

Integrated Core Team Builder

Christopher law December 2012

Executive Summary:

The purpose of my program is to solve an actual business problem. I work for the Marriott School Advisement Center. We serve the 1800 students of the Marriott School in matters of academic programs, internship and job placement, academic advising, and running some of the school's clubs (MUSA). The head of the office is Dean Michael Thompson and Kristen McGregor is the Managing Director.

One of the cyclical tasks our office is responsible for is creating teams for the Integrated Core. The Integrated Core is an envelope of four classes that all Marriott School students must take regardless of their major/emphasis. The core is structured around group work. Because all students have to take all four classes, this usually allows for a student to work with the same team for all the classes. Creating these teams can be very difficult due to the constraints set for determining the ideal team. The ideal team has the following characteristics:

- Size: Five students
- Ethnic Diversity: Spread out the international students; only one international student per team
- Academic Diversity: A team should have no more than one student from each of the Marriott School's nine majors or emphasis
- Gender: A team can either have zero or two females
- Conformity: Team members should be in all the same classes as each other

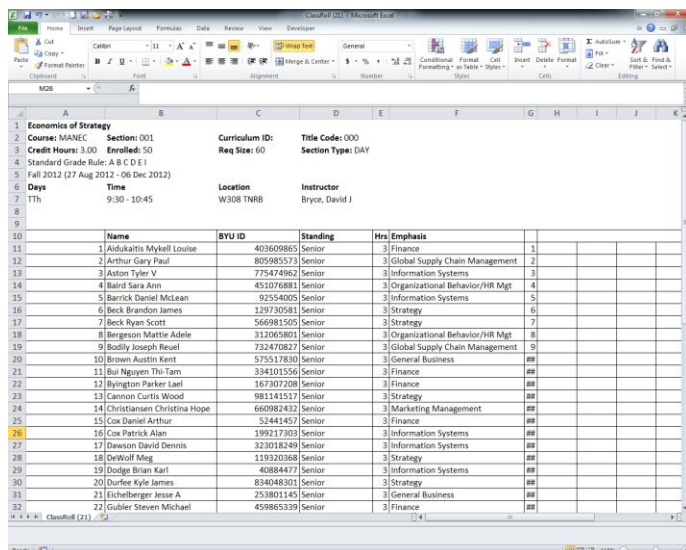
Keep in mind that there are six sections for each of the four classes and that over 400 students are enrolled in the Core. *Assigning the teams manually takes a long time!* My program drastically reduces the amount of time that the full-time staff in the Advisement Center spends on creating these teams by doing the following:

- Take the class lists from each of the 24 sections of the core and place each student in an Access database. The database will assign a unique code to each section that will allow the students to be grouped with other students in their same classes.
- Excel will pull the data from Access and create a data table with the student's name, unique section code created in Access, and emphasis.
- Log on to a password protected BYU site and pull each student's gender and nationality into the data table.
- Create teams based on the parameters listed above.

Implementation

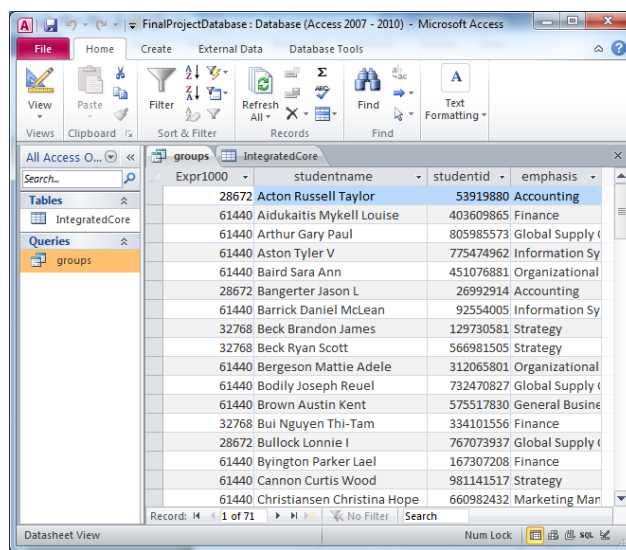
The first task in building this program was to analyze the format of the class lists so I could determine how to iterate through the data and pull it into Access. As you can see in Figure 1, most of the desired data is already in a table. From the table, I pulled the student's name, ID number, and emphasis. From the top of the page, I pulled the class name and section number. I used a loop to go through each row and coupled that information with the class name and section number from above to send an entry to Access. The following SQL command gave Access the correct values:

```
"insert into IntegratedCore values("& counter & ", "& Course & ", "& Section & ", "& code & ", "& studentName & ", "& studentID & ", "& emphasis & ")"
```



Name	BYU ID	Standing	Hrs/Emphasis
1. Aidukaitis Mykell Louise	403609865	Senior	3/Finance 1
2. Arthur Gary Paul	805985573	Senior	3/Global Supply Chain Management 2
3. Aston Tyler V	775474962	Senior	3/Information Systems 3
4. Baird Sara Ann	451076881	Senior	3/Organizational Behavior/HR Mgt 4
5. Barrick Daniel McLean	92554005	Senior	3/Information Systems 5
6. Beck Brandon James	129730581	Senior	3/Strategy 6
7. Beck Ryan Scott	566981505	Senior	3/Strategy 7
8. Bergeson Mattie Adele	312065801	Senior	3/Organizational Behavior/HR Mgt 8
9. Bodily Joseph Reuel	732470827	Senior	3/Global Supply Chain Management 9
10. Brown Austin Kent	575517830	Senior	3/General Business
11. Bui Nguyen Thi-Tam	334101556	Senior	3/Finance
12. Byington Parker Lael	167307208	Senior	3/Finance
13. Cannon Curtis Wood	981141517	Senior	3/Strategy
14. Christiansen Christina Hope	660982432	Senior	3/Marketing Management
15. Cox Daniel Arthur	52441457	Senior	3/Finance
16. Cox Patrick Alan	199217303	Senior	3/Information Systems
17. Dawson David Dennis	323018249	Senior	3/Information Systems
18. DeWolf Meg	119320368	Senior	3/Strategy
19. Dodge Brian Karl	40884477	Senior	3/Information Systems
20. Durfee Kyle James	834048301	Senior	3/Strategy
21. Eichelberger Jesse A	251801145	Senior	3/General Business
22. Gubler Steven Michael	429865339	Senior	3/Finance

Figure 1



studentname	studentid	emphasis
28672 Acton Russell Taylor	53919880	Accounting
61440 Aidukaitis Mykell Louise	403609865	Finance
61440 Arthur Gary Paul	805985573	Global Supply Chain Management
61440 Aston Tyler V	775474962	Information Systems
61440 Baird Sara Ann	451076881	Organizational Behavior/HR Mgt
28672 Bangerter Jason L	26992914	Accounting
61440 Barrick Daniel McLean	92554005	Information Systems
32768 Beck Brandon James	129730581	Strategy
32768 Beck Ryan Scott	566981505	Strategy
61440 Bergeson Mattie Adele	312065801	Organizational Behavior/HR Mgt
61440 Bodily Joseph Reuel	732470827	Global Supply Chain Management
61440 Brown Austin Kent	575517830	General Business
32768 Bui Nguyen Thi-Tam	334101556	Finance
28672 Bullock Lonnie I	767073937	Global Supply Chain Management
61440 Byington Parker Lael	167307208	Finance
61440 Cannon Curtis Wood	981141517	Strategy
61440 Christiansen Christina Hope	660982432	Marketing Management

Figure 2

Next, I needed to group the students by unique codes so I could tell which students had all four classes together. This was done by assigning each section of each class a unique number by taking 2^x . Then, each student's records is combined and the sum of each section's value of 2^x becomes the unique number indicating that any other student with the same sum has the exact schedule. An example of this is shown in Figure 2.

All of the values from Access are then pulled into an Excel data table. Next is the part requiring command of Internet Explorer. In order to find the student's gender and international status, we need to access their Personal Summary page on BYU's system. This page requires a username and password. The code will stall while the user inputs this information on the webpage. Next, a loop provides the webpage with a student ID number, Internet Explorer requests the new page, the gender and international status are copied and pasted from the loaded page back into the Excel data table, and the process repeats with the next student ID number. Please see Figure 3 and Figure 4. The data sorts by the unique section number, gender, and international status of the student.

Now that all of the needed data is in the data table, the only remaining step is to create the teams. I initially thought this part of the project would be one third of the overall program, but it turned out to

be more complicated than I had imagined. In its most basic form, each student is looked at individually to find a team they can fit into. Due to the many constraints, this is a very complicated process.

The first part of this process is to create an array that houses each student. We used a type of array that I am unfamiliar with. This allows the entries of the array to be different variable types. Seven different variables are stored in this two-dimensional array. Four different variable types are used for the seven variables. After this step, we have each student's name, gender, unique section code, citizenship, emphasis, and student ID number saved into an array.

Personal Information

Last Name/Identifier: First Name: GO

Summary Change Log Contact Names Identifiers Personal Relationships Languages Id Card Unlist

Law, Christopher George - [claw2] - (14-277-5076) - 13 Sep 1986 - Male - Senior - STD/PT/ACT

NOTE: Non-Directory information is accessible only to yourself and to university personnel with a legitimate need to know.

Names
Christopher Law
Complete Name: Law, Christopher George

Personal Information
Born: 13 Sep 1986
25 Years old
Gender: Male
Marital Status: Married
Citizenship: United States of America
Religion: Latter-day Saint

Contact Info
Mailing: 909 North 500 West
Apt #3
Provo, UT 84604
Unlisted
480-861-9777
Work: No Work Address
Email: c.law913@gmail.com
Unlisted

Identifiers
BYU ID: 14-277-5076 03
Net ID: claw2
SSN: ***-**-7163

Employee Info
STD/PT/ACT
Dean's Office MSM
STUDENT ADVISOR
Original PT Hire Date: 10 Feb 2012

Student Info
Senior
Classes Fall 2012 - 15.0 credit hours
BUS M 432-001 (Larson)
BUS M 490R-002 (Scoville)
1SYS 520-001 (Allen)
SRL 480-001 (Olson)
1SYS 515-001 (Meservy)

Figure 3

	A	B	C	D	E	F
	Expr1000	studentname	studentid	emphasis	Gender	USofA
1	15	Aidukaitis Mykell Louise	403609865	Finance	Female	TRUE
2	15	Arthur Gary Paul	805985573	Global Supply Chain Management	Male	TRUE
3	15	Aston Tyler V	775474962	Information Systems	Male	TRUE
4	15	Baird Sara Ann	451076881	Organizational Behavior/HR Mgt	Female	TRUE
5	15	Barrick Daniel McLean	92554005	Information Systems	Male	TRUE
6	15	Bergeson Mattie Adele	312065801	Organizational Behavior/HR Mgt	Female	TRUE
7	15	Bodily Joseph Reuel	732470827	Global Supply Chain Management	Male	TRUE
8	15	Brown Austin Kent	575517830	General Business	Male	TRUE
9	15	Byington Parker Lael	167307208	Finance	Male	TRUE
10	15	Cannon Curtis Wood	981141517	Strategy	Male	TRUE
11	15	Christiansen Christina Hope	660982432	Marketing Management	Female	TRUE
12	15	Cox Patrick Alan	199217303	Information Systems	Female	TRUE
13	15	Dawson David Dennis	323018249	Information Systems	Male	TRUE
14	15	DeWolf Meg	119320368	Strategy	Female	TRUE
15	15	Dodge Brian Karl	40884477	Information Systems	Male	TRUE
16	15	Eichelberger Jesse A	253801145	General Business	Male	TRUE
17	15	Gubler Steven Michael	459865339	Finance	Male	TRUE
18	15	Hall Jeffrey Raymond	869635943	Entrepreneurship	Male	TRUE
19	15	Hall Stephen Marc	84518861	Finance	Male	TRUE
20	15	Jackson Preston Taylor	437728801	Strategy	Male	TRUE
21	15	Jarrett Nathan Floyd	60332885	Finance	Male	TRUE
22	15	Krauss Derek Kenneth	308442634	Entrepreneurship	Male	TRUE
23	15	Krueger Ryan Jackson	1364866178	Organizational Behavior/HR Mgt	Male	TRUE
24	15	Lebaron Amber Lee	524311183	Marketing Management	Female	TRUE
25	15	Jensen Blake Russell	933255656	Accounting	Male	TRUE

Figure 4

Now that we have the students array, we can start assigning them to teams. First, we start with international females. Each international female is assigned to an empty team. Then we take the American females. Each American female is added to a team with one international female if such a team exists. If no teams are available that have one international female, then the American female is added to a team that has one American female. If no such team exists, the American female is added to a new team. We next take the international males. International males are added to the first team that doesn't have any international students. After this point, we assign the rest of the American males to teams until each team reaches its capacity of five. This whole time, two constraints are being checked before each student is placed. First, the student's unique section number must match the team's unique section number. Second, if a team already has a student with the same emphasis as the student in question, the student in question goes to the next team. After this loop is created, each student has been added to a team.

This is the extent of the program. The program is not perfect, however. After the code has finished running, the user will have to do some minor manipulations to the teams. For example, inevitably, some teams will need to have two students from the same emphasis to make all the students fit into teams.

For tasks like these, computer programs are not too great. The final product is a worksheet that is shown in Figure 5.

	A	B	C	D	E	F	G	H
1	Team Number 1							
2	Aidukaitis Mykell Louise	Finance	403609865	TRUE	F	15		
3	Arthur Gary Paul	Global Supply Chain	805985573	TRUE	M	15		
4	Aston Tyler V	Information Systems	775474962	TRUE	M	15		
5	Brown Austin Kent	General Business	575517830	TRUE	M	15		
6	Cannon Curtis Wood	Strategy	981141517	TRUE	M	15		
7	Team Number 2							
8	Baird Sara Ann	Organizational Behavior	451076881	TRUE	F	15		
9	Barrick Daniel McLean	Information Systems	92554005	TRUE	M	15		
10	Bodily Joseph Reuel	Global Supply Chain	732470827	TRUE	M	15		
11	Byington Parker Lael	Finance	167307208	TRUE	M	15		
12	Eichelberger Jesse A	General Business	253801145	TRUE	M	15		
13	Team Number 3							
14	Bergeson Mattie Adele	Organizational Behavior	312065801	TRUE	F	15		
15	Christiansen Christina H	Marketing Management	660982432	TRUE	F	15		
16	Dawson David Dennis	Information Systems	323018249	TRUE	M	15		
17	Gubler Steven Michael	Finance	459865339	TRUE	M	15		
18								
19	Team Number 4							
20	Cox Patrick Alan	Information Systems	199217303	TRUE	F	15		
21	DeWolf Meg	Strategy	119320368	TRUE	F	15		
22								

Figure 5

Learnings and Obstacles

This project was extremely beneficial to me because of a few reasons. First of all, the project has significance to me because I knew that it was a practical solution to a problem encountered at my job. This provided more motivation to me than any homework assignment. This also provided an interesting sub-feature of me not being able to modify the project too much if I encountered a difficulty. Sometimes, I was jealous of my friends that were doing a project they were designing for themselves only to meet the requirements of the project because they could modify their project to avoid a feature they found to be too complicated. When I hit features like this, I usually went to my boss to ask for clarification so I could be sure the complication was an essential feature.

One element of the project that I did modify is how the project starts. Initially, I wanted the program to go online and pull each class list that the user had specified from a drop-down menu. This would be run for each of the 24 sections. After consulting first with my boss and then with Professor Allen, I decided that this should be a feature that the user took care of manually. Therefore, one part of this process that is still manual is that the user must download each section and save the class lists to a folder that is at the same level as the Excel file.

The next difficulty that I encountered was iterating through the class lists and putting the data into Access. As I had just finished Project 5, I was familiar with integrating a database with VBA. I hoped to do all of this by myself but eventually needed to get some help from Professor Allen. Something cool that I

learned here was that I could pull the initial data table into Excel through the ribbon and then just include an update feature in the macro to refresh the data. This saved me a lot of coding by not having to create the table from scratch through VBA as was done in Project 5.

The next major problem I had was leveraging the Agent code that Dr. Allen gave us to gather data online. I knew that my understanding of this was pretty shaky from the instruction we had received in class so I knew my only hope was to use Dr. Allen. We got this working so my code would cycle through each student and use their student ID number to get their gender and nationality.

As I mentioned in the Executive Summary, I was surprised how long the last part of the project took me. I initially suspected that the team matching portion would be somewhat simple. I felt that I was capable of designing this part of the project myself. I spent about ten hours on Friday and Saturday trying to make loops that put a student on a worksheet and checked with the cells in each team group to see if the team was a possible fit. One issue that was frustrating was that I had a hard time deciding how to indicate which cells belonged to a specific team. I experimented with different colors and by using borders to indicate where one team started and one stopped. I eventually got frustrated enough that I decided to start this part of the project all over.

My next idea was to make an array of arrays. I had never tried this before but it was the idea of a programmer friend and I thought I would give it a try. In theory, this sounded like an easy way for me to organize the variables and check to see if the team had any students with characteristics that would disqualify the student in question from being put on the team. I really learned a lot about arrays through this method. When we learned about arrays in class, I couldn't really conceptualize them very well. That understanding was solidified after trying this array within arrays approach. Eventually, I got too confused using this approach so I decided to go to Dr. Allen again and start all over for a third time. I spent 13 hours on this approach before moving on.

The final approach proved to be a success. Dr. Allan helped me to create a bunch of constraints that would analyze arrays and find teams that didn't violate any of the conditions. This got to be pretty complicated because we were matching attributes of a two dimensional array to the different elements of the test. Finally though, we got this to work. One interesting part of this approach was that Dr. Allen introduced me to a new way to define variables. The code is as follows:

```
Type student
name As String
emph As String
code As Double
citi As Boolean
gend As String * 1
team As Integer
stid As String
```

These are module level variables that define the array 'student'. The cool part is that this allows 'student' to have multiple types of variables instead of just being a string or Boolean for all values.

Assistance

Honestly, there is no possible way I could have done this project by myself. I relied heavily on the expertise of Dr. Allen for all the main parts of my project. He was the one that gave me the initial structure of how to find if students were in all the same classes as each other. He also helped me to establish the database connection and the Internet Explorer connection. For a while, I was using my mission friend to help with configuring the teams. He is an ISYS student that is great a programming but not familiar with VBA. He worked with me for about five hours before I decided we needed to try a new method. At that point, I went to Dr. Allen once again and he figured out a structure that depended on calling many different sub procedures to see if the team was a good candidate for the student. After that, I needed to figure out how I wanted to display the data. It felt good to figure out something for myself. I wish I had learned more in this class so I was a more capable programmer but I know that I learned a lot that can help me automate different processes in my future career (which obviously will not be programming).