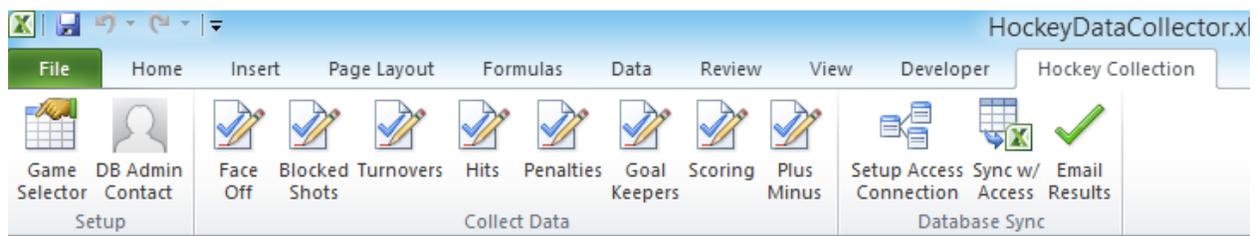


## Executive Summary

For the past eight months I have been interning with High Performance Hockey, a local hockey data analytics company. In my role as an intern I am responsible for aggregating all of the data collected by the data collection volunteers. This year High Performance Hockey collected all of the statistics for the BYU Men's Hockey Team. At the start of my internship I created a series of excel workbooks that served as templates for collecting the various statistics. After I input several thousands of these records by hand, I finally decided that there should be a much simpler way to collect the data. The goal of this project was to create a data collection system consolidated on to one workbook. I wanted this system to make it easy for anyone volunteering to collect data for the team. I also wanted the workbook to be able to sync to a centralized database, so that I never had to worry about consolidating several different excel files again.

My final project features several user forms that allow collectors to collect data in real time if they had their laptop or access to excel on a tablet at a hockey game. My project is adaptable on a game by game basis. Changes to the rosters of either team and the game schedule are reflected in the user forms. There are several different buttons in the "Data Collection" group on the ribbon, each opens a separate user form customized to collect data for that particular statistic. Whenever these forms are opened they create a new worksheet to hold the data, if that sheet does not already exist. Each form has "Previous" and "Next" buttons that allow the user to write data to the selected sheet and go back over the data if they need to correct an error in the previous entry. Once a user is done collecting data they can select the "Setup Access Connection" button to specify the location of the access database they would like to sync to. After the connection is set up they can click the "Sync with Access" button to send their data to a local access database. Before I had created this program I would have users edit a spreadsheet indicating that they collected a piece of data for a particular game. This worked with limited success, I would often find files in our shared DropBox folder that I was not aware of. For this reason I built in the option to notify the database administrator via email once data was synced to the database. The email simply notifies the administrator which tables were updated for which particular game. High Performance Hockey primarily relies on interns to collect their data, so the goal of this notification system was to make it easier to monitor if interns were collecting their assigned data in a timely manner.

## Detailed Description



All of the statistics collected by the various forms are statistics for "your team". The only statistics that collect data for the opposing team scoring. The player options for each statistic are populated accordingly from the rosters sheet. This program was designed to collect data for one particular game at a time to then be synced to the database.

## Set Up

Button Game Selector

Each worksheet has a foreign key variable “GameID” which corresponds to the game id’s set on the “Games” sheet. The “Games” sheet serves as a teams schedule for a given season. On the “Hockey Collection” ribbon tab under the “Set Up” group there is a “Game Selector” button which opens up a user form when clicked.

| Game ID | Home Team | Away Team | Date      | Current Game |
|---------|-----------|-----------|-----------|--------------|
| 1       | BYU       | OppTeam   | 1/31/2014 | 3            |
| 2       | OppTeam   | BYU       | 1/14/2014 |              |
| 3       | BYU       | OppTeam   | 2/24/2014 |              |
| 4       | BYU       | OppTeam   | 2/25/2014 |              |
| 5       | BYU       | OppTeam   | 2/26/2014 |              |
| 6       | BYU       | OppTeam   | 2/27/2014 |              |
| 7       | BYU       | OppTeam   | 2/28/2014 |              |
| 8       | BYU       | OppTeam   | 3/1/2014  |              |

Select Game

Select Game From Current List:

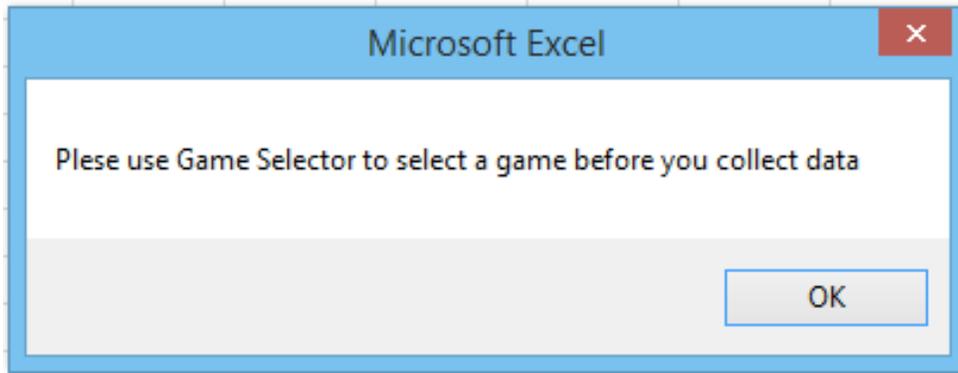
|                                |                                  |                                      |  |  |
|--------------------------------|----------------------------------|--------------------------------------|--|--|
| Game ID                        | Home Team                        | Visiting Team                        | Date                                   |  |
| <input type="text" value="6"/> | <input type="text" value="BYU"/> | <input type="text" value="OppTeam"/> | <input type="text" value="2/27/2014"/> | <input type="button" value="Select Game"/> |

OR

Add New Game:

|                      |                      |                      |   |
|----------------------|----------------------|----------------------|---|
| Home Team            | Visiting Team        | Date                 |   |
| <input type="text"/> | <input type="text"/> | <input type="text"/> | <input type="button" value="Add New Game"/> |

When “Game ID” is selected from the combo box the three fields adjacent populate with the data entries matching the GameID on the games sheet. This user form also gives the user the choice to add a new game to the sheet if they would like. If a new game is added it will select that game as the current game id. This is an important step in the data collection process. If this user form has not been opened and set the current game, when a user tries to open another data collection form a message box will appear informing them that they need to set a current game before collecting any data. Once they click ok the game selector form will automatically populate. After setting a current game, the game selector form will close and their originally selected form will populate.



#### Button DB Admin Contact

This button opens a user form that allows the user to set the contact details of the database administrator. Once these details are set the form will repopulate with the same values if it is opened again. This form also automatically populates if the user tries to send an email without setting the contact information first.

A screenshot of a "Database Administrator Contact" dialog box. The title bar reads "Database Administrator Contact" and has a red close button with a white "X" on the right. The form contains two text input fields. The first field is labeled "Data Base Administrator Name" and contains the text "Scott". The second field is labeled "Email Address" and contains the text "sryvola@gmail.com". At the bottom right of the form, there is a button labeled "Set Preferences".

#### Collect Data Group

Face off Button

When this user form is opened it creates a face-off sheet if it does not already exist. All of the other forms that will be described function in the same way with regards to sheet selection and generation. The game ID box is also visible on all of the other forms. The value represented is the game selected by the game selector form.

The option buttons on the picture of the rink allow the user to set the location of the face-off. These locations are represented as a numeric index in the data. All selected values are written to the sheet and repopulated by clicking the previous button if edits are needed.

#### Blocked Shots Button

## Turnovers

The screenshot shows a window titled "Turnover Collection" with a close button (X) in the top right corner. Inside the window, there is a "Game ID" field with the value "35". Below it is a "Select Player:" label followed by a dropdown menu. Underneath the dropdown are three radio buttons labeled "Period 1", "Period 2", and "Period 3". Below these is a "Zone:" label. Under "Zone:" are three radio buttons labeled "Offensive", "Neutral", and "Defensive". To the left of these is a "Type:" label with three radio buttons labeled "Give", "Take", and "Defensive". At the bottom of the form are two buttons: "Previous" and "Next".

This user form allows the user to track give aways and take aways of the puck by zone and period for each player.

## Hits

The screenshot shows a window titled "Hits Collection" with a close button (X) in the top right corner. Inside the window, there is a "Game ID" field with the value "35". Below it is a "Select Player:" label followed by a dropdown menu. Underneath the dropdown are three radio buttons labeled "Period 1", "Period 2", and "Period 3". Below these are two labels: "Type of Hit:" and "Zone:". Under "Type of Hit:" are two radio buttons labeled "For" and "Against". Under "Zone:" are three radio buttons labeled "Offensive", "Neutral", and "Defensive". At the bottom of the form are two buttons: "Previous" and "Next".

This form is very similar to turnovers. It tracks the hits that players are giving and receiving by zone and period.

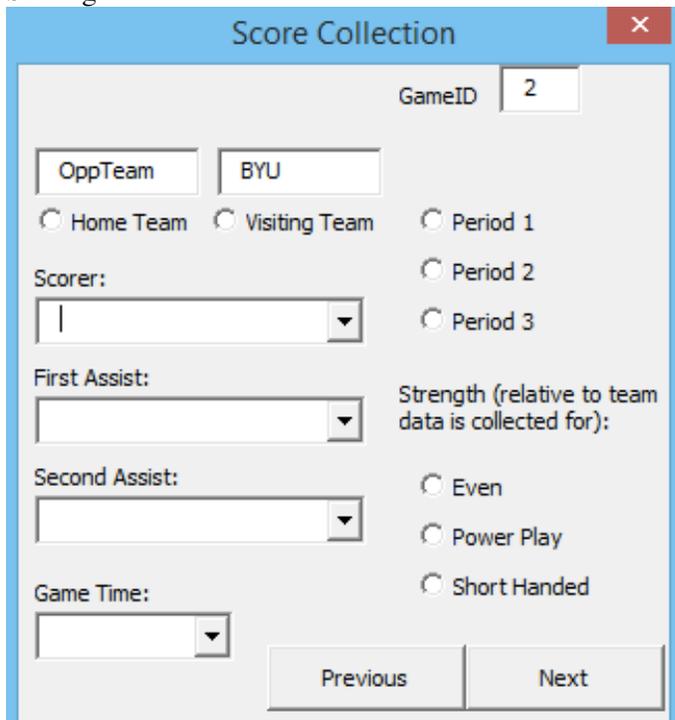
## Penalty

The penalty collector allows a user to record the penalties awarded to your team. The infraction combo box is populated in the code with a list of all the major and minor penalties that are possible. Users can also select how long the penalty was and the game time they entered and exited the box.

#### Goal Keeper

The SOG, Saves, and Goals are all drop down boxes to record the number of each event by period. Since this statistic rapidly occurs the +1 buttons were added to increase the current value by one, as if you were collecting it live.

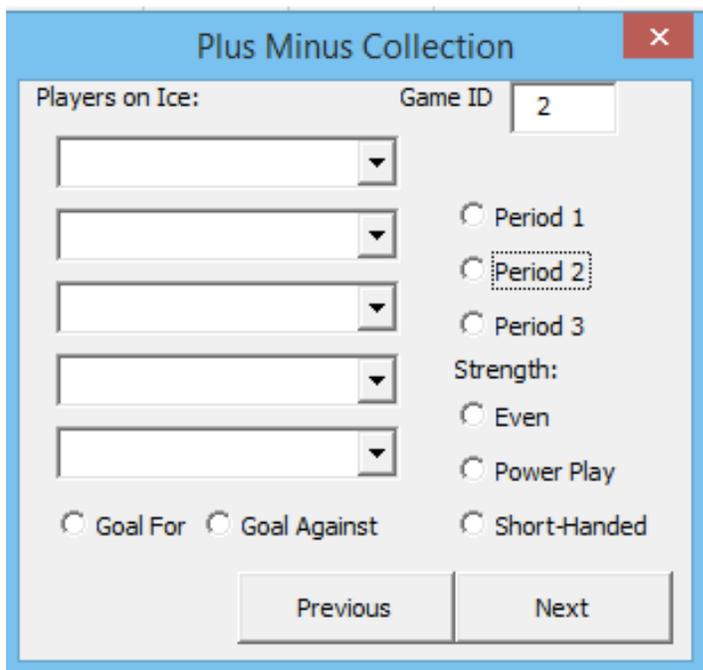
## Scoring



The "Score Collection" dialog box is used for recording game events. It features a title bar with a close button (X) and a "GameID" field containing the value "2". Below the title bar, there are two text boxes for "OppTeam" (containing "BYU") and "Home Team". There are three radio buttons for "Home Team", "Visiting Team", and "Period 1". Below these are three radio buttons for "Period 2" and "Period 3". There are three dropdown menus for "Scorer:", "First Assist:", and "Second Assist:". To the right of these dropdowns is the text "Strength (relative to team data is collected for):". Below this are three radio buttons for "Even", "Power Play", and "Short Handed". At the bottom, there is a "Game Time:" dropdown menu and two buttons labeled "Previous" and "Next".

Home team and Visiting team text boxes are populated from the games sheet. Selecting the option below either will write that team to the record. Scoring, FirstAssist, and SecondAssist are all populated from the rosters sheet for both teams.

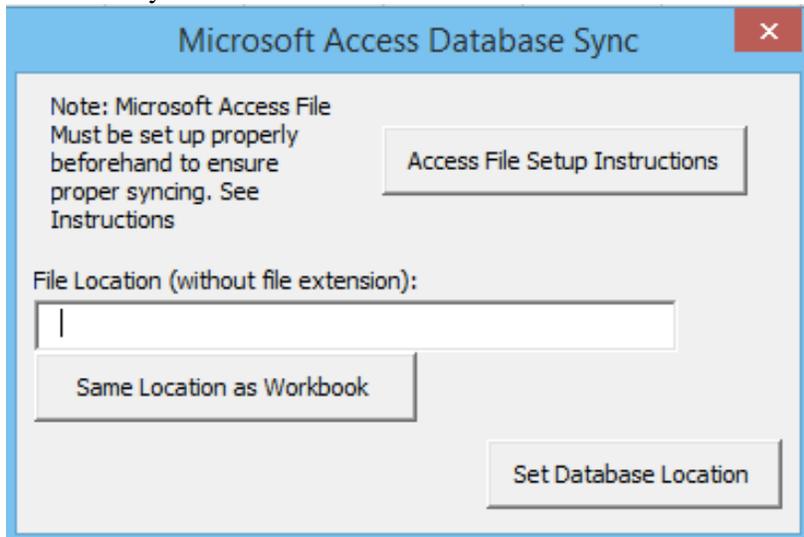
## Plus Minus



The "Plus Minus Collection" dialog box is used for recording plus/minus statistics. It features a title bar with a close button (X) and a "Game ID" field containing the value "2". Below the title bar, there is a "Players on Ice:" label and five empty dropdown menus. To the right of these dropdowns are three radio buttons for "Period 1", "Period 2", and "Period 3". Below these are three radio buttons for "Strength:" with options "Even", "Power Play", and "Short-Handed". At the bottom, there are two radio buttons for "Goal For" and "Goal Against", and two buttons labeled "Previous" and "Next".

Plus Minus is a point system for hockey players that measures their scoring contributions to the team. Players receive a +1 for being on the ice when a goal is scored and a -1 when they are scored on. This form allows the user to record these values. This form writes out a separate record for the number of players selected from the combo boxes. The previous button can only be used to edit the last goal. If there were a consistent number of players on the ice for such events I would have had the previous button be able to scan back through all of the events. However, since I could not count on a specific number of players for each event (sometimes there are 4 or 3 players on the ice in penalty situations) I was only able to get it working for the one event before.

#### Data Base Sync



The screenshot shows a dialog box titled "Microsoft Access Database Sync". It contains a note: "Note: Microsoft Access File Must be set up properly beforehand to ensure proper syncing. See Instructions". There is a button labeled "Access File Setup Instructions". Below this is a text input field labeled "File Location (without file extension):" which is currently empty. There are two buttons below the text field: "Same Location as Workbook" and "Set Database Location".

This form allows the user to set the location of a local access database file. The setup instructions button will populate a instructions worksheet with instructions of the table and column names that are required in the database in order to sync. Clicking the “same location as workbook” button will populate the text box with the current path of the workbook.

My program was designed with file sharing through DropBox in mind, so if the user placed this program along with the access database in a shared DropBox folder multiple users could all write to the same database simultaneously. In the future I would also like to add SQL syncing abilities, but I ran out of time to complete that feature for this project.

The sync with access button will scan through the sheets created by opening the various user forms and only update the tables if there is data present. If the Access Setup form has not been completed yet the sync button will open that form. After the sync is complete the user has the option to send the results to the database administrator with their gmail account. This code is heavily based off of the example we worked through together in class. As the data is syncing it sets boolean variables to True if a table is updated. The body of the email sent to the database administrator is based off of these values.

☆ sryvola@gmail.com Ryvola  
To: Scott Ryvola  
Reply-To: sryvola@gmail.com Ryvola  
New Database Entries



---

I just updated the database tables: Faceoff, Turnovers, Hits, Penalty, Scoring,for GameID 3

## What I learned

I had some frustrating times with this project but in the end I really enjoyed working on it. I learned that a lot can be done with user forms. Working with these forms helped solidify exactly how the project scope for variables works. I also became much more familiar with Access, which I liked because I have not had much experience with it in the past.

I had plans to implement some results on my worksheets that were generated from querying my database but I learned that the execute command only works for action statements in Excel VBA, to my disappointment.

One thing that tripped me up for a while was forgetting to put single quotes around the string variables present in my sql statements, even though I had learned that concept already when I was completing the database project. Now that I spent a couple hours frustrated with that error I won't forget to do that ever again.

Perhaps the largest thing I learned is that both "Time" and "Zone" are reserved words when writing sql statements to an access database. I spent over 8 hours trying to figure out why some of my sheets would update to the database while others would not. This single error is the only reason why I had to turn in my project one day late. I learned that if you want to use reserved words as column names they need to be inclosed in square brackets in your insert into statements.