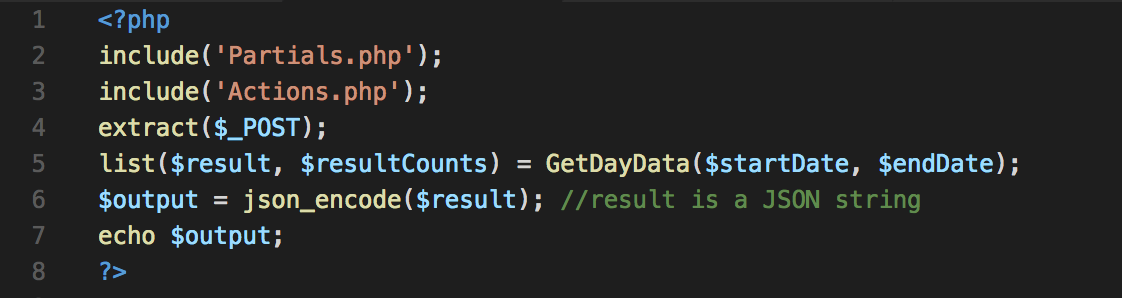
**Coding Improvements Documentation**

PART I: WEBSERVER

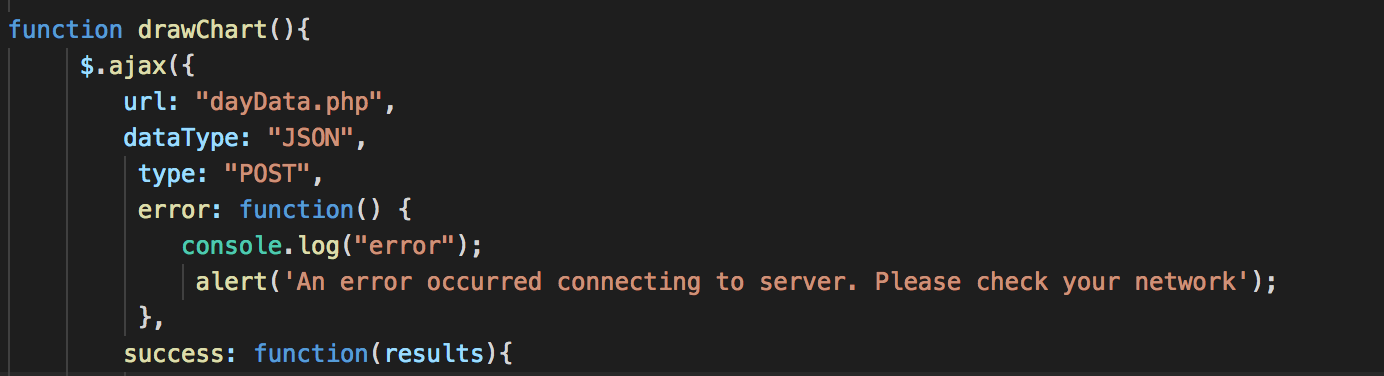
The main enhancement made to the web interface is the addition of the auto-refresh feature. Before successful implementation of this feature, a user would have to manually refresh the page to render the most up-to-date information from the MySQL database. The auto-refresh feature was implemented with the integration of the AJAX/JQuery and Google Charts API.

The ajax works by referencing a php file that is located within the same directory as the javascript file that invokes the ajax call. Four new javascript and php files were created for each time interval (hour, day, week, month). All these files can be found directly within the server directory. In the php file, two other files, Actions.php and Partials.php, are included to retrieve and format the data based on a specific time input range.

 A sample dayData.php is shown below:

As shown above, after the data is retrieved by calling GetDayData, the results are stored in a php variable called $result. Then, $result is cast to a JSON string value by calling json\_encode function. This is important because now dayData.php returns data in a json Format as shown below:



 Now, within the drawChart() in LoadDayData.js, the ajax function is invoked. The picture below shows the parameters/values loaded by the ajax call.

Within the success component of the ajax call, the input JSON data is formatted, a chart is drawn according to user-specified options and google API packages, corechart and bar, are loaded in that order.

The ajax is implemented with a timer which is called with setInterval(drawChart, 5000); this renders the charts every 5 seconds, extracting the changes made to the phpMyAdmin database within that interval. The same method is applied to the other three time intervals (hour, week, month).

A view of DayChart.php is shown below, displaying the count of people entering and exiting on each date. The user can filter the Analysis Table results by selecting a location.

