600.415.FINAL PROJECT

Member:
Xiaoxiao Liu xliu91@jhu.edu
Da Chen dachen78@jhu.edu

Domain:

Through this database, we provide real-time data and comparisons between different attributes of air quality for different cities in China. The purpose is to demonstrate regional air pollution statistic data and potential major causes for different cities.

To view our results:

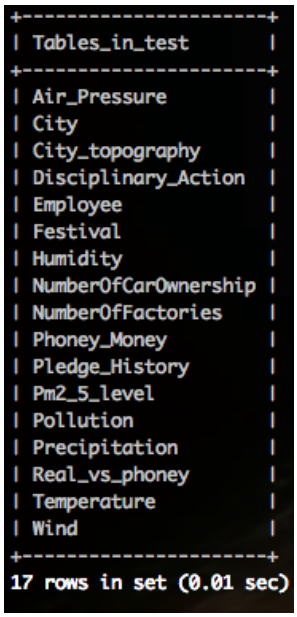
Our design and codes are all under git SVN control,
Github link: https://github.com/xliu59/air_pollution

- **Databases:**

  Uses course-assigned MySQL db:
  username: xliu91
  password: kzpahbbunq
  database: test

  Updated Tables & Schemas:

  The only changes are adding a column 'City_name_eng' and delete 'Province' attribute to City table.



```
+---------------------+
| Tables_in_test      |
+---------------------+
| Air_Pressure        |
| City                |
| City_topography     |
| Disciplinary_Action |
| Employee            |
| Festival            |
| Humidity            |
| NumberOfCarOwnership |
| NumberOfFactories   |
| Phoney_Money        |
| Pledge_History      |
| Pm2_5_level         |
| Pollution           |
| Precipitation       |
| Real_vs_phoney      |
| Temperature         |
| Wind                |
+---------------------+
17 rows in set (0.01 sec)
```

- **Data source**

  Our data source consists two parts:

  1) From public data APIs, examples are hourly-requested *whether & pollution data*, *last-hour pm2.5 index, city air pollution level*:

  - City Air Quality API (Real-time synchronization):
    http://pm25.in/
  - City Weather API (Real-time synchronization):
    http://openweathermap.org
  - Festival API:
    https://www.timeanddate.com/holidays/

  For these data, we wrote data-retrieval python scripts to request and process data

- Github link:
  https://github.com/xliu59/air_pollution/tree/master/air_pollution/data_retrieve

2) Dataset downloaded from internet, examples include *(City_Chinese_name, City_English_name, City_Longitude_Latitude):*

- City locations
  http://wenku.baidu.com/view/31396223482fb4daa58d4b61.html
  http://blog.csdn.net/a497785609/article/details/6405719
- City Names
  http://m.sodocs.net/doc/dc5af962b90d6c85ec3ac69a-27.html
- City -landform:
  http://data.stats.gov.cn/search.htm?s=%E5%9C%B0%E5%BD%A2
- City-automobile ownership
  http://data.stats.gov.cn/easyquery.htm?cn=C01&zb=A0G0I&sj=2014
- City-factory amount
  http://data.stats.gov.cn/search.htm?s=%E5%B7%A5%E5%8E%82%E6%95%B0%E9%87%8F

For these data, we downloaded and use tools like Excel and Sublime to manually format them, then import to our database corresponding tables

- **Run our code**

We use the same LAMP architecture and settings as the in-class HW3 example. To run our project, simply clone our repo https://github.com/xliu59/air_pollution.git to your local webserver root and open <root>/air_pollution/index.html

- **Areas of specialization**

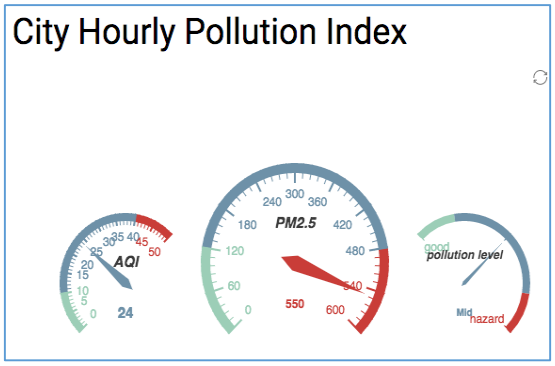
1. REAL-TIME DATA
In our project we use real-time data instead of solely static dataset to make our graph timeliness, always updated. The python scripts we created is responsible for making http request, retrieving data, parsing and interacting with our MySQL database.

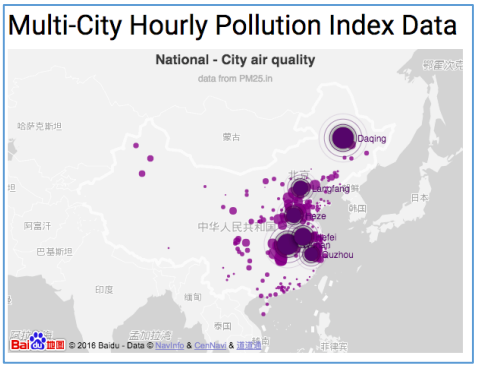
2. DATA VISUALIZATION
We use ECharts API to create fine, animated, interactive charts, graphs and diagrams to clearly demonstrate the trend of data or comparison of multiple attributes.

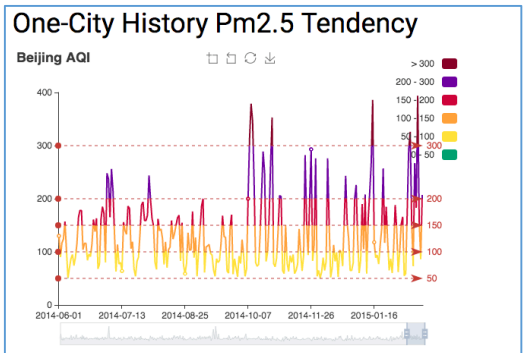
- **Limitations**

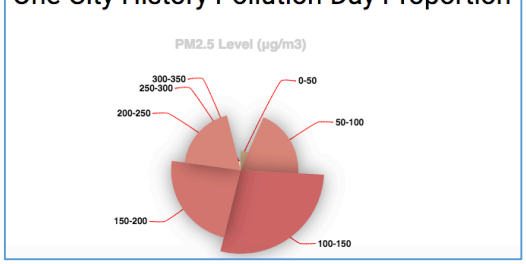
  Even though we have finished real-time data retrieval, stored procedures, front-end data visualization, and some back-end logics to handle request, we met some serious Json format issues in this project and can only show data that is pre-selected. The sidebar for selecting dates and cities have not been implemented due to lack of Javacript experience. If had more time on this project, we will put more into the data communication between front-end and back-end of a web service and we are confident this web project will be useful, accurate and stable.

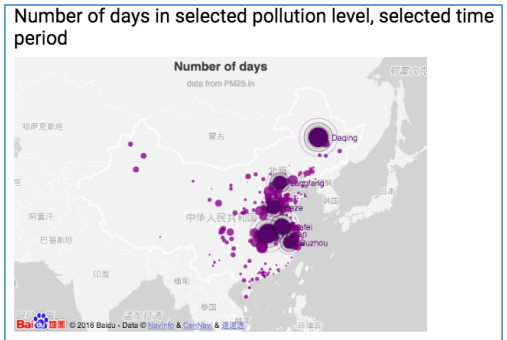
- **Project Output**

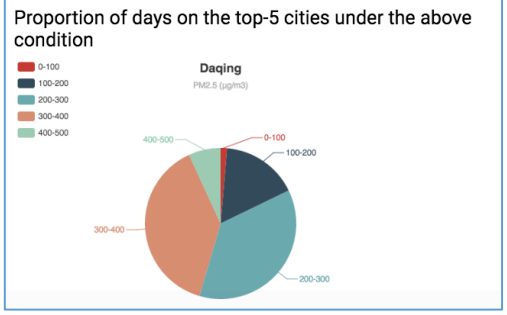


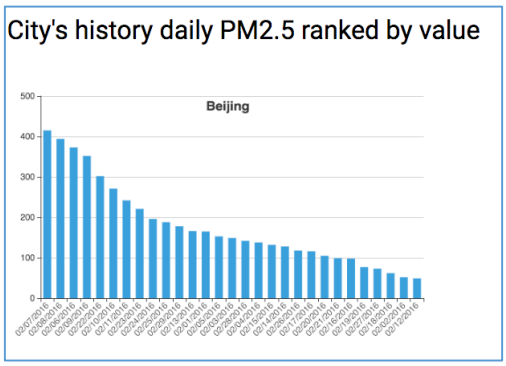


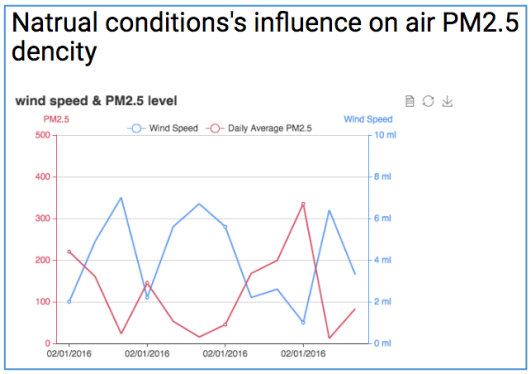


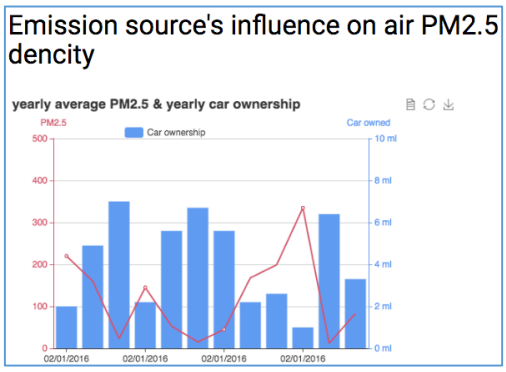


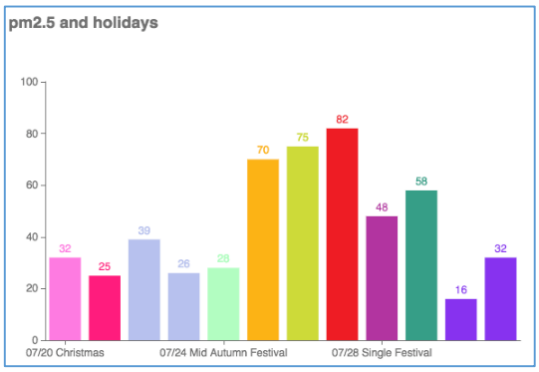












- **Database specification**

```
#City
        DROP TABLE City;
        CREATE TABLE City (
            City_Code INT NOT NULL,
            City_name VARCHAR(200) NOT NULL,
            City_name_en VARCHAR(200) NOT NULL,
            Province VARCHAR(200) NOT NULL,
            Longitude FLOAT,
            Latitude FLOAT,
            PRIMARY KEY (City_Code)
        );

        #Pollution
        DROP TABLE Pollution;
        CREATE TABLE Pollution (
            City_name VARCHAR(200) NOT NULL,
            Time VARCHAR(200) NOT NULL,
            Quality VARCHAR(20),
            Pm2_5 FLOAT,
            Pm2_5_24 FLOAT,
```

```
    AQI FLOAT,
    Primary_pollutant VARCHAR(20),
    PRIMARY KEY (City_name)
);

#Pm2_5_level
DROP TABLE Pm2_5_level;
CREATE TABLE Pm2_5_level (
    Level VARCHAR(20) NOT NULL,
    StartNum FLOAT NOT NULL,
    PRIMARY KEY (Level)
);

#Wind
DROP TABLE Wind;
CREATE TABLE Wind (
    City_name VARCHAR(200) NOT NULL,
    Time INT NOT NULL,
    Speed FLOAT NOT NULL,
    PRIMARY KEY (City_name)
);

#Humidity
DROP TABLE Humidity;
CREATE TABLE Humidity (
    City_name VARCHAR(200) NOT NULL,
    Time INT NOT NULL,
    Humidity FLOAT NOT NULL,
    PRIMARY KEY (City_name)
);

#Precipitation
DROP TABLE Precipitation;
CREATE TABLE Precipitation (
    City_name VARCHAR(200) NOT NULL,
    Time INT NOT NULL,
    Precipitation FLOAT NOT NULL,
    PRIMARY KEY (City_name)
);

#Air_Pressure
DROP TABLE Air_Pressure;
CREATE TABLE Air_Pressure (
    City_name VARCHAR(200) NOT NULL,
    Time INT NOT NULL,
```

```
    Air_Pressure FLOAT NOT NULL,
    PRIMARY KEY (City_name)
);

#Temperature
DROP TABLE Temperature;
CREATE TABLE Temperature (
    City_name VARCHAR(200) NOT NULL,
    Time INT NOT NULL,
    Temperature FLOAT NOT NULL,
    PRIMARY KEY (City_name)
);

#City_topography
DROP TABLE City_topography;
CREATE TABLE City_topography (
    City_name VARCHAR(200) NOT NULL,
    Landform VARCHAR(100) NOT NULL,
    PRIMARY KEY (City_name)
);

#NumberOfCarOwnership
DROP TABLE NumberOfCarOwnership;
CREATE TABLE NumberOfCarOwnership (
    City_name VARCHAR(200) NOT NULL,
    Year VARCHAR(4) NOT NULL,
    CarNumber INT NOT NULL,
    PRIMARY KEY (City_name)
);

#NumberOfFactories
DROP TABLE NumberOfFactories;
CREATE TABLE NumberOfFactories (
    City_name VARCHAR(200) NOT NULL,
    Year VARCHAR(4) NOT NULL,
    Factories INT NOT NULL,
    PRIMARY KEY (City_name)
);

#Festival
DROP TABLE Festival;
CREATE TABLE Festival (
    Festival_name VARCHAR(200) NOT NULL,
    Date DATE NOT NULL,
    PRIMARY KEY (Festival_name)
```

```
);
```

- **SQL COE**
  (see attachment)

- **Complete Archive**
  (see attachment)