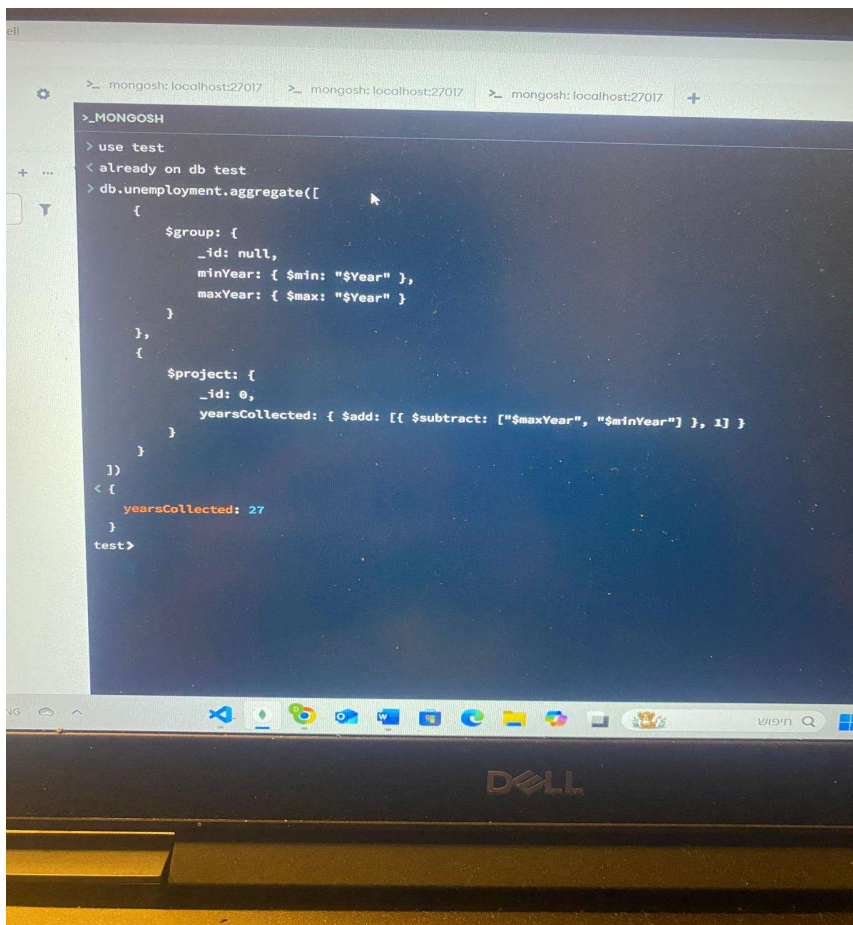


David Schwartzman

22/11/2024

Database Management HW5

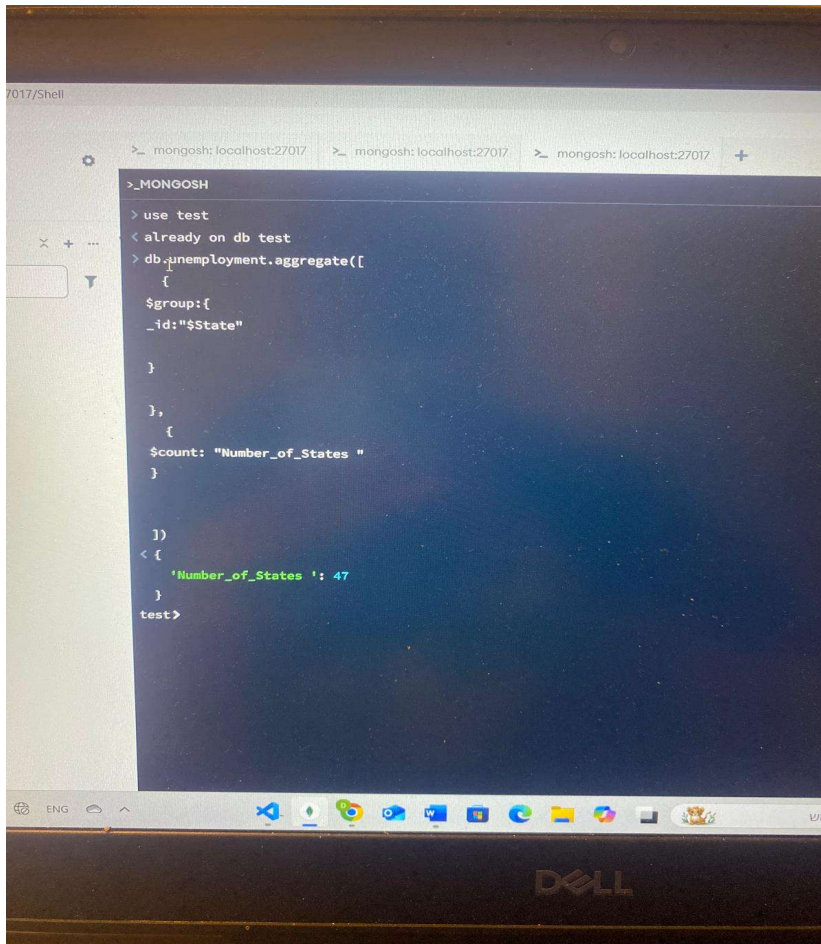
1) Over how many years was the unemployment data collected?



```
> use test
< already on db test
> db.unemployment.aggregate([
  {
    $group: {
      _id: null,
      minYear: { $min: "$Year" },
      maxYear: { $max: "$Year" }
    },
  },
  {
    $project: {
      _id: 0,
      yearsCollected: { $add: [{ $subtract: ["$maxYear", "$minYear"] }, 1] }
    }
  }
])
< {
  yearsCollected: 27
}
test>
```

This query calculates the range of years in the dataset by identifying the minimum and maximum years. It then calculates the total number of years between the minimum and maximum year, inclusive. The result shows how many years was the unemployment data collected.

2) How many states were reported on in this dataset?

A screenshot of a MongoDB Shell window. The window has a title bar with three tabs, each labeled 'mongosh: localhost:27017'. The main area is a dark blue terminal with white text. The text shows the following commands and output:

```
> use test
> <already on db test
> db.unemployment.aggregate([
  {
    $group:{
      _id:"$State"
    }
  },
  {
    $count: "Number_of_States "
  }
])
< {
  'Number_of_States ': 47
}
test>
```

The bottom of the image shows a Windows taskbar with various icons and a Dell logo on the laptop bezel.

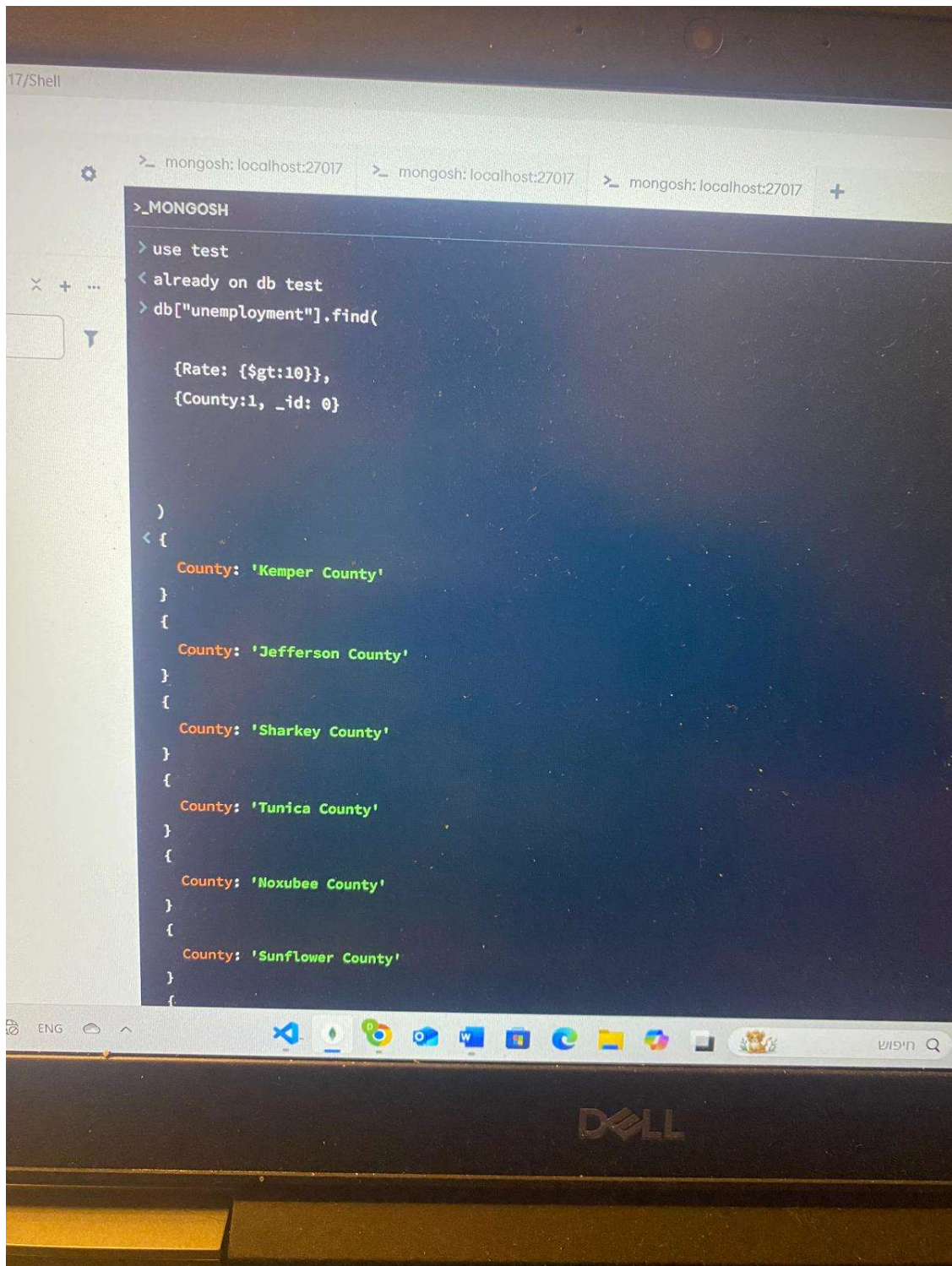
This query counts the number of unique states in the unemployment collection. It first groups the documents by the State field and then uses the \$count stage to count how many unique states are present in the dataset. The result shows the number of unique states there are in the dataset.

3) What does this query compute?

db.unemployment.find({Rate : {\$lt: 1.0}}).count()

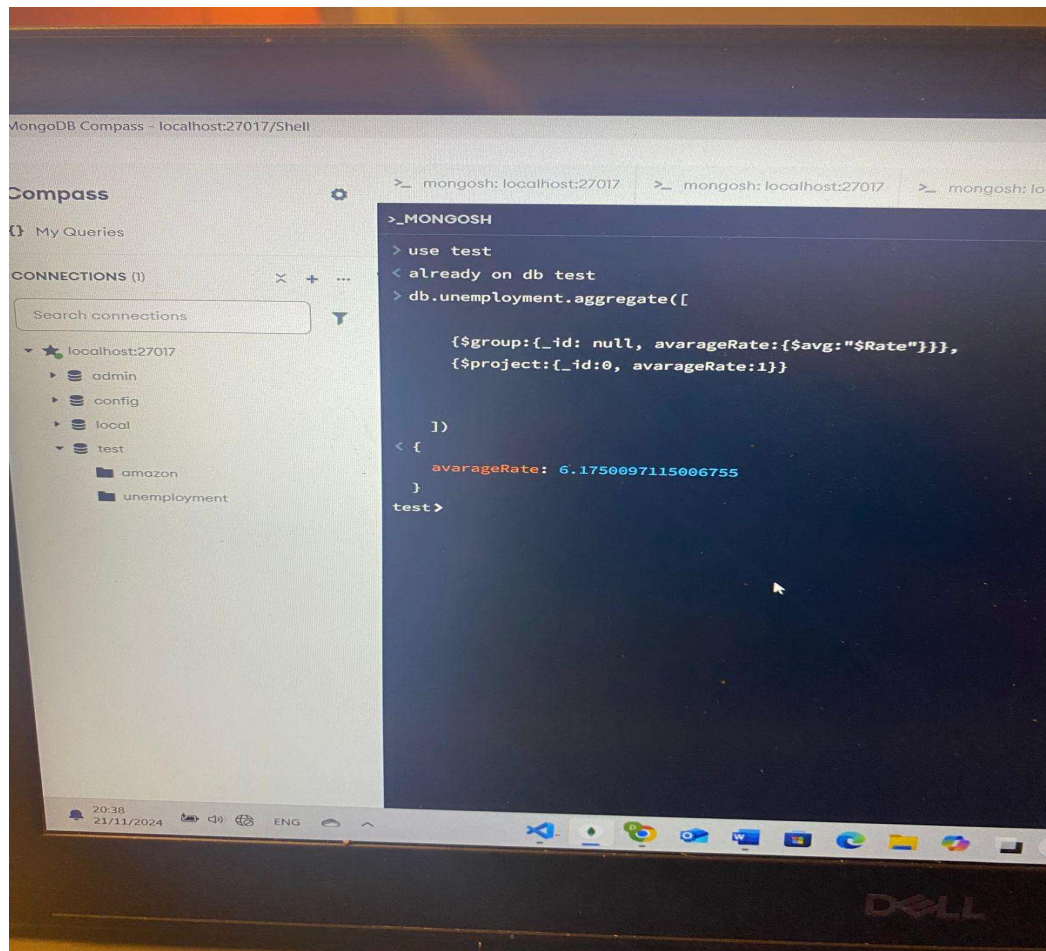
This query computes the number of documents in the unemployment collection where the Rate field is less than 1.0.

4) Find all counties with unemployment rate higher than 10%



This query retrieves all documents from the collection where the unemployment Rate is greater than 10%. The result will be a list of counties where the unemployment rate exceeds 10%.

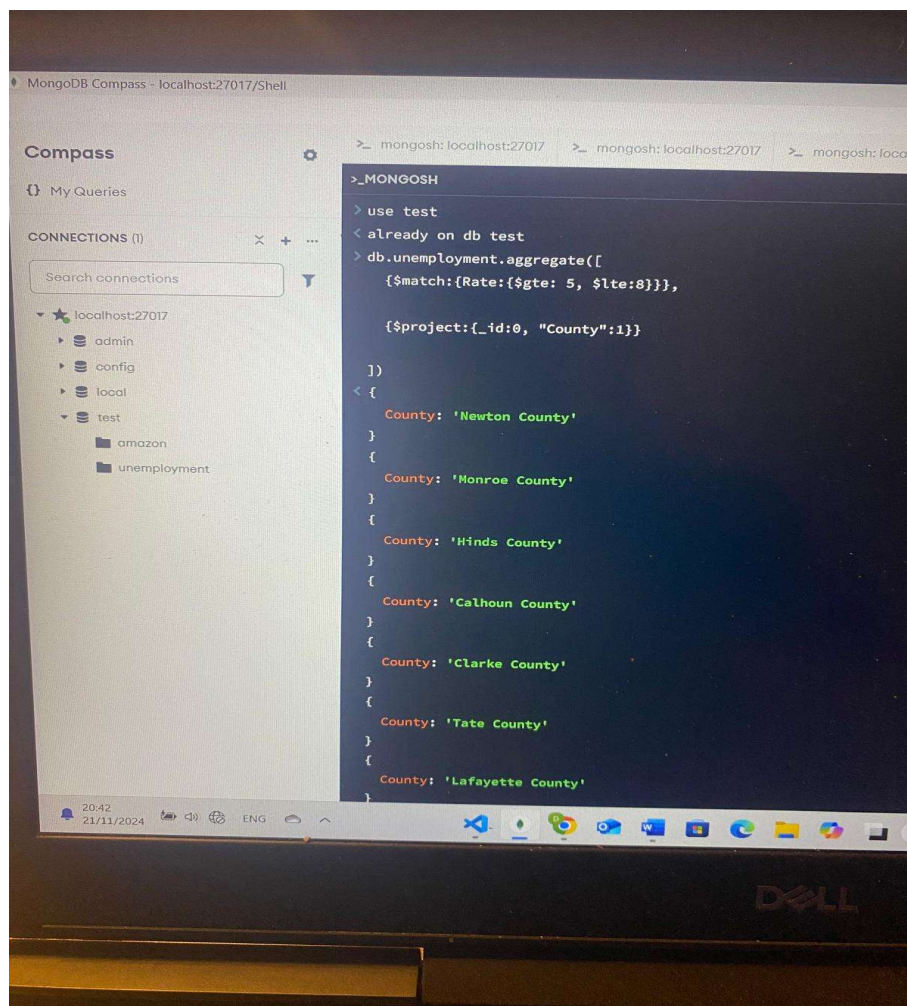
5) Calculate the average unemployment rate across all states.



This query calculates the average unemployment rate across all counties in the collection. It first uses the \$group stage to group all documents into a single group, and then calculates the average of the Rate field using the \$avg operator. The \$project stage is used to exclude the _id field from the output and only include the

computed averageRate. The result will display the overall average unemployment rate for all counties in the collection.

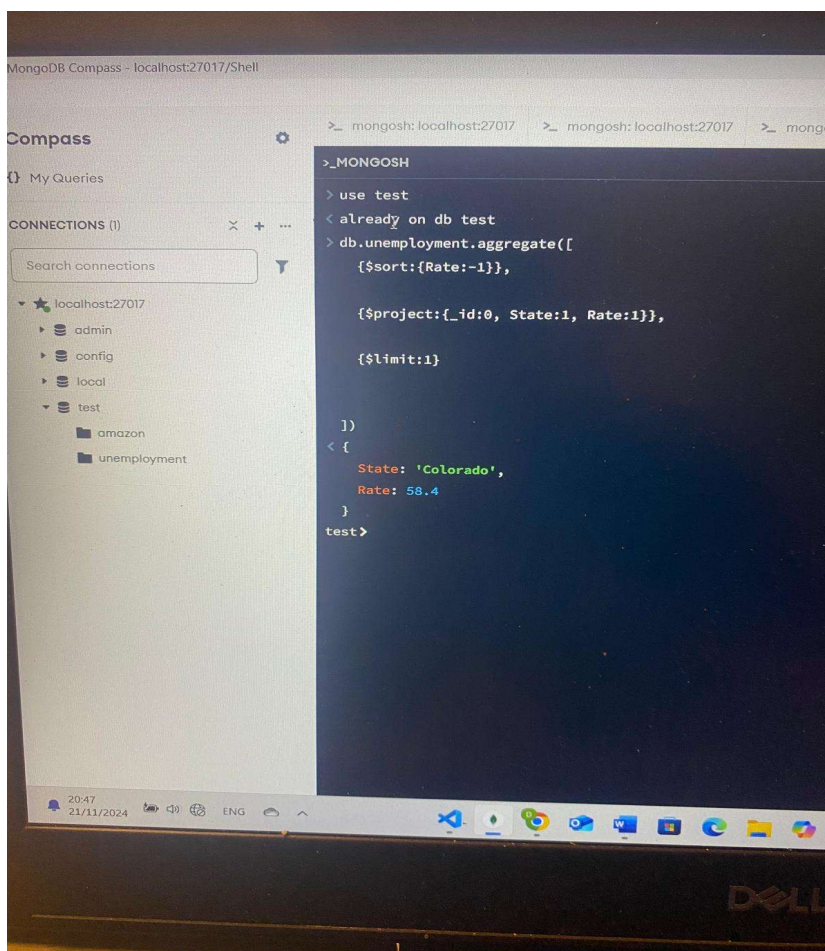
6) Find all counties with an unemployment rate between 5% and 8%.



- Assumption: Assumption: I assume I need to include 5 and 8, if I wouldn't include them it would be \$gt and \$ls.

The query retrieves all counties from the collection where the unemployment rate is between 5% and 8%, inclusive. The \$match stage filters the documents to only include those where the Rate is greater than or equal to 5 and less than or equal to 8. The \$project stage will display a list of counties with unemployment rates in the specified range.

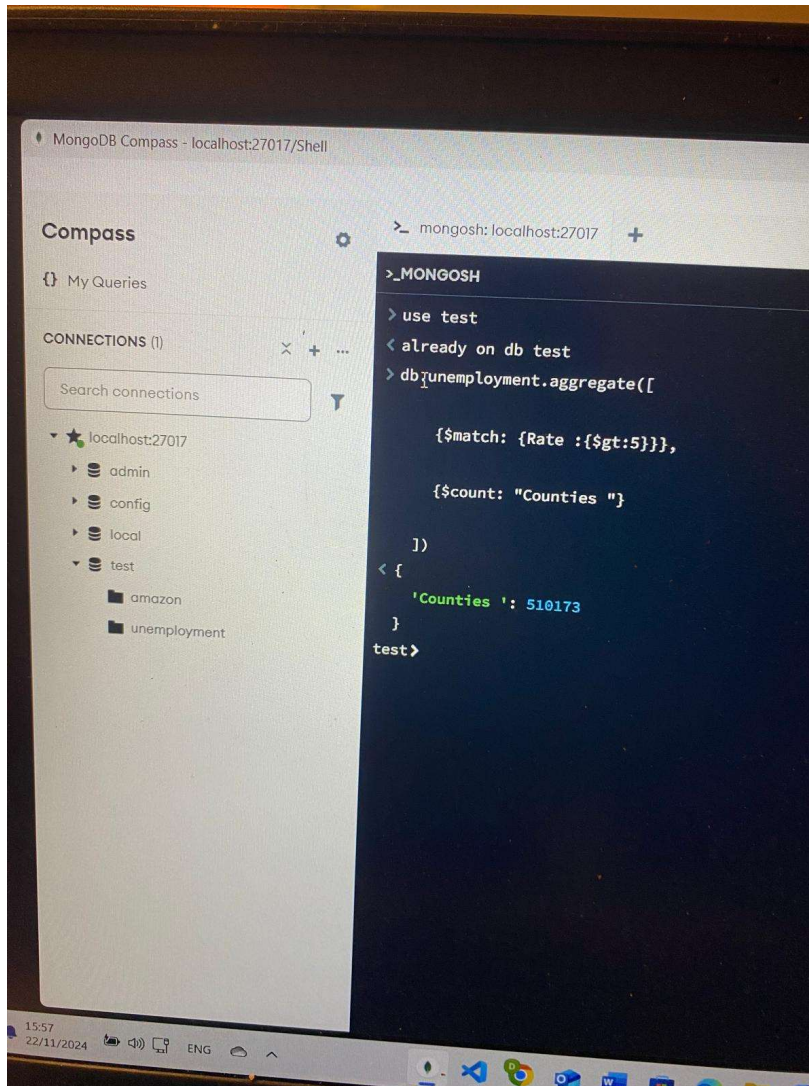
7) Find the state with the highest unemployment rate. Hint. Use { \$limit: 1 }



The query identifies the state with the highest unemployment rate from the collection. It begins by sorting the documents in descending order based on the Rate field, ensuring that the highest rates appear first. Then, it limits the output to a

single document, returning the state with the highest unemployment rate along with its corresponding rate.

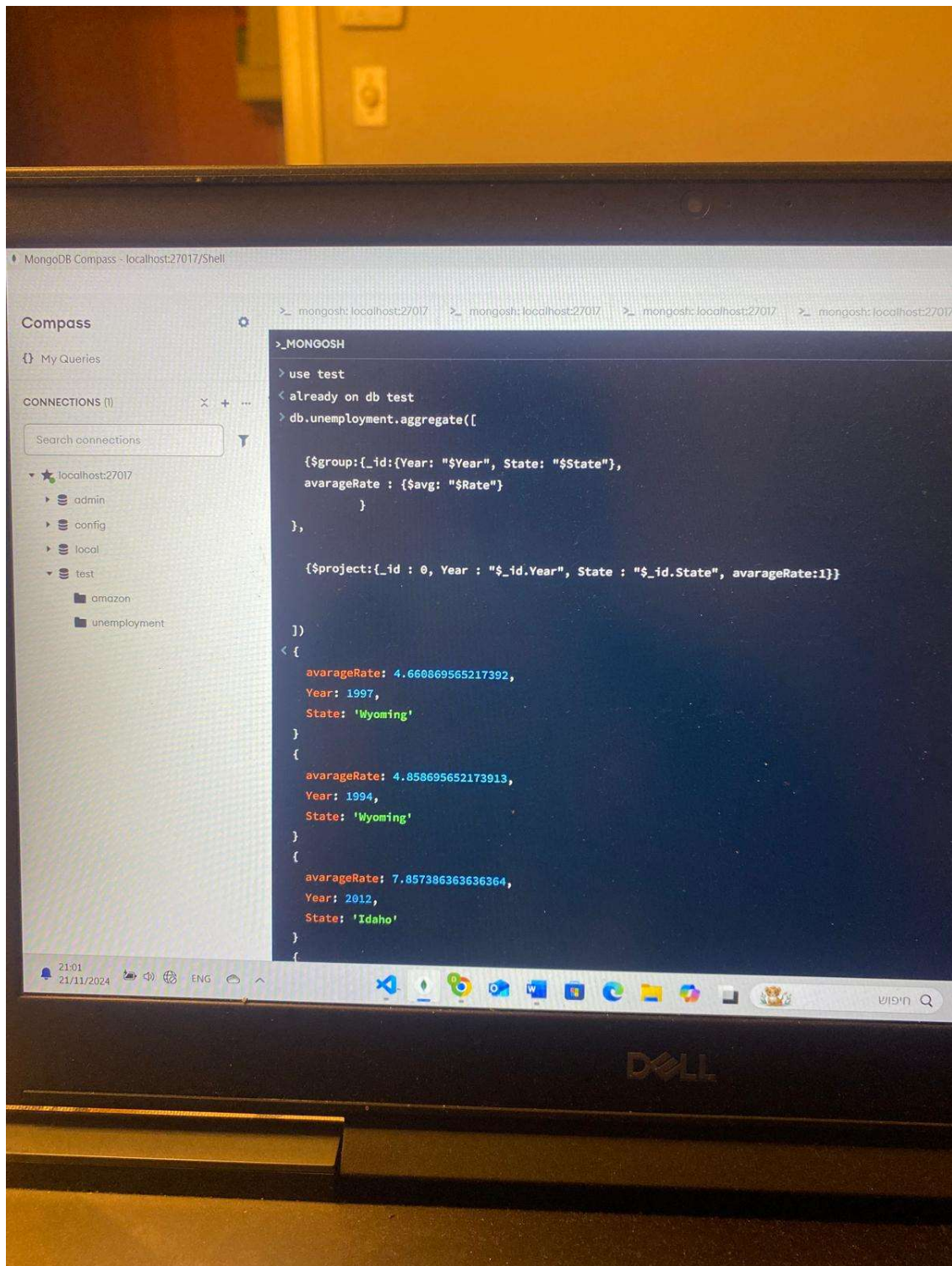
8) Count how many counties have an unemployment rate above 5%.



This query calculates the number of counties with an unemployment rate greater than 5 in the collection. It starts with a \$match stage to filter documents where the

Rate field is greater than 5. Then, it uses the \$count stage to count the number of matching documents and outputs the result with the field name "Counties".

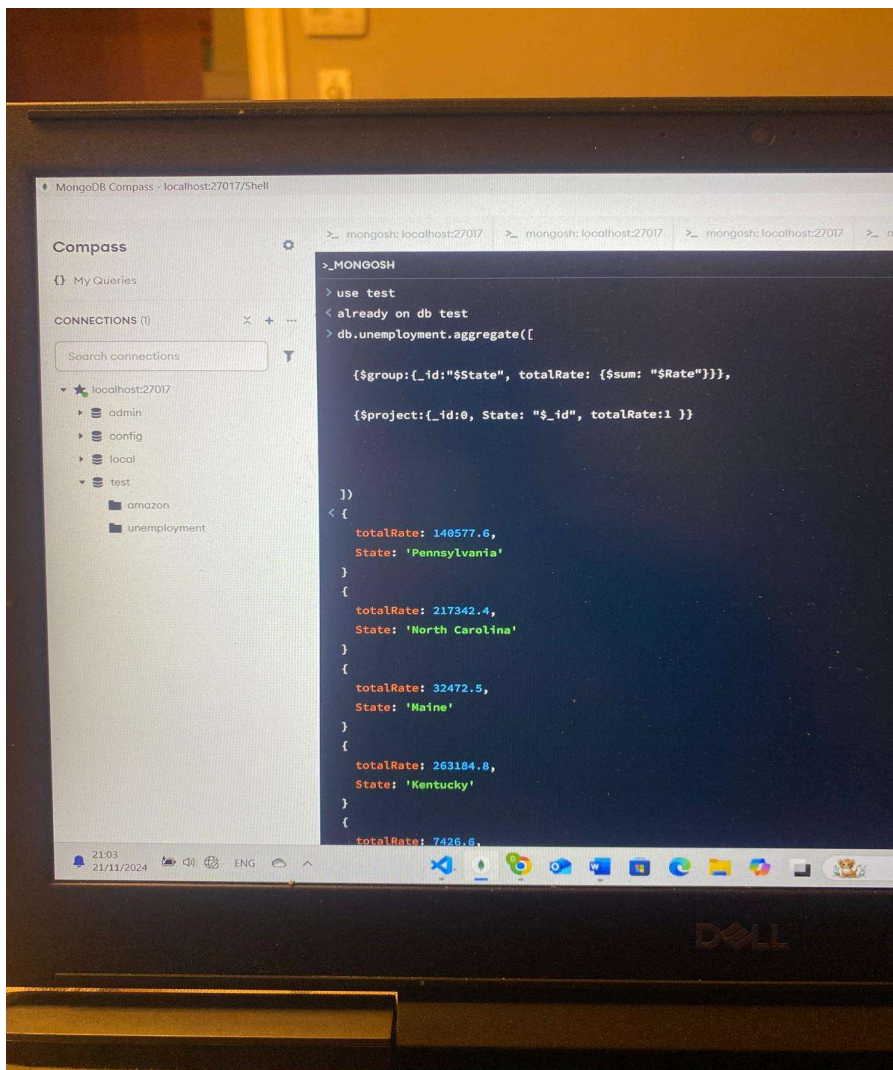
9) Calculate the average unemployment rate per state by year.



This MongoDB aggregation query calculates the average unemployment rate for each state and year in the unemployment collection. It groups documents by Year

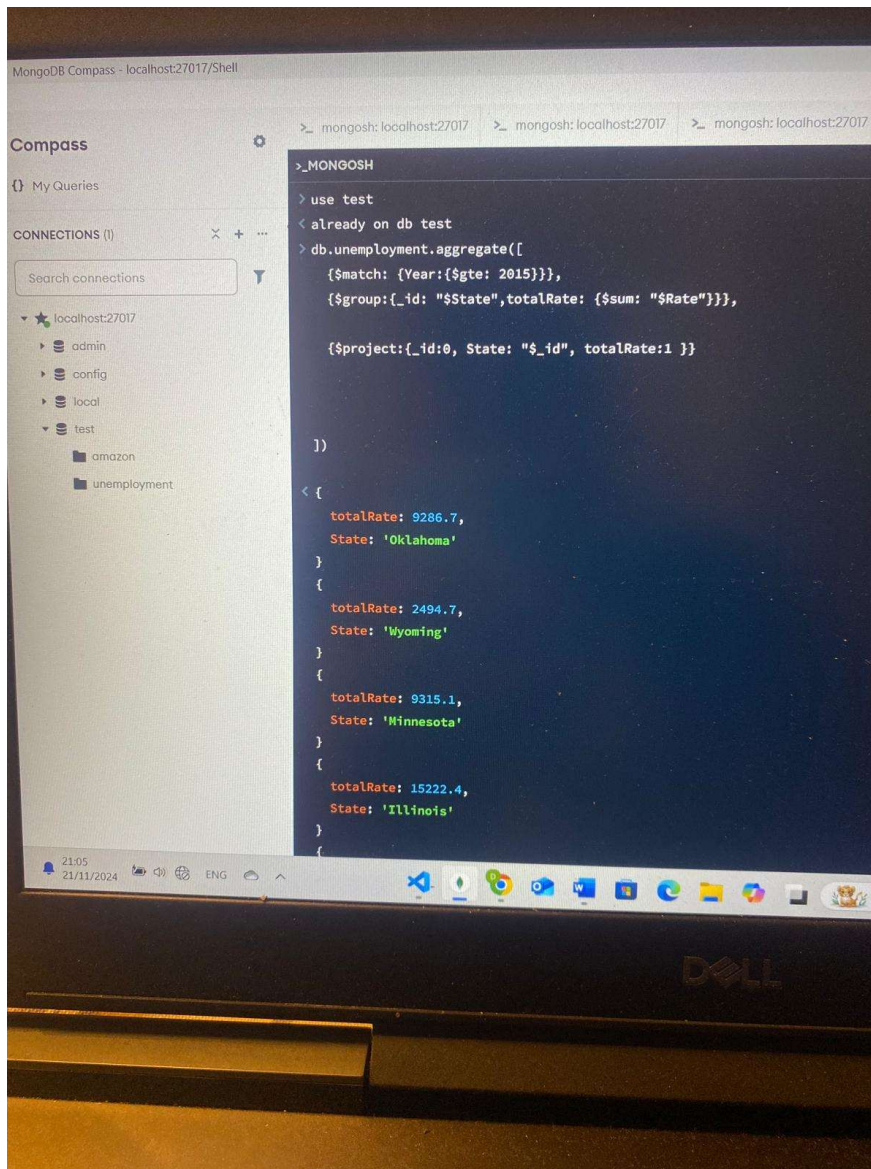
and State, computes the average rate using \$avg, and then reformats the output to include Year, State, and averageRate.

10) (Extra Credit) For each state, calculate the total unemployment rate across all counties (sum of all county rates).



This query calculates the total unemployment rate for each state in the collection. It groups documents by the State field and sums the Rate values for each state using the \$sum operator. The \$project stage then reformats the output to include the State name and its corresponding totalRate.

11) (Extra Credit) The same as Query 10 but for states with data from 2015 onward



This query calculates the total unemployment rate for each state from the year 2015 onward in the unemployment collection. It starts with a \$match stage to filter documents where the Year is greater than or equal to 2015. Next, it groups the filtered documents by State and computes the total Rate for each state using \$sum. Finally, the \$project stage reformats the output to include the State name and its totalRate, excluding the default _id field.

