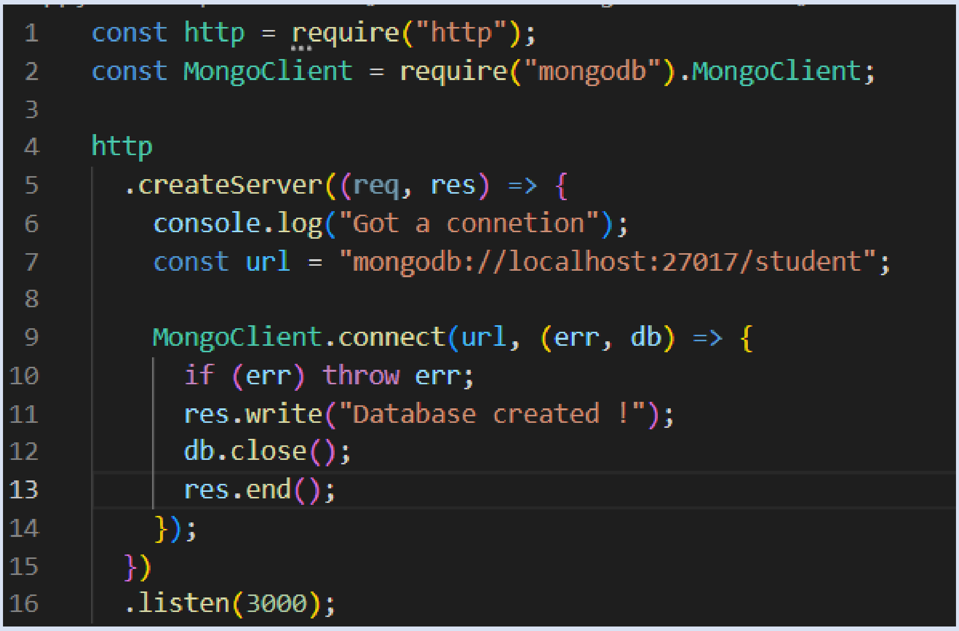
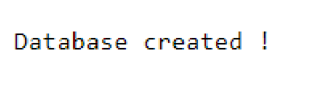
Ukistu02 - MongoDB Ex2

1. Create a Database called **student**?

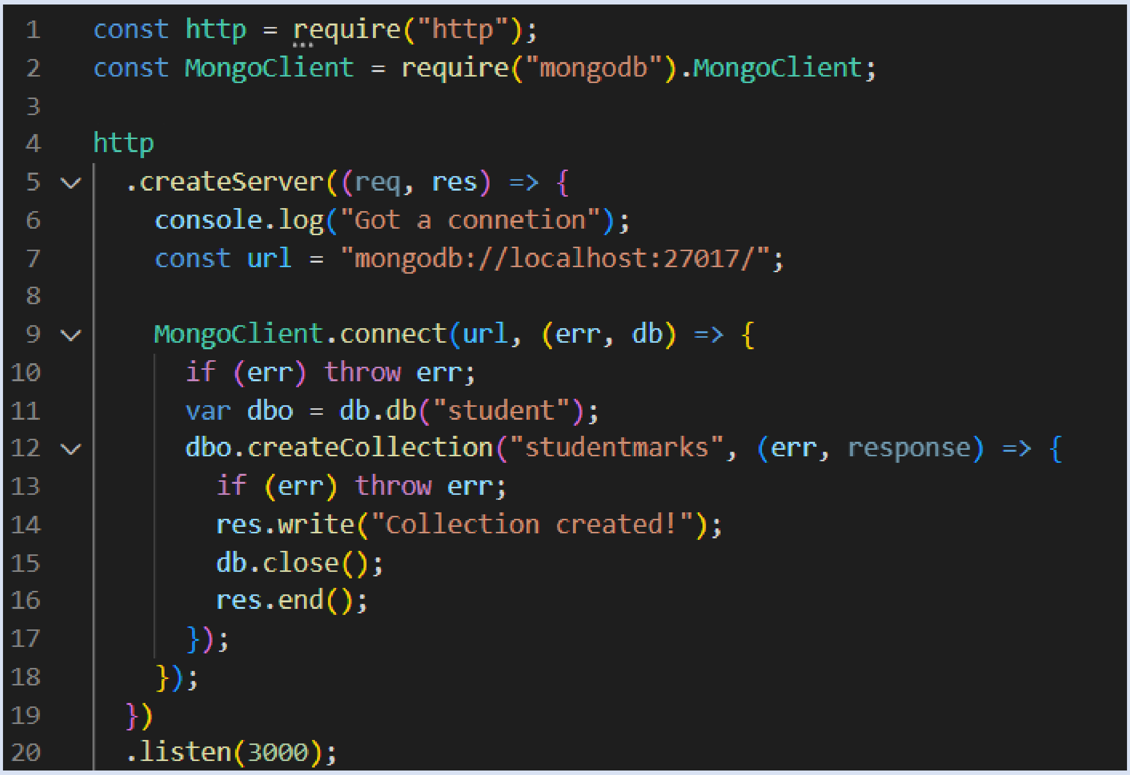


```javascript
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/student";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      res.write("Database created !");
      db.close();
      res.end();
    });
  })
  .listen(3000);
```

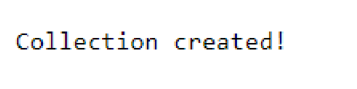


```
Database created !
```

2. Create a collection called studentmarks?

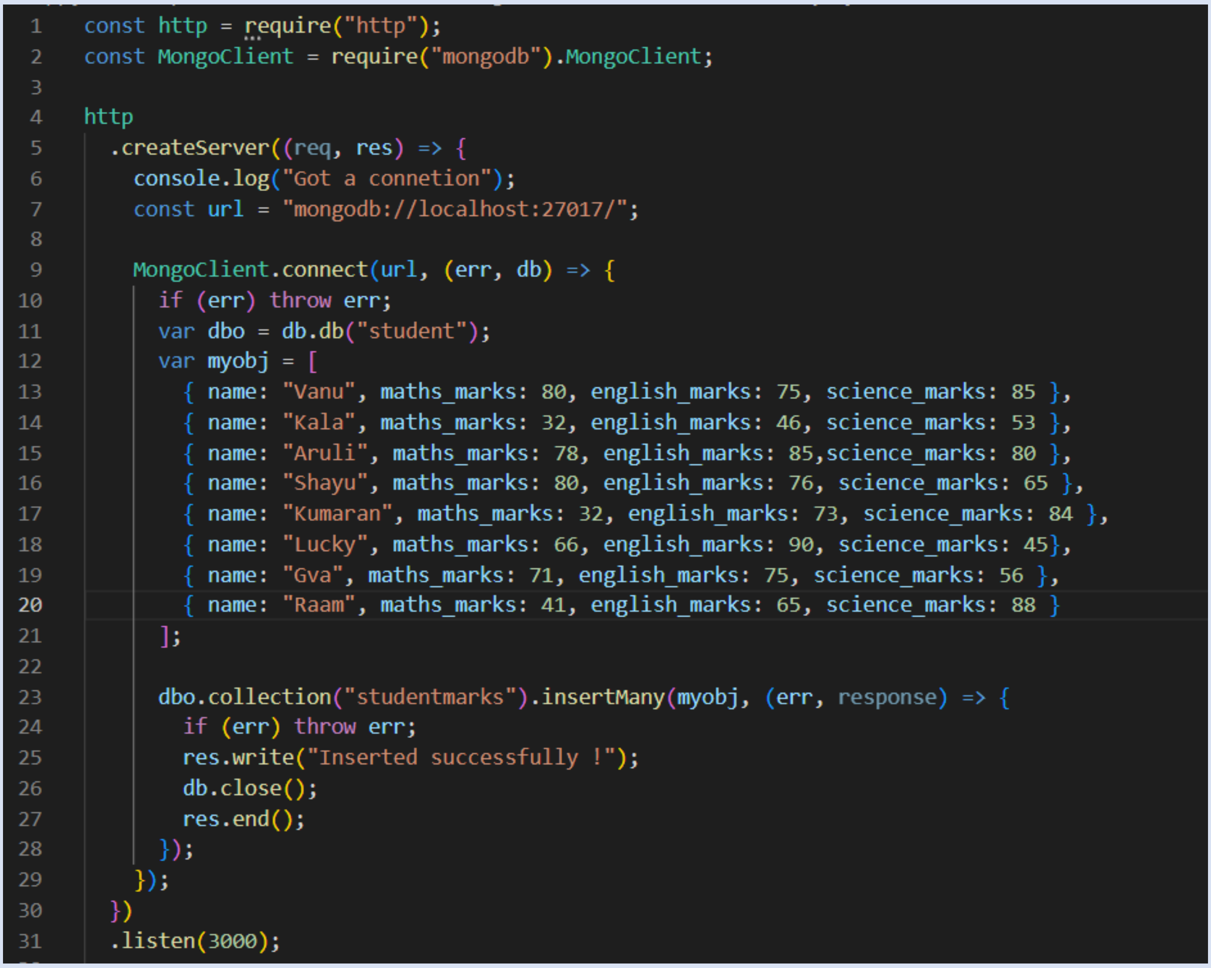


```javascript
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      dbo.createCollection("studentmarks", (err, response) => {
        if (err) throw err;
        res.write("Collection created!");
        db.close();
        res.end();
      });
    });
  })
  .listen(3000);
```

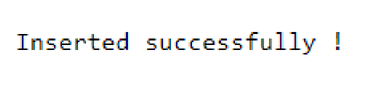


```
Collection created!
```

3. Create the documents listed in above table.

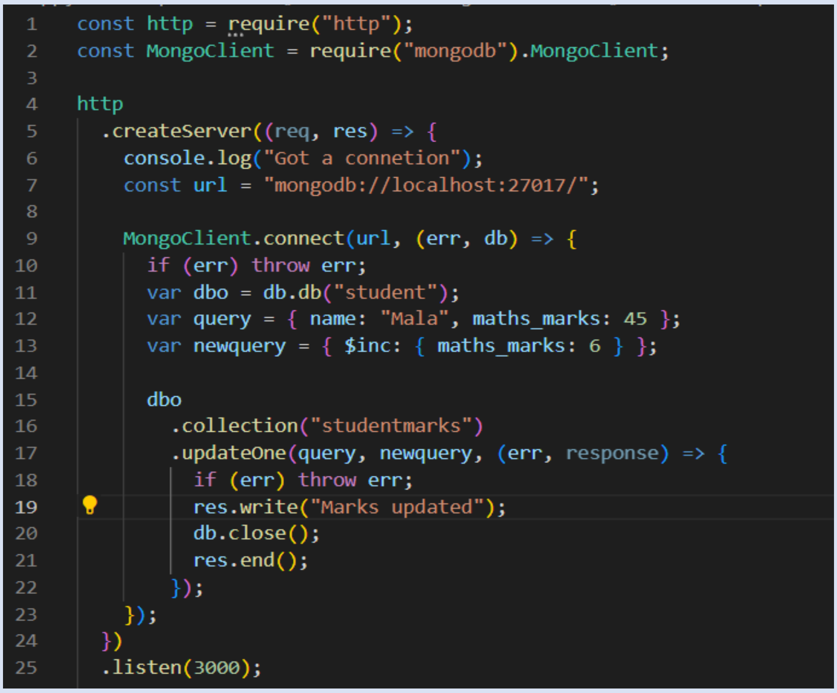


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      var myobj = [
        { name: "Vanu", maths_marks: 80, english_marks: 75, science_marks: 85 },
        { name: "Kala", maths_marks: 32, english_marks: 46, science_marks: 53 },
        { name: "Aruli", maths_marks: 78, english_marks: 85,science_marks: 80 },
        { name: "Shayu", maths_marks: 80, english_marks: 76, science_marks: 65 },
        { name: "Kumaran", maths_marks: 32, english_marks: 73, science_marks: 84 },
        { name: "Lucky", maths_marks: 66, english_marks: 90, science_marks: 45},
        { name: "Gva", maths_marks: 71, english_marks: 75, science_marks: 56 },
        { name: "Raam", maths_marks: 41, english_marks: 65, science_marks: 88 }
      ];

      dbo.collection("studentmarks").insertMany(myobj, (err, response) => {
        if (err) throw err;
        res.write("Inserted successfully !");
        db.close();
        res.end();
      });
    });
  })
  .listen(3000);
```

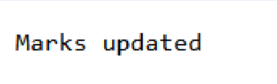


```
Inserted successfully !
```

4. Increase the maths marks of Mala by 6 mark?

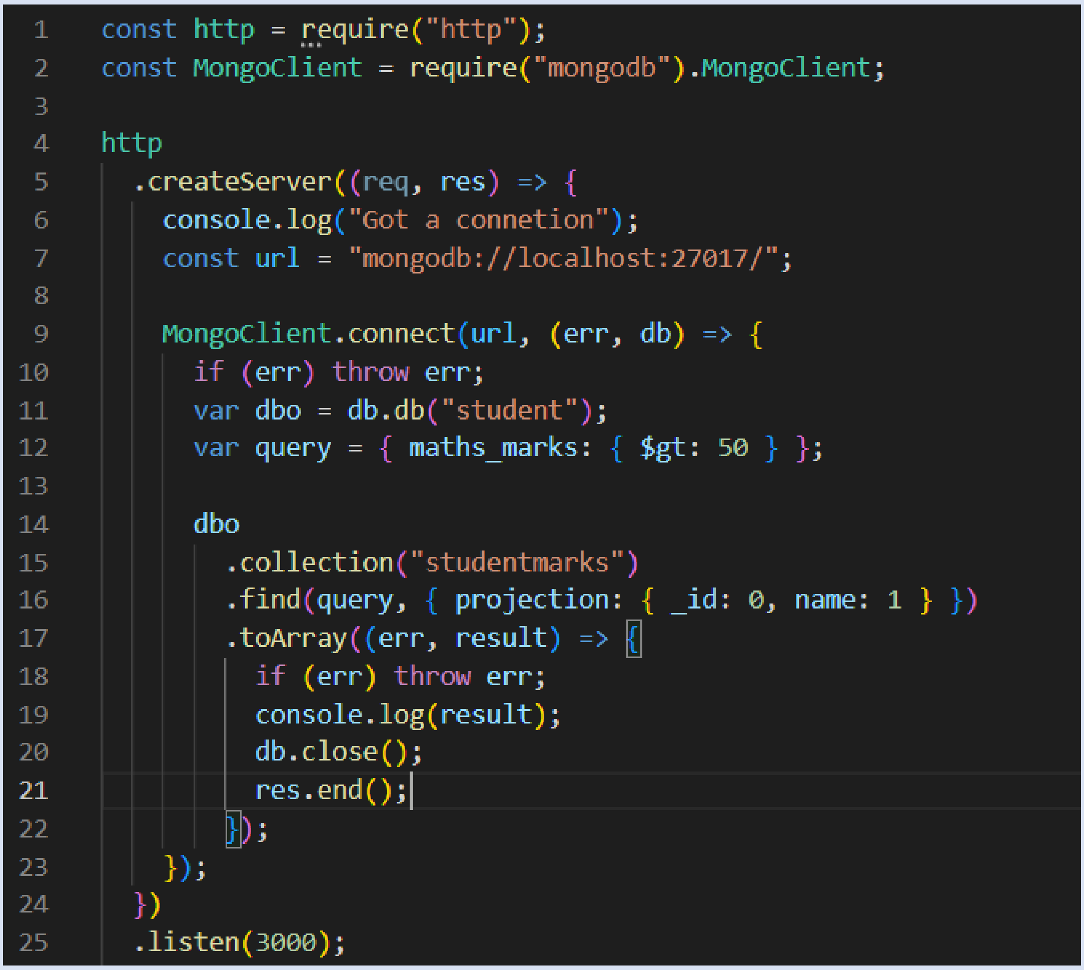


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      var query = { name: "Mala", maths_marks: 45 };
      var newquery = { $inc: { maths_marks: 6 } };

      dbo
        .collection("studentmarks")
        .updateOne(query, newquery, (err, response) => {
          if (err) throw err;
          res.write("Marks updated");
          db.close();
          res.end();
        });
    });
  })
  .listen(3000);
```

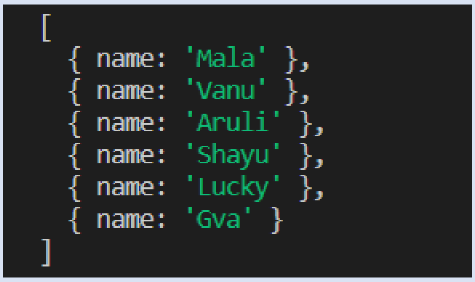


```
Marks updated
```

5. List the names of students who got more than 50 marks in Maths Subject?

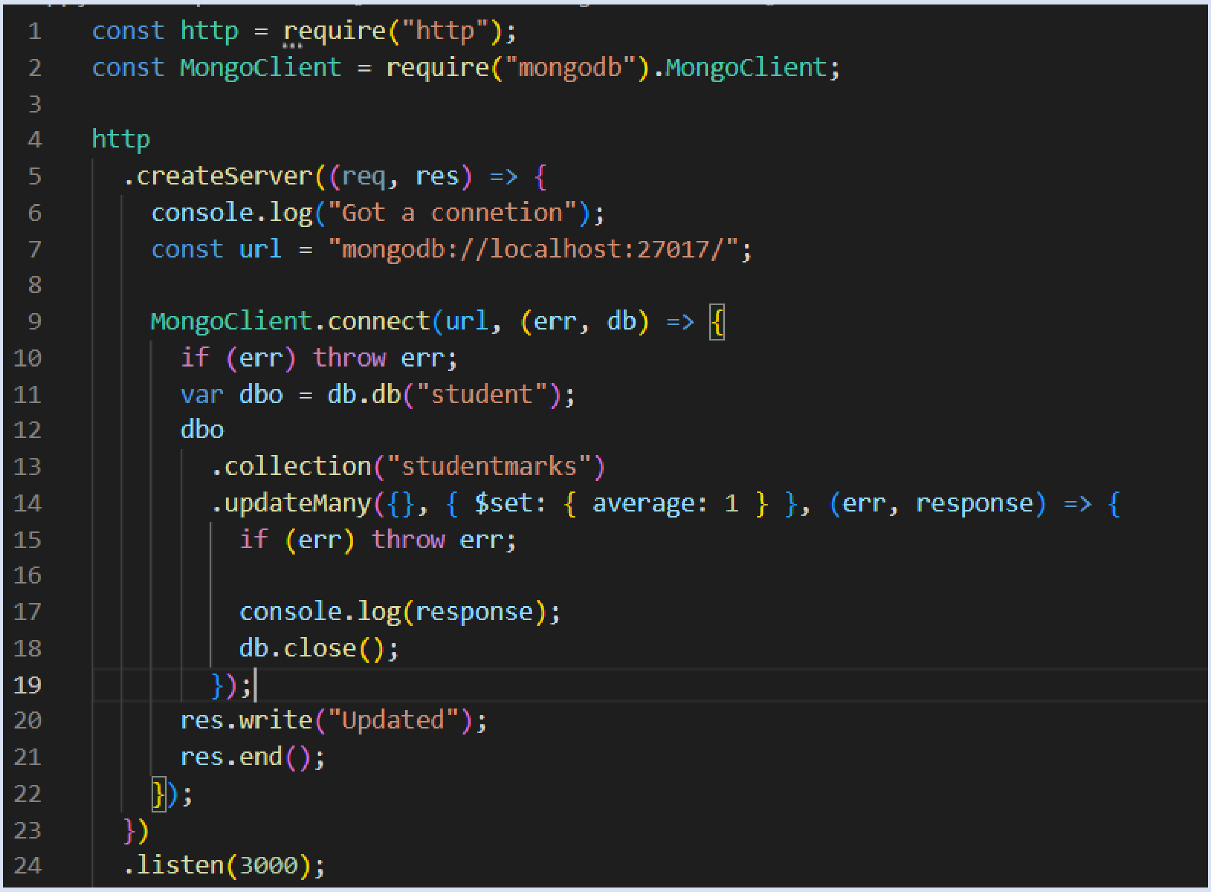


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      var query = { maths_marks: { $gt: 50 } };

      dbo
        .collection("studentmarks")
        .find(query, { projection: { _id: 0, name: 1 } })
        .toArray((err, result) => {
          if (err) throw err;
          console.log(result);
          db.close();
          res.end();
        });
    });
  })
  .listen(3000);
```

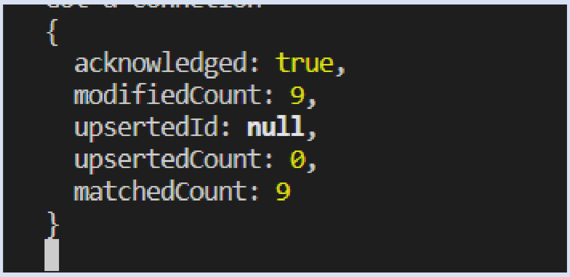


```
[
  { name: 'Mala' },
  { name: 'Vanu' },
  { name: 'Aruli' },
  { name: 'Shayu' },
  { name: 'Lucky' },
  { name: 'Gva' }
]
```

6. Add a new column(field) for Average for all students.

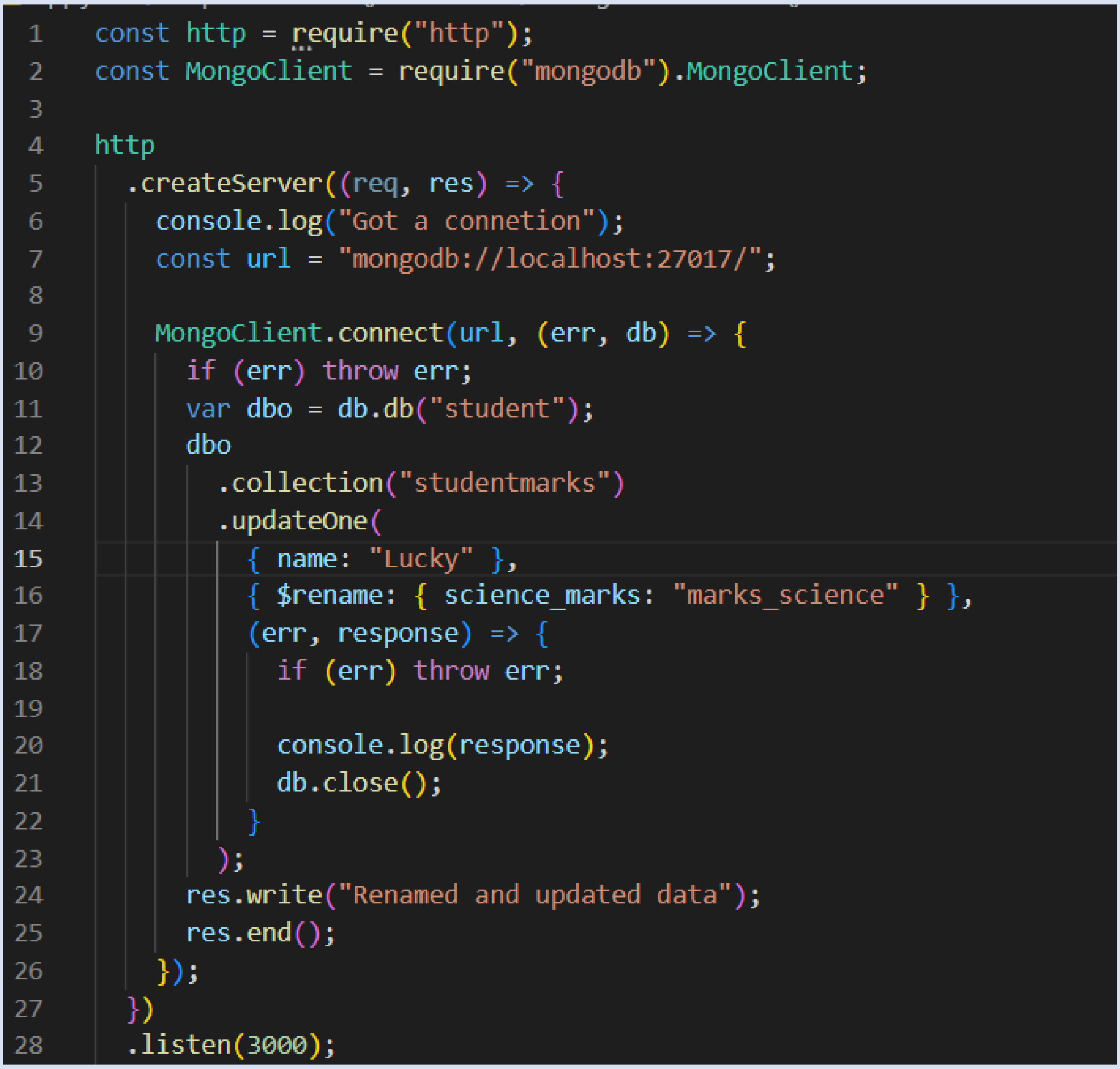


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      dbo
        .collection("studentmarks")
        .updateMany({}, { $set: { average: 1 } }, (err, response) => {
          if (err) throw err;

          console.log(response);
          db.close();
        });
      res.write("Updated");
      res.end();
    });
  })
  .listen(3000);
```

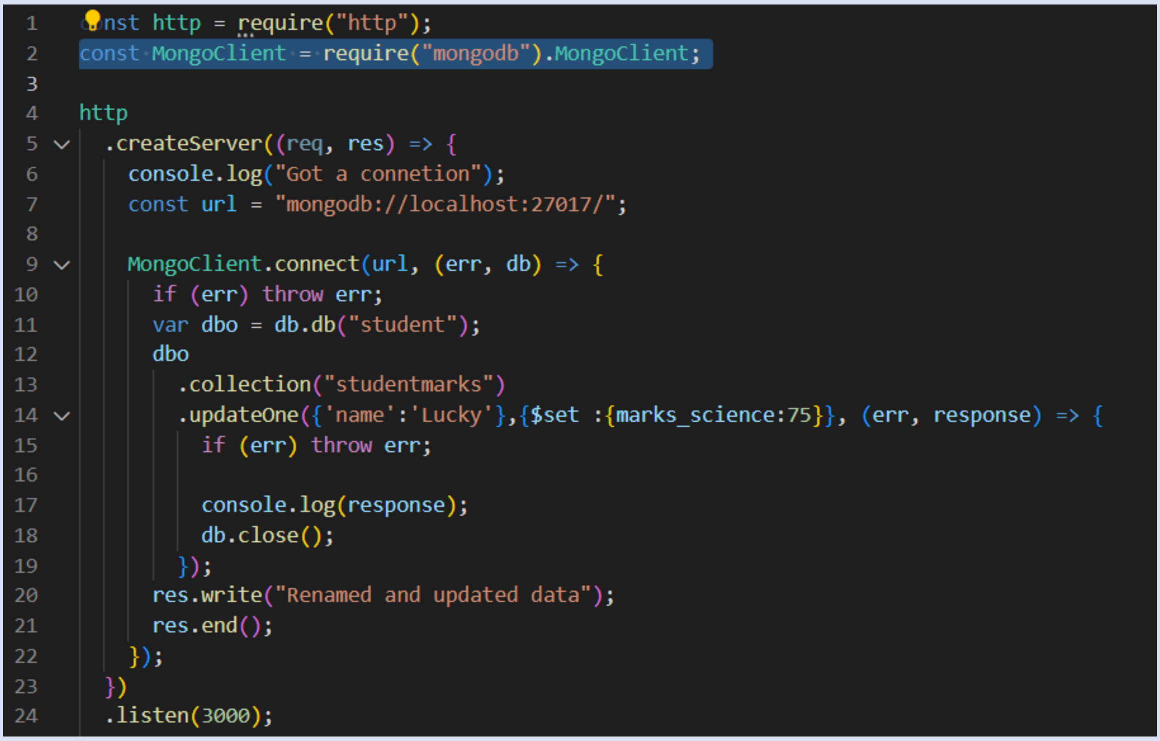


```
{
  acknowledged: true,
  modifiedCount: 9,
  upsertedId: null,
  upsertedCount: 0,
  matchedCount: 9
}
```

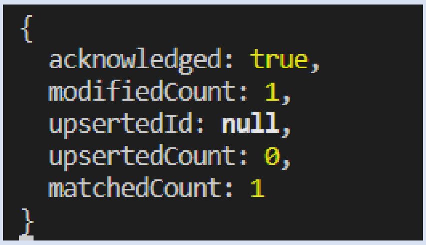
7. Update Marks_Science=75 to Lucky



```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      dbo
        .collection("studentmarks")
        .updateOne(
          { name: "Lucky" },
          { $rename: { science_marks: "marks_science" } },
          (err, response) => {
            if (err) throw err;

            console.log(response);
            db.close();
          }
        );
      res.write("Renamed and updated data");
      res.end();
    });
  })
  .listen(3000);
```

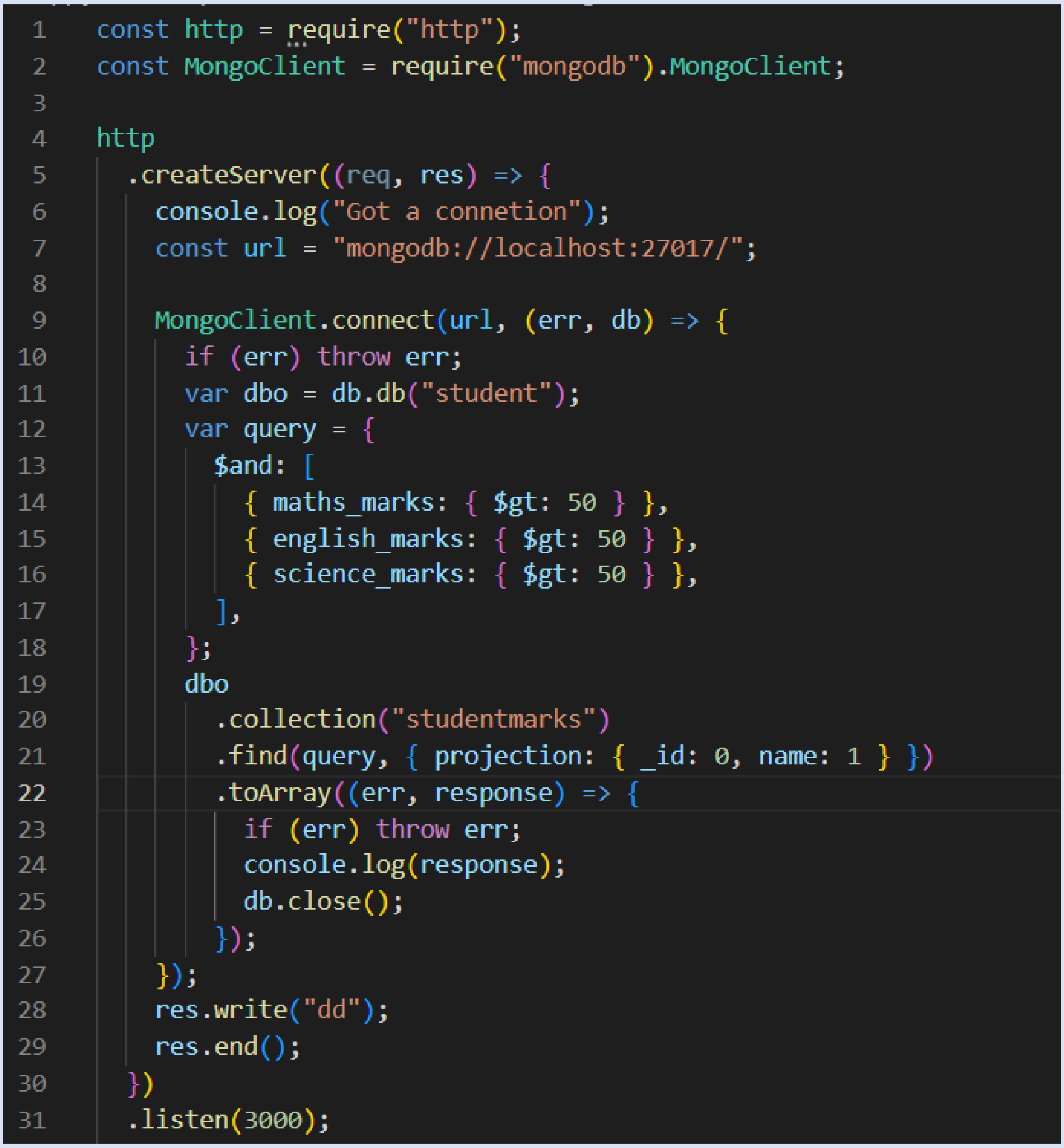


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      dbo
        .collection("studentmarks")
        .updateOne({'name':'Lucky'},{$set :{marks_science:75}}, (err, response) => {
          if (err) throw err;

          console.log(response);
          db.close();
        });
      res.write("Renamed and updated data");
      res.end();
    });
  })
  .listen(3000);
```

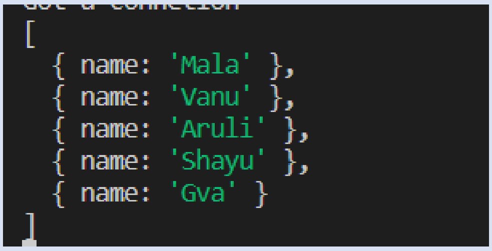


```
{
  acknowledged: true,
  modifiedCount: 1,
  upsertedId: null,
  upsertedCount: 0,
  matchedCount: 1
}
```

8. List the names who got more than 50 marks in all subjects.

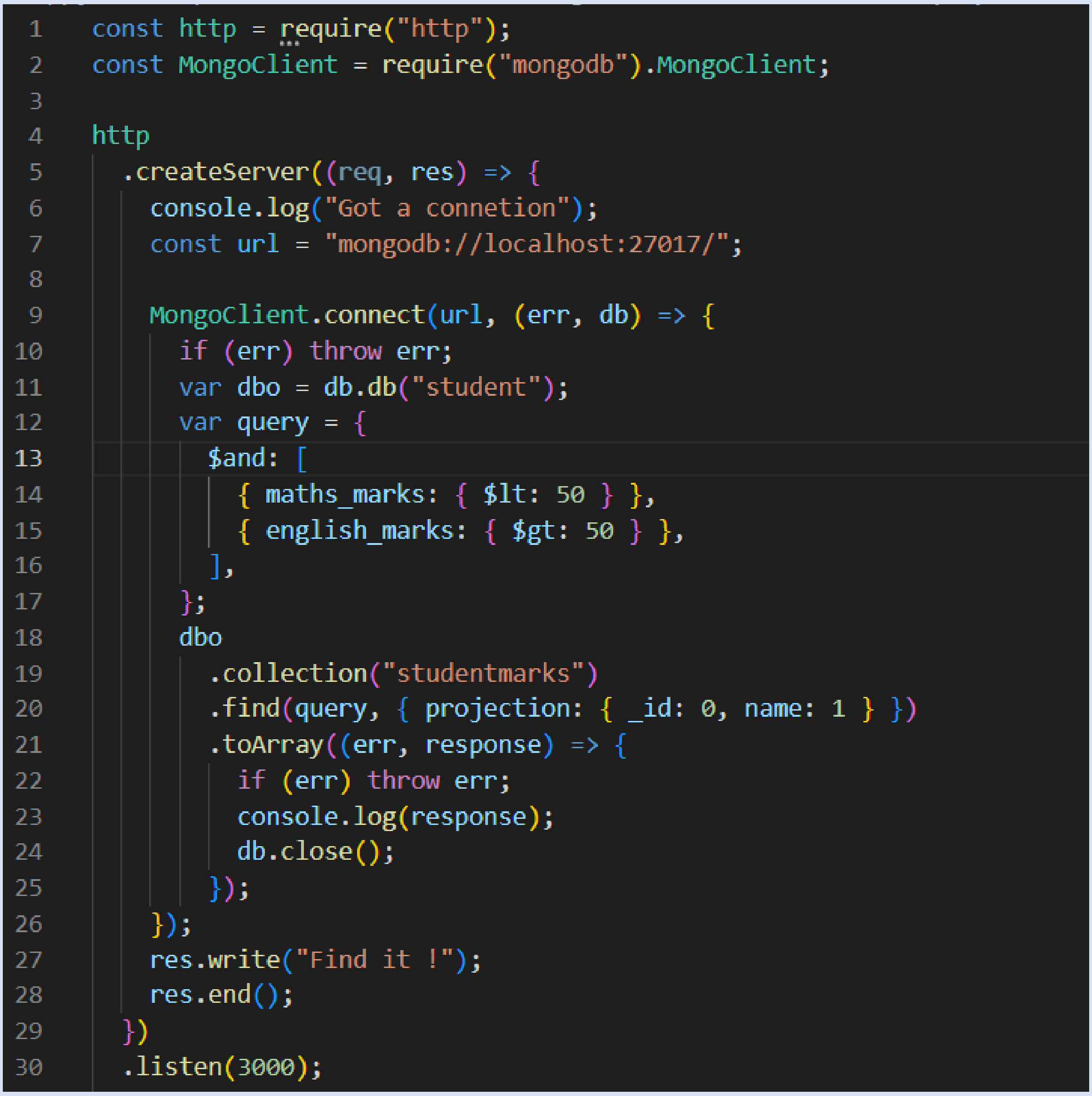


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      var query = {
        $and: [
          { maths_marks: { $gt: 50 } },
          { english_marks: { $gt: 50 } },
          { science_marks: { $gt: 50 } },
        ],
      };
      dbo
        .collection("studentmarks")
        .find(query, { projection: { _id: 0, name: 1 } })
        .toArray((err, response) => {
          if (err) throw err;
          console.log(response);
          db.close();
        });
    });
    res.write("dd");
    res.end();
  })
  .listen(3000);
```

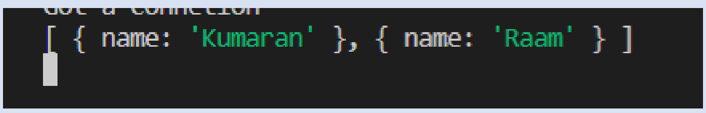


```
[
  { name: 'Mala' },
  { name: 'Vanu' },
  { name: 'Aruli' },
  { name: 'Shayu' },
  { name: 'Gva' }
]
```

9. List the names who got less than 50 marks in Maths subject and more than 50 marks in English

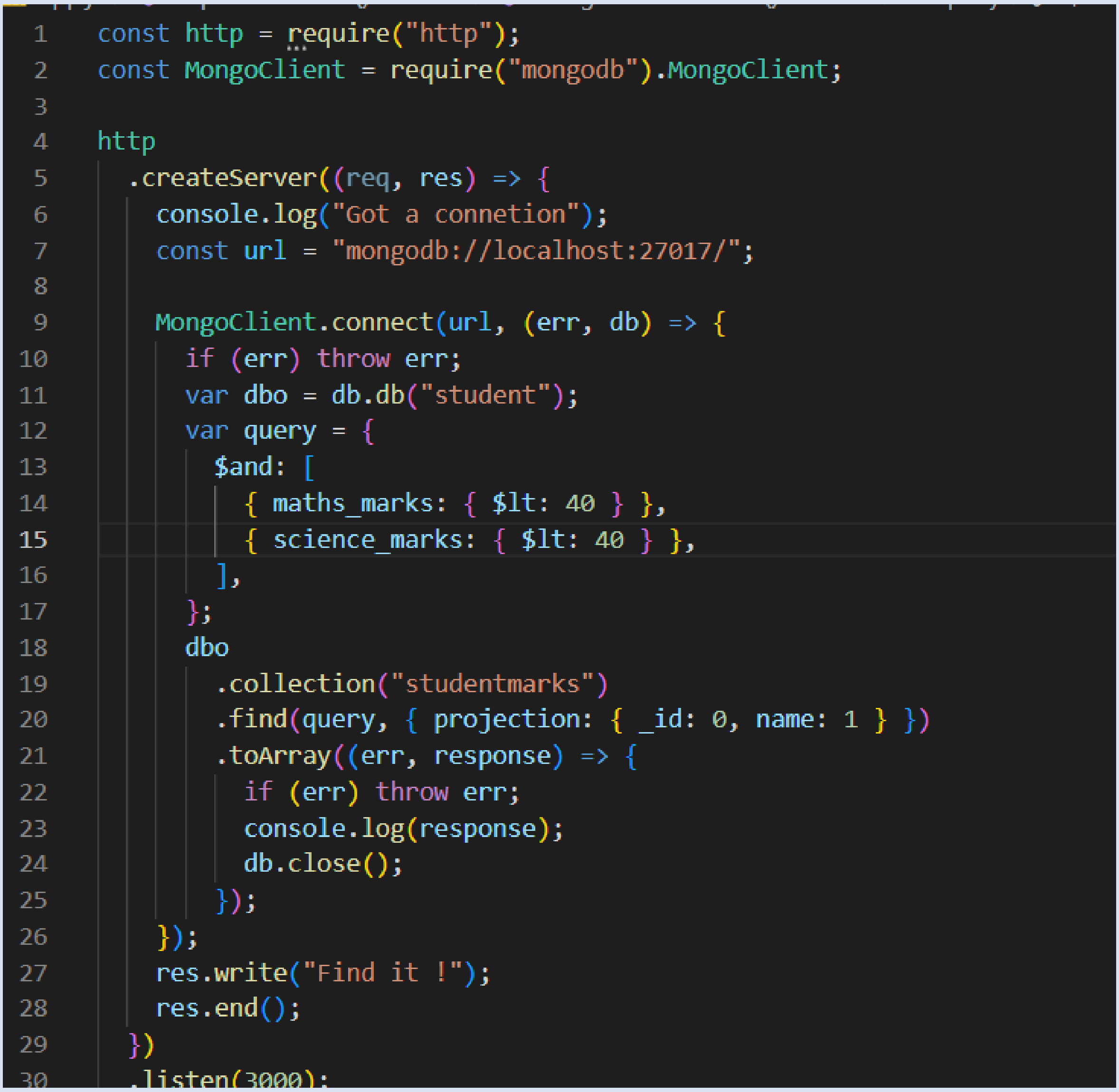


```javascript
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      var query = {
        $and: [
          { maths_marks: { $lt: 50 } },
          { english_marks: { $gt: 50 } },
        ],
      };
      dbo
        .collection("studentmarks")
        .find(query, { projection: { _id: 0, name: 1 } })
        .toArray((err, response) => {
          if (err) throw err;
          console.log(response);
          db.close();
        });
    });
    res.write("Find it !");
    res.end();
  })
  .listen(3000);
```

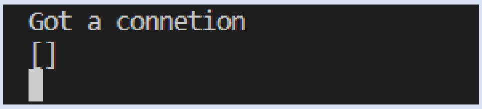


```
Got a connetion
[ { name: 'Kumaran' }, { name: 'Raam' } ]
```

10. List the names who got less than 40 in both Maths and Science.

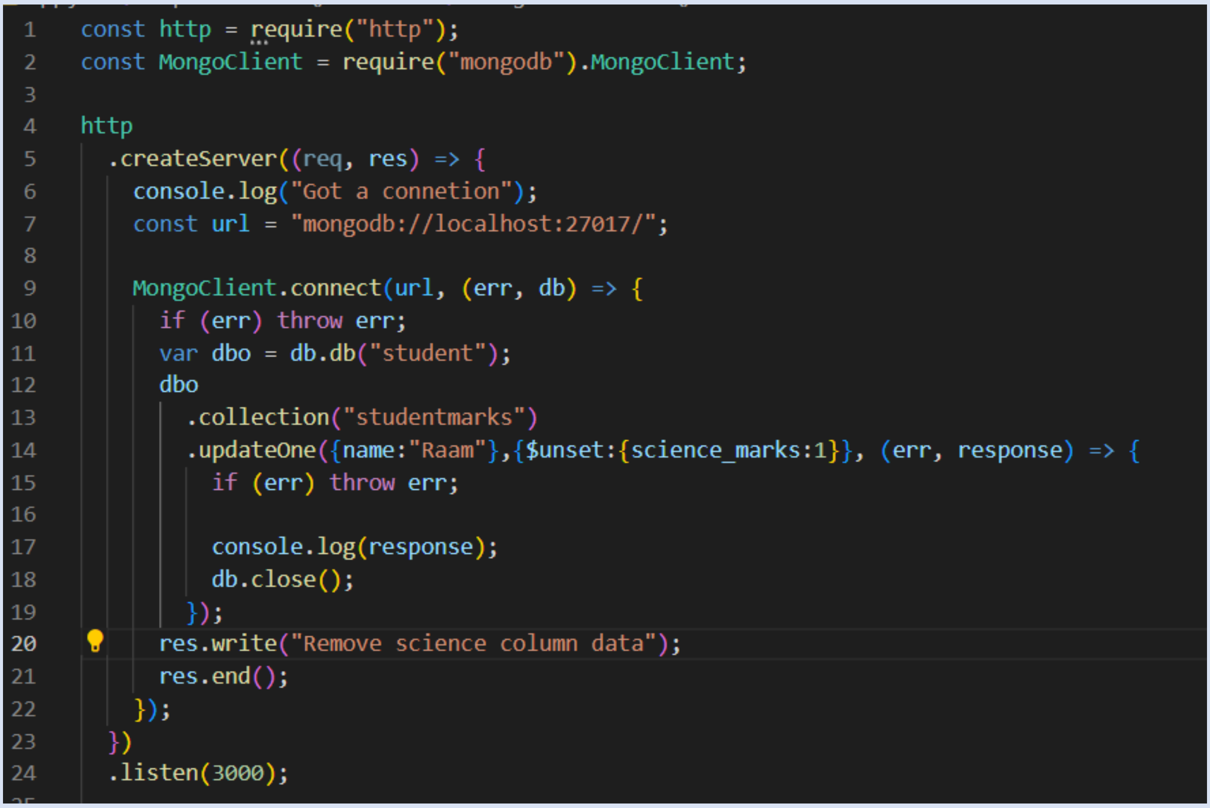


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      var query = {
        $and: [
          { maths_marks: { $lt: 40 } },
          { science_marks: { $lt: 40 } },
        ],
      };
      dbo
        .collection("studentmarks")
        .find(query, { projection: { _id: 0, name: 1 } })
        .toArray((err, response) => {
          if (err) throw err;
          console.log(response);
          db.close();
        });
    });
    res.write("Find it !");
    res.end();
  })
  .listen(3000);
```

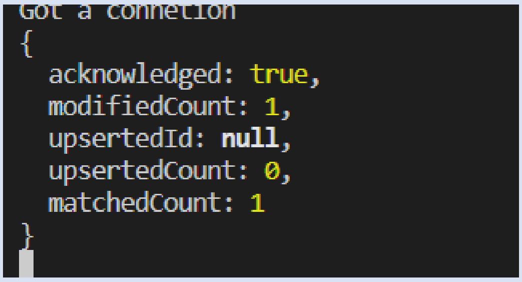


```
Got a connetion
[]
```

11. List the names who got less than 40 in both Maths and Science.

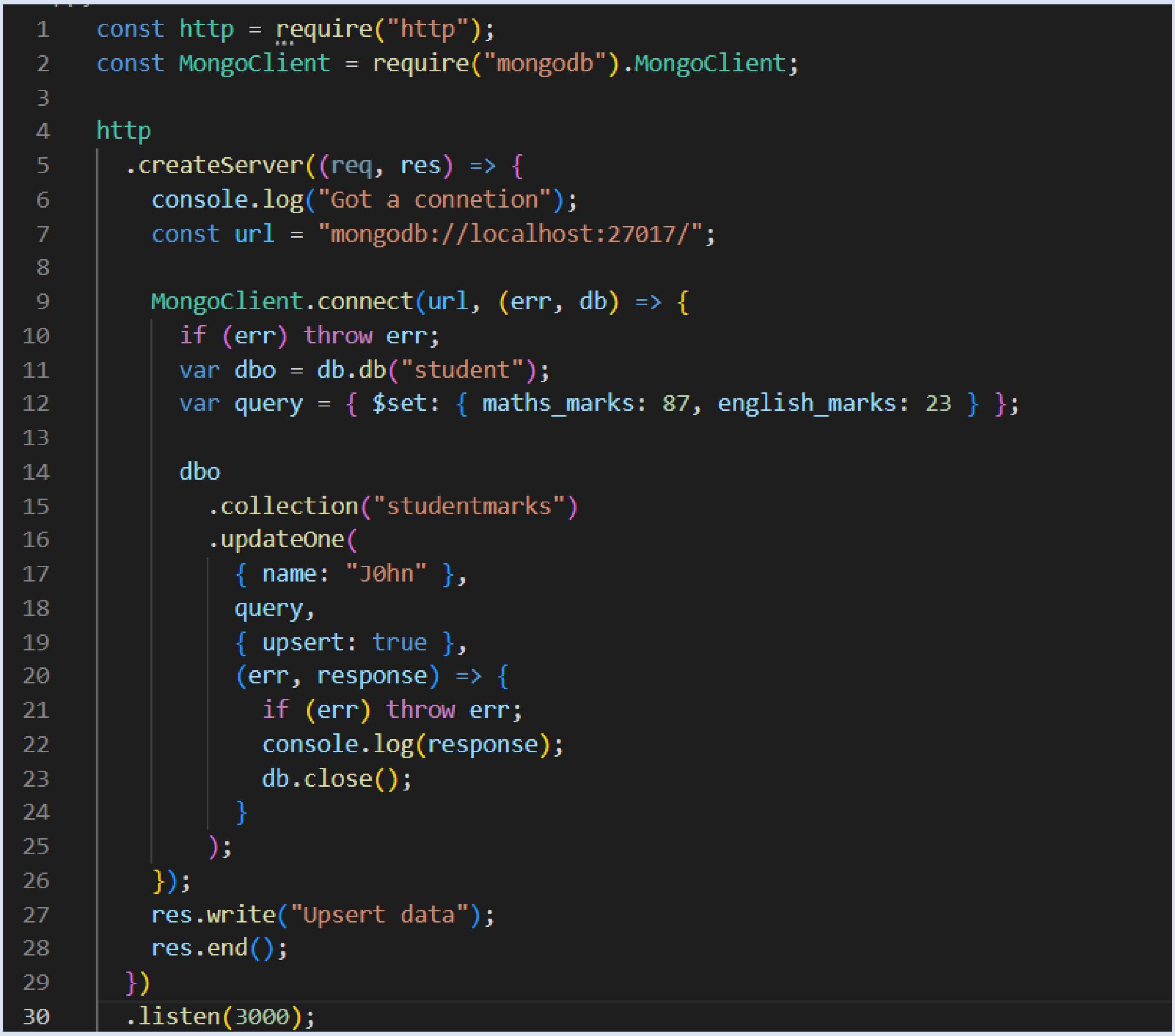


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      dbo
        .collection("studentmarks")
        .updateOne({name:"Raam"},{$unset:{science_marks:1}}, (err, response) => {
          if (err) throw err;

          console.log(response);
          db.close();
        });
      res.write("Remove science column data");
      res.end();
    });
  })
  .listen(3000);
```

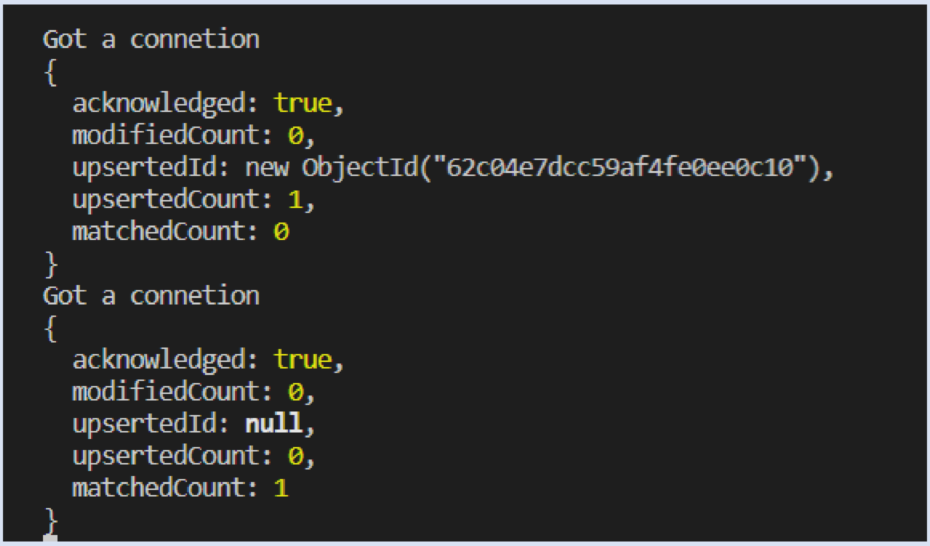


```
Got a connetion
{
  acknowledged: true,
  modifiedCount: 1,
  upsertedId: null,
  upsertedCount: 0,
  matchedCount: 1
}
```

12. Update John’s Math mark as 87 and English mark as 23, if john not available upsert.

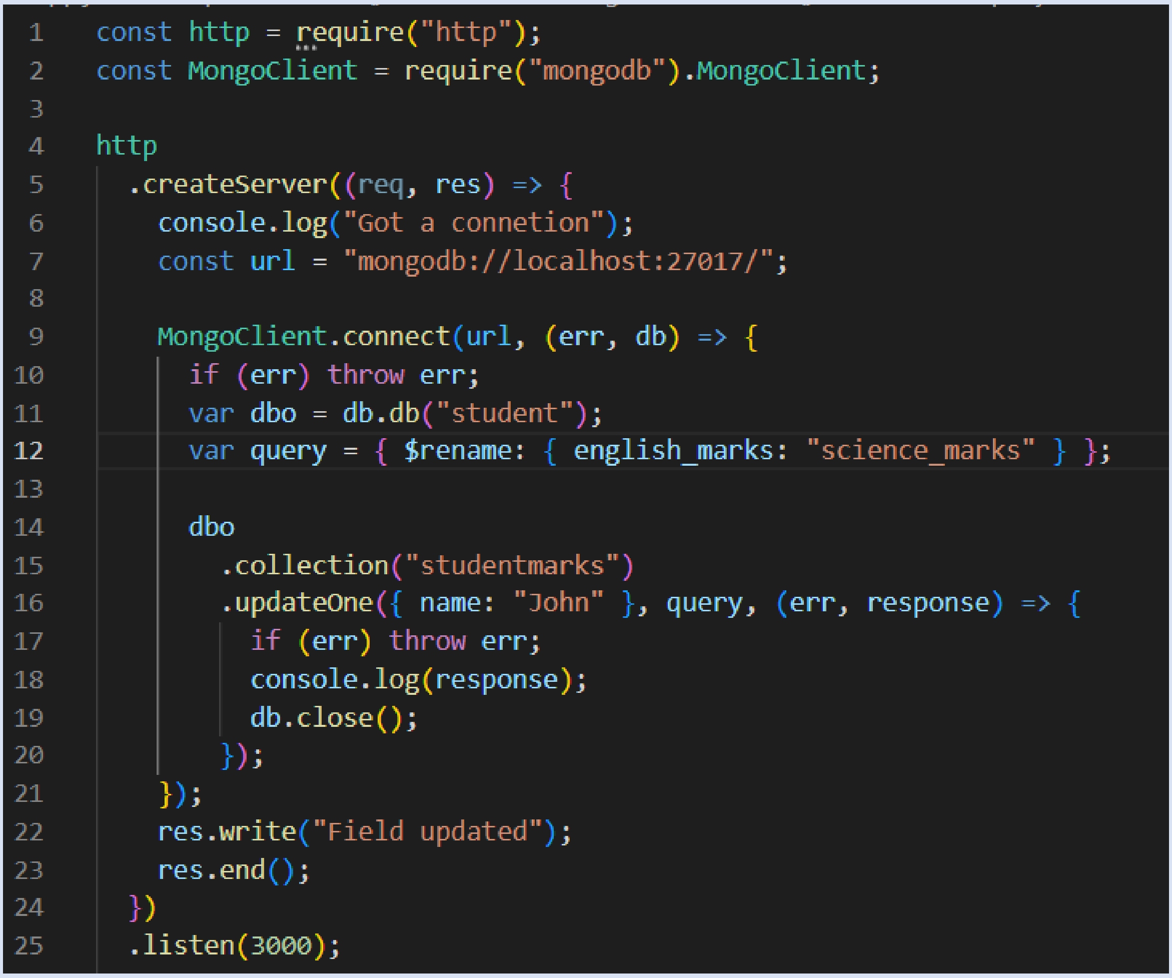


```javascript
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      var query = { $set: { maths_marks: 87, english_marks: 23 } };

      dbo
        .collection("studentmarks")
        .updateOne(
          { name: "J0hn" },
          query,
          { upsert: true },
          (err, response) => {
            if (err) throw err;
            console.log(response);
            db.close();
          }
        );
    });
    res.write("Upsert data");
    res.end();
  })
  .listen(3000);
```

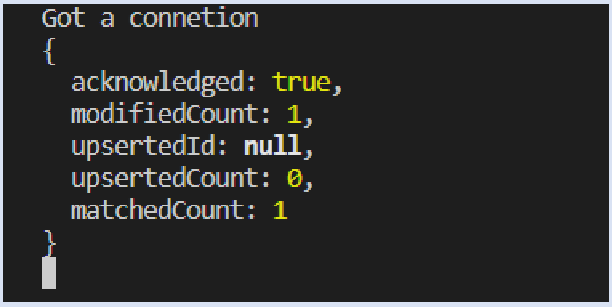


```
Got a connetion
{
  acknowledged: true,
  modifiedCount: 0,
  upsertedId: new ObjectId("62c04e7dcc59af4fe0ee0c10"),
  upsertedCount: 1,
  matchedCount: 0
}
Got a connetion
{
  acknowledged: true,
  modifiedCount: 0,
  upsertedId: null,
  upsertedCount: 0,
  matchedCount: 1
}
```

13. Rename the english_marks column/field for John to science_marks

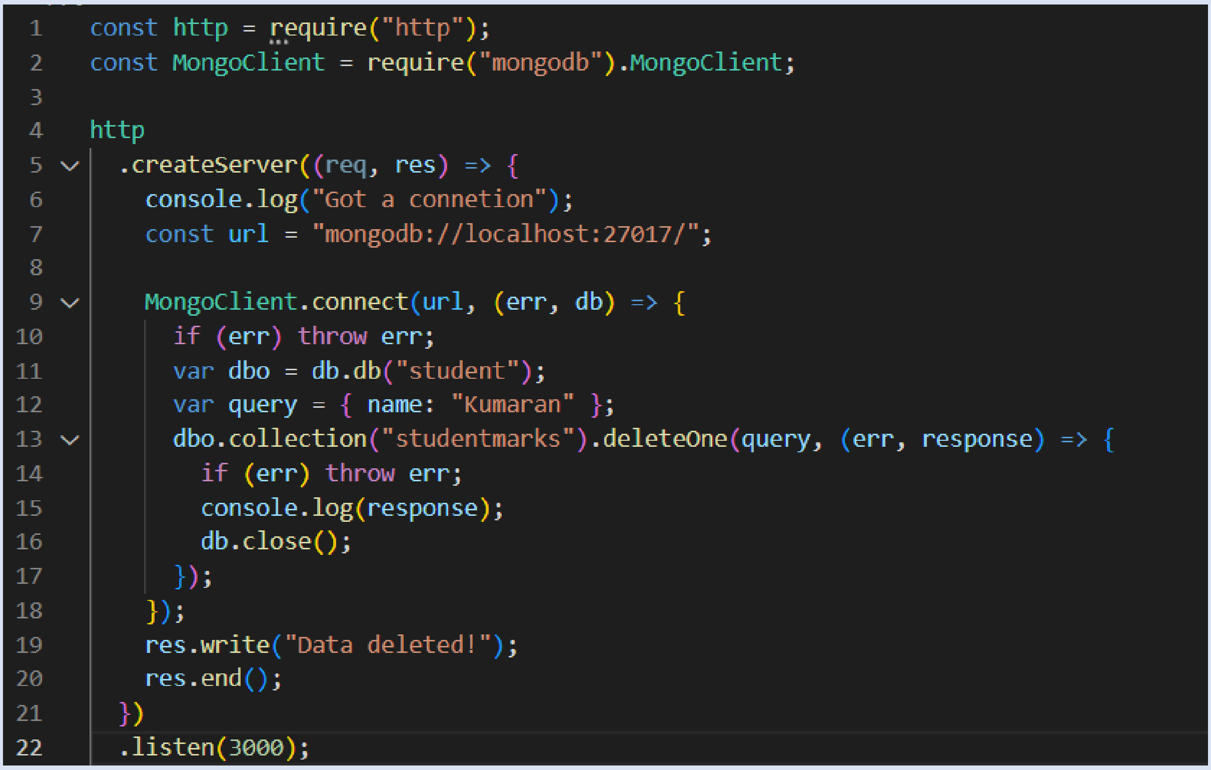


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      var query = { $rename: { english_marks: "science_marks" } };

      dbo
        .collection("studentmarks")
        .updateOne({ name: "John" }, query, (err, response) => {
          if (err) throw err;
          console.log(response);
          db.close();
        });
    });
    res.write("Field updated");
    res.end();
  })
  .listen(3000);
```

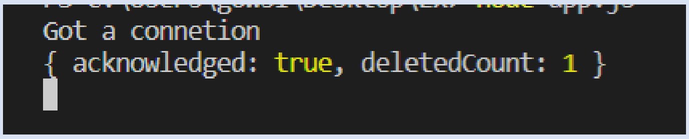


```
Got a connetion
{
  acknowledged: true,
  modifiedCount: 1,
  upsertedId: null,
  upsertedCount: 0,
  matchedCount: 1
}
```

14. Remove Kumaran’s document from collection

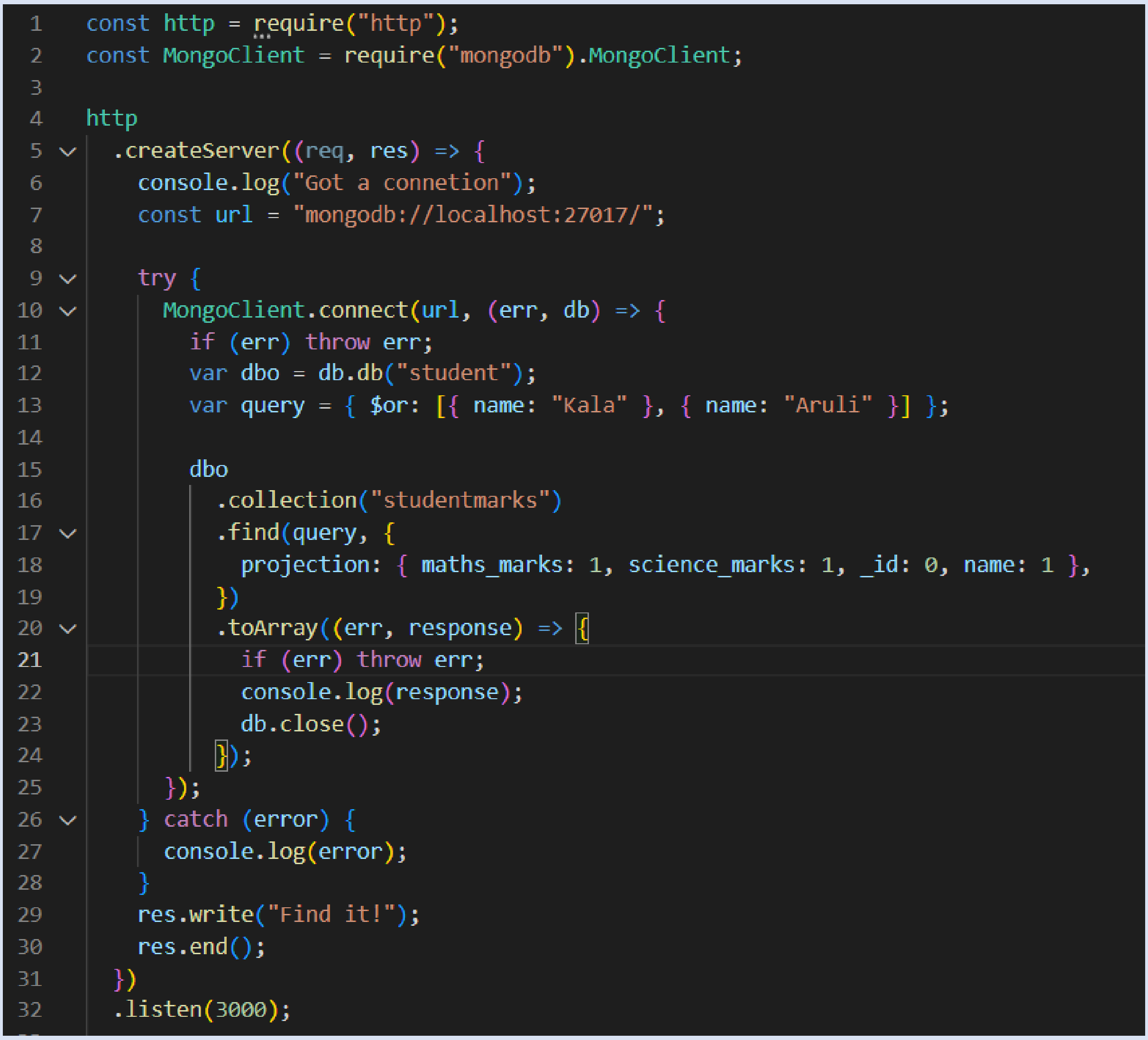


```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    MongoClient.connect(url, (err, db) => {
      if (err) throw err;
      var dbo = db.db("student");
      var query = { name: "Kumaran" };
      dbo.collection("studentmarks").deleteOne(query, (err, response) => {
        if (err) throw err;
        console.log(response);
        db.close();
      });
    });
    res.write("Data deleted!");
    res.end();
  })
  .listen(3000);
```

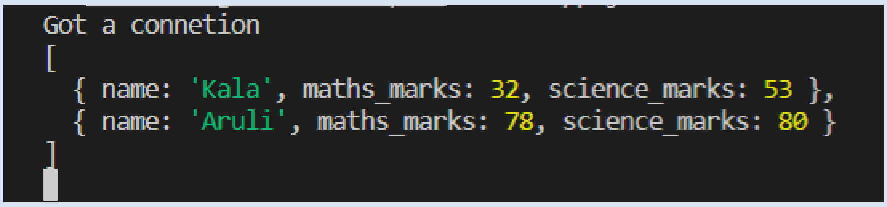


```
Got a connetion
{ acknowledged: true, deletedCount: 1 }
```

15. Find Kala’s or Aruli’s math_marks and science_marks



```
const http = require("http");
const MongoClient = require("mongodb").MongoClient;

http
  .createServer((req, res) => {
    console.log("Got a connetion");
    const url = "mongodb://localhost:27017/";

    try {
      MongoClient.connect(url, (err, db) => {
        if (err) throw err;
        var dbo = db.db("student");
        var query = { $or: [{ name: "Kala" }, { name: "Aruli" }] };

        dbo
          .collection("studentmarks")
          .find(query, {
            projection: { maths_marks: 1, science_marks: 1, _id: 0, name: 1 },
          })
          .toArray((err, response) => {
            if (err) throw err;
            console.log(response);
            db.close();
          });
      });
    } catch (error) {
      console.log(error);
    }
    res.write("Find it!");
    res.end();
  })
  .listen(3000);
```



```
Got a connetion
[
  { name: 'Kala', maths_marks: 32, science_marks: 53 },
  { name: 'Aruli', maths_marks: 78, science_marks: 80 }
]
```