Manual to Use SQL via TeamViewer

# Table of Content

[Table of Content 1](#_Toc427940067)

[Getting Started 2](#_Toc427940068)

[Using TeamViewer 3](#_Toc427940069)

[Understanding SQL 4](#_Toc427940070)

[Example of SQL codes 6](#_Toc427940071)

[Case Study # 1: 7](#_Toc427940072)

[Case Study # 2: 10](#_Toc427940073)

[Case Study # 3: 12](#_Toc427940074)

[Case Study # 4: 13](#_Toc427940075)

# Getting Started

**Website** (for reference): <http://www.w3schools.com/sql/default.asp>   

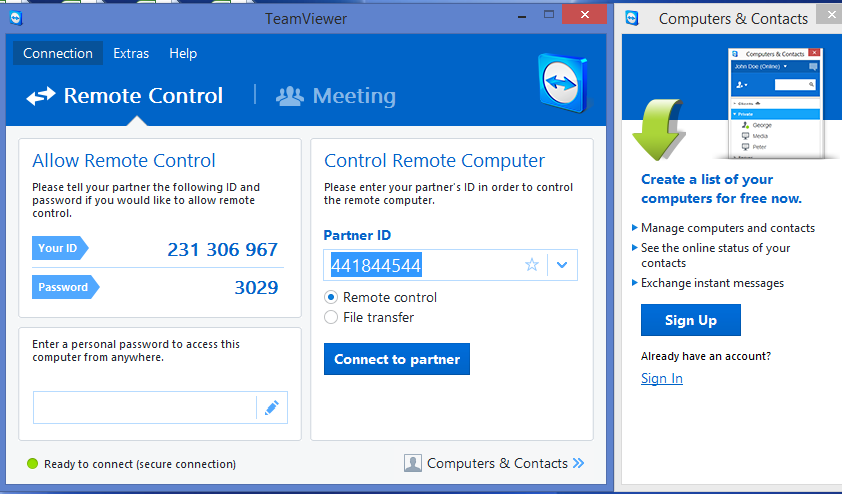
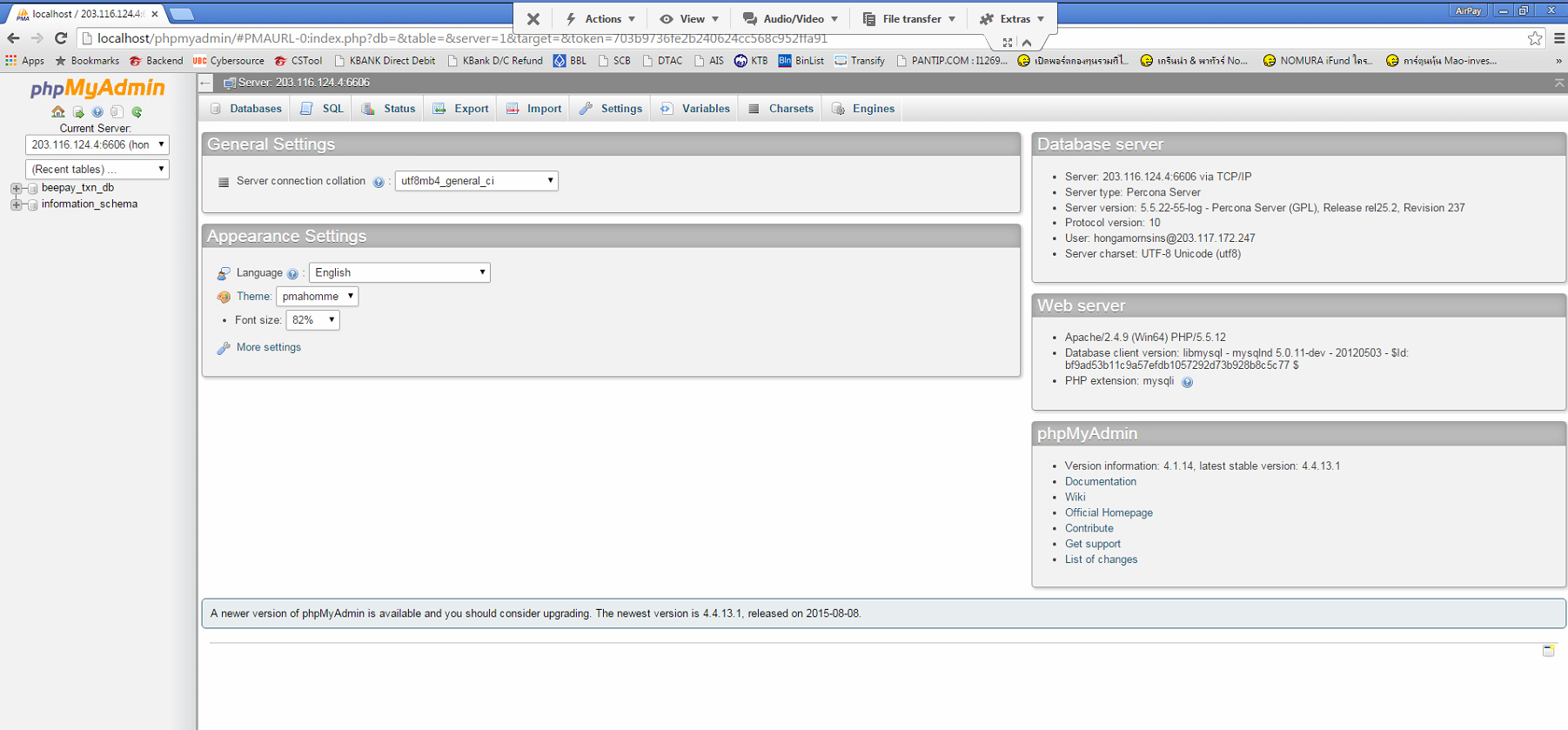

**Programs:**

1. **Sublime Text 2** (<http://www.sublimetext.com/2>)   
   Note that Sublime Text 2 is for writing the SQL code.   
      
   Click on link “Windows 64 bit” to start the download.
2. **TeamViewer 10** (<https://www.teamviewer.com/en/download/windows.aspx>)   
   (Used to look up confidential data)  
   

Note! Before using TeamViewer, remember that it is more convenient if you can find anything under [https://admin.airpay.in.th/manager/real\_time/#](https://admin.airpay.in.th/manager/real_time/).

**Refer to** airpaySQL.sql for examples of completed SQL codes that I have created using Sublime Text 2.

# Using TeamViewer

1. Open TeamViewer. (should look similar to the screenshot below)  
   
2. **Partner ID**: 441844544
3. **Password**: airpay69
4. Once the connection is established, the bottom-right-hand corner will look like this:   
   Note that if there would be two icons shown: a green icon indicating that “start WampServer64” has been opened, and “lock key icon with green background” from “OpenVPN GUI” program. [If not, open “OpenVPN GUI” and type down the same password. (airpay69)]
5. Open Google Chrome and type down “localhost/phpmyadmin” in the address bar. When connected: 

# Understanding SQL

The difficult part is writing the SQL codes and figuring which part could be done easier in Microsoft Excel.

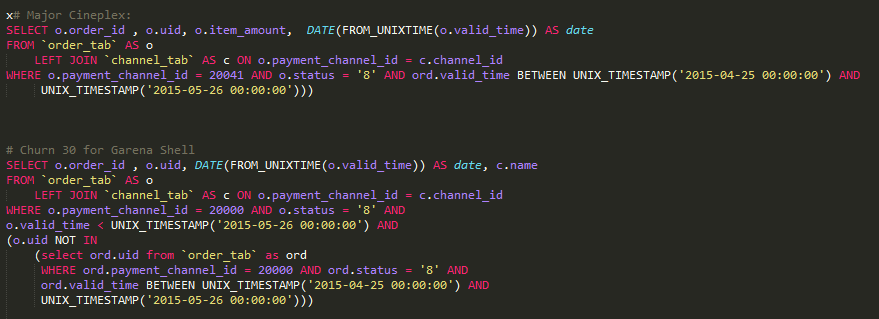
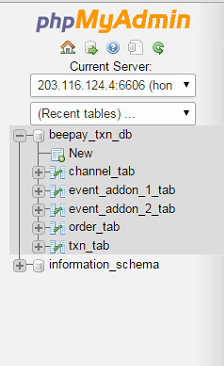


Figure : Screenshot of SQL codes

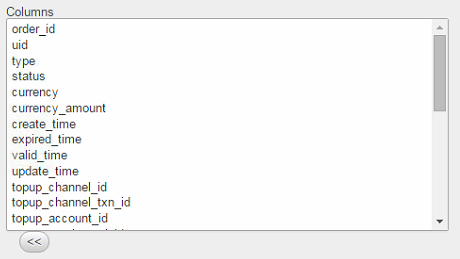
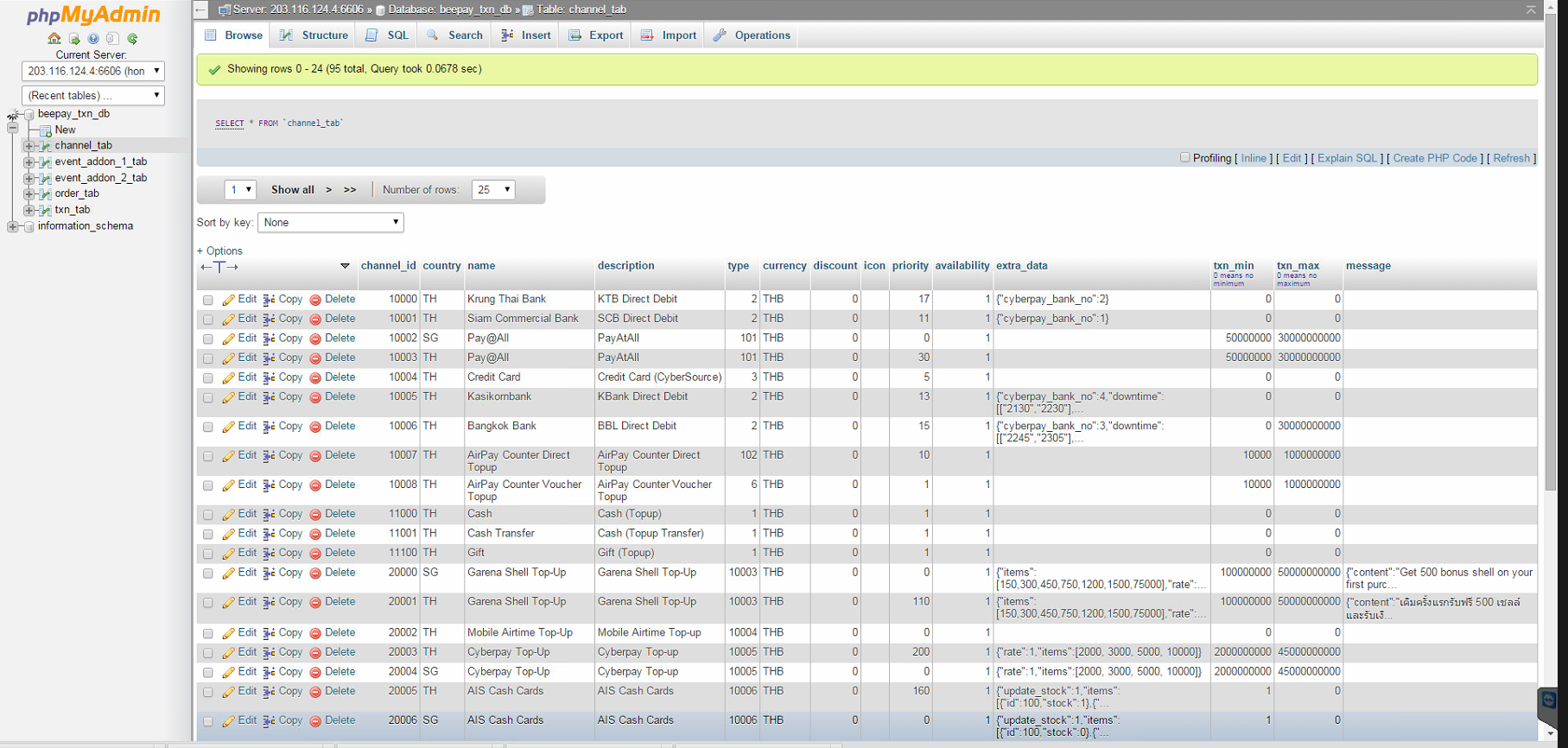
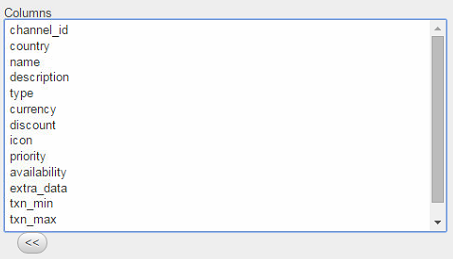
As seen in Figure 1, SQL codes used to extract data begin with “**SELECT**”. To put it simply, “SELECT” tells the computer what kind of information we desire. The syntax of the most basic SQL statements is as follows: “SELECT \* FROM X”.

The word “FROM” specifies which file or database we want the information from. Since we are only interested in transactions, the database that we are mostly interested in is “order\_tab” database.   
(5 main databases: “order\_tab”, “channel\_tab”, “txn\_tab”, “event\_addon\_1\_tab”, and event\_addon\_2\_tab”)



# Two Important Databases

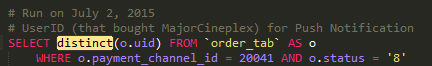
There are only two main databases that we extract data from.

1. order\_tab   
   Data: (uid, order\_id, valid\_time)  
   
2. channel\_tab  
     
   Data:  
     
   “channel\_tab” is used to filter down the data by using “LEFT JOIN” in SQL.

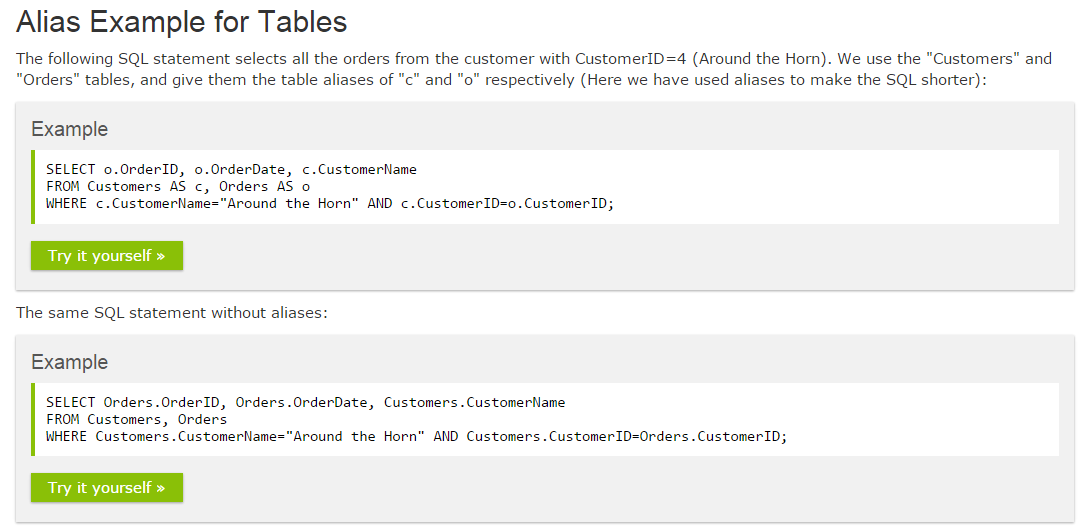
From Mind’s example, the only reference to “channel\_tab” is to using LEFT JOIN to filter down the data.

1. txn\_tab (never used)
2. event\_addon\_1\_tab (never used)
3. event\_addon\_2\_tab (never used)

# Example of SQL codes

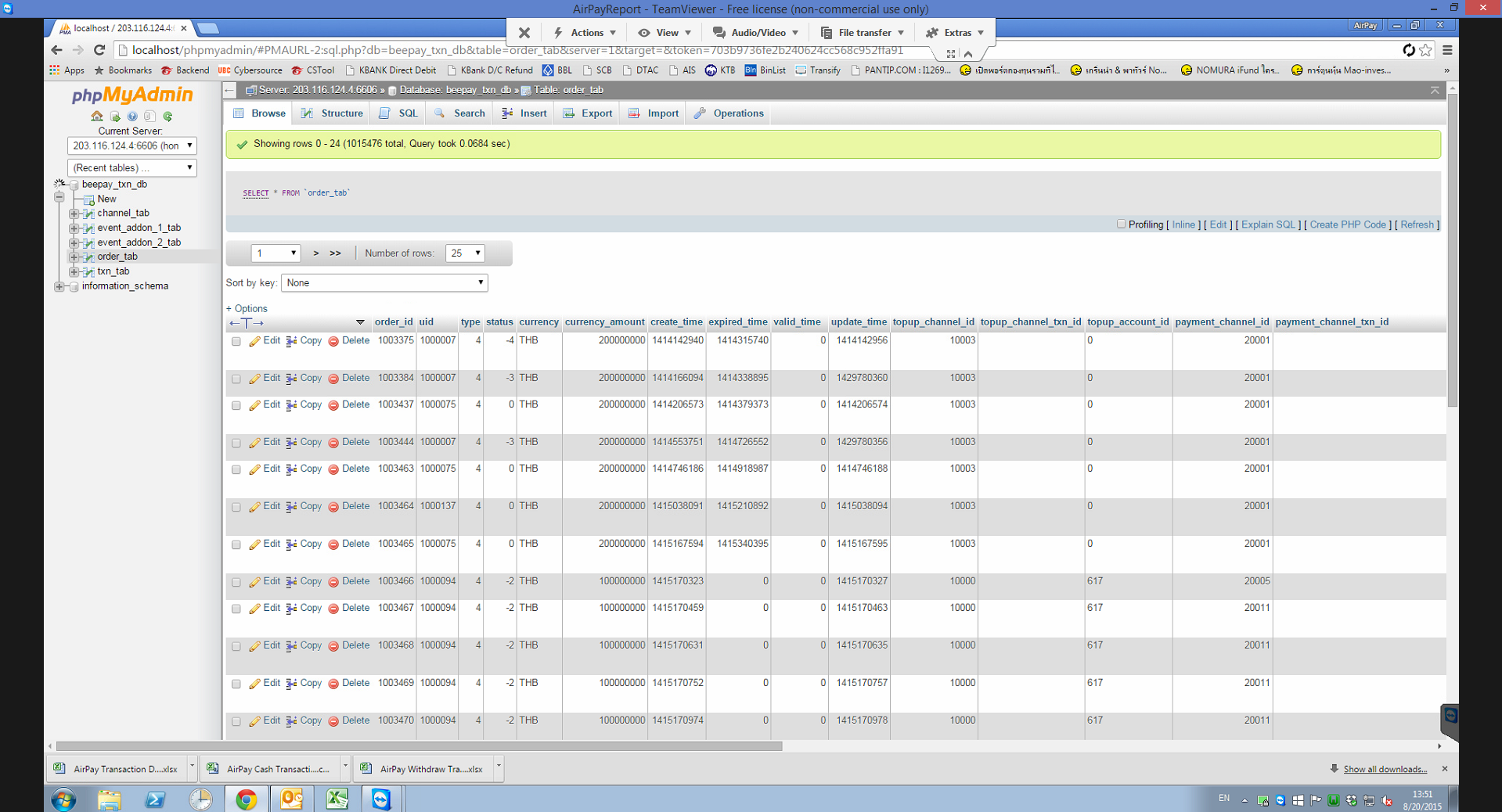
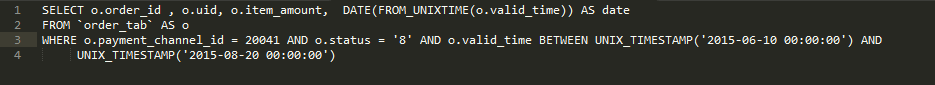
  
  
This simple example shows you the basic syntax for SQL codes. As written in the comments, we want the list of User IDs who have bought Major Cineplex tickets; therefore, we only “**SELECT**” user ids, or “**o.uid**”. In this case, the word “**distinct**” indicating how we do not want duplicated User IDs is used, because a user may make two transactions for Major Cineplex tickets, but we only want to include his User IDs once.

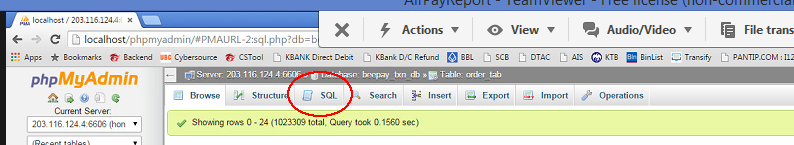
The word “**AS**” which stands for ‘aliases’ is used to shorten the SQL codes by renaming a column heading.   
[from “SELECT distinct(order\_tab.uid) FROM ‘order\_tab’”]

Regarding the rest of the code, the word “**WHERE**” is used to narrow down the list. In this case, we are only interested in UserID that have bought tickets from Major Cineplex only, so we use the specify “**o.payment\_channel\_id = 20041**”. Lastly, “**o.status ‘8’**” means that the users have purchased the tickets successfully.   
  
Look at the screenshot or w3schools for more information. (<http://www.w3schools.com/sql/sql_alias.asp>)   


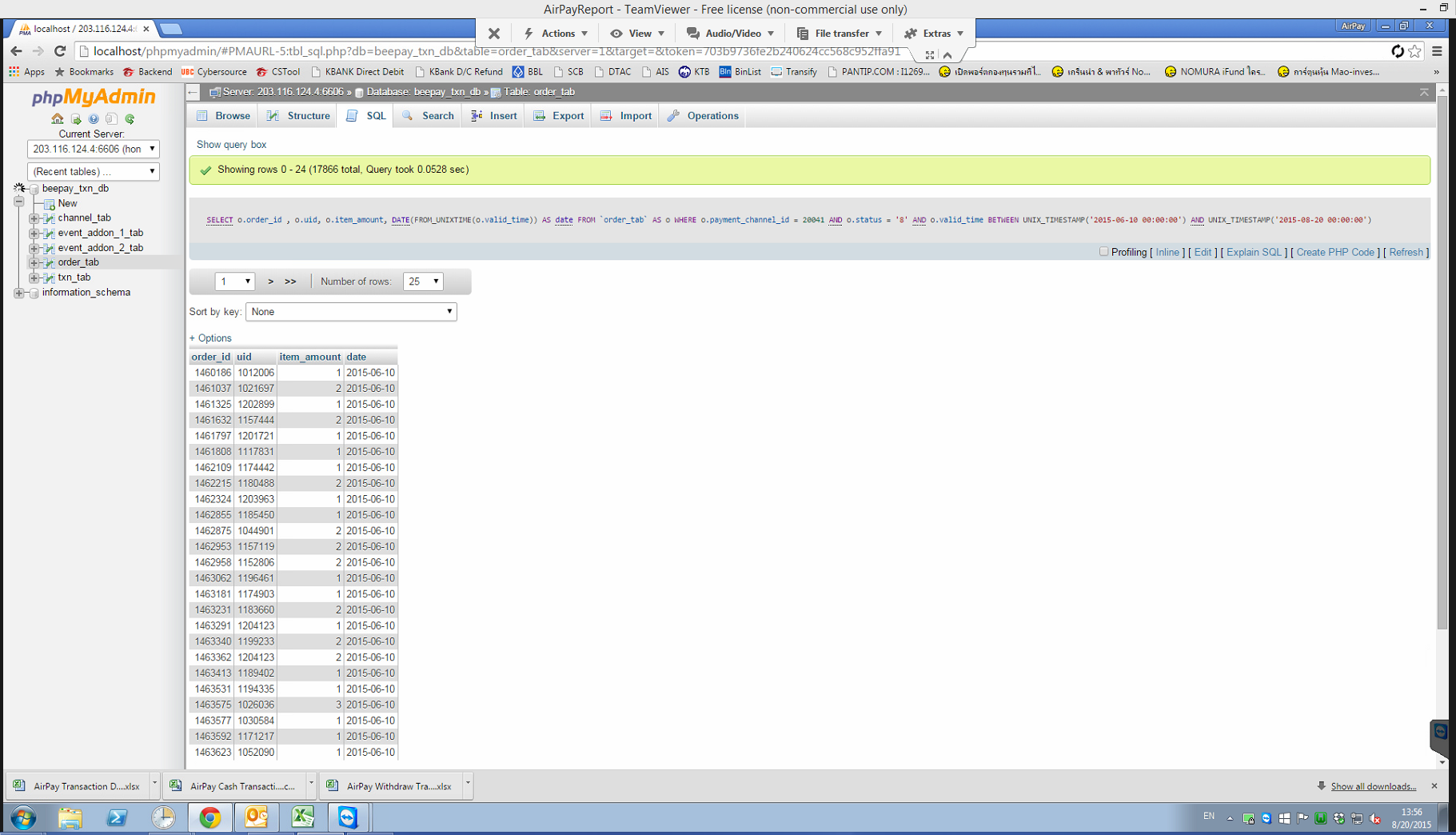
# Case Study # 1:

Prompt: “Find list of UserIDs and number of tickets MajorCineplex Tickets purchased from June 10 till August 20”  
Note: Please read this first (most descriptive). See “#1-majorcineplexTicket.xlsx”.

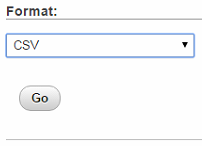
1. Once localhost/phpmyadmin/ is opened, click “order\_tab” under “beepay\_txn\_db”. The page will look like this:  
   
2. Create the SQL code to use. For this one, we can enter the following code:   
   
3. Then, click on “SQL”. (between “Structure” and “Search” tab)



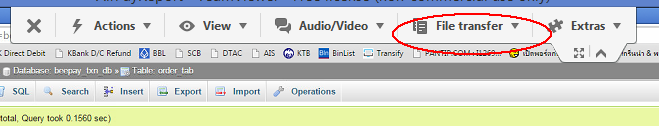
1. Paste your code and click “Go”. After clicking “Go”:



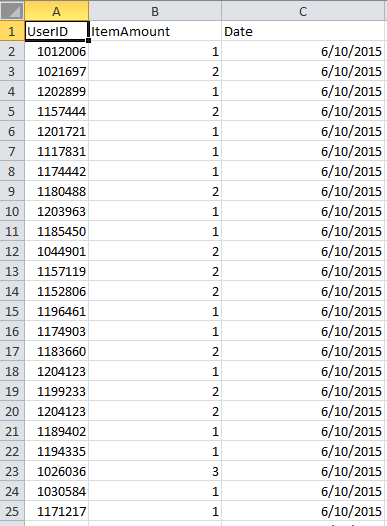
1. Scroll down and click on “Export”. Select “CSV” and click “Go.”  
   As a note, CSV stands for “comma separated values” file and “allows data to be saved in a table structured format.”



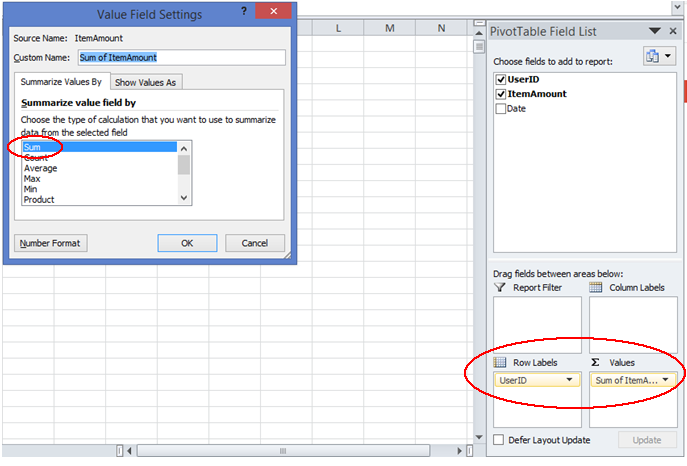
1. A new file will be downloaded.
2. Click “File transfer” under “File Transfer” [feature of TeamViewer] to transfer the file to your computer. .

Your window will look similar to this:   


1. Select the name of the file that is to be exported on P Leng’s computer under “Remote Computer”. Choose the destination to whether the file is to be exported on your computer under “Local Computer”.   
   [e.g. Desktop, Download]
2. Click “Receive” to transfer the file.
3. Open the .csv file in your computer using Microsoft Excel.
4. Insert an additional row in the first column and create a label for each specific column respectively.  
   For instance,



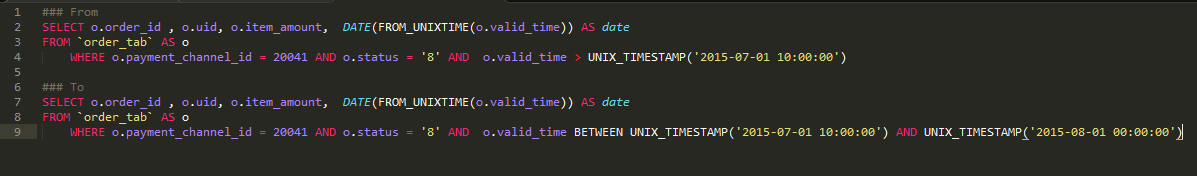
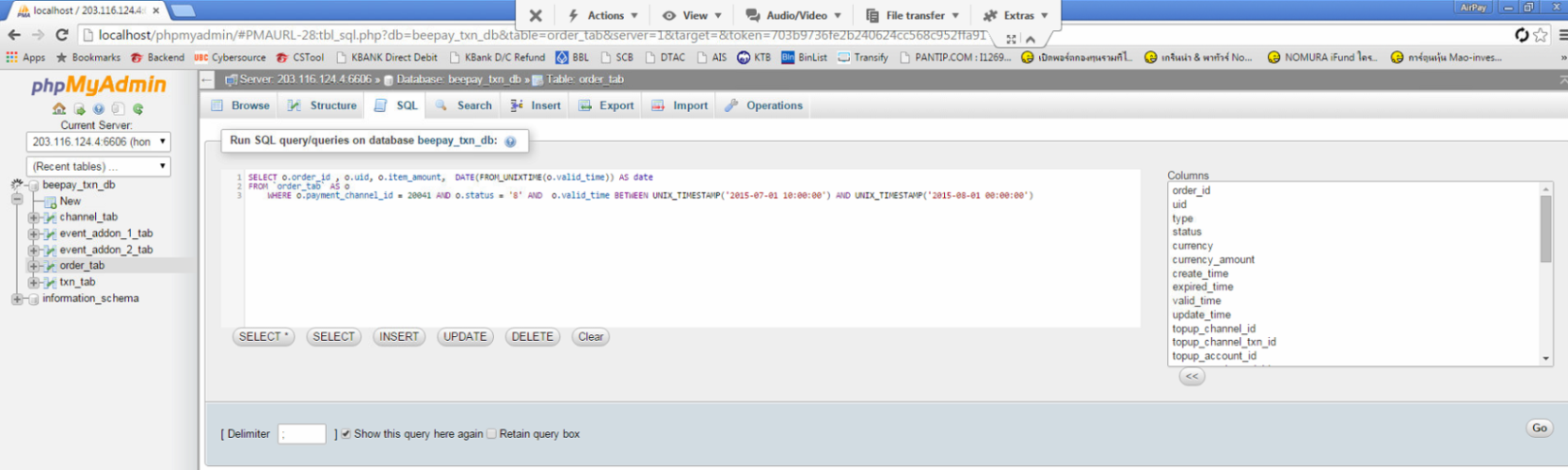
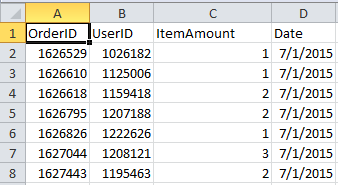
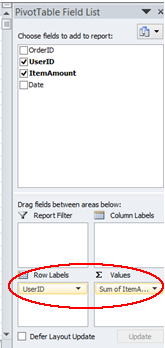
1. Use Pivot table to find the cumulative # of tickets purchased by each person during the interval.
2. Drag UserID to “Row Labels” and “ItemAmount” to “Sum Values” and edit “Value Field Settings” by right-clicking “ItemAmount”. Choose “Sum” since we want the sum of the tickets purchased and click “OK”.



1. Congratulation! Now you are done.   
   Additional Note: click “save as” to save in excel format.

# Case Study # 2:

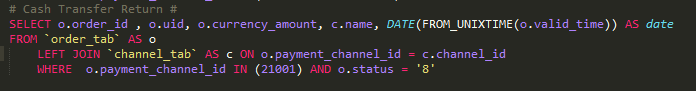
Prompt: “Find Top movie ticket buyers for Maroon 5 Event”  
See “#2-maroon5.xlsx”.

1. Create the SQL code.
2. Edited the code to the right time interval. Note the difference before and after in the picture above.
3. Pasted the new code under “SQL” section in “order\_tab”.
4. Transfer the file to your computer using the same method as above.   
   (File transfer 🡪 receive)
5. Open the .csv file and label each column.   
   
6. Use “Pivot Table”. We want to calculate the total tickets bought for each UserID.  
   Hence, drag UserID to Row Labels, and Item Amount to Sum Values.   
   
7. Copy the data into a separate sheet. Filter the second row from largest to smallest.
8. Congratulation! You are done.

# Case Study # 3:

Prompt: “Find total users who use cash transfer and the percentage of users who only use cash transfer once.”  
See “#3-CashTransfer.xlsx”.

1. Create the SQL code.



1. Pasted the new code under “SQL” section in “order\_tab”.
2. Transfer the file to your computer using the same method as above.   
   (File transfer 🡪 receive)
3. Open the .csv file and label each column.

  
Note: In this case, the “FinalAmount” is calculated by dividing the value of “Currency” by 10^6.

1. Use “Pivot Table”. We want to calculate the total tickets bought for each UserID.  
   Hence, drag UserID to Row Labels, and OrderID to Sum Values.   
   Remember to edit the “Value Field Settings” from “Sum” to “Count”.
2. Copy the raw data into a separate worksheet.
3. Delete repeated values using “Remove Duplicates” under Data Tools. (only select UserID)
4. Copy the “Count of OrderID” and paste it to a new row in this worksheet.
5. Use “Pivot Table” a second time. Drag “Counter of OrderID” to Row Labels, and UserID to Sum Values.   
   Remember to edit the “Value Field Settings” from “Sum” to “Count”.
6. Congratulation! You are done.   
   Total is from the “Grand Table” from the Pivot table (the 2nd time). Top-up once is from the “1” row.   
   Top-up more than once is calculated by subtracting Total with Top-up once.  
   To draw the graph, copy the values from the Pivot table into new rows. Then, drag over the data values and click insert graph.

# Case Study # 4:

Prompt: “Find total users who use airtime top-up and the percentage of users who only use airtime top-up once.”  
See “#4-Airtime.xlsx”.

1. Create the SQL code. (similar to last one)
2. Pasted the new code under “SQL” section in “order\_tab”.
3. Transfer the file to your computer using the same method as above.   
   (File transfer 🡪 receive)
4. Open the .csv file and label each column.

  
Note: In this case, the “FinalAmount” is calculated by dividing the value of “Currency” by 10^6.

1. Use “Pivot Table”. We want to calculate the total tickets bought for each UserID.  
   Hence, drag UserID to Row Labels, and OrderID to Sum Values.   
   Remember to edit the “Value Field Settings” from “Sum” to “Count”.
2. Copy the raw data into a separate worksheet.
3. Delete repeated values using “Remove Duplicates” under Data Tools. (only select UserID)
4. Copy the “Count of OrderID” and paste it to a new row in this worksheet.
5. Use “Pivot Table” a second time. Drag “Counter of OrderID” to Row Labels, and UserID to Sum Values.   
   Remember to edit the “Value Field Settings” from “Sum” to “Count”.
6. Congratulation! You are done.   
   Total is from the “Grand Table” from the Pivot table (the 2nd time). Top-up once is from the “1” row.   
   Top-up more than once is calculated by subtracting Total with Top-up once.  
   To draw the graph, copy the values from the Pivot table into new rows. Then, drag over the data values and click insert graph.