled out the procedure fills out the student’s profile appropriately by pulling the private lesson charge from the “Data” sheet. The next button “Add Group Lessons” calls the sub “addNewGroup” which does exactly the same thing, but for group lessons.

The “48-Hr Notice” (of cancellation) button calls the “fourtyEightHourNotice” sub procedure. This opens the “frmFourtyEightHrCancel” user form that allows the user to enter the date the instructor was notified of the cancel and the date of the originally scheduled lesson (little did I know as I created this that I was misspelling “forty” as “fourty”). The month and year are also entered by the user for invoice purposes. After the form is filled out it then adds the cancellation to the student’s profile and credits back the student 40% of his/her regularly scheduled lesson charge. Profiles are sorted based on date of transactions before the procedure ends.

The “Reschedule” button calls the “reschedule” procedure and opens the “frmReschedule” user form. This form is identical to the “frmFourtyEightHrCancel.” After the relevant data is entered into the form the procedure reschedules the lesson on the student’s profile. The “No Show” button calls the “noShow” sub which opens the “frmNoShow” user form. After the necessary data are entered into the user form, then the sub adds a No-Show entry to the student’s file.

The “Cancel” button calls the “cancellation” procedure which opens the “frmCancel” user form. This requires the input of the date of notification of cancel, the time of notification, and the date of the regularly scheduled lesson. The time is important here because it will distinguish between a forty eight hour notice of cancellation and a forty seven hour notice of cancellation, as these are treated differently. The “Late Pay Fee” ($10) and “Rtrn Check Fee” ($25) buttons call the “latepaymentfee” and “returnedCheckFee” sub procedures which call the “frmLatePay” and “frmReturnCheck” user forms. These assess the relevant fees to the student’s profile as given in the company contract.

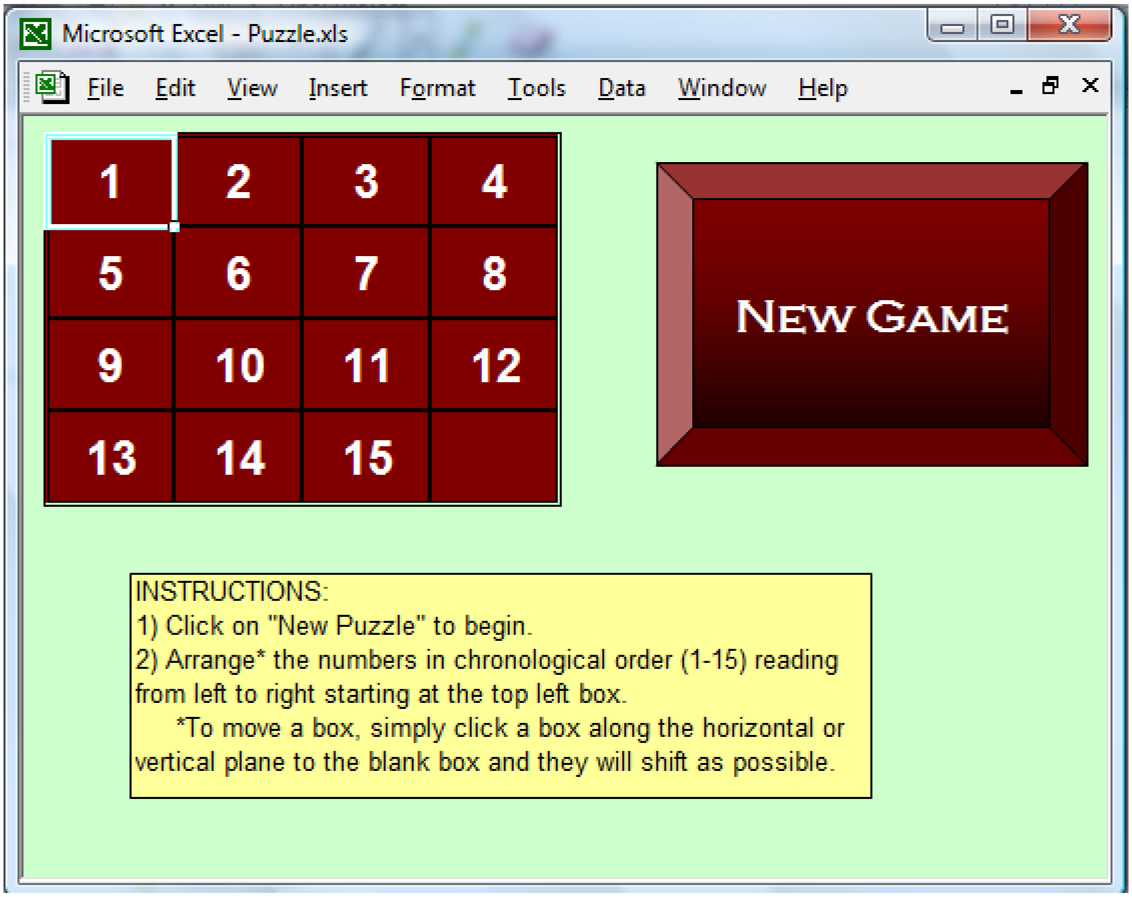
Puzzle

The second file I included in this project was done to give me a brief break from the Violin Studio workbook. As shown below it is a very simple game that uses sub procedures to move the numbers around. The first image below is what it looks like when it is opened (it automatically hides all toolbars, worksheet tabs, status bars, ribbons (if 2007), etc, and resizes the window to where it’s only big enough to include the game). The second image is an example of what it looks like after you press the “NEW GAME” button.

The sub procedures are mostly simple other than the private sub “Worksheet_SelectionChange(ByVal Target As Range)” which allows for the “moveCells” sub procedure to be called simply by clicking any cell within the 4 x 4 block of cells. This private sub I learned how to do from my good friend Google. It basically consists of one If-Then statement asking if the target address is equal to any cell within that 4 x 4 block, then call “moveCell.”

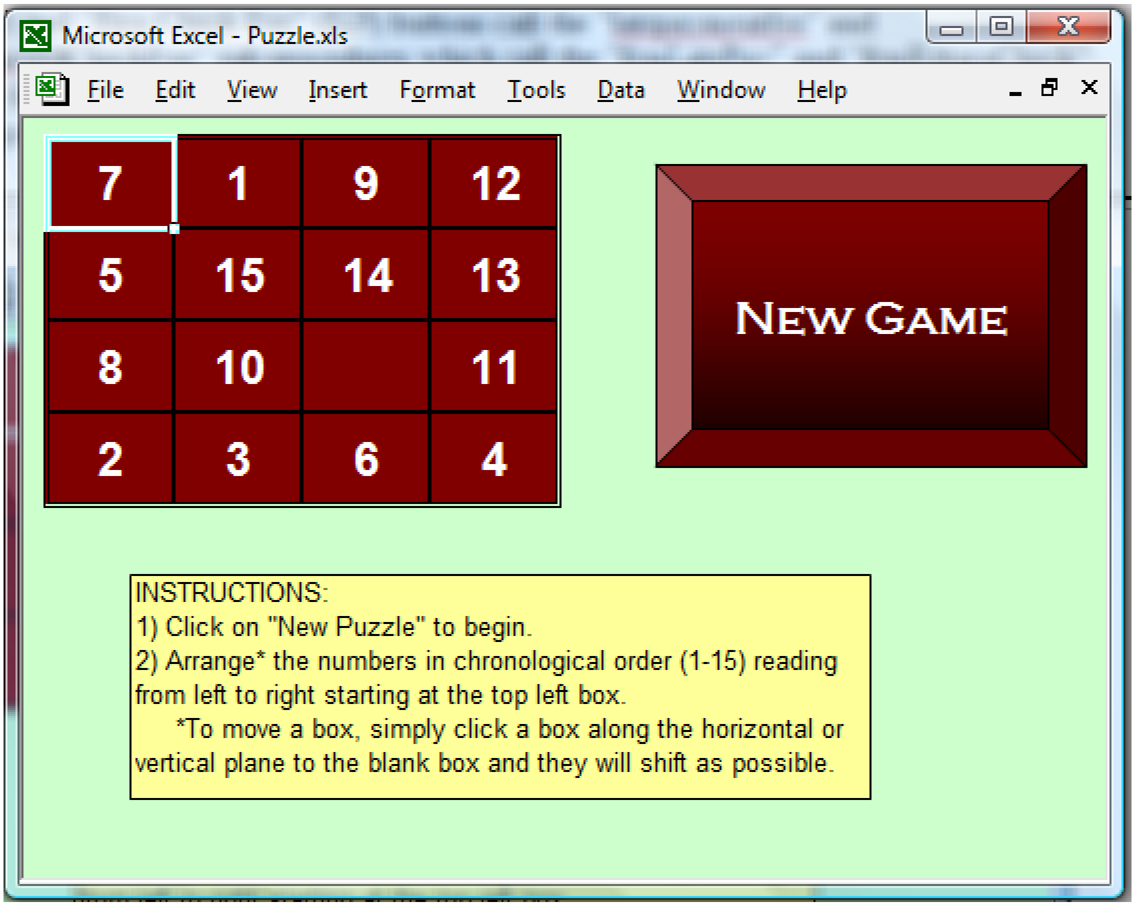
The “moveCells” sub procedure determines if there is a “” (blank) cell on the same row or column as the cell pressed, and if so then it shifts the numbers so as to fill in that blank cell and make the selected cell the new blank cell. Although this one looks lengthy and confusing, it really was pretty simple to do. You’ll have to give this one a try and feel free to look at the code. If anything it will give you a break from grading.

The “newGame” sub procedure first unlocks the worksheet and then randomly places the numbers 1-15 into the 4 x 4 block of cells. After this is finished it relocks the worksheet and the user is ready to play.



1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

INSTRUCTIONS:
1) Click on "New Puzzle" to begin.
2) Arrange* the numbers in chronological order (1-15) reading from left to right starting at the top left box.
*To move a box, simply click a box along the horizontal or vertical plane to the blank box and they will shift as possible.



7	1	9	12
5	15	14	13
8	10		11
2	3	6	4

INSTRUCTIONS:
1) Click on "New Puzzle" to begin.
2) Arrange* the numbers in chronological order (1-15) reading from left to right starting at the top left box.
*To move a box, simply click a box along the horizontal or vertical plane to the blank box and they will shift as possible.

2.3 Discussion of learning and conceptual difficulties encountered. Let me know what you learned by going through this project. If there are elements you wanted to include but could not get to work, discuss these in this section. Please be sure that you have tried to solve the problem, including asking the professor for assistance, before giving up.

1. It is crucial to get a solid understanding of what the final project is to look like and include before you begin. This includes writing out “blueprints” of each form, each worksheet, and a general layout of each sub procedure (this I of course failed to do). I had to make changes countless times to EVERY form and sub procedure as new elements were introduced and thought of by the instructor.

2. Having the instructor there to make changes from time to time could either be a wonderful help, or it could be a catastrophe. I found that as the project evolved, the instructor over and over again changed her mind regarding how she wanted things to look and what capabilities she wanted the workbook to have. I learned that there are ways to create your sub procedures in such a way as to be *flexible* to changes in user preferences along the way. I first wished that I did not have to show the project to the user until I had a final product, but gradually changed my mind as I was able to make her changes along the way before I had gotten too far along.

3. One, of the many, technical lessons I learned from this project was how to control sub procedures based on inputs through user forms. Perhaps I missed this section during the class presentation on user forms, but I had the hardest time figuring out how to make a sub procedure, that had called a user form, to exit when the user pressed "Cancel" on the user form. After asking the professor I found that it was a very simple solution and one that I needed for *each* of my user forms in order for them to function correctly within their respective sub procedures.

4. I learned that it really is easy, and important, to have as much done in VBA as possible, as opposed to having your code input temporary data into cells and work with it there. There are a couple of instances where this seemed impossible to me, but after brief research with Google I was able to find ways around my issues.

5. I learned how to work with dates in Excel, specifically this function, "=date(year(C2),month(C2)-3,day(C2))," for example. This is an excellent way to track periods of time that was necessary for my 3-month and 2-month period tracking.

6. Spelling is critical when naming things because there is no spell check. It could potentially be embarrassing if a coworker were to see that you named a sub procedure as "fourtyEightHrCancel" as opposed to "fortyEightHrCancel" (as I obviously did).

7. Another very important lesson I learned is the power of Google. It is so simple to learn how to do random things in VBA by searching on-line and seeing how other people are doing it.

2.4 Post your write-up to the VBA Projects blog (vbaprojects.blogspot.com). The write-up should be of sufficient detail that the professor could grade your project just by examining the write up. That is, the write up should stand on its own merits in discussing the project without needing to refer the reader to any of the workbook for clarification. Your project write-up will be publicly available on the internet. The professor will provide a location for you to upload your project files so you can link to them in the blog post. If your project contains confidential data, you will need to make alternate arrangements with the professor for submission.