**PHP/MYSQL Related API Documentation**

**INTRODUCTION**

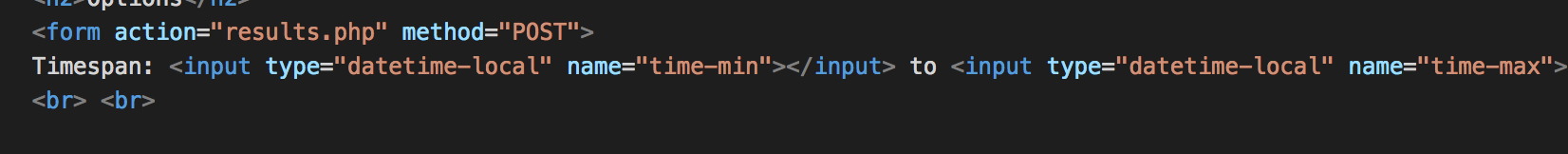
The purpose of this document is to demonstrate the relationship and functionality between the webserver (PHP/HTML interface) and the MYSQL table stored in phpMyAdmin. PHP offers a variety of MySQL drivers and plugins to handle and access MySQL.

**** PHP offers three extensions (mysql, mysqli, PDO) to connect to the MySQL database. The mysqli extension was chosen for this project due to its relatively easy functionality and inclusion with PHP 5.X and above. The code snippet below creates a connection to the MySQL server running on database.cs.tamu.edu, using the specified username and password. A query is then run to establish a secure connection.

The default time zone was also set to ‘america/chicago’ to aid in time/date calculations. Let’s switch to the HTML interface to discuss a few important concepts.

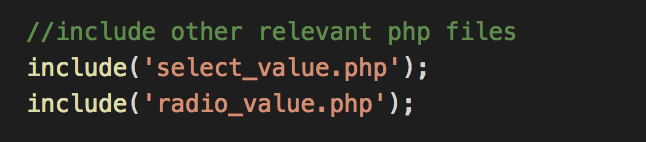
**FORM.HTML**

This is the interface that designs the layout and overall functionality of the web user interface. It loads in a few css style sheets to add visualization features such as customized color, font, and background of the website.

 The form also performs and HTTP POST request to a PHP form named ‘results.php’. This means that when a user enters the date-time ranges into the textbox input fields, the form redirects to `results.php` to display the results.

**RESULTS.PHP**

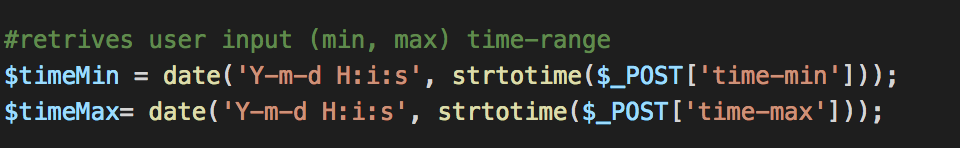
This php file includes two other php files:

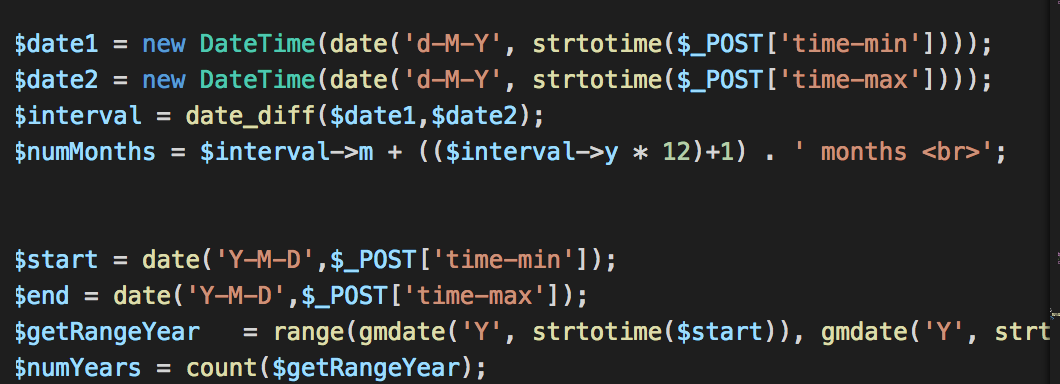


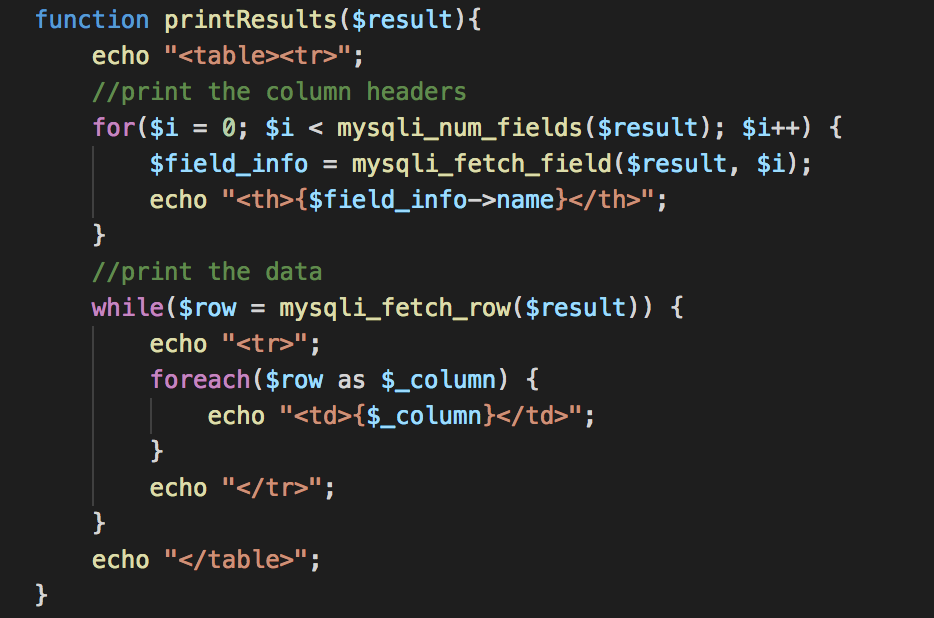
1. **select\_value.php:** when a select option (daily,weekly,monthly,yearly) is selected, it will be registered and passed to results.php.
2. **radio\_value.php:** when any radio button (count,average,statistics) is selected, it will be registered and passed to results.php.

***Global Variables***

The time-values entered in by the user are stored in php variables $timeMin and $timeMax accordingly. These values are formatted in date-time format [‘Y-m-d H:i:s’].



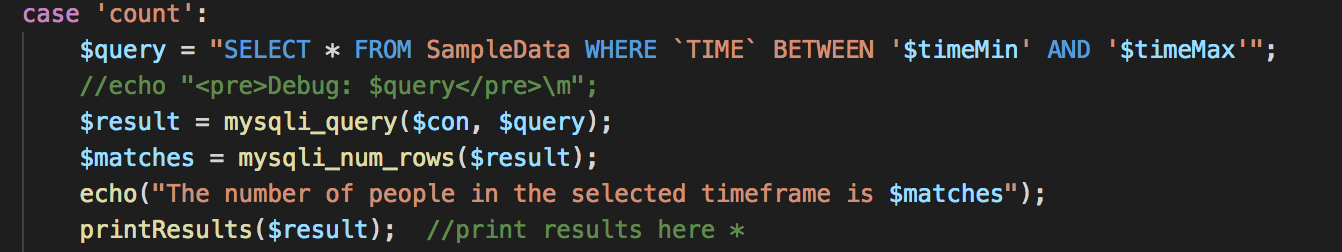
 Two new DateTime objects called **$date1**, **$date2** are instantiated to extract only the date part of **$timeMin** and **$timeMax** respectively. The $interval variable calculates the difference between the two dates and **$numMonths** records the number of months between the two dates. The **$numYears** variable is calculated in a similar manner and indicates the number of years between two given dates.

***function printResults($query)***

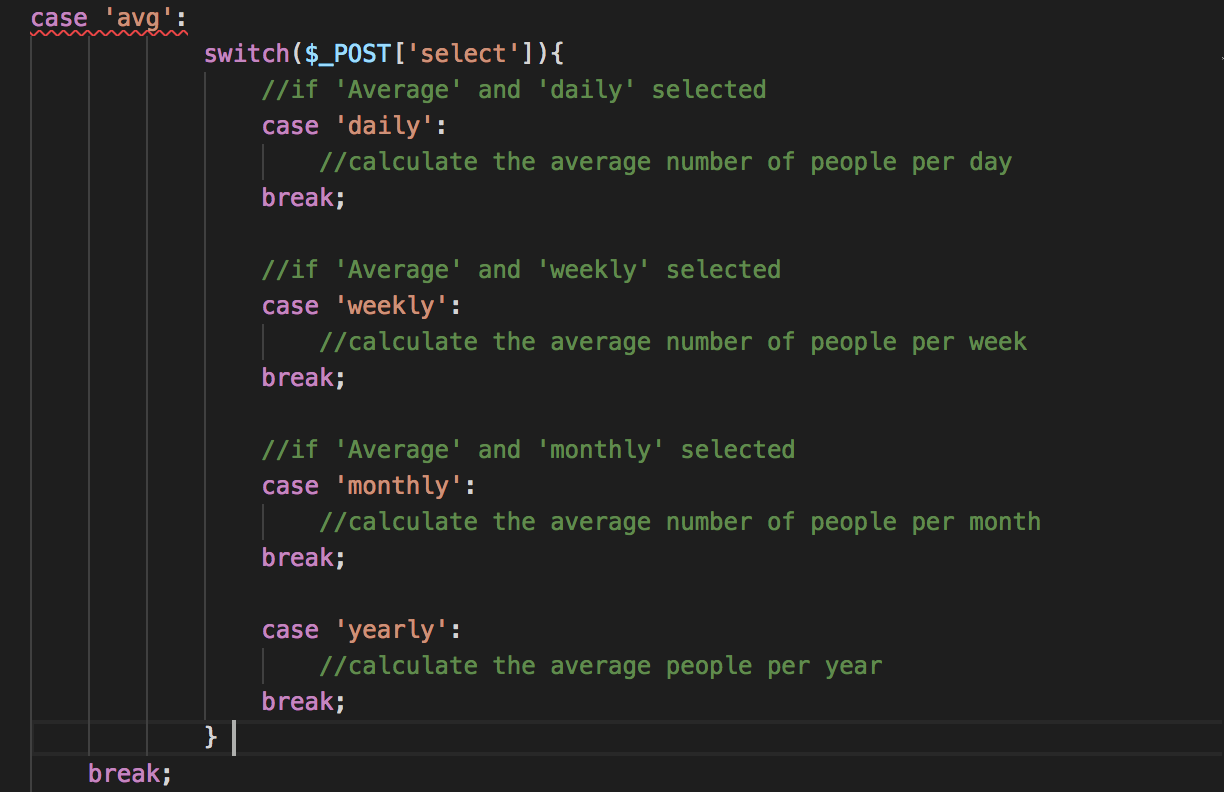
* The purpose of this function is to print the results of a query, which is taken in as an argument.
* It loops through the number of fields in the MySQL table and fetches the field information such as header labels and data that matches the query. It then outputs all relevant information that’s been collected.

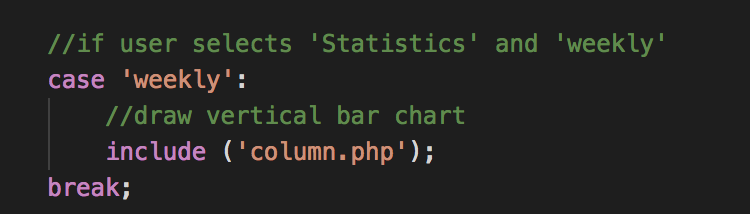
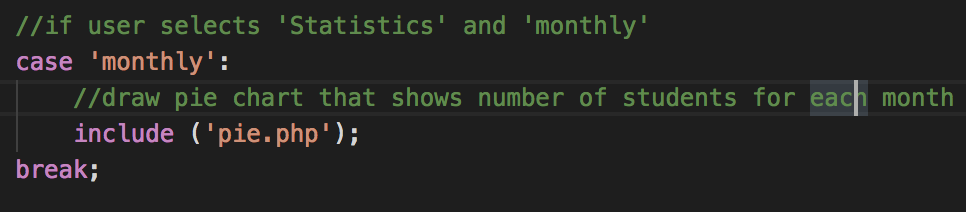
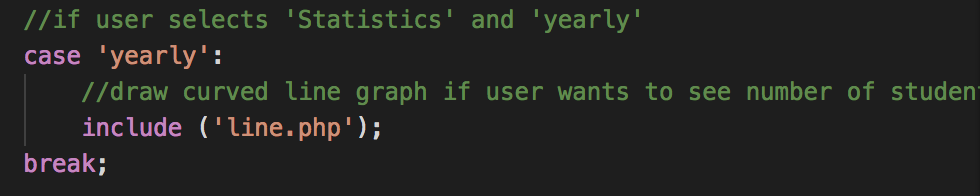
Following the printResults function declaration is the switch case statement which decides which queries to run based on user selection.

* If the user selects ‘Count’, then the following query is executed.



* + The query selects all the rows that are between the time range. The variable **$matches** counts the number of rows that match the result of the query. Then, a simple echo statement follows, outputting to the screen the number of students in the selected timeframe.
* If the user selects ‘Average’, then the result varies based on if the user wants to figure out the average number of students by day, week, month or year.



* + The code snippet above shows the general structure of the switch case statement for average calculations. For the logic of each case implementation, please refer to results.php.
* If the user selects ‘Statistics’, then the form outputs a different kind of graph based on the unit time frame selection: weekly, monthly, yearly.
  + For case ‘weekly’, a bar chart is drawn by including the file ‘column.php’. This shows the number of students per day for the week selected.
  + For case ‘monthly’, a pie chart is drawn by including the file ‘pie.php’. This shows the number of students for each month.
  + For case ‘yearly’, a line chart is drawn by including ‘line.php’. This shows the number of students for each year in the selected time range.

See the table below for more information on the google charts php files.

|  |  |
| --- | --- |
| **Include File Name** | **Purpose** |
| column.php | google charts tool that establishes a connection with the database and outputs the results of a query in the form of column chart; this is used for statistics weekly. |
| pie.php | google charts tool that outputs the results of a query in the form of pie chart; this is used for statistics monthly. |
| line.php | google charts tool that outputs the results of a query in the form of line graph; this is used for statistics yearly. |