

Part 1

Part 1 of this assignment includes extracting several fields of information from the MongoDB data provided. Aggregations requested are rather straightforward. They were created using MongoDB's built in LLM that is designed to generate aggregations in Mongo, as the assignment suggested. Each query is included in the [HW5.JS](#) with a comment at the front that designates what the query is returning. Only one query is missing from those that were suggested in the assignment. That would be the one about air accidents in certain hours, since it led to an error by the LLM and is used as an example of an error below.

Part 2

Data Type Confusion: For this part, I managed to confuse it for one of the previous queries. My prompt was: “Are accidents significantly more common during daylight hours (6:00 to 18:00) or nighttime hours (18:00-6:00)?” The LLM was unable to execute the proper commands resulting in a typing error. There is a picture attached, but the aggregation is not in the HW5 file. We see that the LLM got tripped up failing to parse through a number, which was not an expected outcome at all. Rather, this should have resulted in a successful query.

The screenshot shows the MongoDB aggregation pipeline interface. At the top, there is a search bar with the query: "Are accidents significantly more common during daylight hours (6:00 to 18:00) or nighttime hours (18:00-6:00)?". Below the search bar are tabs for \$addFields, \$bucket, \$match, \$project, and \$group. The \$addFields tab is selected. The pipeline consists of two stages:

- Stage 1 (\$addFields):** Contains the following aggregation stage:

```
1 ▶ {  
2   ▶ hour: {  
3     ▶ $toInt: {  
4       ▶ $substr: ["$Time", 0, 2]  
5     }  
6   }  
7 }
```

Output preview after \$addFields stage (Sample of 0 documents): No Preview Documents

A red warning box indicates an executor error: "Executor error during aggregate command on namespace: test.planes :: caused by :: Failed to parse number " in \$convert with no onError value: No digits".
- Stage 2 (\$bucket):** Contains the following aggregation stage:

```
1 ▶ {  
2   ▶ groupBy: "$hour",  
3   ▶ boundaries: [0, 6, 18, 24],  
4   ▶ default: "Unknown",  
5   ▶ output: {  
6     ▶ count: {
```

Output preview after \$bucket stage (Sample of 0 documents)

Ambiguous Context: For this part, I tried to get the AI to hallucinate something that wasn't real. My prompt was: "Write a query that returns the plane with the longest wingspan that crashed." Obviously, wingspan is not one of the things recorded in this data, but the LLM hallucinated it anyways. The resulting query is noted in the [HW5.JS](#) file. The expected outcome was that it would hallucinate, and that was exactly what happened since LLMs usually have a hallucination problem.

Complex Aggregation Structure: This query was a little tricky to get to work. Eventually I was able to trip it up with the following query: "Write an aggregation that groups documents by their Route, while filtering out any records with short Location fields inside the same grouping stage." It seems like it was a little easier after I made the prompt a bit longer. The query is in the [HW5.JS](#) file. There is a picture below. The

The screenshot shows a MongoDB aggregation query editor interface. At the top, there are buttons for 'SAVE' and 'CREATE NEW', and a 'EXPORT TO LANGUAGE' button. The main area contains the following code:

```
3   {
4     $group: {
5       _id: "$Route",
6       documents: {
7         $push: {
8           $cond: [
9             { $gt: [ { $strLenCP: "$Location" }, 3 ] },
10            "$$ROOT",
11            null
12          ]
13        }
14      }
15    }
16  }
```

Below the code, a red box highlights an error message:

**⚠ Executor error during aggregate command on namespace:
test.planes :: caused by :: \$strLenCP requires a string argument**