Workshop

mongoDB

สถาบัน ไอเอ็มซี

CRUD

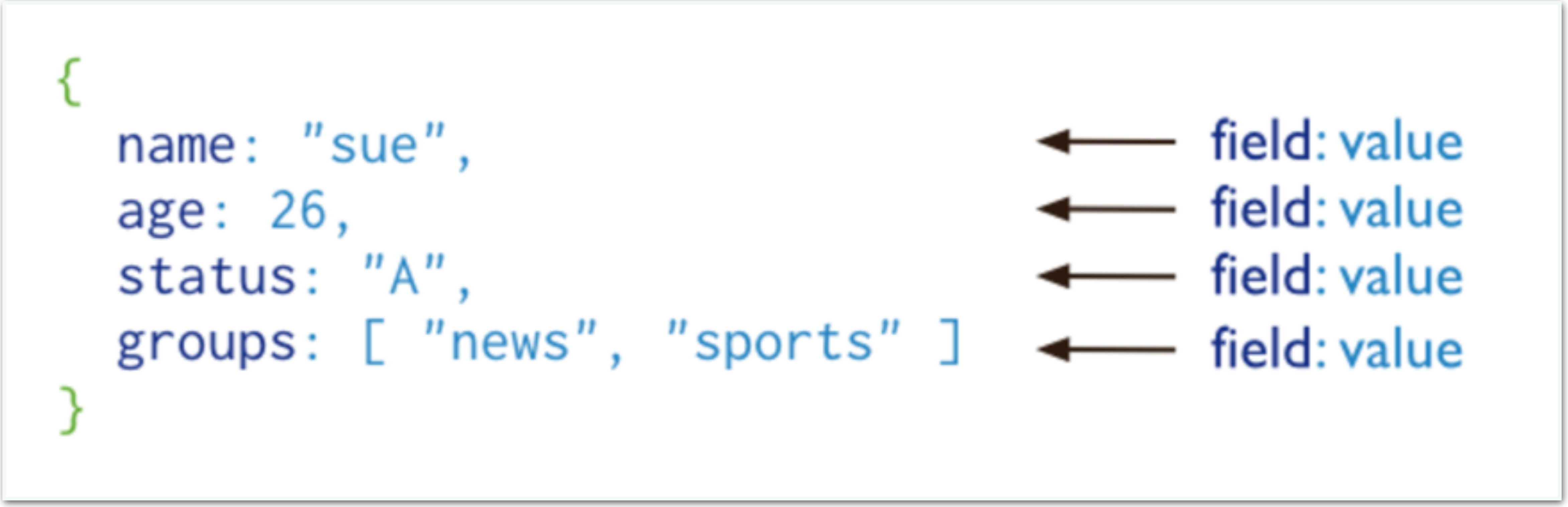
C = Create

R = Read

U = Update

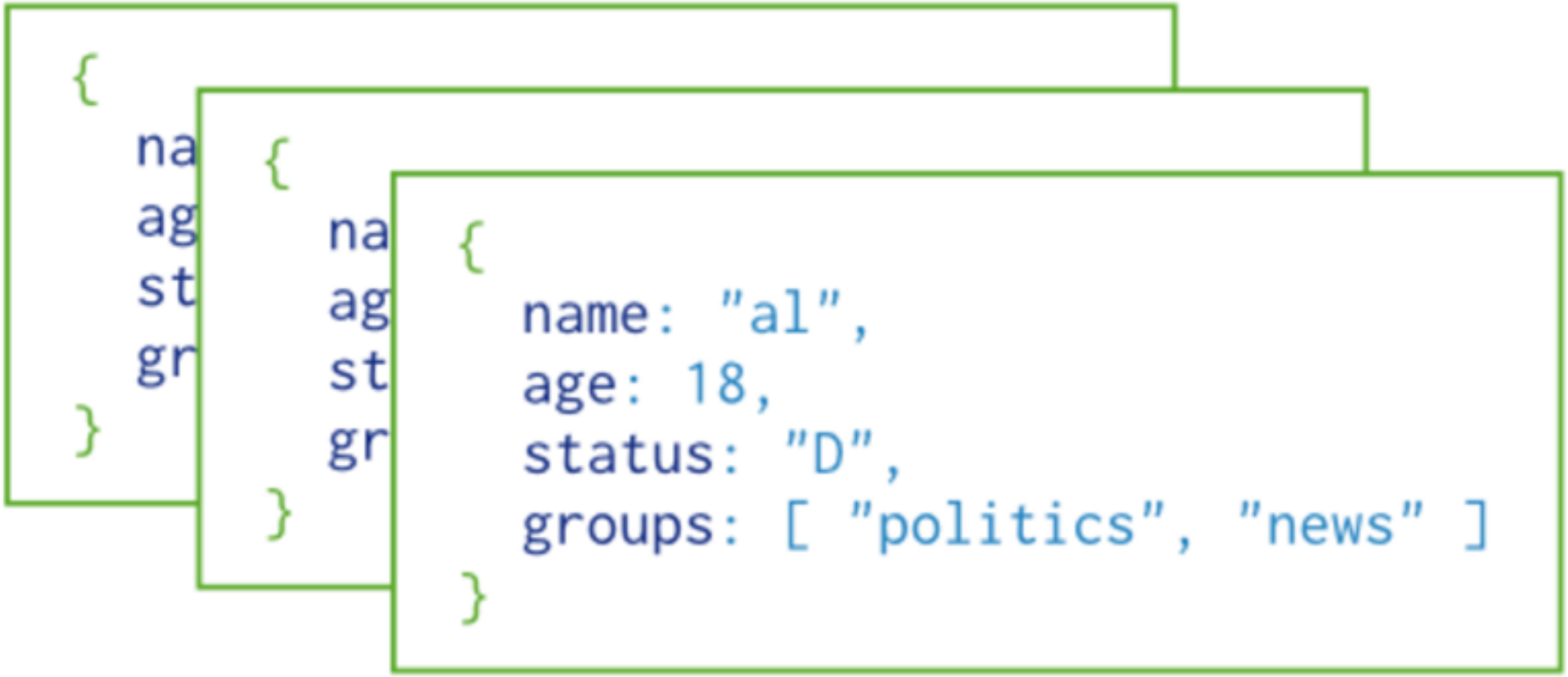
D = Delete

Document



```
{
   name: "sue",
   age: 26,
   status: "A",
   groups: [ "news", "sports" ]
}
```

Collection



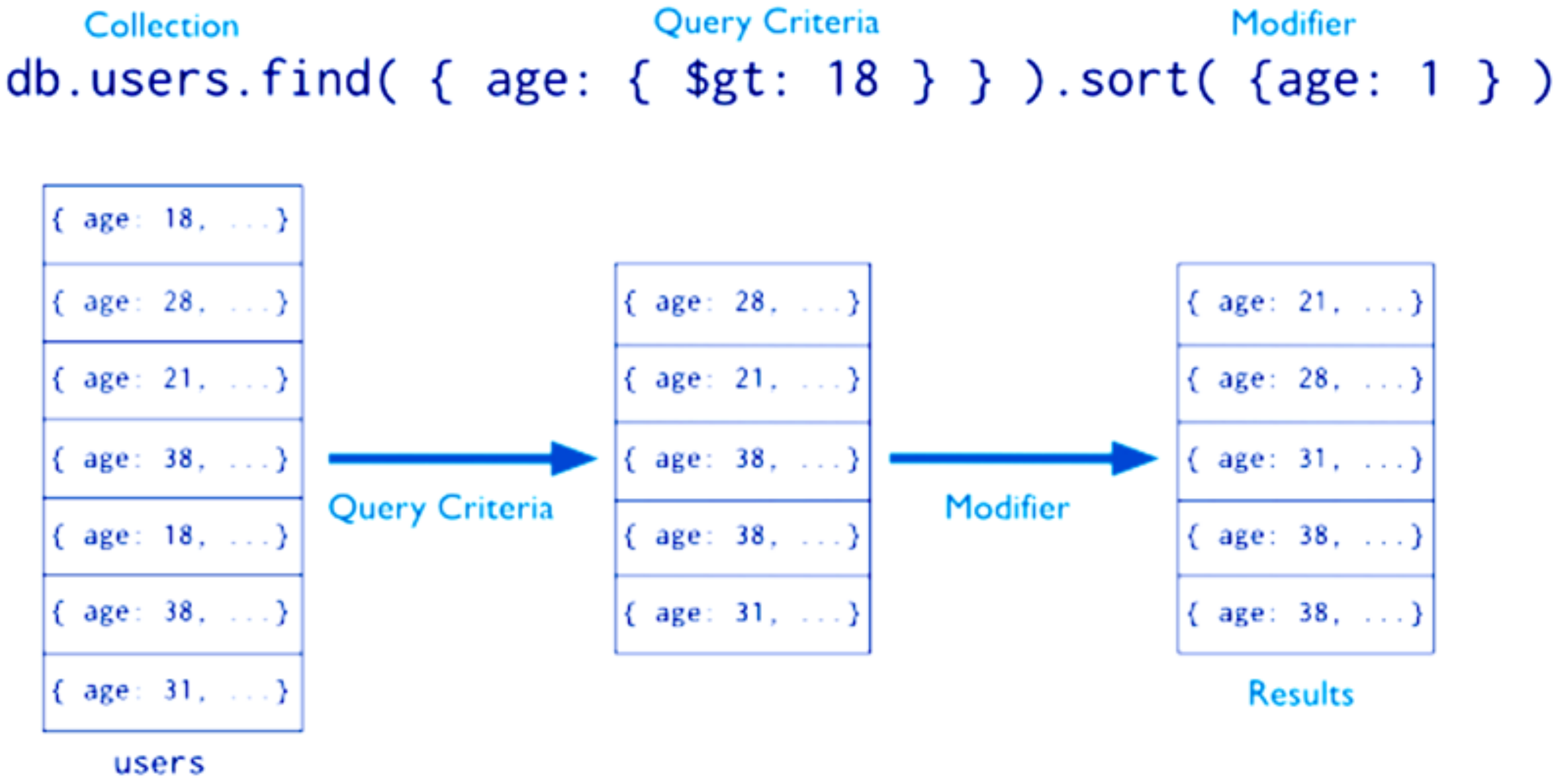
```
{
   name: "al",
   age: 18,
   status: "D",
   groups: [ "politics", "news" ]
}
```

Collection

Create data

```
>db.person.insert(
{
    "name": "somkiat",
    "age": 35,
    "status": "A",
    "groups": [ "news", "sport" ]
}
)
```

Query data diagram



```
db.users.find( { age: { $gt: 18 } } ).sort( {age: 1 } )
```

```
{ age: 18, ...}
{ age: 28, ...}
{ age: 21, ...}
{ age: 38, ...}
{ age: 18, ...}
{ age: 38, ...}
{ age: 31, ...}
```

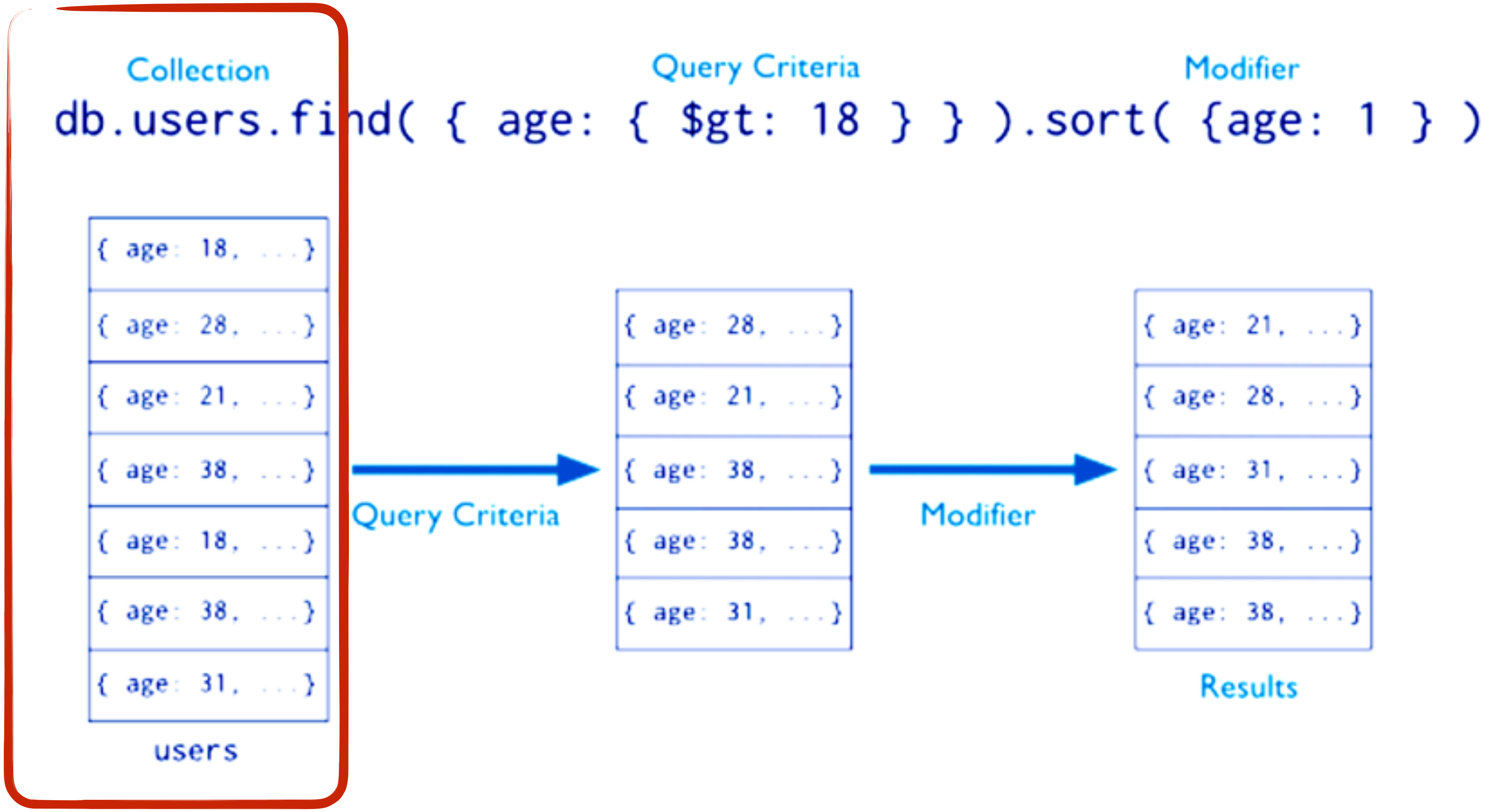
users

```
{ age: 28, ...}
{ age: 21, ...}
{ age: 38, ...}
{ age: 38, ...}
{ age: 31, ...}
```

```
{ age: 21, ...}
{ age: 28, ...}
{ age: 31, ...}
{ age: 38, ...}
{ age: 38, ...}
```

Results

Query data diagram



Collection

Query Criteria

Modifier

```
db.users.find( { age: { $gt: 18 } } ).sort( {age: 1 } )
```

{ age: 18, ...}
{ age: 28, ...}
{ age: 21, ...}
{ age: 38, ...}
{ age: 18, ...}
{ age: 38, ...}
{ age: 31, ...}

users

Query Criteria

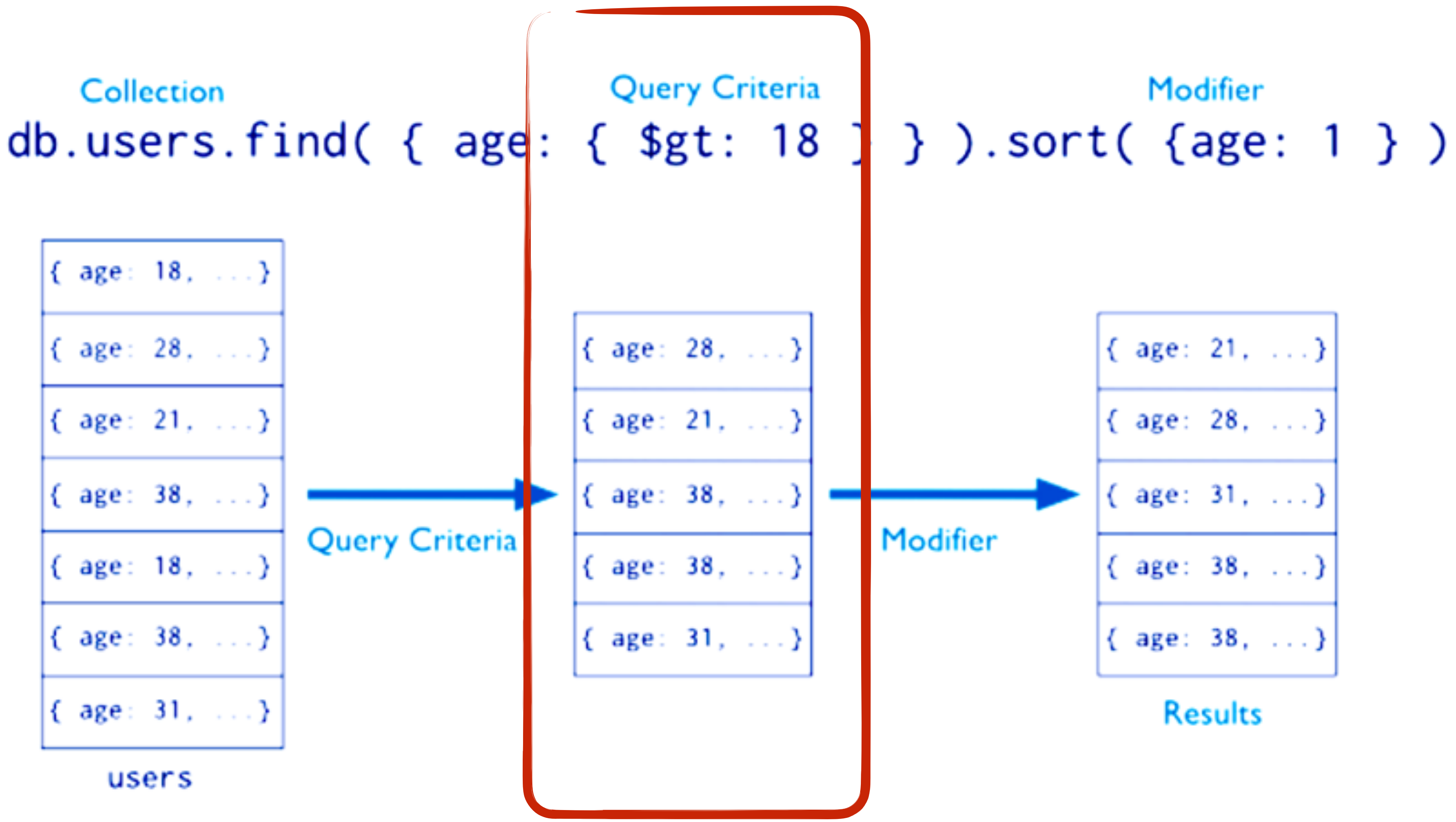
{ age: 28, ...}
{ age: 21, ...}
{ age: 38, ...}
{ age: 38, ...}
{ age: 31, ...}

Modifier

{ age: 21, ...}
{ age: 28, ...}
{ age: 31, ...}
{ age: 38, ...}
{ age: 38, ...}

Results

Query data diagram



Collection Query Criteria Modifier

```
db.users.find( { age: { $gt: 18 } } ).sort( {age: 1 } )
```

{ age: 18, ...}
{ age: 28, ...}
{ age: 21, ...}
{ age: 38, ...}
{ age: 18, ...}
{ age: 38, ...}
{ age: 31, ...}

users

Query Criteria

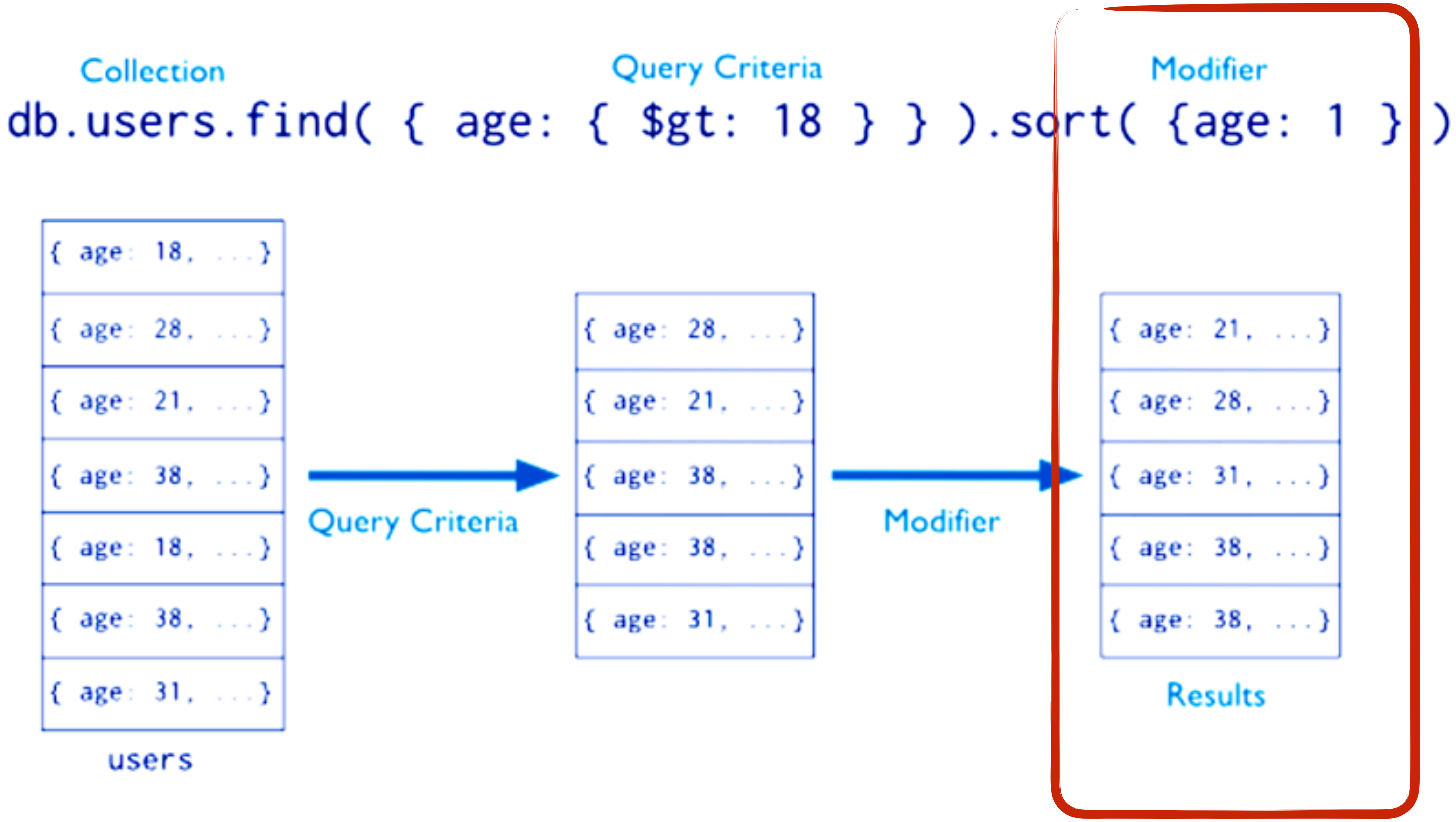
{ age: 28, ...}
{ age: 21, ...}
{ age: 38, ...}
{ age: 38, ...}
{ age: 31, ...}

Modifier

{ age: 21, ...}
{ age: 28, ...}
{ age: 31, ...}
{ age: 38, ...}
{ age: 38, ...}

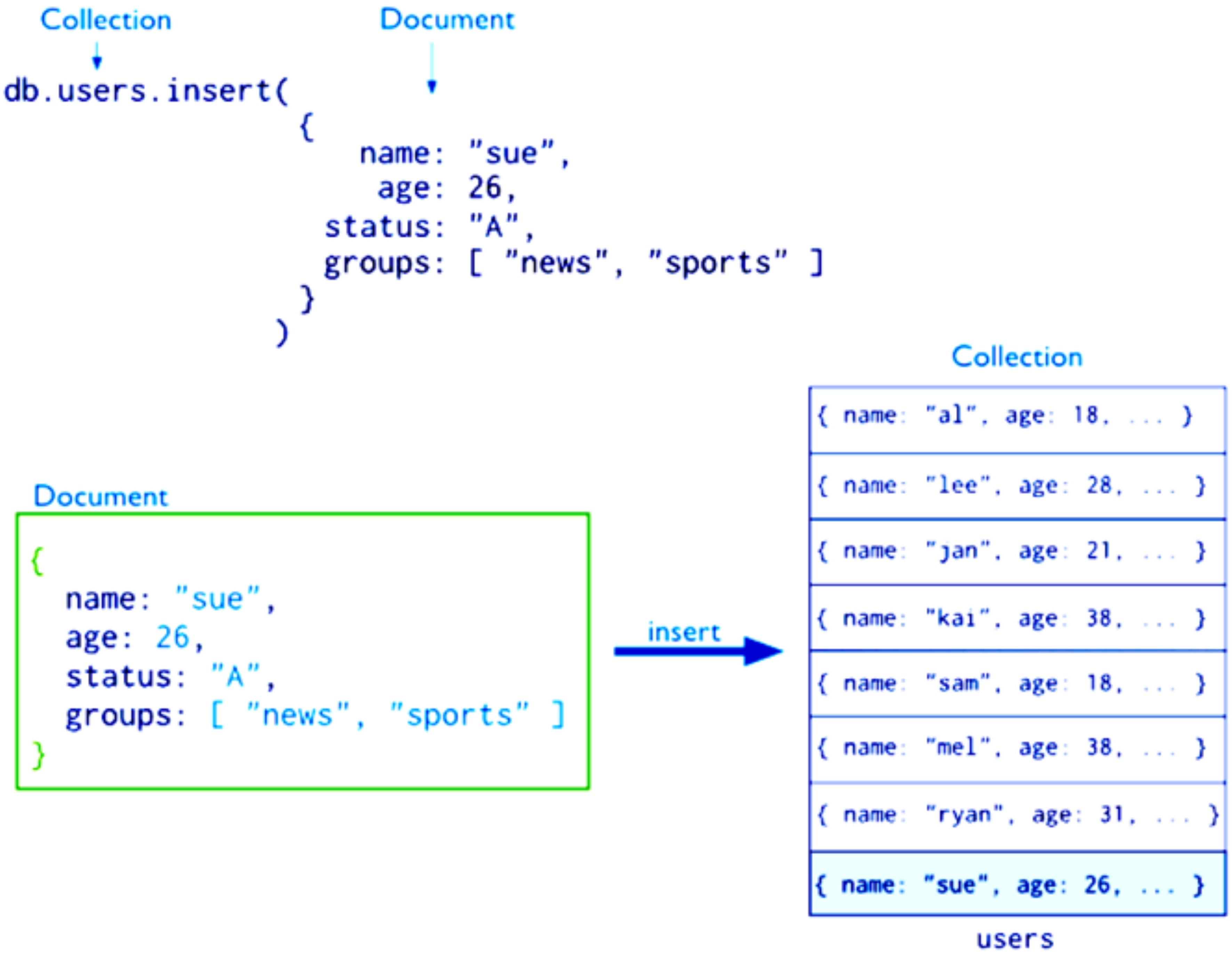
Results

Query data diagram



```
db.users.find( { age: { $gt: 18 } } ).sort( {age: 1 } )
```

Data modification

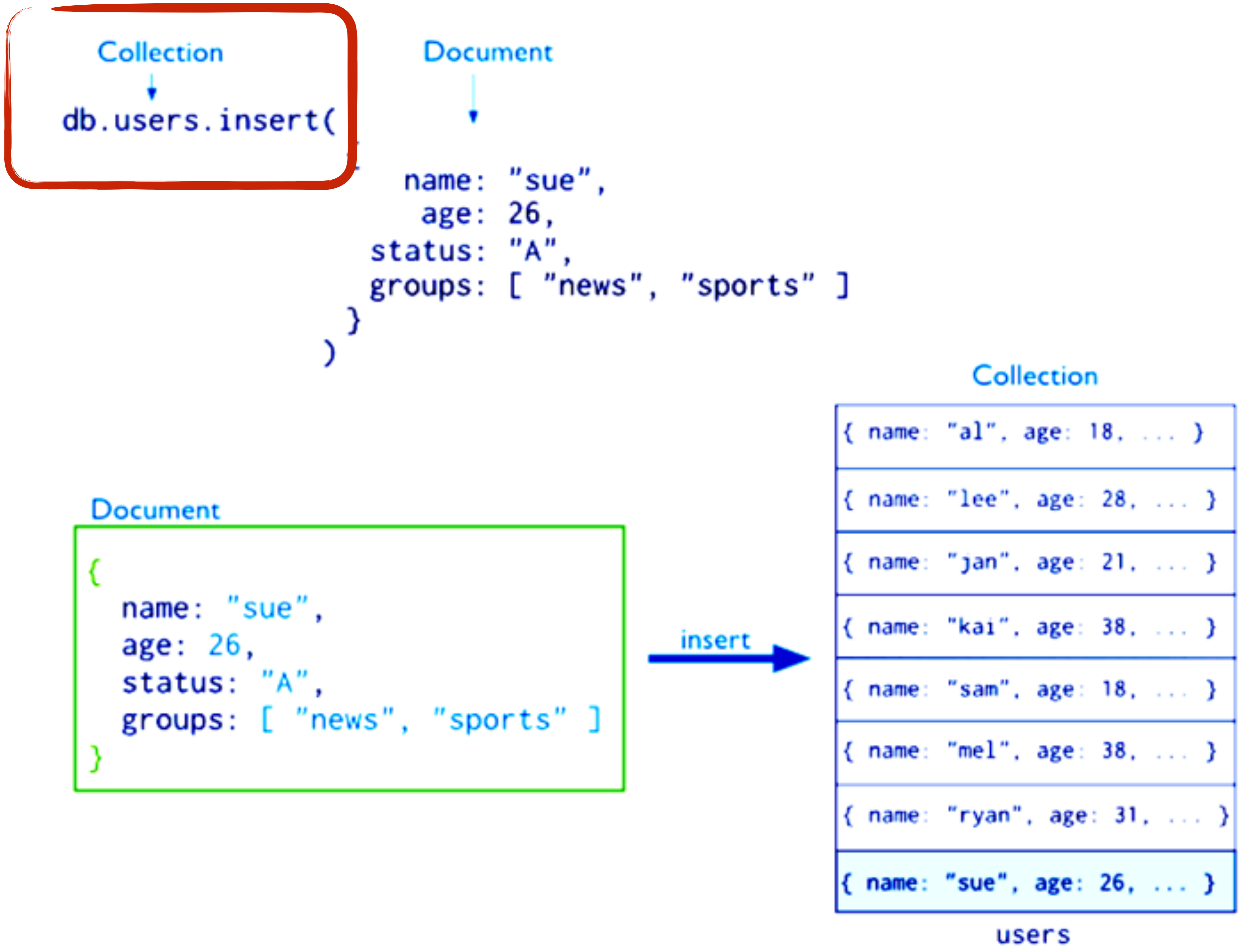


```
db.users.insert(
                {
                   name: "sue",
                    age: 26,
                 status: "A",
                 groups: [ "news", "sports" ]
                }
               )
```

```
{
  name: "sue",
  age: 26,
  status: "A",
  groups: [ "news", "sports" ]
}
```

```
{ name: "al", age: 18, ... }
{ name: "lee", age: 28, ... }
{ name: "jan", age: 21, ... }
{ name: "kai", age: 38, ... }
{ name: "sam", age: 18, ... }
{ name: "mel", age: 38, ... }
{ name: "ryan", age: 31, ... }
{ name: "sue", age: 26, ... }
```

Data modification



```
db.users.insert(
  {
     name: "sue",
      age: 26,
   status: "A",
   groups: [ "news", "sports" ]
  }
)
```

```
{
  name: "sue",
  age: 26,
  status: "A",
  groups: [ "news", "sports" ]
}
```

```
{ name: "al", age: 18, ... }
{ name: "lee", age: 28, ... }
{ name: "jan", age: 21, ... }
{ name: "kai", age: 38, ... }
{ name: "sam", age: 18, ... }
{ name: "mel", age: 38, ... }
{ name: "ryan", age: 31, ... }
{ name: "sue", age: 26, ... }
```

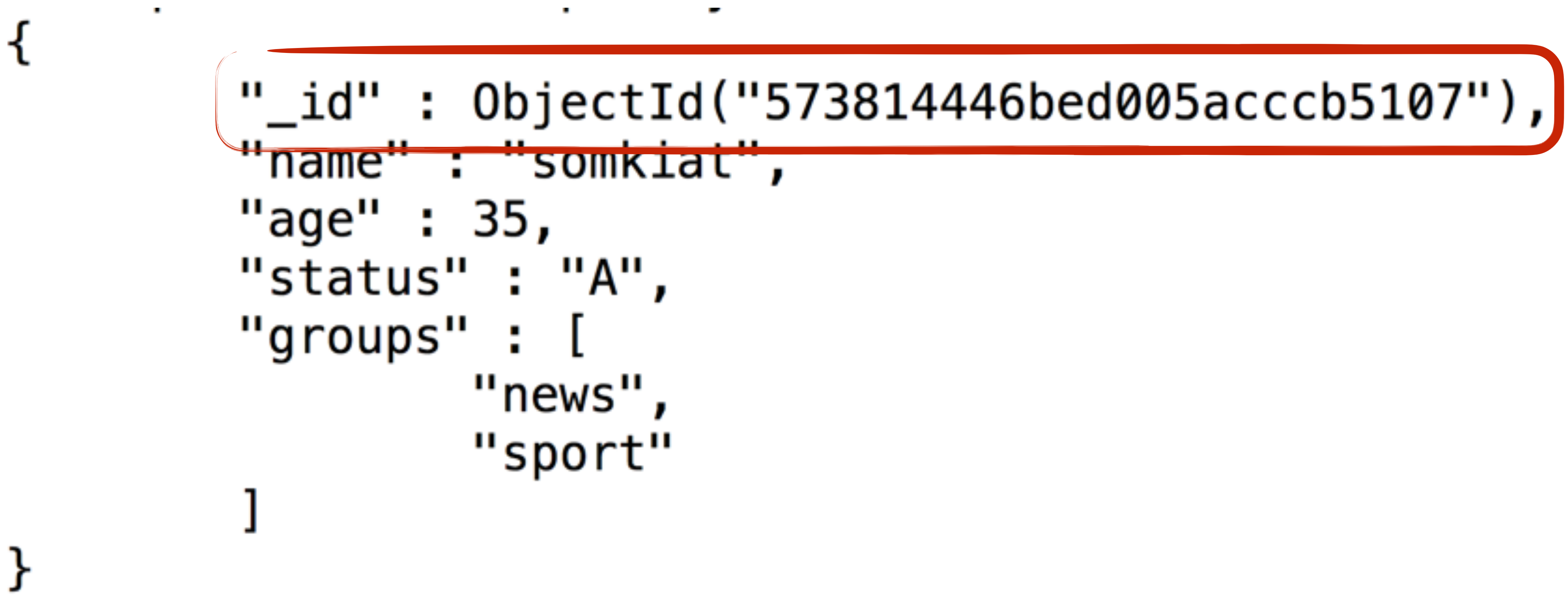
CreateRUD

Create data

```
>db.person.insert(
{
    "name": "somkiat",
    "age": 35,
    "status": "A",
    "groups": [ "news", "sport" ]
}
)
```

Check inserted data

>db.person.find().pretty()

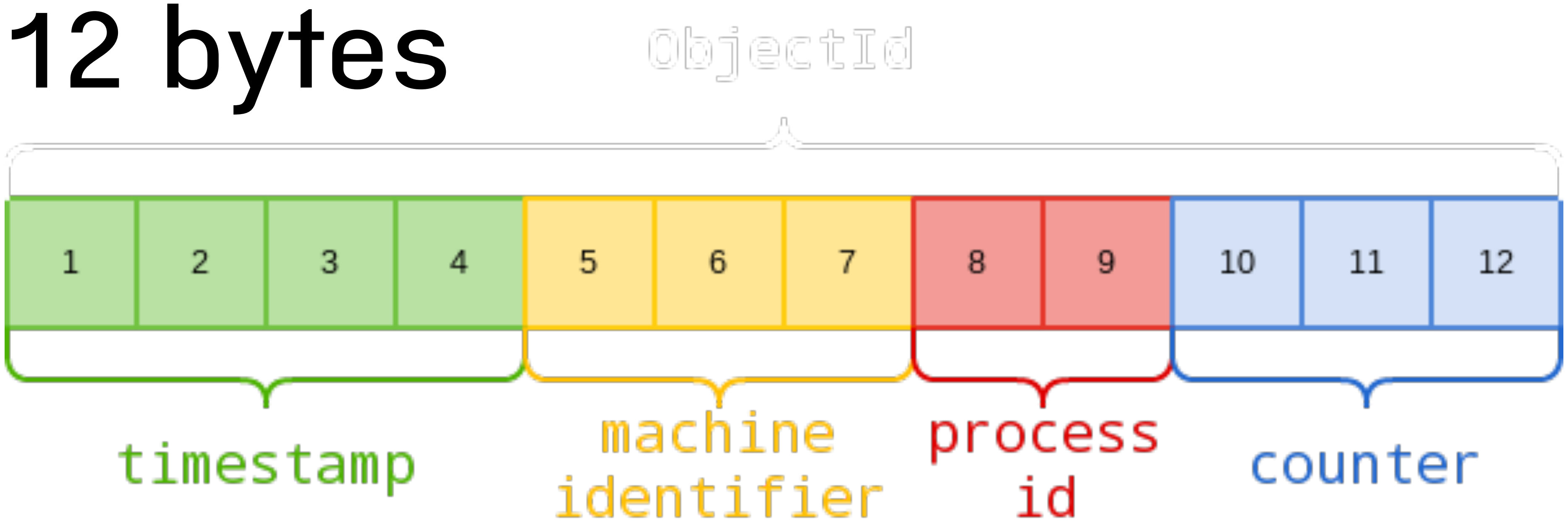


```
{
        "_id" : ObjectId("573814446bed005acccb5107"),
        "name" : "somkiat",
        "age" : 35,
        "status" : "A",
        "groups" : [
                "news",
                "sport"
        ]
}
```

ObjectId

Hexadecimal

12 bytes



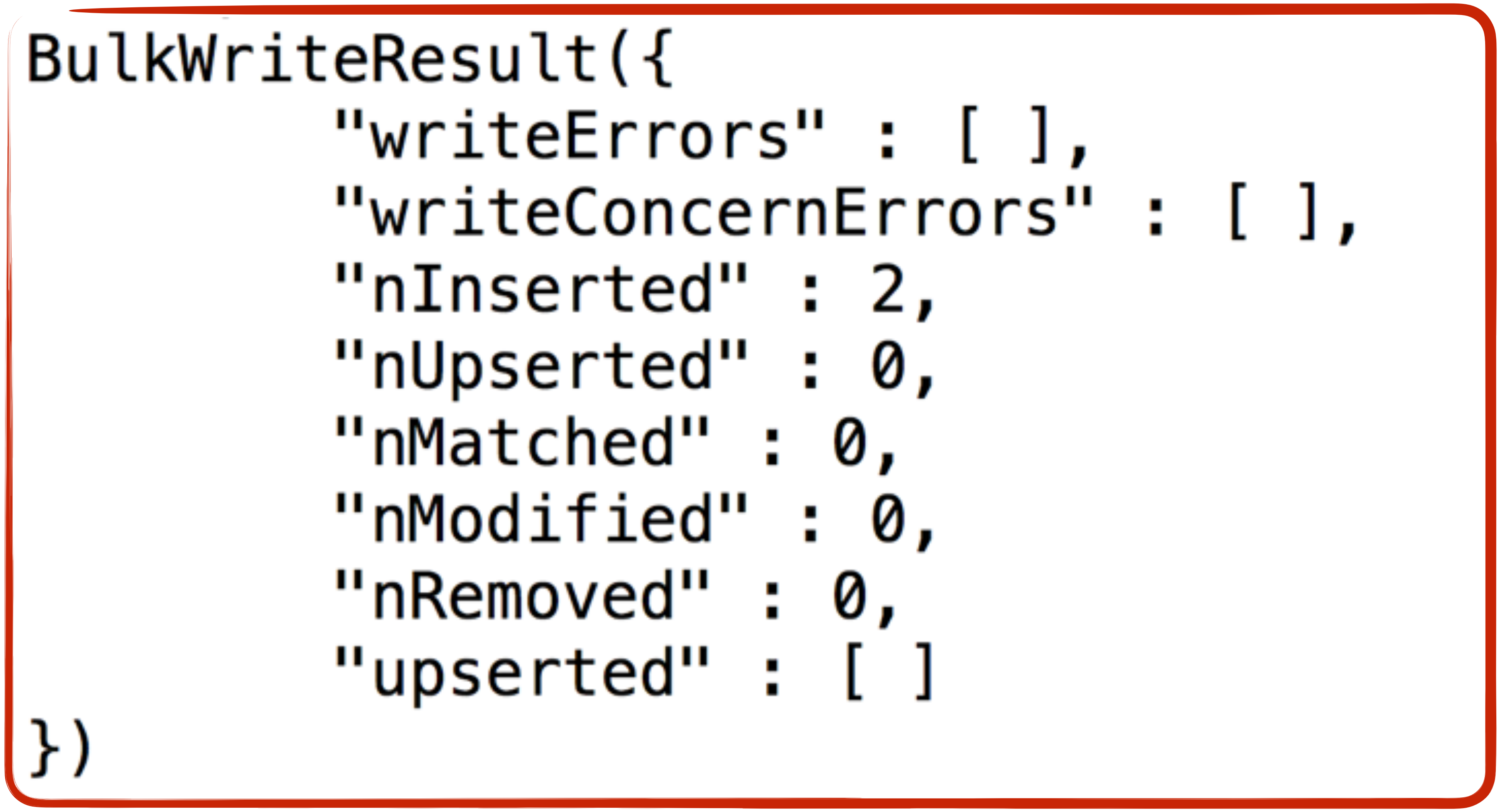
https://docs.mongodb.com/manual/reference/method/ObjectId/

Create more data

```
var datas =
[
    {
        "name": "somkiat",
        "age": 35,
        "status": "A",
        "groups": [ "news", "sport" ]
    },
    {
        "name": "other",
        "age": 40,
        "status": "A",
        "groups": [ "news", "sport" ]
    }
]
```

Create more data

```
>db.person.insert( datas )
```

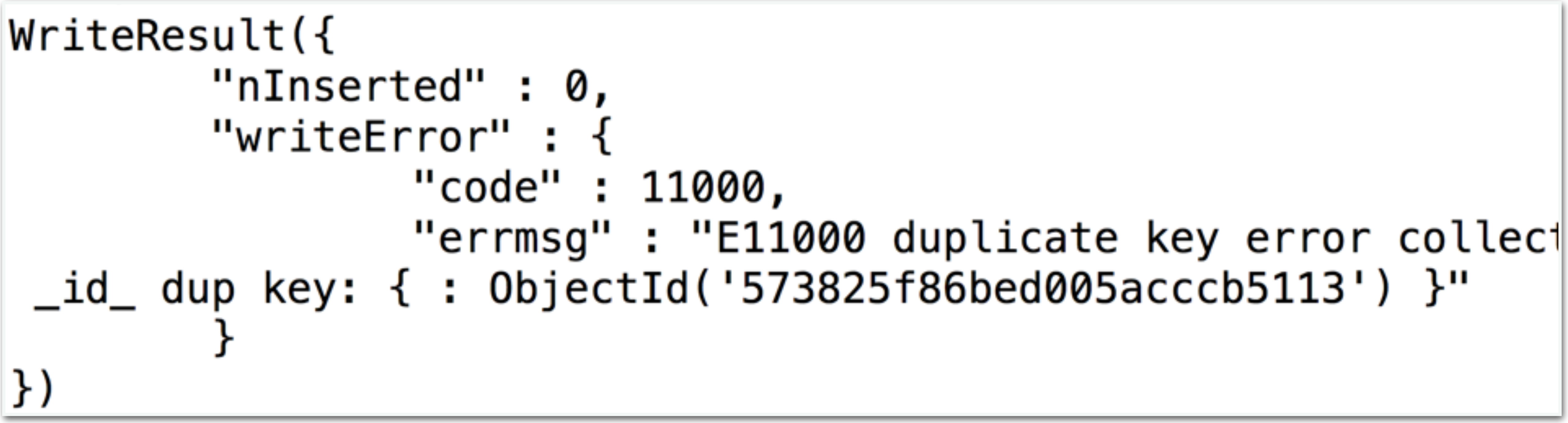


```
BulkWriteResult({
        "writeErrors" : [ ],
        "writeConcernErrors" : [ ],
        "nInserted" : 2,
        "nUpserted" : 0,
        "nMatched" : 0,
        "nModified" : 0,
        "nRemoved" : 0,
        "upserted" : [ ]
})
```

Existing Document

```
>db.person.insert(
{
    "_id": ObjectId("573825f86bed005acccb5113"),
    "name": "somkiat",
    "age": 35,
    "status": "A",
    "groups": [ "news", "sport" ]
}
)
```

Existing Document



```
WriteResult({
        "nInserted" : 0,
        "writeError" : {
                "code" : 11000,
                "errmsg" : "E11000 duplicate key error collect
 _id_ dup key: { : ObjectId('573825f86bed005acccb5113') }"
        }
})
```

Unordered insert

With unordered inserts, if an error occurs during an insert of one of the documents

Unordered insert

```
>db.person.insert(
  [
    { _id: 1, name: "people 1"},
    { _id: 2, name: "people 2"},
    { _id: 3, name: "people 3"}
  ],
  { ordered: false }
)
```

Unordered insert

```
>db.person.insert(
   [
     { _id: 1, name: "people 1"},
     { _id: 4, name: "people 4"},
     { _id: 3, name: "people 3"}
   ],
   { ordered: false }
)
```

Unordered insert

```
>db.person.insert(
  [
    { _id: 1, name: "people 1"},
    { _id: 5, name: "people 5"},
    { _id: 3, name: "people 3"}
  ],
  { ordered: true }
)
```

Create multiple docs with Bulk

Create multiple docs with Bulk

```
>var bulk =
db.person.initializeUnorderedBulkOp()
```

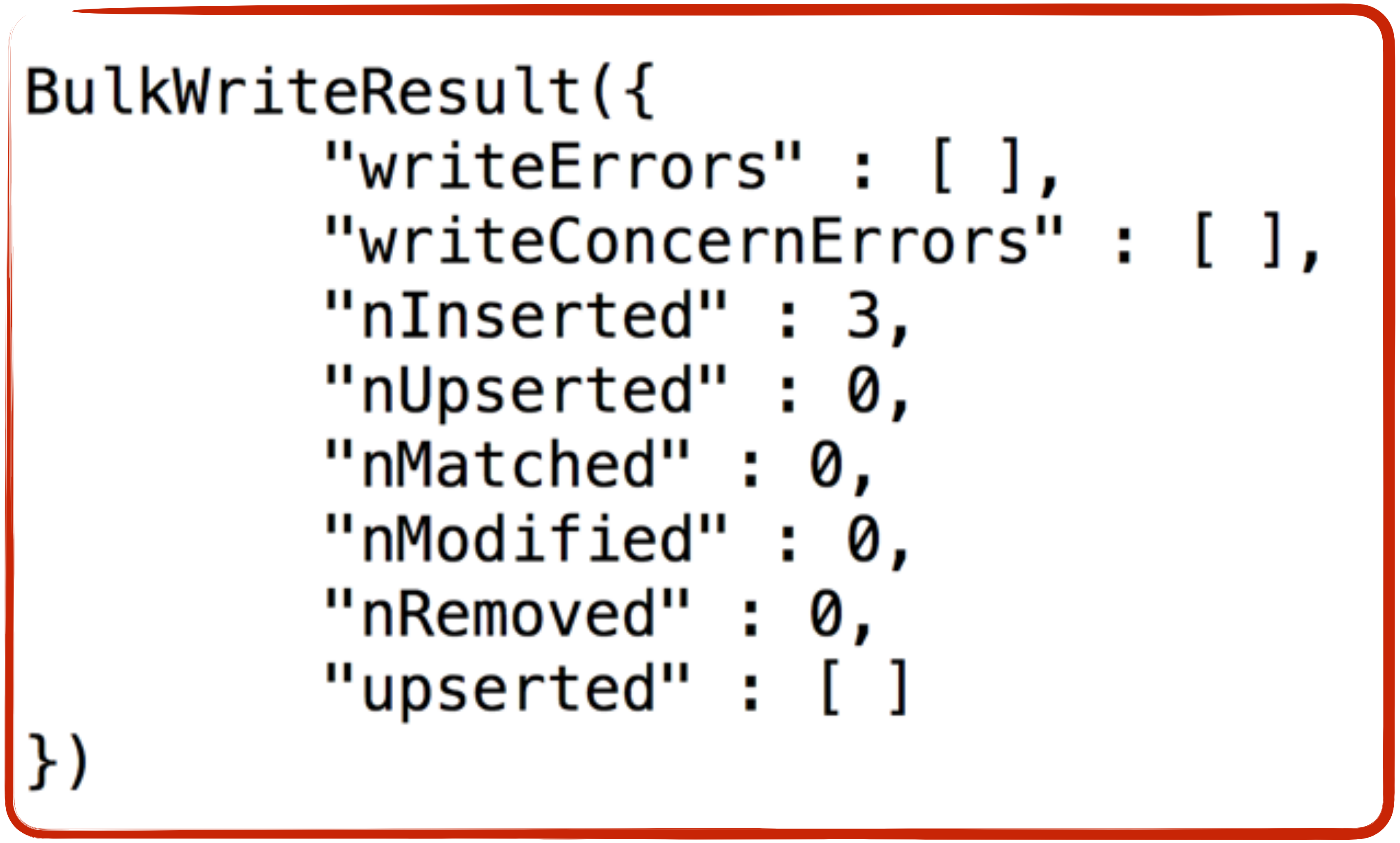
```
>var bulk =
db.person.initializeOrderedBulkOp()
```

Create multiple docs with Bulk

```
>bulk.insert( { _id: 1, name: "people 1"} )
>bulk.insert( { _id: 2, name: "people 2"} )
>bulk.insert( { _id: 3, name: "people 3"} )
```

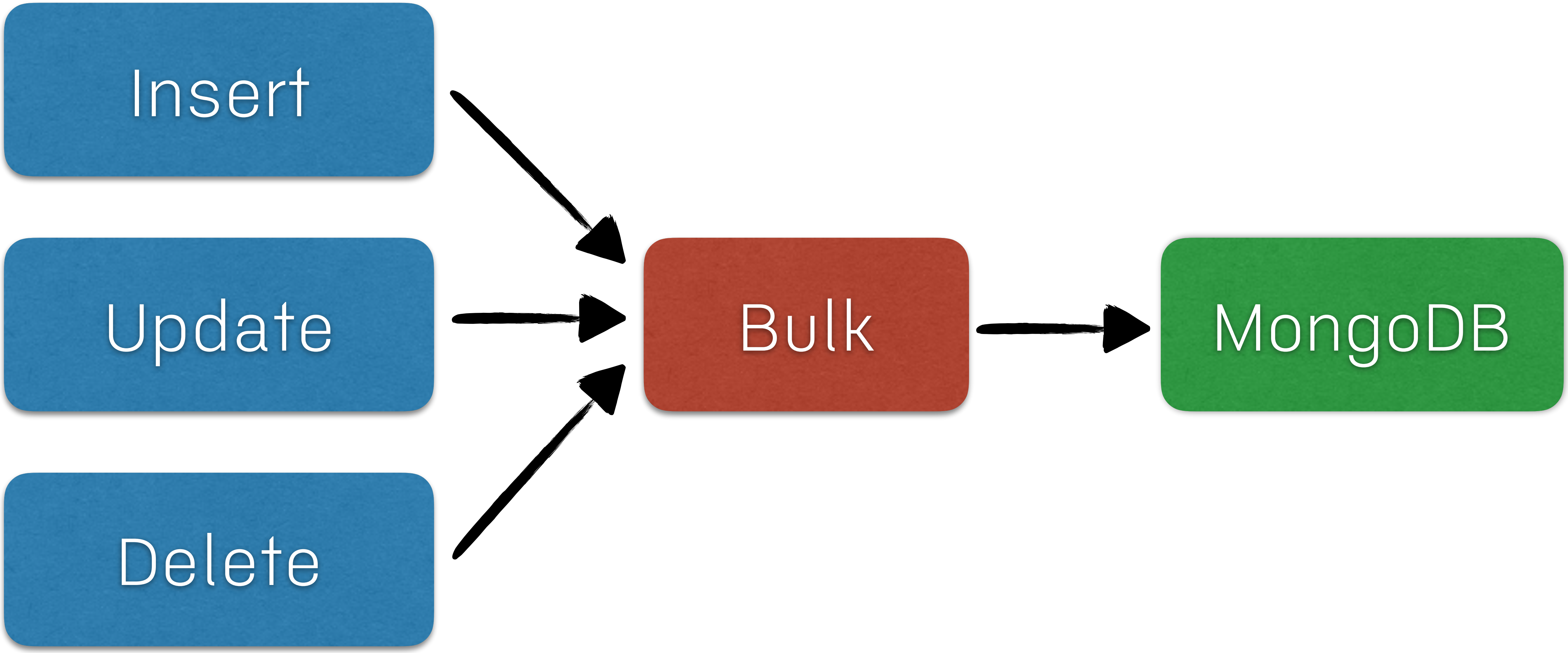
Create multiple docs with Bulk

>bulk.execute()



```
BulkWriteResult({
        "writeErrors" : [ ],
        "writeConcernErrors" : [ ],
        "nInserted" : 3,
        "nUpserted" : 0,
        "nMatched" : 0,
        "nModified" : 0,
        "nRemoved" : 0,
        "upserted" : [ ]
})
```

Try with multi operation



Insert

Update

Delete

Bulk

MongoDB

CReadUD

All documents

```
>db.food.find()

>db.food.find( {} )
```

CRUpdateD

CRUDelete

All documents

```
>db.food.remove({})
```

Delete collections

```
>db.food.drop()
```

Resources

https://github.com/up1/course-introduction-mongodb