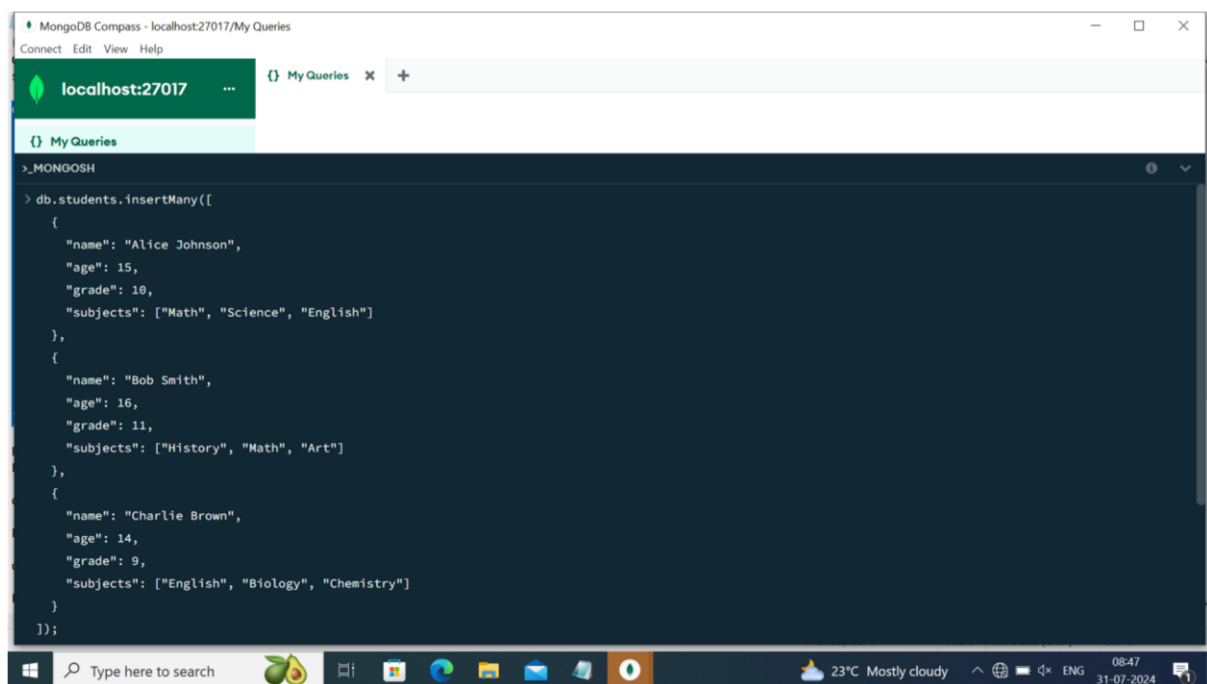


DATE:30/07/2024

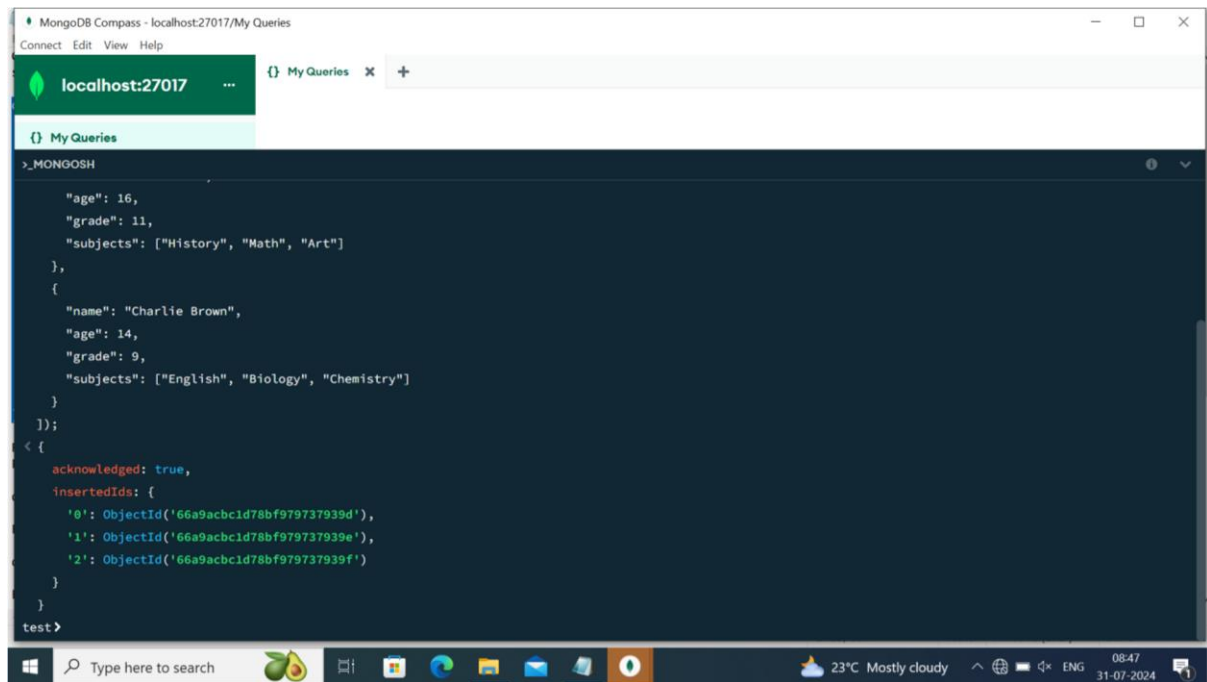
1. Create: Insert multiple documents into a collection called students with fields such as name, age, grade, and subjects.



The screenshot shows the MongoDB Compass interface. The top bar indicates the connection to 'localhost:27017'. Below the top bar, there is a tab labeled 'My Queries'. The main area displays a MongoDB query in the 'MongoShell' tab, which is used to insert three documents into the 'students' collection. The query is as follows:

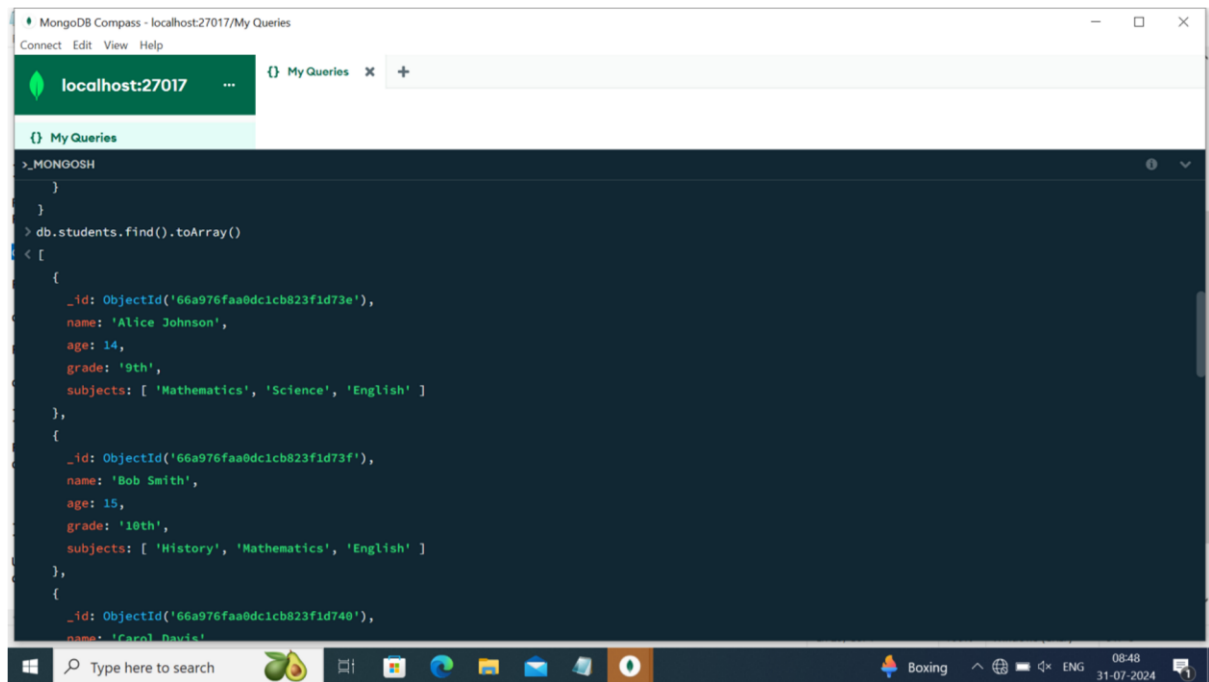
```
> db.students.insertMany([
  {
    "name": "Alice Johnson",
    "age": 15,
    "grade": 10,
    "subjects": ["Math", "Science", "English"]
  },
  {
    "name": "Bob Smith",
    "age": 16,
    "grade": 11,
    "subjects": ["History", "Math", "Art"]
  },
  {
    "name": "Charlie Brown",
    "age": 14,
    "grade": 9,
    "subjects": ["English", "Biology", "Chemistry"]
  }
]);
```

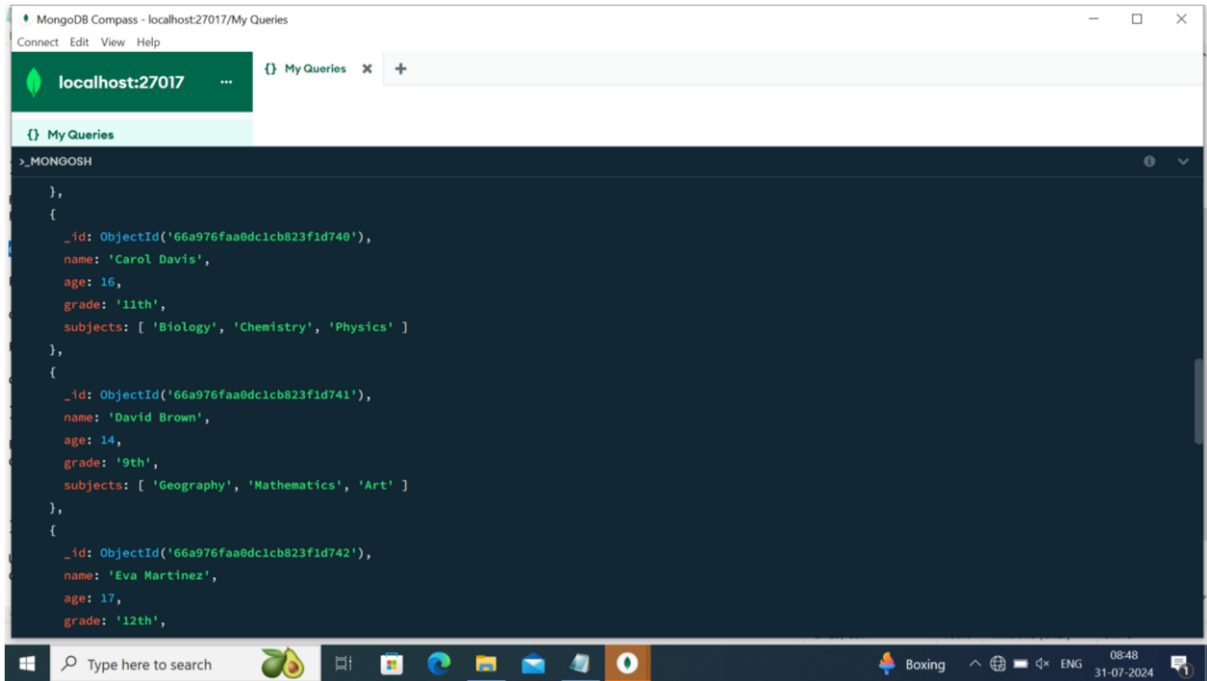
The Windows taskbar at the bottom shows the system time as 08:47 on 31-07-2024, with a weather forecast of 23°C Mostly cloudy.



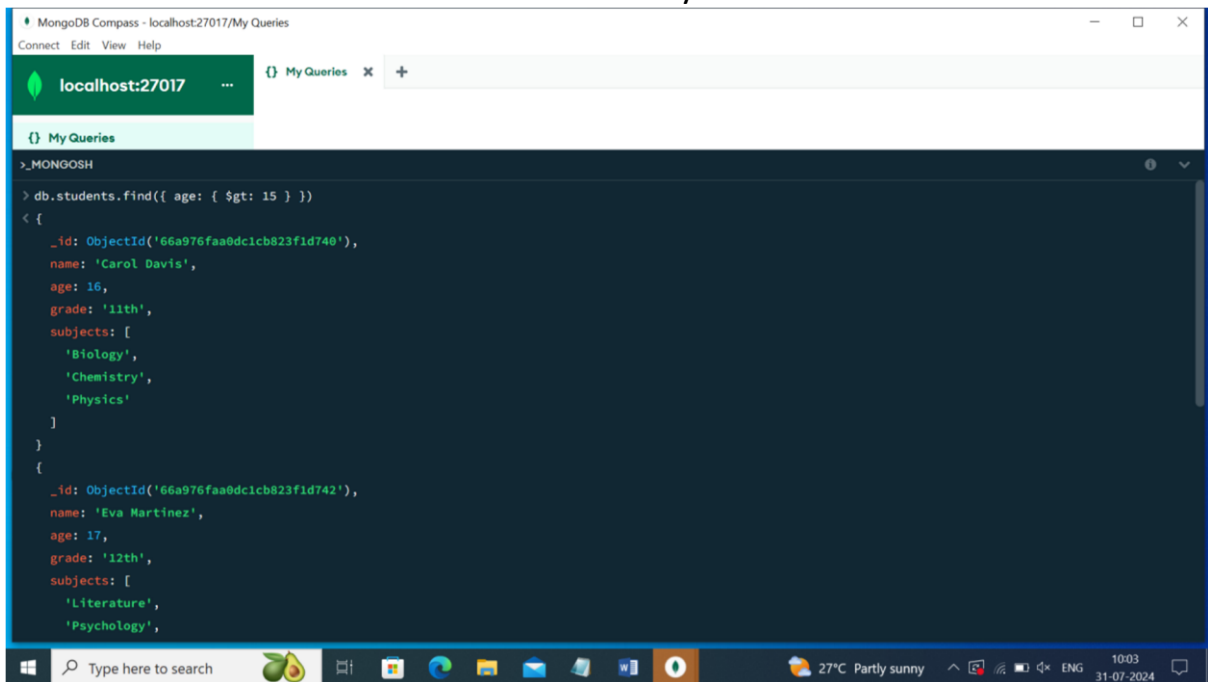
Find all Read:

Write queries to: students.





2. Find students who are older than 15 years



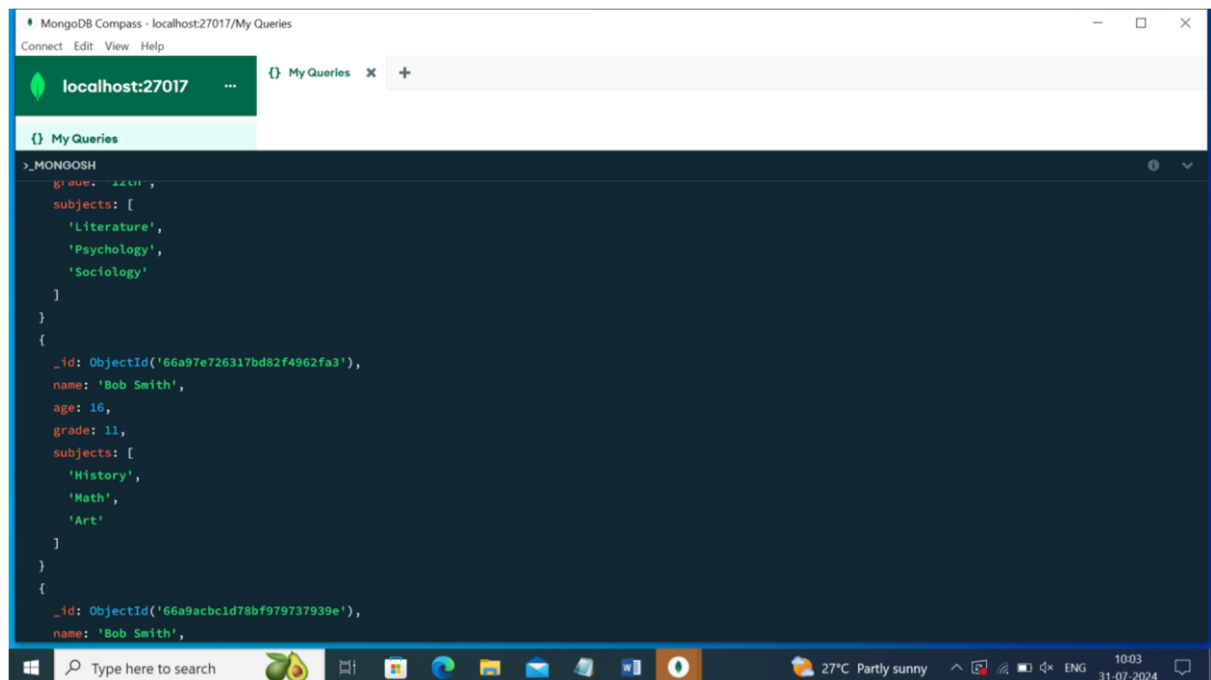
The screenshot shows the MongoDB Compass interface. The top bar indicates the connection to 'localhost:27017'. The 'My Queries' tab is active, showing a query in the 'My Queries' editor. The query is:

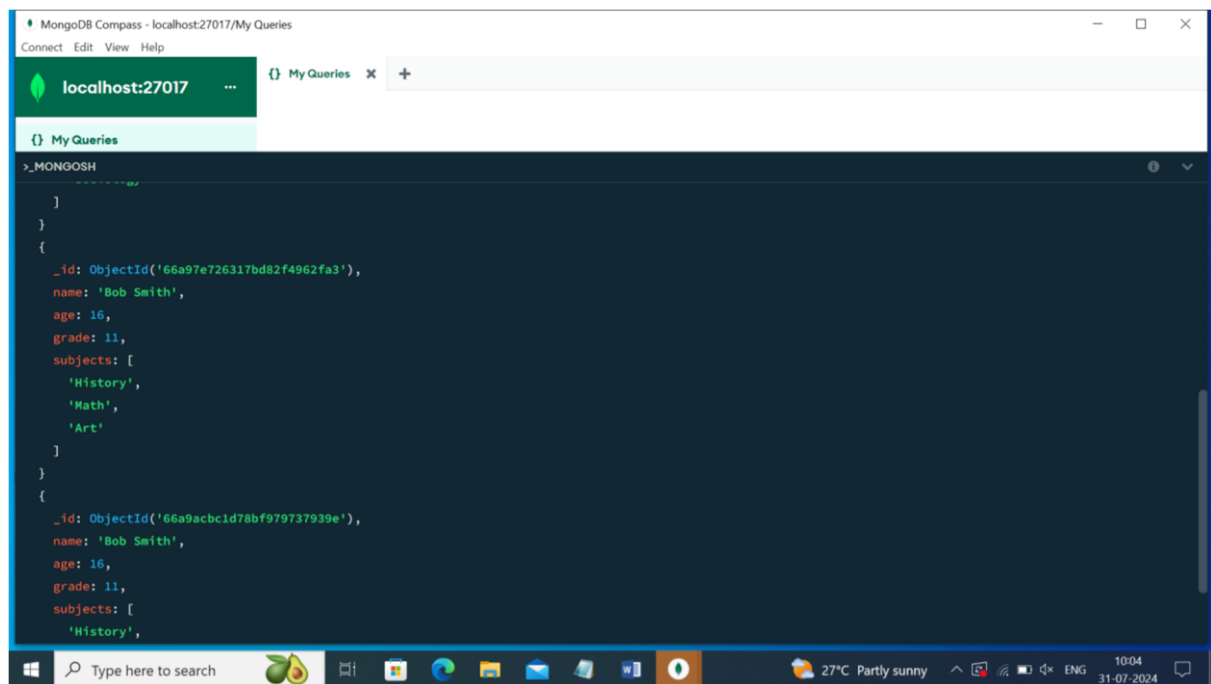
```
> db.students.find({ age: { $gt: 15 } })
```

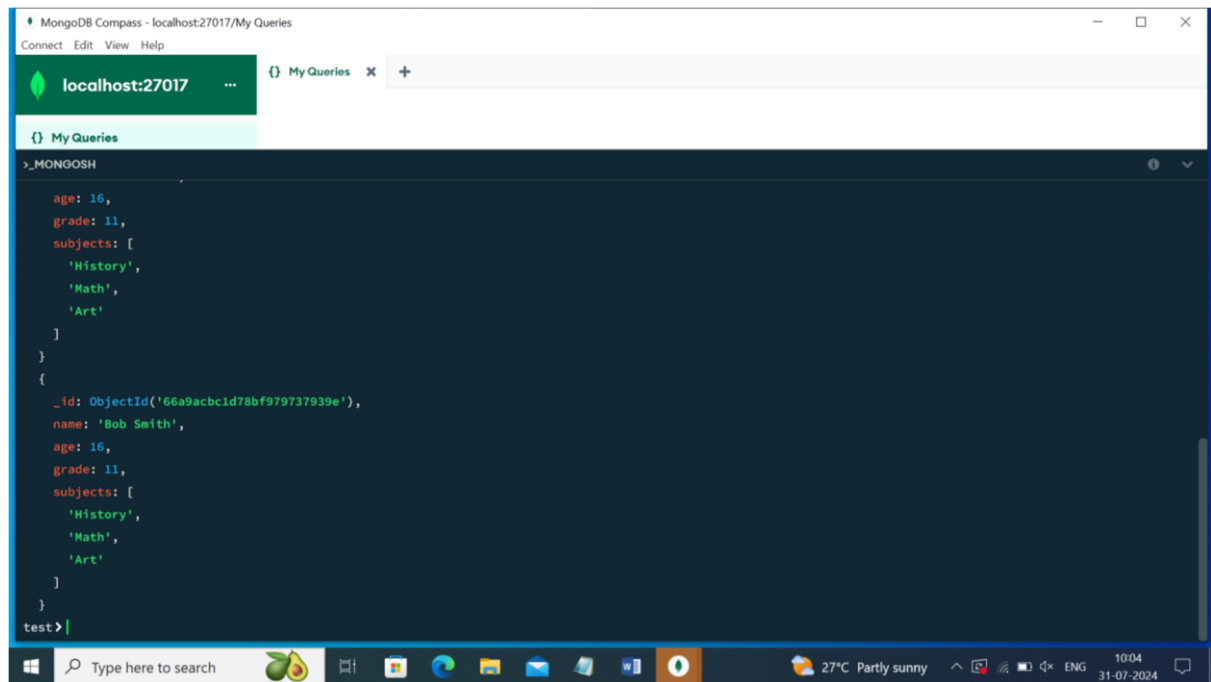
The results are displayed in a JSON format, showing two documents:

```
< {
  _id: ObjectId('66a976faa0dc1cb823f1d740'),
  name: 'Carol Davis',
  age: 16,
  grade: '11th',
  subjects: [
    'Biology',
    'Chemistry',
    'Physics'
  ]
}
{
  _id: ObjectId('66a976faa0dc1cb823f1d742'),
  name: 'Eva Martinez',
  age: 17,
  grade: '12th',
  subjects: [
    'Literature',
    'Psychology'
  ]
}
```

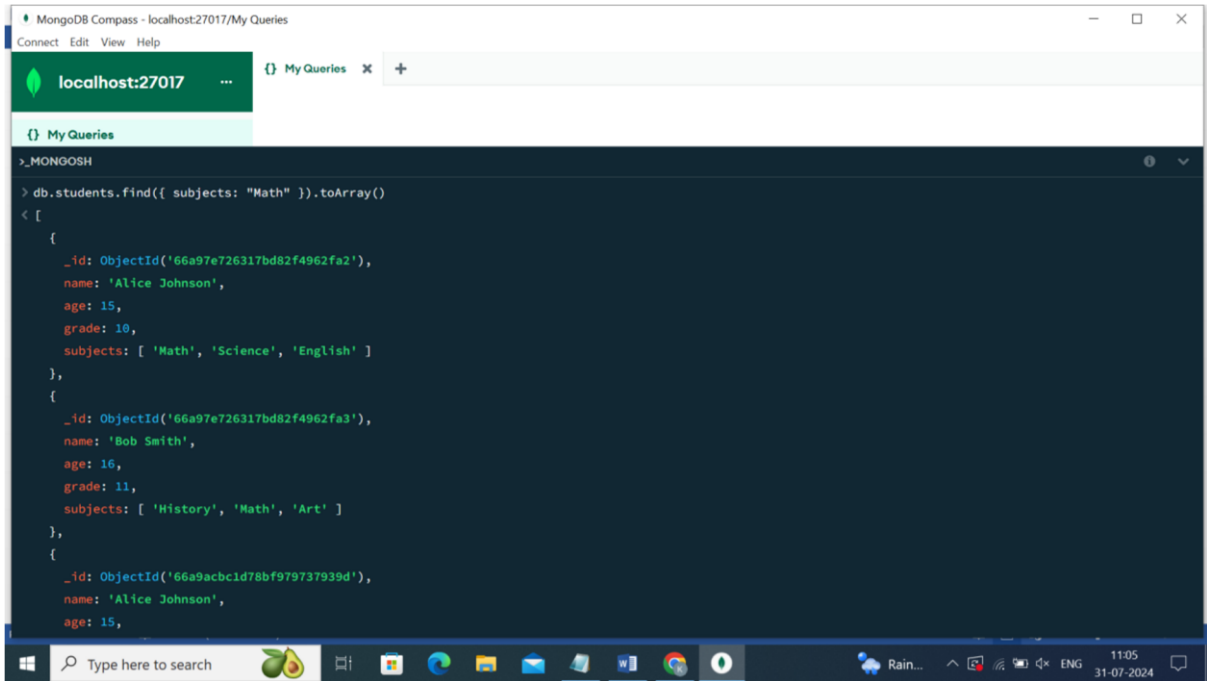
The Windows taskbar at the bottom shows the system clock as 10:03 on 31-07-2024, and the weather as 27°C Partly sunny.

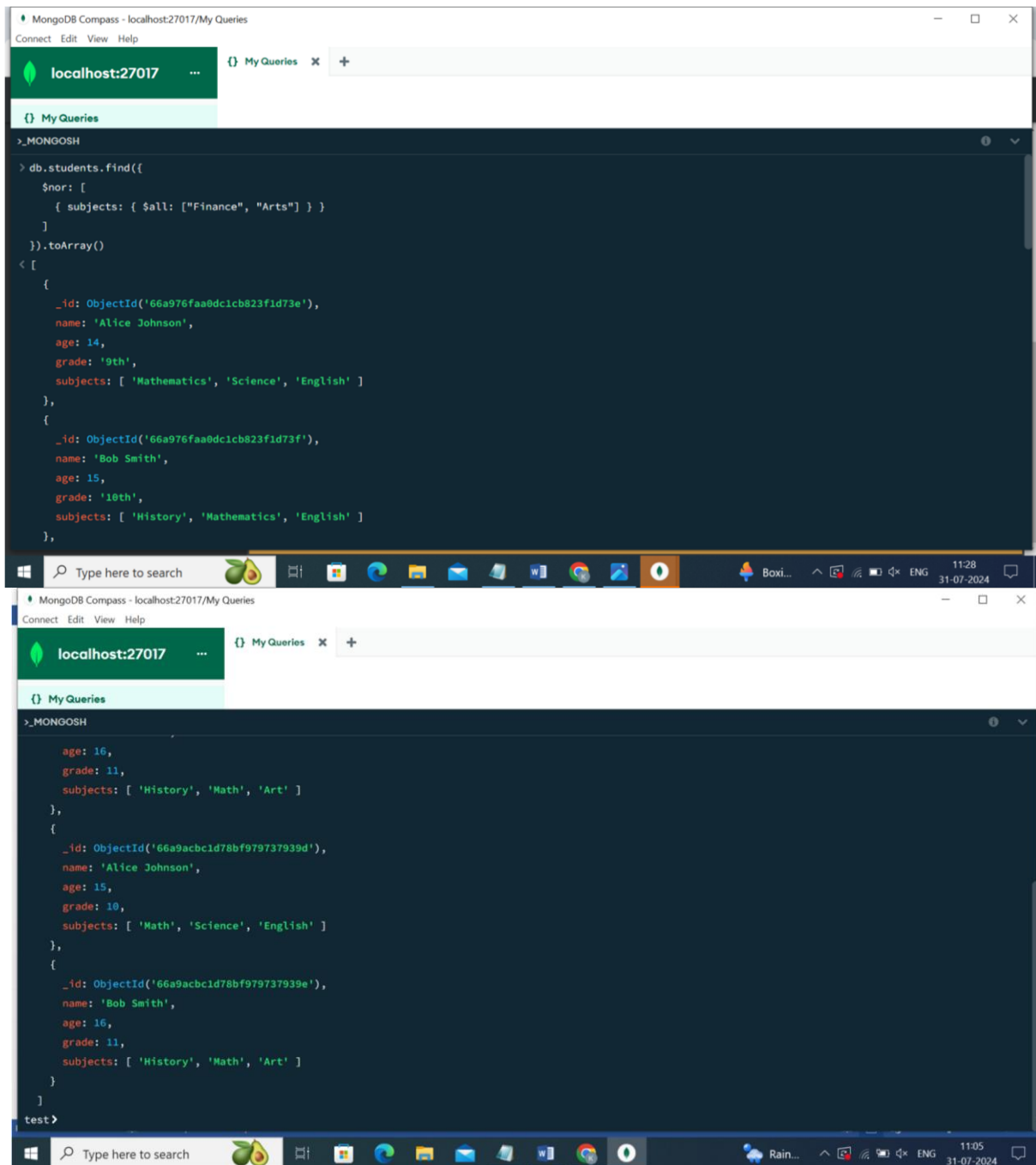




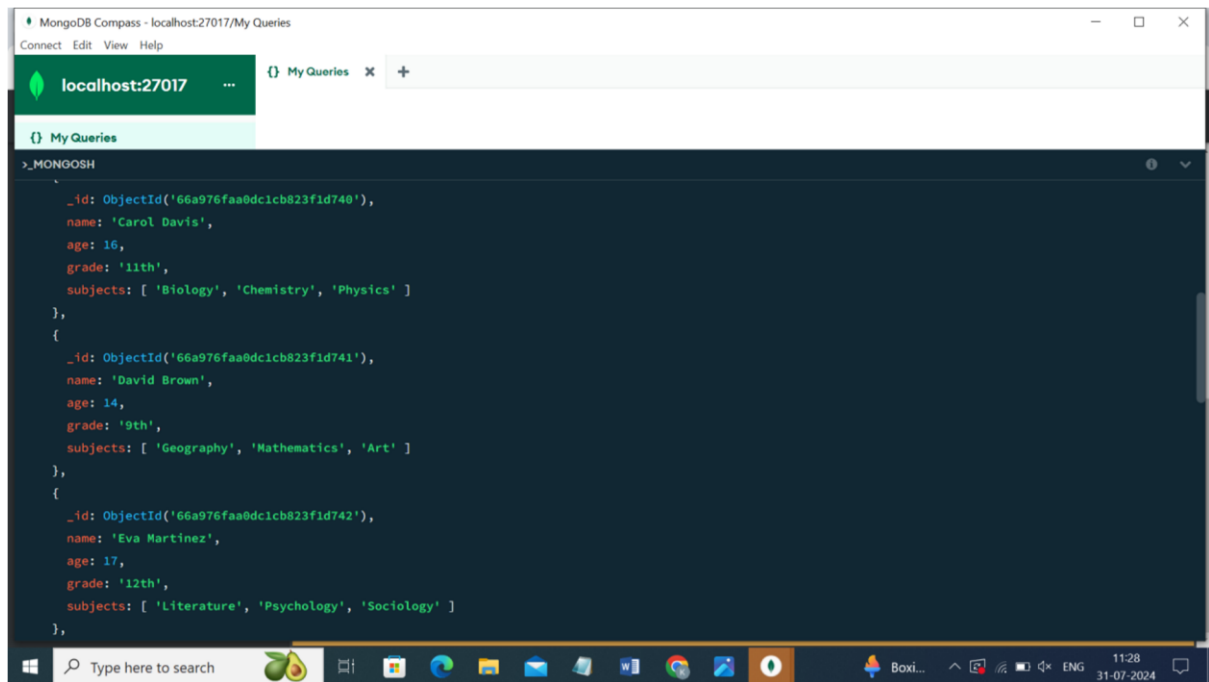


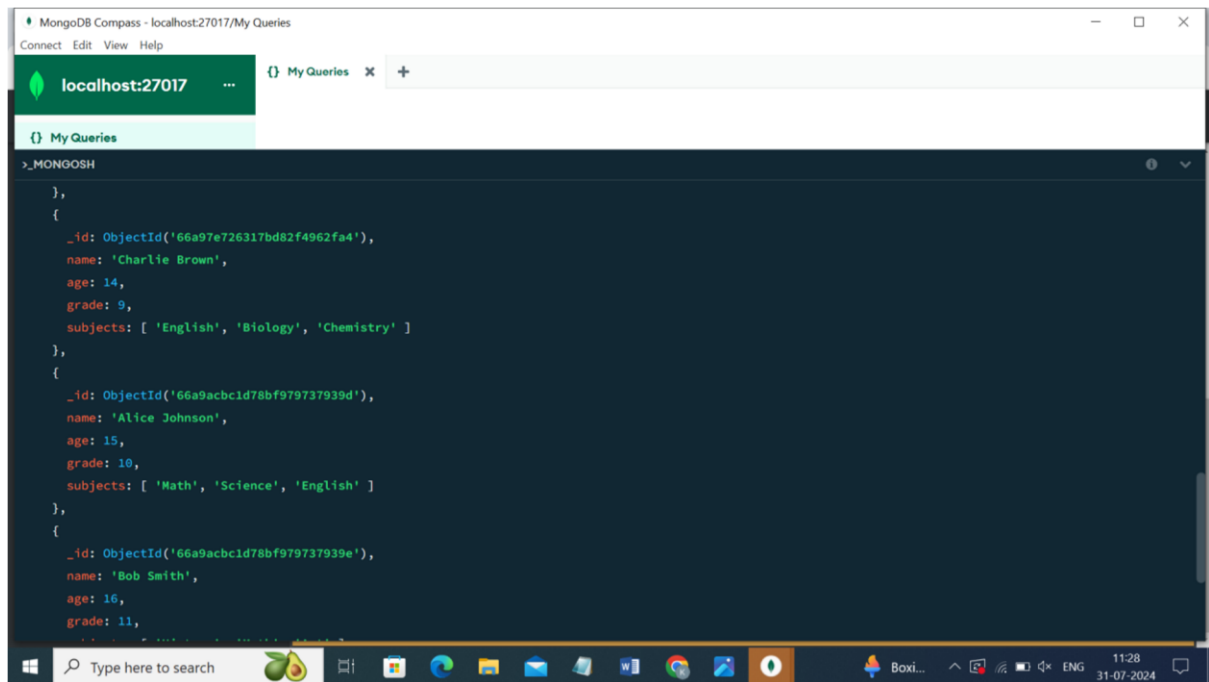
3.Find students who have "Math" as one of their subjects.

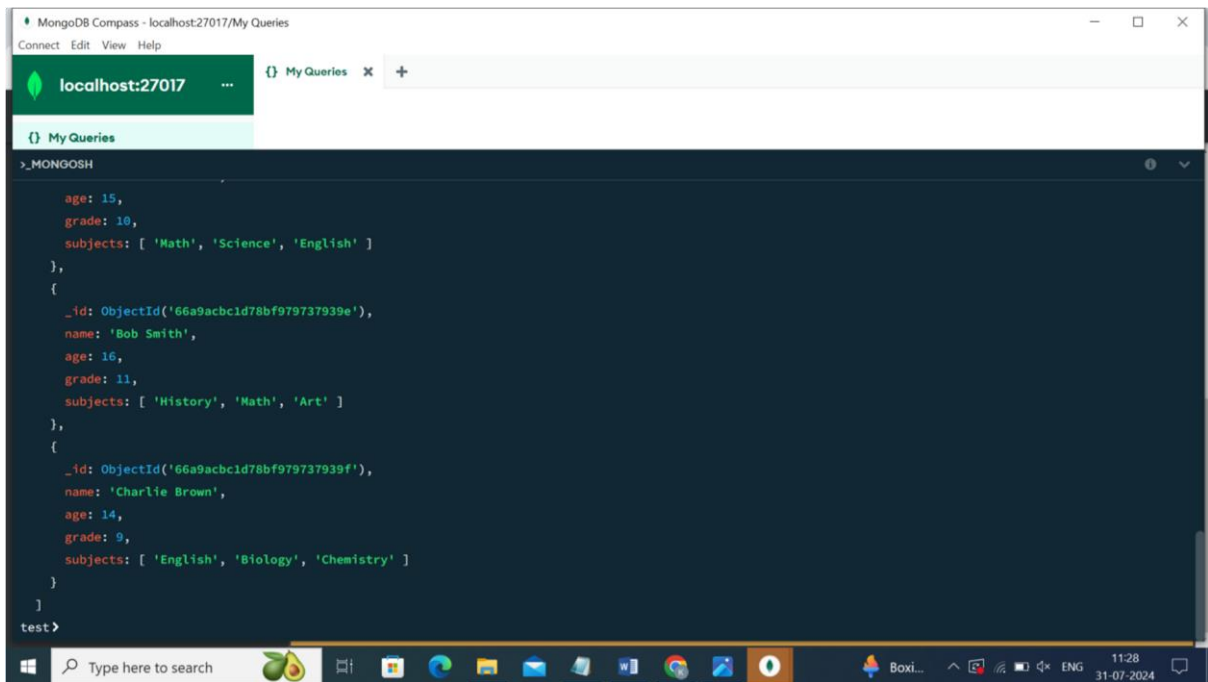




4. Find students who not have both "Finance" and "Arts" subjects



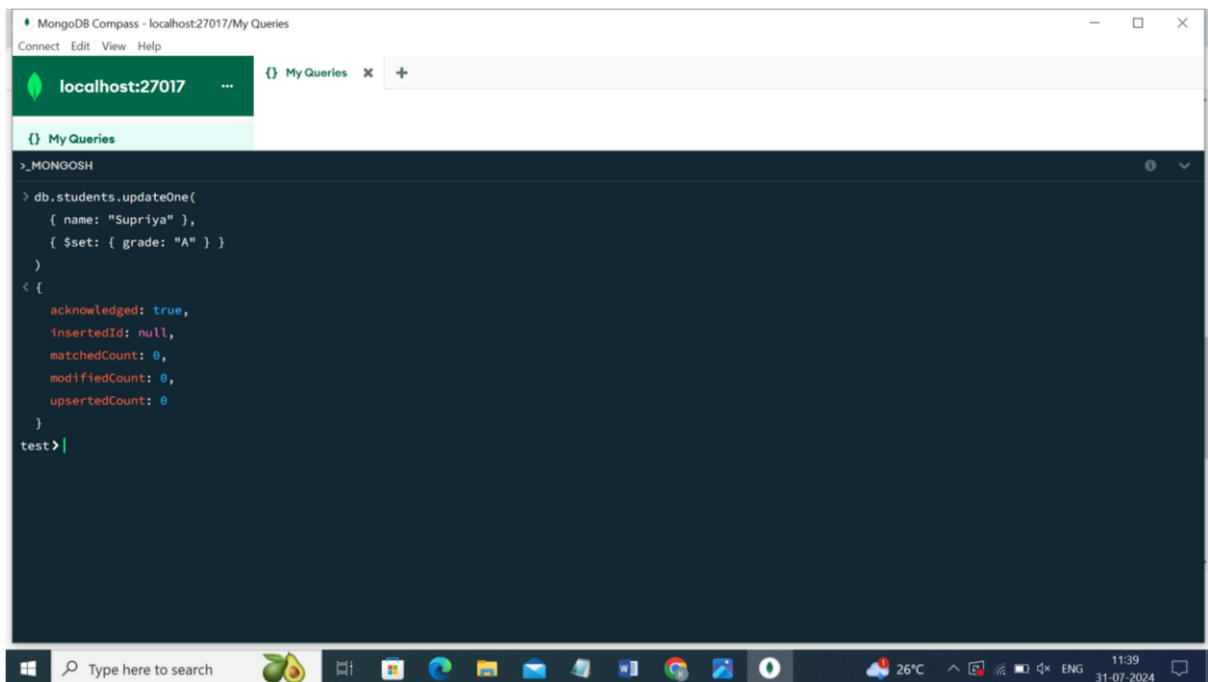




The screenshot shows the MongoDB Compass interface with a connection to localhost:27017. The 'My Queries' tab is active, displaying a JSON array of three student documents. The first document is for a student named 'Bob Smith' in grade 10, the second for 'Charlie Brown' in grade 11, and the third for an unnamed student in grade 9. The interface includes a top menu bar, a sidebar with the 'My Queries' tab, and a main editor area with a dark theme. The Windows taskbar at the bottom shows the time as 11:28 on 31-07-2024.

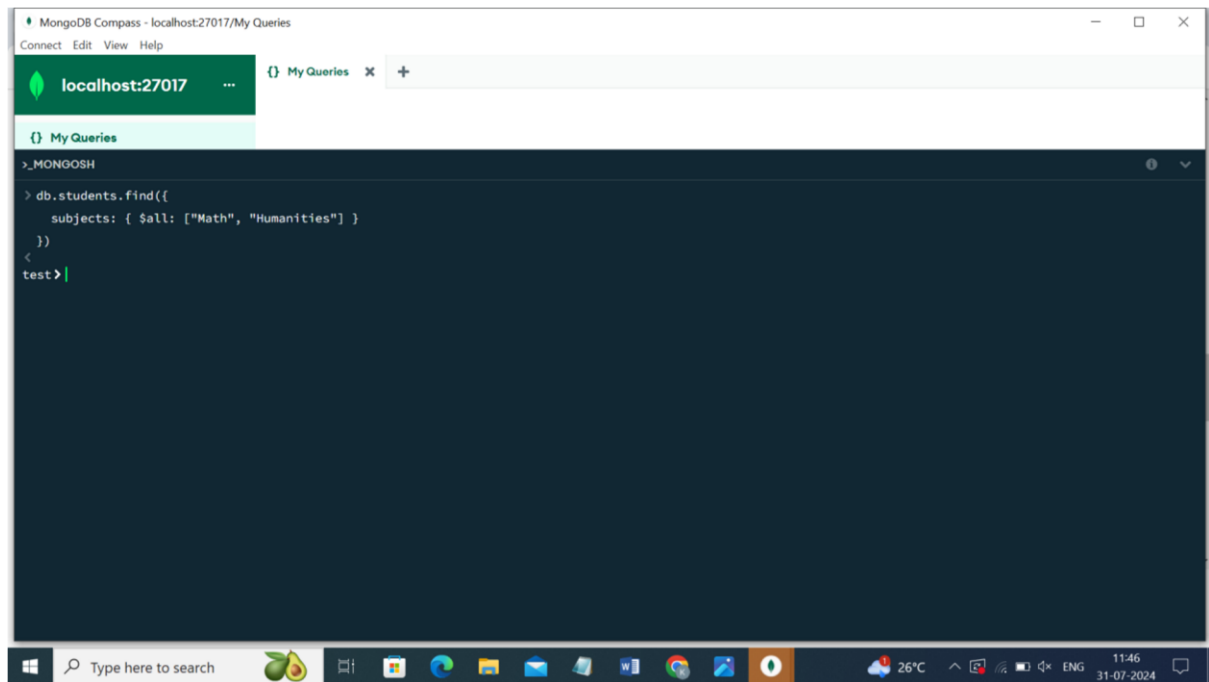
```
>_MONGOSH
[
  {
    age: 15,
    grade: 10,
    subjects: [ 'Math', 'Science', 'English' ]
  },
  {
    _id: ObjectId('66a9acbcd78bf979737939e'),
    name: 'Bob Smith',
    age: 16,
    grade: 11,
    subjects: [ 'History', 'Math', 'Art' ]
  },
  {
    _id: ObjectId('66a9acbcd78bf979737939f'),
    name: 'Charlie Brown',
    age: 14,
    grade: 9,
    subjects: [ 'English', 'Biology', 'Chemistry' ]
  }
]
test>
```

5. Update: Update the grade of a student named "Supriya" to "A"



The screenshot shows the MongoDB Compass interface with the same connection. The 'My Queries' tab is active, displaying a MongoDB update query. The query uses the updateOne method to set the grade of the student named 'Supriya' to 'A'. The result shows that the update was successful, with the matchedCount being 1 and the modifiedCount being 1. The interface includes a top menu bar, a sidebar with the 'My Queries' tab, and a main editor area with a dark theme. The Windows taskbar at the bottom shows the time as 11:39 on 31-07-2024.

```
>_MONGOSH
> db.students.updateOne(
  { name: "Supriya" },
  { $set: { grade: "A" } }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
test>
```



6. Find students who have "Math" and “Humanities” subjects.