

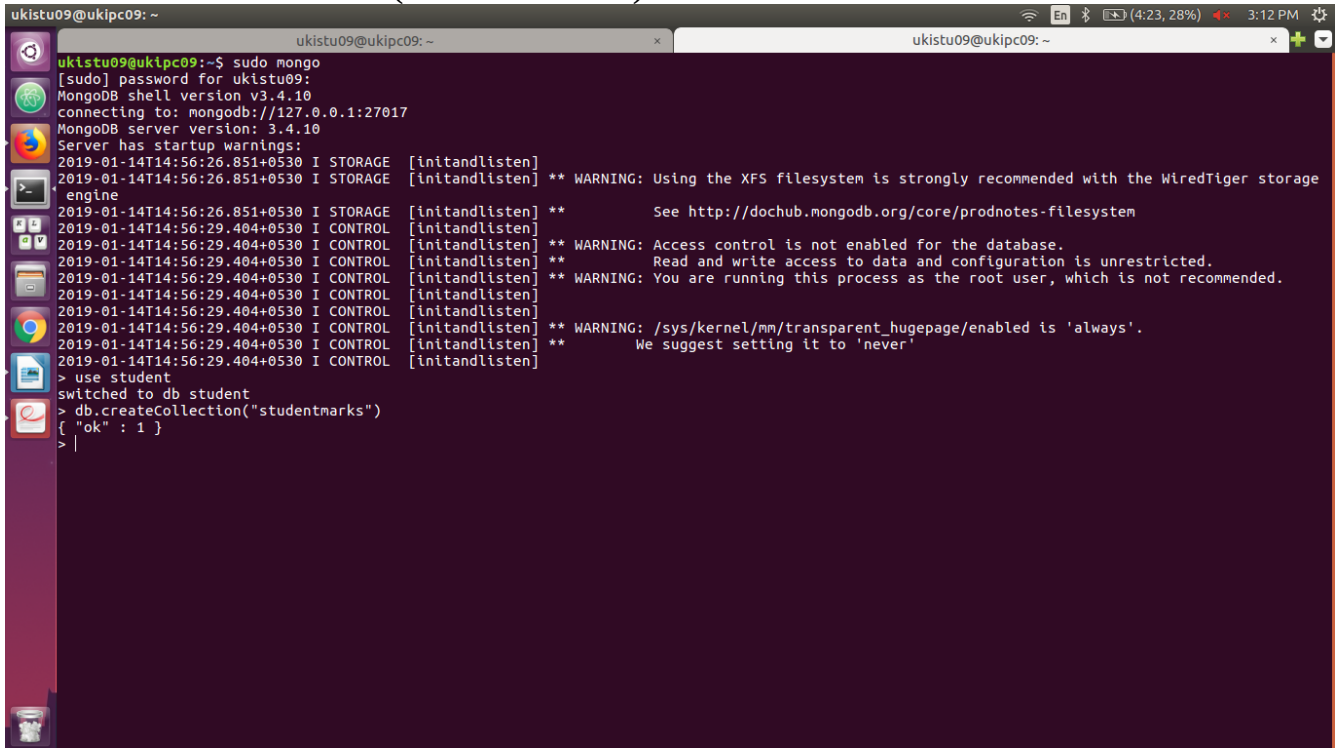
MongoDB Exercise 2

1) Create a Database called student use student

```
ukistu09@ukipc09: ~  
ukistu09@ukipc09: ~  
ukistu09@ukipc09:~$ sudo mongo  
[sudo] password for ukistu09:  
MongoDB shell version v3.4.10  
connecting to: mongodb://127.0.0.1:27017  
MongoDB server version: 3.4.10  
Server has startup warnings:  
2019-01-16T10:11:56.488+0530 I STORAGE [initandlisten]  
2019-01-16T10:11:56.488+0530 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage  
engine  
2019-01-16T10:11:56.488+0530 I STORAGE [initandlisten] **  
2019-01-16T10:12:11.131+0530 I CONTROL [initandlisten] See http://dochub.mongodb.org/core/prodnotes-filesystem  
2019-01-16T10:12:11.131+0530 I CONTROL [initandlisten] **  
2019-01-16T10:12:11.131+0530 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.  
2019-01-16T10:12:11.131+0530 I CONTROL [initandlisten] ** Read and write access to data and configuration is unrestricted.  
2019-01-16T10:12:11.131+0530 I CONTROL [initandlisten] ** WARNING: You are running this process as the root user, which is not recommended.  
2019-01-16T10:12:11.131+0530 I CONTROL [initandlisten] **  
2019-01-16T10:12:11.132+0530 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.  
2019-01-16T10:12:11.132+0530 I CONTROL [initandlisten] ** We suggest setting it to 'never'  
2019-01-16T10:12:11.132+0530 I CONTROL [initandlisten] **  
> use student  
switched to db student  
> |
```

2) Create a collection called studentmarks

`db.createCollection("studentmarks")`



```
ukistu09@ukipc09: ~  
ukistu09@ukipc09: ~  
ukistu09@ukipc09:~$ sudo mongo  
[sudo] password for ukistu09:  
MongoDB shell version v3.4.10  
connecting to: mongodb://127.0.0.1:27017  
MongoDB server version: 3.4.10  
Server has startup warnings:  
2019-01-14T14:56:26.851+0530 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine  
2019-01-14T14:56:26.851+0530 I STORAGE [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** Read and write access to data and configuration is unrestricted.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: You are running this process as the root user, which is not recommended.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten]  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten]  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] We suggest setting it to 'never'  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten]  
> use student  
switched to db student  
> db.createCollection("studentmarks")  
{ "ok" : 1 }  
> |
```

3) Create the documents listed in above table.

```
db.studentmarks.insert({name: "Mala", maths_marks:45,english_marks:53,science_marks:72})
```

```
db.studentmarks.insert({name: "Vanu", maths_marks:80,english_marks:75,science_marks:85})
```

```
db.studentmarks.insert({name: "Kala", maths_marks:32,english_marks:40,science_marks:53})
```

```
db.studentmarks.insert({name: "Aruli", maths_marks:78,english_marks:85,science_marks:80})
```

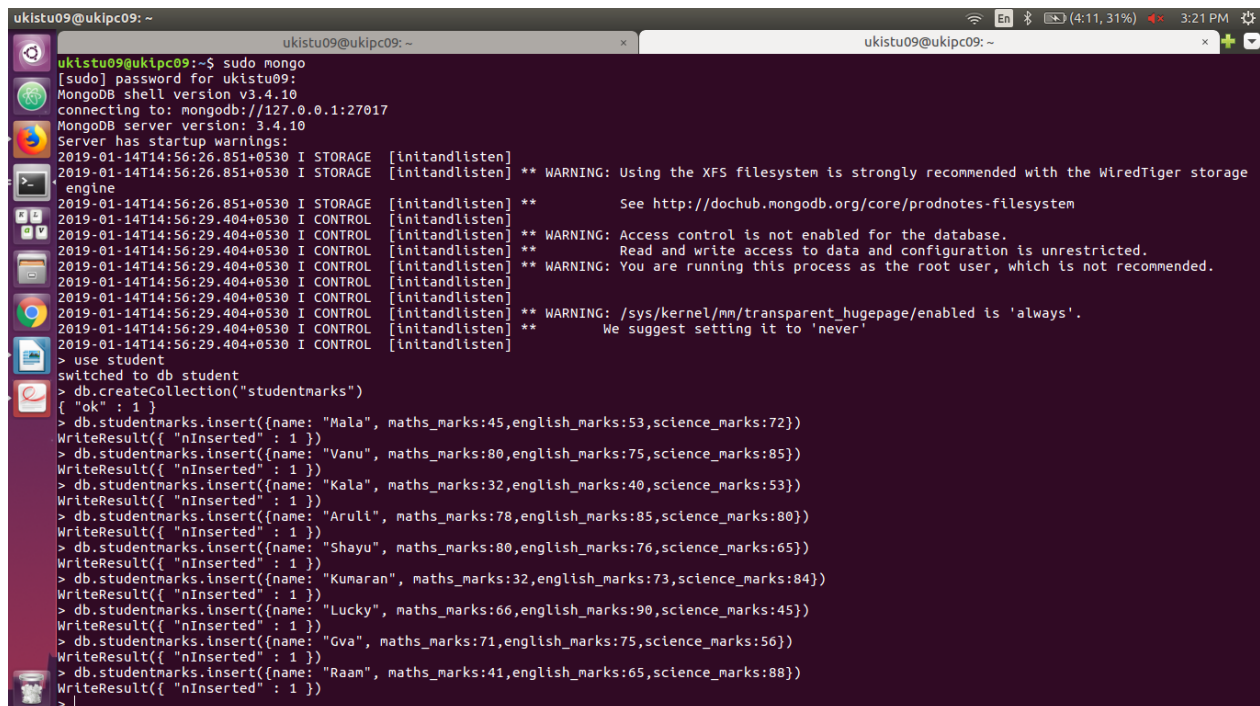
```
db.studentmarks.insert({name: "Shayu", maths_marks:80,english_marks:76,science_marks:65})
```

```
db.studentmarks.insert({name: "Kumaran", maths_marks:32,english_marks:73,science_marks:84})
```

```
db.studentmarks.insert({name: "Lucky", maths_marks:66,english_marks:90,science_marks:45})
```

```
db.studentmarks.insert({name: "Gva", maths_marks:71,english_marks:75,science_marks:56})
```

```
db.studentmarks.insert({name: "Raam", maths_marks:41,english_marks:65,science_marks:88})
```

A screenshot of a terminal window on a Linux system. The terminal shows the user 'ukistu09' at the prompt '~'. They run 'sudo mongo', which prompts for a password. After entering the password, the MongoDB shell version 'v3.4.10' is displayed. The user connects to 'mongodb://127.0.0.1:27017'. The MongoDB server version '3.4.10' is shown. Several startup warnings are displayed, including a recommendation to use the XFS filesystem and a warning about access control. The user then switches to the 'student' database and creates a collection named 'studentmarks'. Finally, they insert ten documents into the 'studentmarks' collection, each representing a student's marks in maths, english, and science. Each insertion is confirmed with a 'WriteResult' message showing 'nInserted: 1'.

```
ukistu09@ukipc09: ~  
ukistu09@ukipc09: ~  
ukistu09@ukipc09:~$ sudo mongo  
[sudo] password for ukistu09:  
MongoDB shell version v3.4.10  
connecting to: mongodb://127.0.0.1:27017  
MongoDB server version: 3.4.10  
Server has startup warnings:  
2019-01-14T14:56:26.851+0530 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine  
2019-01-14T14:56:26.851+0530 I STORAGE [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: Read and write access to data and configuration is unrestricted.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: You are running this process as the root user, which is not recommended.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] **  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** We suggest setting it to 'never'  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] **  
> use student  
switched to db student  
> db.createCollection("studentmarks")  
{ "ok" : 1 }  
> db.studentmarks.insert({name: "Mala", maths_marks:45,english_marks:53,science_marks:72})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Vanu", maths_marks:80,english_marks:75,science_marks:85})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Kala", maths_marks:32,english_marks:40,science_marks:53})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Aruli", maths_marks:78,english_marks:85,science_marks:80})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Shayu", maths_marks:80,english_marks:76,science_marks:65})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Kumaran", maths_marks:32,english_marks:73,science_marks:84})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Lucky", maths_marks:66,english_marks:90,science_marks:45})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Gva", maths_marks:71,english_marks:75,science_marks:56})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Raam", maths_marks:41,english_marks:65,science_marks:88})  
WriteResult({ "nInserted" : 1 })  
>
```

4) Increase the maths marks of Mala by 6 marks

```
db.studentmarks.update({name : "Mala"},{$inc:{maths_marks:6}})
```

```
ukistu09@ukipc09: ~  
MongoDB server version: 3.4.10  
Server has startup warnings:  
2019-01-14T14:56:26.851+0530 I STORAGE [initandlisten]  
2019-01-14T14:56:26.851+0530 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage  
engine  
2019-01-14T14:56:26.851+0530 I STORAGE [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten]  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: Read and write access to data and configuration is unrestricted.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: You are running this process as the root user, which is not recommended.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten]  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten]  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten]  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten] ** We suggest setting it to 'never'  
2019-01-14T14:56:29.404+0530 I CONTROL [initandlisten]  
> use student  
switched to db student  
> db.createCollection("studentmarks")  
{ "ok" : 1 }  
> db.studentmarks.insert({name: "Mala", maths_marks:45,english_marks:53,science_marks:72})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Vanu", maths_marks:80,english_marks:75,science_marks:85})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Kala", maths_marks:32,english_marks:40,science_marks:53})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Aruli", maths_marks:78,english_marks:85,science_marks:80})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Shayu", maths_marks:80,english_marks:76,science_marks:65})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Kumaran", maths_marks:32,english_marks:73,science_marks:84})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Lucky", maths_marks:66,english_marks:90,science_marks:45})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Gva", maths_marks:71,english_marks:75,science_marks:56})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.insert({name: "Raam", maths_marks:41,english_marks:65,science_marks:88})  
WriteResult({ "nInserted" : 1 })  
> db.studentmarks.update({name : "Mala"},{$inc:{maths_marks:6}})  
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })  
>  
>  
>
```

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


```
db.studentmarks.find( { maths_marks: { $gt: 50 } } )
```

```
ukistu09@ukipc09: ~
ukistu09@ukipc09: ~
ukistu09@ukipc09: ~

}
{
  "science_marks" : 65
}
{
  "_id" : ObjectId("5c3c5ae8c8ced0abb6425f90"),
  "name" : "Kunaran",
  "maths_marks" : 32,
  "english_marks" : 73,
  "science_marks" : 84
}
{
  "_id" : ObjectId("5c3c5b04c8ced0abb6425f91"),
  "name" : "Lucky",
  "maths_marks" : 66,
  "english_marks" : 90,
  "science_marks" : 45
}
{
  "_id" : ObjectId("5c3c5b19c8ced0abb6425f92"),
  "name" : "Gva",
  "maths_marks" : 71,
  "english_marks" : 75,
  "science_marks" : 56
}
{
  "_id" : ObjectId("5c3c5b44c8ced0abb6425f93"),
  "name" : "Raam",
  "maths_marks" : 41,
  "english_marks" : 65,
  "science_marks" : 88
}
}
>
>
> db.studentmarks.find( { maths_marks: { $gt: 50 } } )
{ "_id" : ObjectId("5c3c5a23c8ced0abb6425f8b"), "name" : "Mala", "maths_marks" : 51, "english_marks" : 53, "science_marks" : 72 }
{ "_id" : ObjectId("5c3c5a6ec8ced0abb6425f8c"), "name" : "Vanu", "maths_marks" : 80, "english_marks" : 75, "science_marks" : 85 }
{ "_id" : ObjectId("5c3c5aa9c8ced0abb6425f8e"), "name" : "Aruli", "maths_marks" : 78, "english_marks" : 85, "science_marks" : 80 }
{ "_id" : ObjectId("5c3c5ac7c8ced0abb6425f8f"), "name" : "Shayu", "maths_marks" : 80, "english_marks" : 76, "science_marks" : 65 }
{ "_id" : ObjectId("5c3c5b04c8ced0abb6425f91"), "name" : "Lucky", "maths_marks" : 66, "english_marks" : 90, "science_marks" : 45 }
{ "_id" : ObjectId("5c3c5b19c8ced0abb6425f92"), "name" : "Gva", "maths_marks" : 71, "english_marks" : 75, "science_marks" : 56 }
```

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

```
db.studentmarks.update({},{$set:{"Average":"null"}},{multi:true})
```

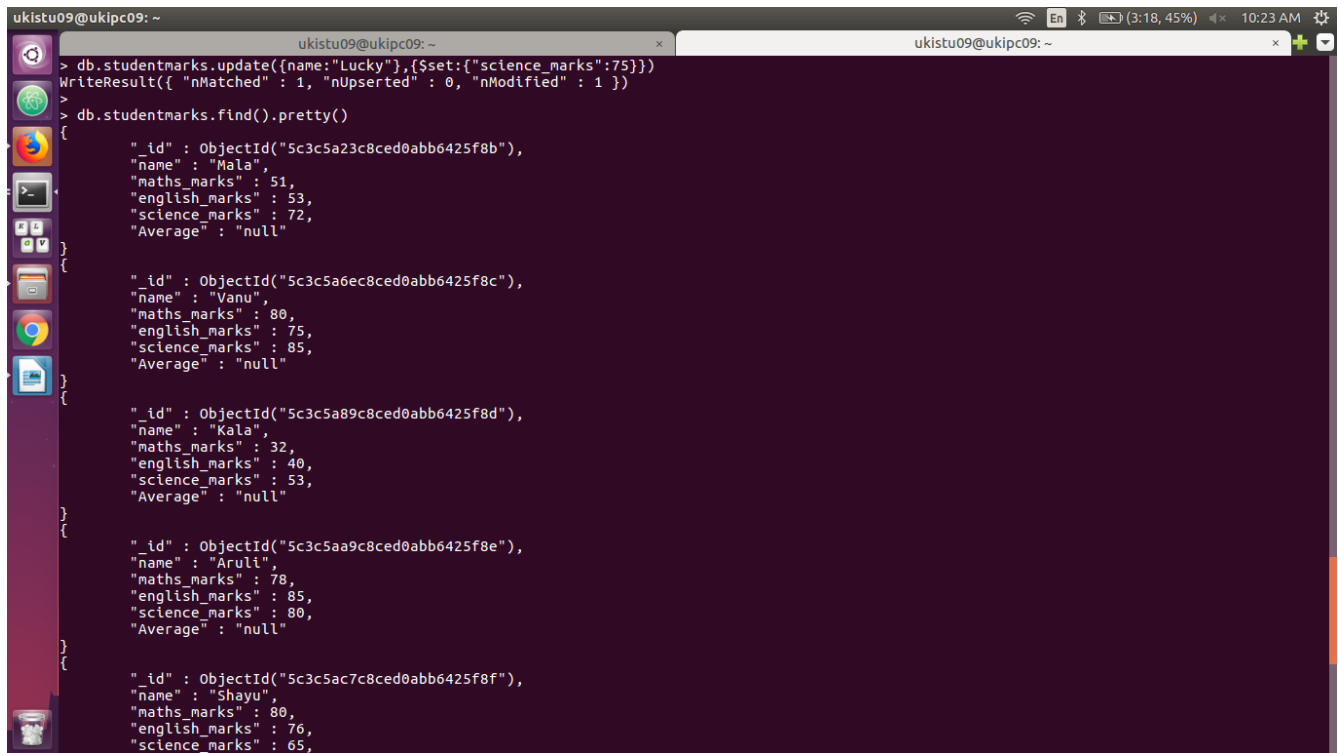


The image shows a terminal window with a dark purple background. The prompt is 'ukistu09@ukipc09: ~'. The terminal displays the following content:

```
{
  "_id" : ObjectId("5c3c5b04c8ced0abb6425f91"),
  "name" : "Lucky",
  "maths_marks" : 66,
  "english_marks" : 90,
  "science_marks" : 45
}
{
  "_id" : ObjectId("5c3c5b19c8ced0abb6425f92"),
  "name" : "Gva",
  "maths_marks" : 71,
  "english_marks" : 75,
  "science_marks" : 56
}
{
  "_id" : ObjectId("5c3c5b44c8ced0abb6425f93"),
  "name" : "Raam",
  "maths_marks" : 41,
  "english_marks" : 65,
  "science_marks" : 88
}
> db.studentmarks.update({},{$set:{"Average":"null"}},{multi:true});
WriteResult({ "nMatched" : 9, "nUpserted" : 0, "nModified" : 9 })
>
> db.studentmarks.find().pretty()
{
  "_id" : ObjectId("5c3c5a23c8ced0abb6425f8b"),
  "name" : "Mala",
  "maths_marks" : 51,
  "english_marks" : 53,
  "science_marks" : 72,
  "Average" : "null"
}
{
  "_id" : ObjectId("5c3c5a6ec8ced0abb6425f8c"),
  "name" : "Vanu",
  "maths_marks" : 80,
  "english_marks" : 75,
  "science_marks" : 85,
  "Average" : "null"
}
```

7) Update Marks_Science=75 to Lucky .

```
db.studentmarks.update({name:"Lucky"},{$set:{"science_marks":75}})
```

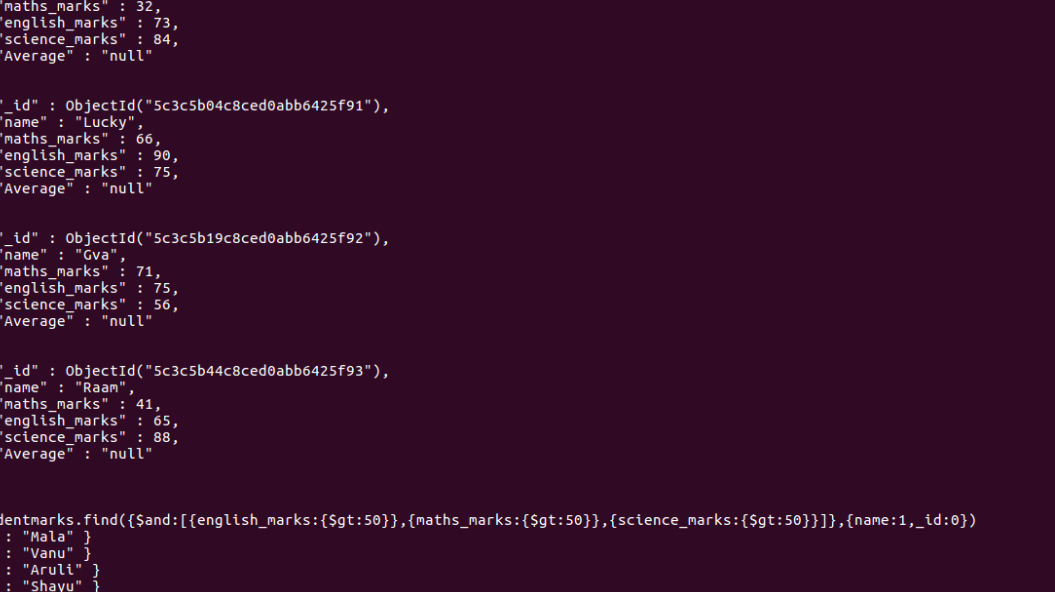


The image shows a terminal window with a dark purple background. The prompt is 'ukistu09@ukipc09: ~'. The user has entered two commands: a MongoDB update command to set 'science_marks' to 75 for the student named 'Lucky', and a find command to display all student records. The output shows five students: Mala, Vanu, Kala, Aruli, and Shayu, each with their respective marks in maths, english, and science, and an 'Average' field set to null.

```
ukistu09@ukipc09: ~  
> db.studentmarks.update({name:"Lucky"},{$set:{"science_marks":75}})  
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })  
> db.studentmarks.find().pretty()  
{  
  "_id" : ObjectId("5c3c5a23c8ced0abb6425f8b"),  
  "name" : "Mala",  
  "maths_marks" : 51,  
  "english_marks" : 53,  
  "science_marks" : 72,  
  "Average" : "null"  
}  
{  
  "_id" : ObjectId("5c3c5a6ec8ced0abb6425f8c"),  
  "name" : "Vanu",  
  "maths_marks" : 80,  
  "english_marks" : 75,  
  "science_marks" : 85,  
  "Average" : "null"  
}  
{  
  "_id" : ObjectId("5c3c5a89c8ced0abb6425f8d"),  
  "name" : "Kala",  
  "maths_marks" : 32,  
  "english_marks" : 40,  
  "science_marks" : 53,  
  "Average" : "null"  
}  
{  
  "_id" : ObjectId("5c3c5aa9c8ced0abb6425f8e"),  
  "name" : "Aruli",  
  "maths_marks" : 78,  
  "english_marks" : 85,  
  "science_marks" : 80,  
  "Average" : "null"  
}  
{  
  "_id" : ObjectId("5c3c5ac7c8ced0abb6425f8f"),  
  "name" : "Shayu",  
  "maths_marks" : 80,  
  "english_marks" : 76,  
  "science_marks" : 65,
```

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

```
db.studentmarks.find({$and:[{english_marks:{$gt:50}},{maths_marks:{$gt:50}},{science_marks:{$gt:50}}]}, {name:1,_id:0})
```



```
ukistu09@ukipc09: ~  
{"name": "Kumaran",  
 "maths_marks": 32,  
 "english_marks": 73,  
 "science_marks": 84,  
 "Average": "null"  
}  
{"_id": ObjectId("5c3c5b04c8ced0abb6425f91"),  
 "name": "Lucky",  
 "maths_marks": 66,  
 "english_marks": 90,  
 "science_marks": 75,  
 "Average": "null"  
}  
{"_id": ObjectId("5c3c5b19c8ced0abb6425f92"),  
 "name": "Gva",  
 "maths_marks": 71,  
 "english_marks": 75,  
 "science_marks": 56,  
 "Average": "null"  
}  
{"_id": ObjectId("5c3c5b44c8ced0abb6425f93"),  
 "name": "Raam",  
 "maths_marks": 41,  
 "english_marks": 65,  
 "science_marks": 88,  
 "Average": "null"  
}  
}  
> db.studentmarks.find({$and: [{english_marks: {$gt: 50}}, {maths_marks: {$gt: 50}}, {science_marks: {$gt: 50}}], {name: 1, _id: 0})  
{ "name": "Mala" }  
{ "name": "Vanu" }  
{ "name": "Aruli" }  
{ "name": "Shayu" }  
{ "name": "Lucky" }  
{ "name": "Gva" }
```



```
db.studentmarks.find({$and:[{english_marks:{$gt:50}},{maths_marks:
{$lt:50}}]}, {name:1,_id:0})
```

A screenshot of a Linux desktop environment. The desktop background is a dark purple color. On the left side, there is a vertical dock containing several application icons: a terminal, a file manager, a web browser (Firefox), a text editor (Gedit), and a few others. The top of the screen shows a panel with system status icons (Wi-Fi, battery, time) and the date. The terminal window is open, showing a MongoDB command prompt. The command entered is: `db.studentmarks.find({'$and': [{'english_marks': {'$gt': 50}}, {'maths_marks': {'$lt': 50}}]}, {'name': 1, '_id': 0})`. The output of the query is displayed below the command: `{ "name" : "Kumaran" }` and `{ "name" : "Raam" }`. The terminal window has two tabs open, both labeled `ukistu09@ukipc09: ~`. The system status bar at the top right shows the time as 10:33 AM and the battery level as 48%.

```
db.studentmarks.find({$and:[{science_marks:{$gt:40}},{maths_marks:
{$lt:40}}]}, {name:1,_id:0})
```


```

ukistu09@ukipc09: ~
{
  "_id" : ObjectId("5c3c5b04c8ced0abb6425f91"),
  "name" : "Lucky",
  "maths_marks" : 66,
  "english_marks" : 90,
  "science_marks" : 75,
  "Average" : ""
}
{
  "_id" : ObjectId("5c3c5b19c8ced0abb6425f92"),
  "name" : "Gva",
  "maths_marks" : 71,
  "english_marks" : 75,
  "science_marks" : 56,
  "Average" : ""
}
{
  "_id" : ObjectId("5c3c5b44c8ced0abb6425f93"),
  "name" : "Raam",
  "maths_marks" : 41,
  "english_marks" : 65,
  "Average" : ""
}
{
  "_id" : ObjectId("5c3ebf5e266fa04f2f34584b"),
  "name" : "John",
  "maths_marks" : 87,
  "science_marks" : 23,
  "Average" : ""
}
> db.studentmarks.find({'$and':[{'science_marks':{'$lt:40'}},{'maths_marks':{'$lt:40'}},]},{'name:1,_id:0'})
> db.studentmarks.find({'$or':[{'science_marks':{'$lt:40'}},{'maths_marks':{'$lt:40'}},]},{'name:1,_id:0'})
{ "name" : "Kala" }
{ "name" : "John" }
>
> db.studentmarks.find({'$or':[{'science_marks':{'$lt:40'}},{'maths_marks':{'$lt:40'}},]},{'name:1,_id:0'})
{ "name" : "Kala" }
{ "name" : "John" }
>

```

11) Remove Science column/field for Raam

```
db.studentmarks.update({name:"Raam"},{$unset:{science_marks:88}})
```

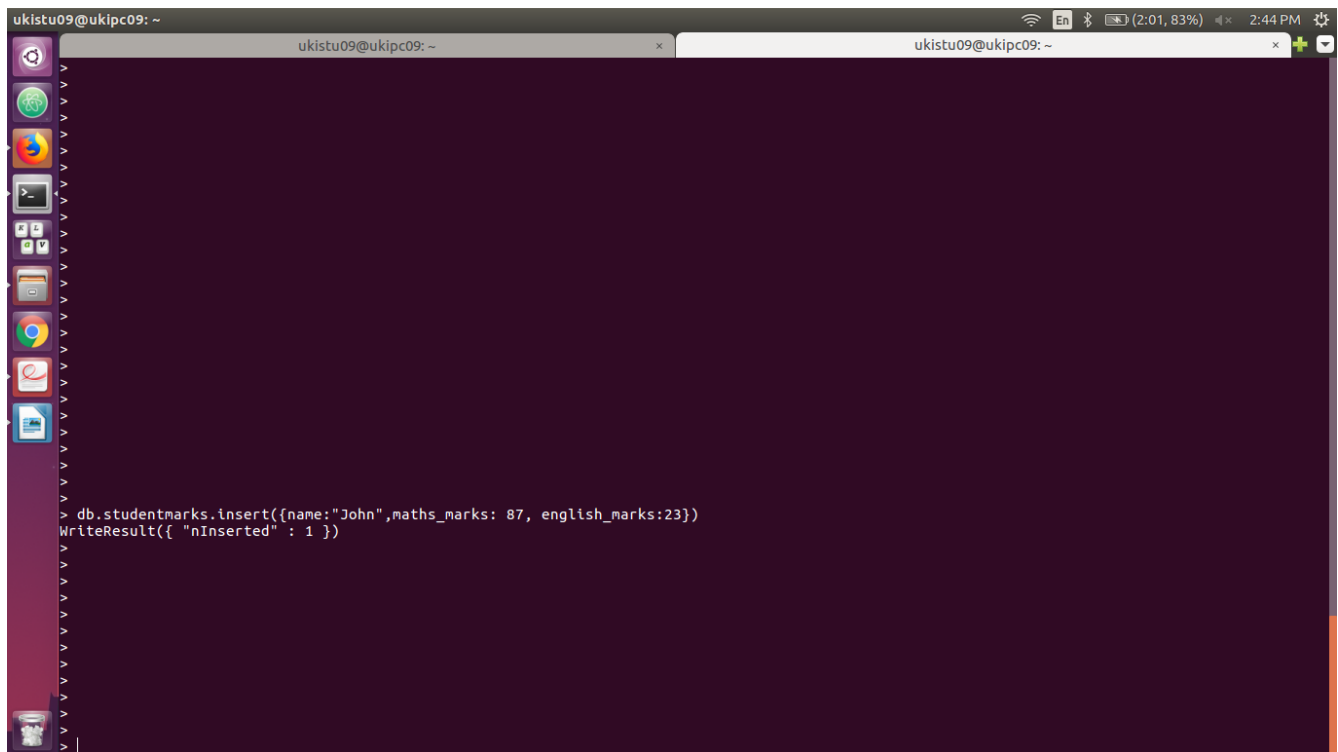


The screenshot shows a terminal window with a dark purple background. The terminal displays a series of MongoDB commands and their outputs. The commands are as follows:

```
db.studentmarks.find({$and:[{english_marks:{>50}},{maths_marks:{<50}}]}, {name:1,_id:0})
{ "name" : "Kumaran" }
{ "name" : "Raam" }
```

The first command returns two documents. The second command, `db.studentmarks.update({name:"Raam"},{$set:{science_marks:88}})`, results in a `SyntaxError: missing) in parenthetical` message. The third command, `db.studentmarks.update({name:"Raam"},{$unset:{science_marks:88}})`, also results in a `SyntaxError: missing) in parenthetical` message. The fourth command, `db.studentmarks.update({name:"Raam"},{$set:{science_marks:88}})`, is highlighted in white and results in a `WriteResult` object: `{ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 }`.

12) Update John's Math mark as 87 and English mark as 23, if john not available upsert.
`db.studentmarks.insert({name:"John",maths_marks: 87, english_marks:23})`



The image shows a terminal window with a dark purple background. The prompt is `ukistu09@ukipc09: ~`. The command `> db.studentmarks.insert({name:"John",maths_marks: 87, english_marks:23})` has been entered. The output is `WriteResult({ "nInserted" : 1 })`. The terminal window is part of a desktop environment with a sidebar on the left containing various application icons. The top of the window shows a status bar with network, battery, and time information.

```
ukistu09@ukipc09: ~  
> db.studentmarks.insert({name:"John",maths_marks: 87, english_marks:23})  
WriteResult({ "nInserted" : 1 })
```

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

```
db.studentmarks.update({name:"John"},{$rename:
{english_marks:"science_marks"}})
```

```
ukistu09@ukipc09: ~  
ukistu09@ukipc09: ~  
ukistu09@ukipc09: ~  
{"science_marks": 65,  
 "Average": "null"  
}  
{  
  "_id": ObjectId("5c3c5b04c8ced0abb6425f91"),  
  "name": "Lucky",  
  "maths_marks": 66,  
  "english_marks": 90,  
  "science_marks": 75,  
  "Average": "null"  
}  
{  
  "_id": ObjectId("5c3c5b19c8ced0abb6425f92"),  
  "name": "Gva",  
  "maths_marks": 71,  
  "english_marks": 75,  
  "science_marks": 56,  
  "Average": "null"  
}  
{  
  "_id": ObjectId("5c3c5b44c8ced0abb6425f93"),  
  "name": "Raam",  
  "maths_marks": 41,  
  "english_marks": 65,  
  "Average": "null"  
}  
{  
  "_id": ObjectId("5c3ebf5e266fa04f2f34584b"),  
  "name": "John",  
  "maths_marks": 87,  
  "english_marks": 23  
}  
>  
>  
>  
> db.studentmarks.update({name:"John"},{$rename:{english_marks:science_marks}})  
2019-01-16T11:02:18.406+0530 E QUERY [thread1] ReferenceError: science_marks is not defined :  
@(shell):1:48  
> db.studentmarks.update({name:"John"},{$rename:{english_marks:"science_marks"}})  
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })  
>  
>
```

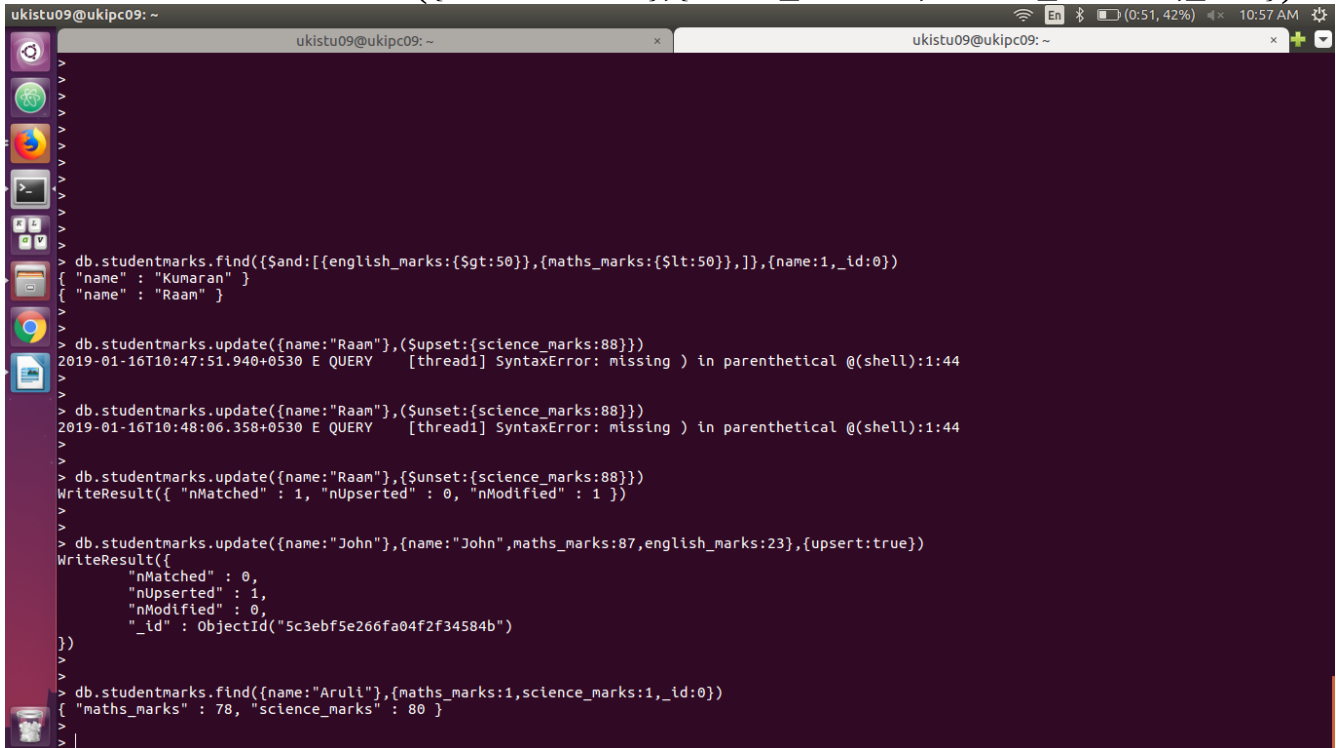
14) Remove Kumaran's document from collection

```
db.studentmarks.remove({name:"Kumaran"})
```

```
ukistu09@ukipc09: ~  
ukistu09@ukipc09: ~  
ukistu09@ukipc09: ~  
> db.studentmarks.remove({name:"Kumaran"})  
WriteResult({ "nRemoved" : 1 })  
> db.studentmarks.find().pretty()  
{  
  "_id" : ObjectId("5c3c5a23c8ced0abb6425f8b"),  
  "name" : "Mala",  
  "maths_marks" : 51,  
  "english_marks" : 53,  
  "science_marks" : 72,  
  "Average" : "null"  
}  
{  
  "_id" : ObjectId("5c3c5a6ec8ced0abb6425f8c"),  
  "name" : "Vanu",  
  "maths_marks" : 80,  
  "english_marks" : 75,  
  "science_marks" : 85,  
  "Average" : "null"  
}  
{  
  "_id" : ObjectId("5c3c5a89c8ced0abb6425f8d"),  
  "name" : "Kala",  
  "maths_marks" : 32,  
  "english_marks" : 40,  
  "science_marks" : 53,  
  "Average" : "null"  
}  
{  
  "_id" : ObjectId("5c3c5aa9c8ced0abb6425f8e"),  
  "name" : "Aruli",  
  "maths_marks" : 78,  
  "english_marks" : 85,  
  "science_marks" : 80,  
  "Average" : "null"  
}  
{  
  "_id" : ObjectId("5c3c5ac7c8ced0abb6425f8f"),  
  "name" : "Shayu",  
  "maths_marks" : 80,  
  "english_marks" : 76,  
  "science_marks" : 65,
```

15) Find Kala's or Aruli's math_marks and science_marks

```
db.studentmarks.find({name:"Aruli"},{maths_marks:1,science_marks:1,_id:0})
```

A terminal window with a dark purple background and white text. The window title is 'ukistu09@ukipc09: ~'. The terminal shows a series of MongoDB commands and their outputs. The commands include finding documents by name and marks, updating documents with specific marks, and finding documents by name and marks. The outputs show the results of these operations, including the number of documents matched, updated, or inserted, and the specific document details.

```
ukistu09@ukipc09: ~  
> db.studentmarks.find({$and:[{english_marks:{$gt:50}},{maths_marks:{$lt:50}},]}, {name:1,_id:0})  
{ "name" : "Kumaran" }  
{ "name" : "Raam" }  
> db.studentmarks.update({name:"Raam"},{$set:{science_marks:88}})  
2019-01-16T10:47:51.940+0530 E QUERY [thread1] SyntaxError: missing ) in parenthetical @(shell):1:44  
> db.studentmarks.update({name:"Raam"},{$unset:{science_marks:88}})  
2019-01-16T10:48:06.358+0530 E QUERY [thread1] SyntaxError: missing ) in parenthetical @(shell):1:44  
> db.studentmarks.update({name:"Raam"},{$unset:{science_marks:88}})  
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })  
> db.studentmarks.update({name:"John"},{name:"John",maths_marks:87,english_marks:23},{upsert:true})  
WriteResult({  
  "nMatched" : 0,  
  "nUpserted" : 1,  
  "nModified" : 0,  
  "_id" : ObjectId("5c3ebf5e266fa04f2f34584b")  
})  
> db.studentmarks.find({name:"Aruli"},{maths_marks:1,science_marks:1,_id:0})  
{ "maths_marks" : 78, "science_marks" : 80 }  
>
```