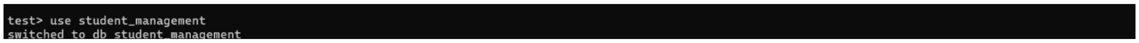
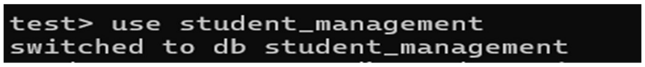
MongoDB Assignment – 1

1. Create a new database called student_management.



```
test> use student_management
switched to db student_management
```

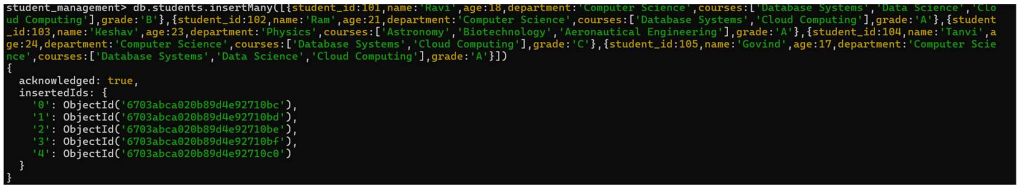
2. Create a collection called students in the student_management database.



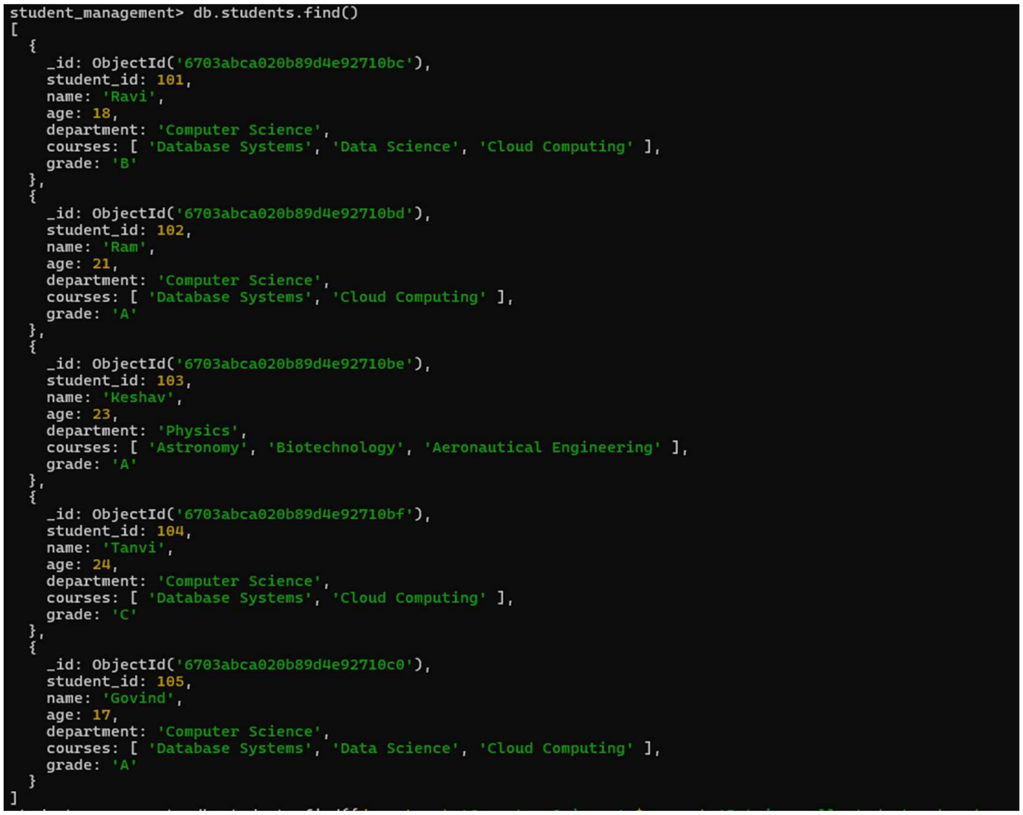
```
test> use student_management
switched to db student_management
```

3. Insert at least five student records into the students collection. Each record should have the following fields:

- student_id (integer)
- name (string)
- age (integer)
- department (string)
- courses (array of strings)
- grade (string)



```
student_management> db.students.insertMany([{student_id:101,name:'Ravi',age:18,department:'Computer Science',courses:['Database Systems','Data Science','Clo
ud Computing'],grade:'B'},{student_id:102,name:'Ram',age:21,department:'Computer Science',courses:['Database Systems','Cloud Computing'],grade:'A'},{student
_id:103,name:'Keshav',age:23,department:'Physics',courses:['Astronomy','Biotechnology','Aeronautical Engineering'],grade:'A'},{student_id:104,name:'Tanvi',a
ge:24,department:'Computer Science',courses:['Database Systems','Cloud Computing'],grade:'C'},{student_id:105,name:'Govind',age:17,department:'Computer Scie
nce',courses:['Database Systems','Data Science','Cloud Computing'],grade:'A'}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6703abca020b89d4e92710bc'),
    '1': ObjectId('6703abca020b89d4e92710bd'),
    '2': ObjectId('6703abca020b89d4e92710be'),
    '3': ObjectId('6703abca020b89d4e92710bf'),
    '4': ObjectId('6703abca020b89d4e92710c0')
  }
}
```

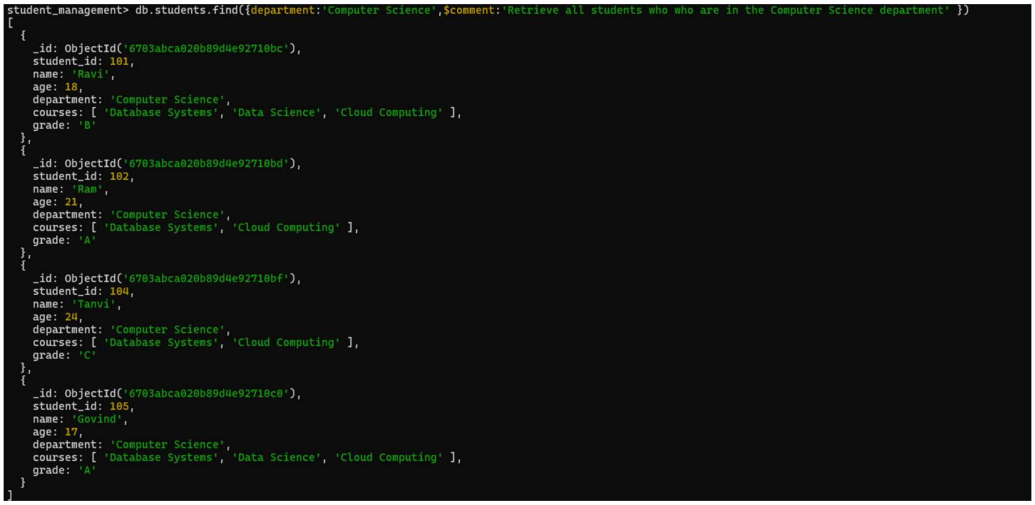


```
student_management> db.students.find()
[
  {
    _id: ObjectId('6703abca020b89d4e92710bc'),
    student_id: 101,
    name: 'Ravi',
    age: 18,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Data Science', 'Cloud Computing' ],
    grade: 'B'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710bd'),
    student_id: 102,
    name: 'Ram',
    age: 21,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing' ],
    grade: 'A'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710be'),
    student_id: 103,
    name: 'Keshav',
    age: 23,
    department: 'Physics',
    courses: [ 'Astronomy', 'Biotechnology', 'Aeronautical Engineering' ],
    grade: 'A'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710bf'),
    student_id: 104,
    name: 'Tanvi',
    age: 24,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing' ],
    grade: 'C'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710c0'),
    student_id: 105,
    name: 'Govind',
    age: 17,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Data Science', 'Cloud Computing' ],
    grade: 'A'
  }
]
```

4. Query the Collection:

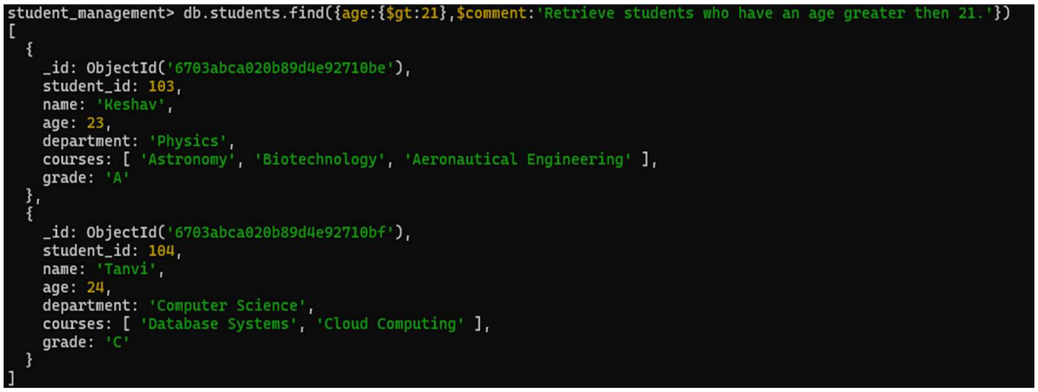
Write queries to perform the following tasks:

- Retrieve all students who are in the "Computer Science" department.



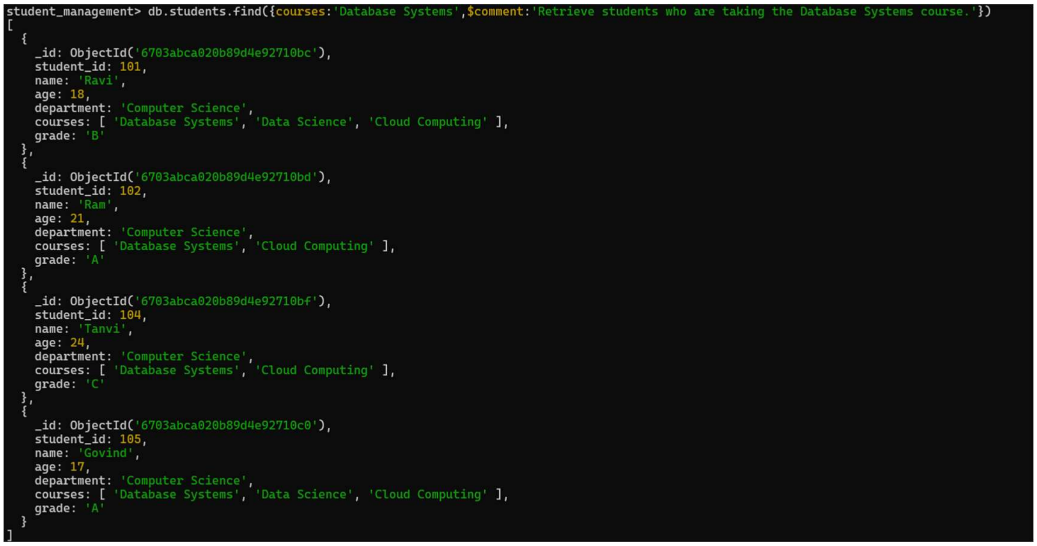
```
student_management> db.students.find({department:'Computer Science',$comment:'Retrieve all students who who are in the Computer Science department' })
[
  {
    _id: ObjectId('6703abca020b89d4e92710bc'),
    student_id: 101,
    name: 'Ravi',
    age: 18,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Data Science', 'Cloud Computing' ],
    grade: 'B'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710bd'),
    student_id: 102,
    name: 'Ram',
    age: 21,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing' ],
    grade: 'A'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710bf'),
    student_id: 104,
    name: 'Tanvi',
    age: 24,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing' ],
    grade: 'C'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710c0'),
    student_id: 105,
    name: 'Govind',
    age: 17,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Data Science', 'Cloud Computing' ],
    grade: 'A'
  }
]
```

- Retrieve students who have an age greater than 21.



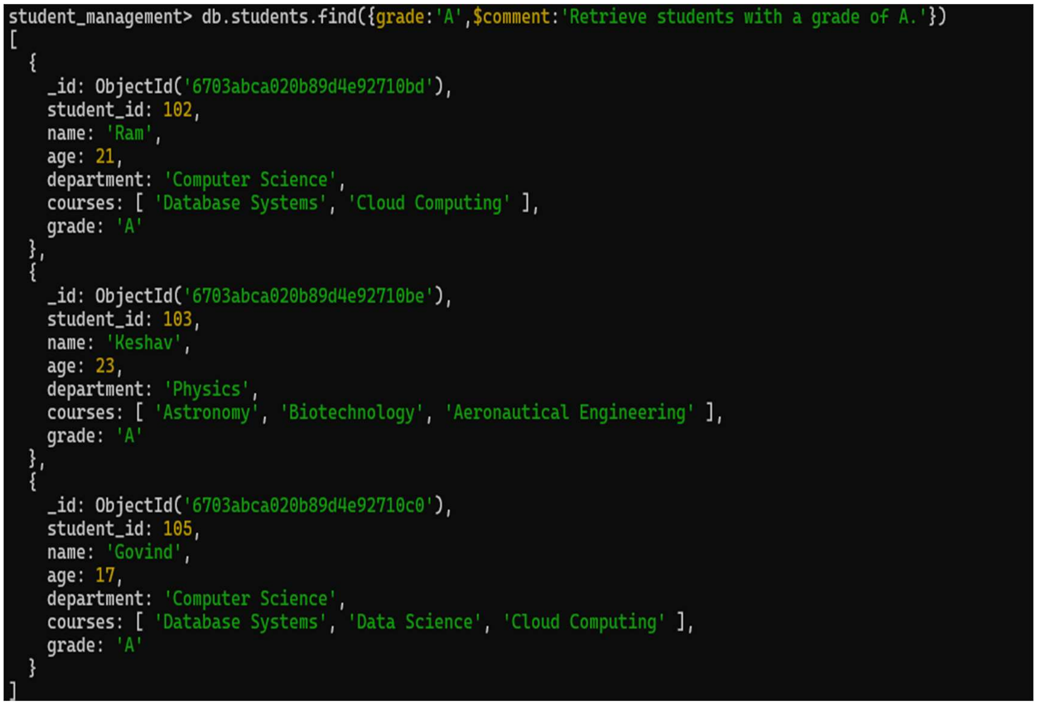
```
student_management> db.students.find({age:{$gt:21},$comment:'Retrieve students who have an age greater then 21.'})
[
  {
    _id: ObjectId('6703abca020b89d4e92710be'),
    student_id: 103,
    name: 'Keshav',
    age: 23,
    department: 'Physics',
    courses: [ 'Astronomy', 'Biotechnology', 'Aeronautical Engineering' ],
    grade: 'A'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710bf'),
    student_id: 104,
    name: 'Tanvi',
    age: 24,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing' ],
    grade: 'C'
  }
]
```

- Retrieve students who are taking the "Database Systems" course.



```
student_management> db.students.find({courses:'Database Systems',$comment:'Retrieve students who are taking the Database Systems course.'})
[
  {
    _id: ObjectId('6703abca020b89d4e92710bc'),
    student_id: 101,
    name: 'Ravi',
    age: 18,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Data Science', 'Cloud Computing' ],
    grade: 'B'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710bd'),
    student_id: 102,
    name: 'Ram',
    age: 21,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing' ],
    grade: 'A'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710bf'),
    student_id: 104,
    name: 'Tanvi',
    age: 24,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing' ],
    grade: 'C'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710c0'),
    student_id: 105,
    name: 'Govind',
    age: 17,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Data Science', 'Cloud Computing' ],
    grade: 'A'
  }
]
```

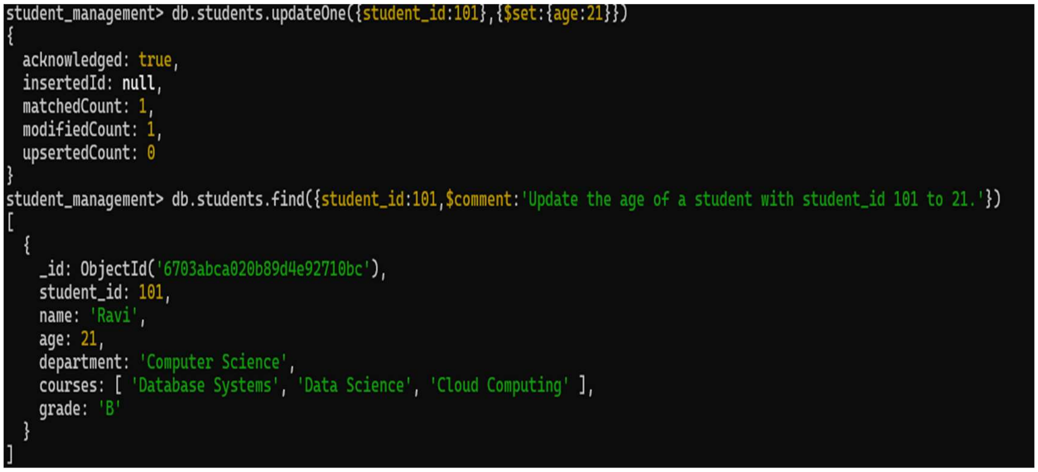
- Retrieve students with a grade of "A".



```
student_management> db.students.find({grade:'A',$comment:'Retrieve students with a grade of A.'})
[
  {
    _id: ObjectId('6703abca020b89d4e92710bd'),
    student_id: 102,
    name: 'Ram',
    age: 21,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing' ],
    grade: 'A'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710be'),
    student_id: 103,
    name: 'Keshav',
    age: 23,
    department: 'Physics',
    courses: [ 'Astronomy', 'Biotechnology', 'Aeronautical Engineering' ],
    grade: 'A'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710c0'),
    student_id: 105,
    name: 'Govind',
    age: 17,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Data Science', 'Cloud Computing' ],
    grade: 'A'
  }
]
```

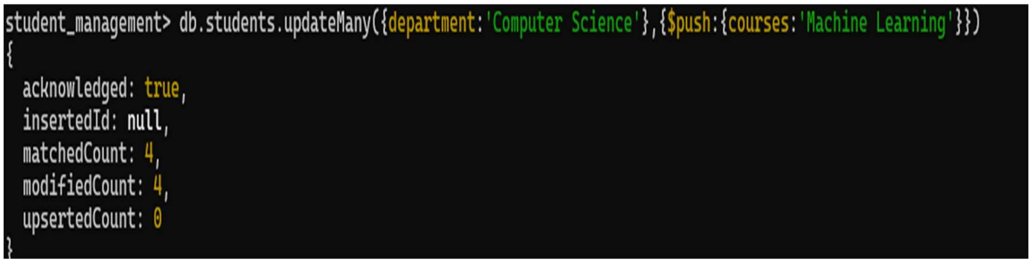
5. Update Documents:

- Update the age of a student with student_id 101 to 21.



```
student_management> db.students.updateOne({student_id:101},{$set:{age:21}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
student_management> db.students.find({student_id:101,$comment:'Update the age of a student with student_id 101 to 21.'})
[
  {
    _id: ObjectId('6703abca020b89d4e92710bc'),
    student_id: 101,
    name: 'Ravi',
    age: 21,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Data Science', 'Cloud Computing' ],
    grade: 'B'
  }
]
```

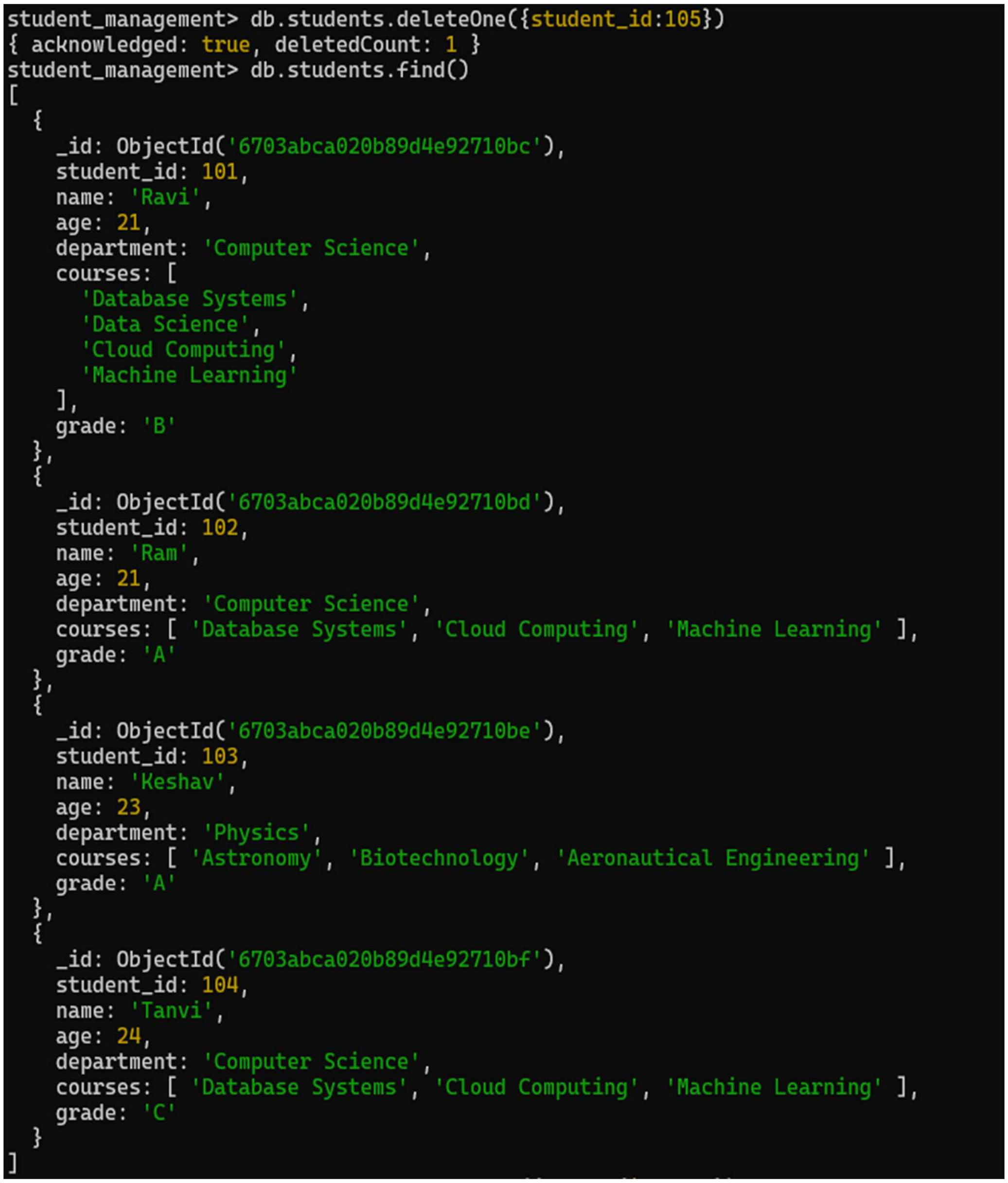
- Add a new course, "Machine Learning", to the courses array for students in the "Computer Science" department.



```
student_management> db.students.updateMany({department:'Computer Science'},{$push:{courses:'Machine Learning'}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 4,
  modifiedCount: 4,
  upsertedCount: 0
}
```

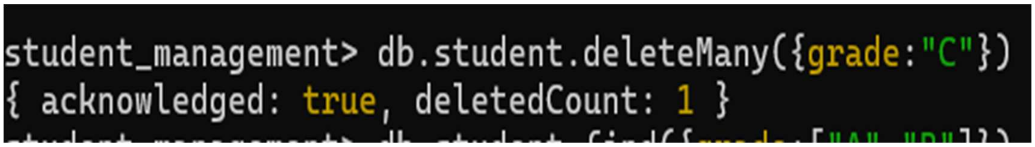
6. **Delete Documents:**

- Delete a student record with student_id 105.

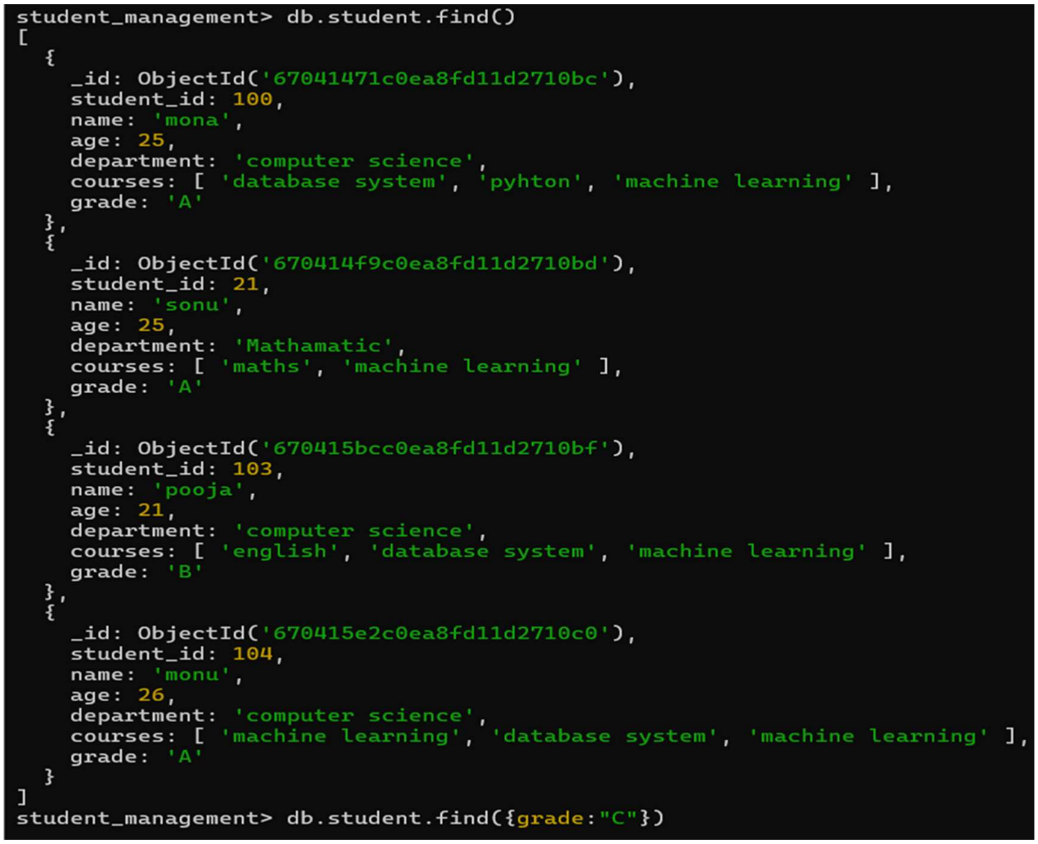


```
student_management> db.students.deleteOne({student_id:105})
{ acknowledged: true, deletedCount: 1 }
student_management> db.students.find()
[
  {
    _id: ObjectId('6703abca020b89d4e92710bc'),
    student_id: 101,
    name: 'Ravi',
    age: 21,
    department: 'Computer Science',
    courses: [
      'Database Systems',
      'Data Science',
      'Cloud Computing',
      'Machine Learning'
    ],
    grade: 'B'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710bd'),
    student_id: 102,
    name: 'Ram',
    age: 21,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing', 'Machine Learning' ],
    grade: 'A'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710be'),
    student_id: 103,
    name: 'Keshav',
    age: 23,
    department: 'Physics',
    courses: [ 'Astronomy', 'Biotechnology', 'Aeronautical Engineering' ],
    grade: 'A'
  },
  {
    _id: ObjectId('6703abca020b89d4e92710bf'),
    student_id: 104,
    name: 'Tanvi',
    age: 24,
    department: 'Computer Science',
    courses: [ 'Database Systems', 'Cloud Computing', 'Machine Learning' ],
    grade: 'C'
  }
]
```

- Delete all students who have a grade lower than "C".



```
student_management> db.student.deleteMany({grade:"C"})
{ acknowledged: true, deletedCount: 1 }
student_management> db.student.find({grade:["A","B"]})
```



```
student_management> db.student.find()
[
  {
    _id: ObjectId('67041471c0ea8fd11d2710bc'),
    student_id: 100,
    name: 'mona',
    age: 25,
    department: 'computer science',
    courses: [ 'database system', 'pyhton', 'machine learning' ],
    grade: 'A'
  },
  {
    _id: ObjectId('670414f9c0ea8fd11d2710bd'),
    student_id: 21,
    name: 'sonu',
    age: 25,
    department: 'Mathamatic',
    courses: [ 'maths', 'machine learning' ],
    grade: 'A'
  },
  {
    _id: ObjectId('670415bcc0ea8fd11d2710bf'),
    student_id: 103,
    name: 'pooja',
    age: 21,
    department: 'computer science',
    courses: [ 'english', 'database system', 'machine learning' ],
    grade: 'B'
  },
  {
    _id: ObjectId('670415e2c0ea8fd11d2710c0'),
    student_id: 104,
    name: 'monu',
    age: 26,
    department: 'computer science',
    courses: [ 'machine learning', 'database system', 'machine learning' ],
    grade: 'A'
  }
]
student_management> db.student.find({grade:"C"})
```