

## **Project 2 - Airline Database**

ISTM 622 - 601

Group 3

Brian Newman, Bhavishya Tyagi, Chuchu Yao, Pooja Vaswani, Anjali Shukla

## **Table of Contents**

<b>Installing MongoDB</b>	<b>3</b>
<b>Configure Mongo DB</b>	<b>3</b>
<b>Application Code</b>	<b>4</b>
<b>Insert and Update Code</b>	<b>6</b>
<b>HTML Code</b>	<b>7</b>
<b>Website Example</b>	<b>10</b>

# Installing MongoDB

- <https://docs.mongodb.com/v3.2/tutorial/install-mongodb-on-windows/>
- Installed on local computer

## Configure MongoDB

- Create DB
    - use airport
  - Import Data into Mongo
    - Open CMD and change directory to C:\Program Files\MongoDB\Server\4.2\bin
    - Run this to import
      - `mongoimport -d airport -c flight --type csv --file "D:\Google Drive\_Fall 2019\ISTM 622\Group1-3\flight.csv" --headerline`
- This shows the data loading into the database.*

```
C:\Windows\System32\cmd.exe - mongoimport -d airport -c flight --type csv --file "D:\Google Drive\_Fall 2019\ISTM 622\Group1-3\flight.csv" --head...
Microsoft Windows [Version 10.0.18362.449]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Program Files\MongoDB\Server\4.2\bin>mongoimport -d airport -c flight --type csv --file "D:\Google Drive\_Fall 2019\ISTM 622\Group1-3\flight.csv" --headerline
2019-11-09T11:39:04.860-0600 error validating settings: only one positional argument is allowed

C:\Program Files\MongoDB\Server\4.2\bin>mongoimport -d airport -c flight --type csv --file "D:\Google Drive\_Fall 2019\ISTM 622\Group1-3\flight.csv" --headerline
2019-11-09T11:39:26.543-0600 connected to: mongodb://localhost/
2019-11-09T11:39:29.545-0600 [.....] airport.flight 32.3MB/1.33GB (2.4%)
2019-11-09T11:39:32.545-0600 [#.....] airport.flight 63.6MB/1.33GB (4.7%)
2019-11-09T11:39:35.545-0600 [#####] airport.flight 96.1MB/1.33GB (7.1%)
2019-11-09T11:39:38.545-0600 [###.....] airport.flight 128MB/1.33GB (9.4%)
2019-11-09T11:39:41.545-0600 [##.....] airport.flight 160MB/1.33GB (11.8%)
2019-11-09T11:39:44.545-0600 [####.....] airport.flight 192MB/1.33GB (14.1%)
2019-11-09T11:39:47.545-0600 [#####] airport.flight 224MB/1.33GB (16.5%)
2019-11-09T11:39:50.544-0600 [#####] airport.flight 256MB/1.33GB (18.8%)
2019-11-09T11:39:53.545-0600 [#####] airport.flight 288MB/1.33GB (21.2%)
2019-11-09T11:39:56.545-0600 [#####] airport.flight 321MB/1.33GB (23.5%)
```

```
C:\Windows\System32\cmd.exe
2019-11-09T11:40:17.543-0600 [#####] airport.flight 546MB/1.33GB (40.1%)
2019-11-09T11:40:20.543-0600 [#####] airport.flight 578MB/1.33GB (42.5%)
2019-11-09T11:40:23.543-0600 [#####] airport.flight 610MB/1.33GB (44.8%)
2019-11-09T11:40:26.543-0600 [#####] airport.flight 642MB/1.33GB (47.2%)
2019-11-09T11:40:29.543-0600 [#####] airport.flight 675MB/1.33GB (49.6%)
2019-11-09T11:40:32.543-0600 [#####] airport.flight 707MB/1.33GB (51.9%)
2019-11-09T11:40:35.543-0600 [#####] airport.flight 740MB/1.33GB (54.3%)
2019-11-09T11:40:38.543-0600 [#####] airport.flight 772MB/1.33GB (56.7%)
2019-11-09T11:40:41.543-0600 [#####] airport.flight 804MB/1.33GB (59.1%)
2019-11-09T11:40:44.543-0600 [#####] airport.flight 836MB/1.33GB (61.4%)
2019-11-09T11:40:47.543-0600 [#####] airport.flight 869MB/1.33GB (63.8%)
2019-11-09T11:40:50.543-0600 [#####] airport.flight 901MB/1.33GB (66.2%)
2019-11-09T11:40:53.543-0600 [#####] airport.flight 933MB/1.33GB (68.6%)
2019-11-09T11:40:56.543-0600 [#####] airport.flight 966MB/1.33GB (70.9%)
2019-11-09T11:40:59.543-0600 [#####] airport.flight 998MB/1.33GB (73.3%)
2019-11-09T11:41:02.543-0600 [#####] airport.flight 1.01GB/1.33GB (75.7%)
2019-11-09T11:41:05.543-0600 [#####] airport.flight 1.04GB/1.33GB (78.0%)
2019-11-09T11:41:08.543-0600 [#####] airport.flight 1.07GB/1.33GB (80.4%)
2019-11-09T11:41:11.543-0600 [#####] airport.flight 1.10GB/1.33GB (82.7%)
2019-11-09T11:41:14.543-0600 [#####] airport.flight 1.13GB/1.33GB (85.1%)
2019-11-09T11:41:17.543-0600 [#####] airport.flight 1.16GB/1.33GB (87.4%)
2019-11-09T11:41:20.543-0600 [#####] airport.flight 1.19GB/1.33GB (89.8%)
2019-11-09T11:41:23.543-0600 [#####] airport.flight 1.22GB/1.33GB (92.1%)
2019-11-09T11:41:26.543-0600 [#####] airport.flight 1.26GB/1.33GB (94.5%)
2019-11-09T11:41:29.543-0600 [#####] airport.flight 1.29GB/1.33GB (96.9%)
2019-11-09T11:41:32.543-0600 [#####] airport.flight 1.32GB/1.33GB (99.2%)
2019-11-09T11:41:33.532-0600 [#####] airport.flight 1.33GB/1.33GB (100.0%)
2019-11-09T11:41:33.532-0600 12751637 document(s) imported successfully. 0 document(s) failed to import.

C:\Program Files\MongoDB\Server\4.2\bin>
```

# Application Code

```
#Import
from flask import Flask, render_template, request
from pymongo import MongoClient

#Create Flask Env
app = Flask(__name__)

#####
myclient = MongoClient()
mydb = myclient["airport"]
mycol = mydb["flight"]

query1 = [{
    "$group": {
        "_id": "null",
        "avg_CARRIER_DELAY": { "$avg": "$CARRIER_DELAY" },
        "avg_WEATHER_DELAY": { "$avg": "$WEATHER_DELAY" },
        "avg_NAS_DELAY": { "$avg": "$NAS_DELAY" },
        "avg_SECURITY_DELAY": { "$avg": "$SECURITY_DELAY" },
        "avg_LATE_AIRCRAFT_DELAY": { "$avg": "$LATE_AIRCRAFT_DELAY" } } } ]

query1list = list(mycol.aggregate(query1))
result = query1list[0]
query1keys = list(result.keys())
query1values = list(result.values())

query1avg = ((query1values[1] + query1values[2] + query1values[3] + query1values[4] + query1values[5]) / 5)

query2 = mycol.find( { "DISTANCE": { "$gt": 1000 }, "CANCELLED": 1 } ).count()
print(query2)

query3 = mycol.find( { "DISTANCE": { "$lte": 1000 }, "CANCELLED": 1 } ).count()
print(query3)

query5 = [{
    "$group": {
        "_id": "null",
        "sum_CARRIER_DELAY": { "$sum": "$CARRIER_DELAY" },
        "sum_WEATHER_DELAY": { "$sum": "$WEATHER_DELAY" },
        "sum_NAS_DELAY": { "$sum": "$NAS_DELAY" },
        "sum_SECURITY_DELAY": { "$sum": "$SECURITY_DELAY" },
        "sum_LATE_AIRCRAFT_DELAY": { "$sum": "$LATE_AIRCRAFT_DELAY" } } } ]

query5list = list(mycol.aggregate(query5))
result = query5list[0]
query5keys = list(result.keys())
query5values = list(result.values())
```

```

query6 = mycol.find( { "CARRIER_DELAY": { "$gt" : 0 } }).count()
print(query6
)
query7 = mycol.find( { "WEATHER_DELAY" : { "$gt" : 0 } } ).count()
print(query7)
query8 = mycol.find( { "NAS_DELAY" : { "$gt" : 0 } } ).count()
print(query8)
query9 = mycol.find( { "SECURITY_DELAY" : { "$gt" : 0 } } ).count()
print(query9)
query10= mycol.find( { "LATE_AIRCRAFT_DELAY" : { "$gt" : 0 } } ).count()
print(query10)

```

```
#####
```

```

# Run the App
@app.route('/')
def dashboard():
    # Display website
    # Send values over
    return render_template('dashboard.html', query1values = query1values, query1avg = query1avg, query2 =
query2, query3 = query3, query5values = query5values, query6 = query6, query7 = query7
        , query8 = query8, query9 = query9, query10 = query10)

if __name__ == '__main__':
    app.run(debug=False)

```

# Insert and Update Code

```
from pymongo import MongoClient
```

```
myclient = MongoClient()
mydb = myclient["airport"]
mycol = mydb["flight"]
```

```
#####
#                                     INSERT                                #
#####
flight_rec1 = {
    "FL_DATE": "2019-11-15",
    "OP_UNIQUE_CARRIER": "AA",
    "OP_CARRIER_FL_NUM": 6,
    "ORIGIN_AIRPORT_ID": 13830,
    "DEST_AIRPORT_ID": 11298,
    "CANCELLED": 1,
    "DIVERTED": 0,
    "ACTUAL_ELAPSED_TIME": 421,
    "AIR_TIME": 402,
    "FLIGHTS": 1,
    "DISTANCE": 5400,
    "CARRIER_DELAY": 0,
    "WEATHER_DELAY": 0,
    "NAS_DELAY": 0,
    "SECURITY_DELAY": 0,
    "LATE_AIRCRAFT_DELAY": 0
}
# Insert Data
rec_id1 = mycol.insert_one(flight_rec1)

print("Data inserted with record ids", rec_id1)

query2 = mycol.find({"DISTANCE": {"$gt": 1000}, "CANCELLED": 1}).count()
print(query2)

#####
#                                     UPDATE                                #
#####
rec_id1 = mycol.update_one({"FL_DATE": "2019-11-15"}, {"$set": {"CANCELLED": 0}}, upsert=False)

print("Data inserted with record ids", rec_id1)

query2 = mycol.find({"DISTANCE": {"$gt": 1000}, "CANCELLED": 1}).count()
print(query2)
```

# HTML Code

```
<!DOCTYPE html>
<html>
<head>
  <title>AirLine</title>
</head>
<!-- Adding some style to table -->
<style type="text/css">
  th, tr {
    color: blue;
  }
  tr:nth-of-type(2n) {
    border: 1px solid black;
    background-color: rgba(150, 150, 150, 0.5);
  }
  td {
    padding: 8px 8px;
    border: 1px solid black;
  }
  body {
    background-color: powderblue;
    text-align: center;
    color: black;
    font-family: Arial, Helvetica, sans-serif;
  }
</style>
<body>
<h1>Airline Stats </h1>
<h2>Data is from January to June from 2017,2018,2019! </h2>
<p>This is a dashboard!</p>
<table style="margin-left: 20px;">
  <p></p>
<!-- Table headers for Query1AVG -->
  <th>
    <tr style="color: Black; font-weight: bold;">
      <td>Average flight delays</td>
    </tr>
    <tr style="color: Black;">
      <td>{{ query1avg }} Minutes</td>
    </tr>
  </th>
</table>
<p></p>
<table style="margin-left: 20px;">
<p></p>
```

```

<!-- Table headers for Query1 -->
<th>
  <tr style="color: Black; font-weight: bold;">
    <td colspan="5">Showing flight average delay in minutes per category</td>
  </tr>
  <tr style="color: Black; font-weight: bold;">
    <td>Carrier Delay</td>
    <td>Weather Delay</td>
    <td>NAS Delay</td>
    <td>Security Delay</td>
    <td>Late Craft Delay</td>
  </tr>

</th>

<!-- table rows -->
<tr>
  <td>{{ query1 values[1] }}</td>
  <td>{{ query1 values[2] }}</td>
  <td>{{ query1 values[3] }}</td>
  <td>{{ query1 values[4] }}</td>
  <td>{{ query1 values[5] }}</td>

</tr>
</table>
<p></p>
<table style="margin-left: 20px;">
<!-- Table headers for Query2 -->
<th>
  <tr style="color: Black; font-weight: bold; ">
    <td>Showing number of flights whose distance is greater than 1000 that were cancelled.</td>
  </tr>
  <tr style="color: Black; ">
    <td>{{ query2 }}</td>
  </tr>

</th>

</table>

<p></p>
<table style="margin-left: 20px;">
<!-- Table headers for Query3 -->
<th>
  <tr style="color: Black; font-weight: bold; ">
    <td>Showing number of flights whose distance is less than 1000 that were cancelled.</td>
  </tr>
  <tr style="color: Black; ">
    <td>{{ query3 }}</td>
  </tr>

</th>

</table>

```



```

<p></p>
<table style="margin-left: 20px;">
<!-- Table headers for Queries -->
<th>
  <tr style="color: Black; font-weight: bold;">
    <td colspan="5">Showing flight delayed by categories in numbers</td>
  </tr>
  <tr style="color: Black; font-weight: bold;">
    <td>Carrier Delay</td>
    <td>Weather Delay</td>
    <td>NAS Delay</td>
    <td>Security Delay</td>
    <td>Late Craft Delay</td>
  </tr>

</th>

<tr>
  <td>{{query6}}</td>
  <td>{{query7}}</td>
  <td>{{query8}}</td>
  <td>{{query9}}</td>
  <td>{{query10}}</td>
</tr>
</table>

<table style="margin-left: 20px;">
<p></p>
<!-- Table headers for Query5 -->
<th>
  <tr style="color: Black; font-weight: bold;">
    <td colspan="5">Showing total flight delay in minutes per category</td>
  </tr>
  <tr style="color: Black; font-weight: bold;">
    <td>Carrier Delay</td>
    <td>Weather Delay</td>
    <td>NAS Delay</td>
    <td>Security Delay</td>
    <td>Late Craft Delay</td>
  </tr>
</th>
<!-- table rows -->
<tr>
  <td>{{query5values[1]}}</td>
  <td>{{query5values[2]}}</td>
  <td>{{query5values[3]}}</td>
  <td>{{query5values[4]}}</td>
  <td>{{query5values[5]}}</td>
</tr>
</table>
<p></p>

</body>
</html>

```

# Website Example

Airline Stats				
Data is from January to June from 2017,2018,2019!				
This is a dashboard!				
Average flight delays				
2.433128998415416 Minutes				
Showing flight average delay in minutes per category				
Carrier Delay	Weather Delay	NAS Delay	Security Delay	Late Craft Delay
3.7451870014258586	0.6346137830081745	2.9476742316698106	0.016208222739490075	4.821961753233747
Showing number of flights whose distance is greater than 1000 that were cancelled.				
46207				
Showing number of flights whose distance is less than 1000 that were cancelled.				
181952				
Showing flight delayed by categories in numbers				
Carrier Delay	Weather Delay	NAS Delay	Security Delay	Late Craft Delay
1183150	138324	1289841	7514	1240743
Showing total flight delay in minutes per category				
Carrier Delay	Weather Delay	NAS Delay	Security Delay	Late Craft Delay
47757179.0	8092350.0	37587604.0	206681.0	61487795.0