

Assignment 6.2 - Aggregate Queries (Hands-On 6.1)

Author: Chris Newell

Date: October 12, 2025

WEB-335: MongoDB Design and Development

This report demonstrates the successful execution of MongoDB aggregate queries, data manipulation, and projection operations for Assignment 6.2.

Figure 1: Connection to MongoDB Atlas and creation of houses/students collection

```
web-335 — mongosh mongodb+srv://<credentials>@web335cluster.sbfdyc0.mongodb.net/?retryWrites=true&w=ma...

e=web335Cluster
Using MongoDB:      8.0.14
Using Mongosh:      2.5.8

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

Atlas atlas-n7wwok-shard-0 [primary] test> use web335DB
| load("week-6/houses.js")
| db.houses.find()
| db.students.find()
|
[] load("week-6/newell_assignment6.2.js")
switched to db web335DB
Atlas atlas-n7wwok-shard-0 [primary] web335DB> load("week-6/houses.js")
| db.houses.find()
| db.students.find()
|
[] load("week-6/newell_assignment6.2.js")
true
[Atlas atlas-n7wwok-shard-0 [primary] web335DB> load("week-6/houses.js")
true
[Atlas atlas-n7wwok-shard-0 [primary] web335DB> db.houses.find()
[
  {
    _id: ObjectId('68ec0e207acca404625104d4'),
    houseId: 1,
    name: 'Gryffindor',
    mascot: 'Lion'
  },
  {
    _id: ObjectId('68ec0e207acca404625104d5'),
    houseId: 2,
    name: 'Hufflepuff',
    mascot: 'Badger'
  },
  {
    _id: ObjectId('68ec0e207acca404625104d6'),
    houseId: 3,
    name: 'Slytherin',
    mascot: 'Serpent'
  },
  {
    _id: ObjectId('68ec0e207acca404625104d7'),
    houseId: 4,
    name: 'Ravenclaw',
    mascot: 'Eagle'
  }
]
[Atlas atlas-n7wwok-shard-0 [primary] web335DB> db.students.find()
[
  {
    _id: ObjectId('68ec0e207acca404625104d8'),
    studentId: 1001,
    firstName: 'Harry',
    lastName: 'Potter',
    houseId: 1
  },
  {
    _id: ObjectId('68ec0e207acca404625104d9'),
    studentId: 1002,
    firstName: 'Hermione',
    lastName: 'Granger',
    houseId: 1
  },
  {
    _id: ObjectId('68ec0e207acca404625104da'),
    studentId: 1003,
```

Figure 2: Query results confirming data inserted into houses and students collections

```
web-335 — mongosh mongodb+srv://<credentials>@web335cluster.sbfdyc0.mongodb.net/?retryWrites=true&w=ma...

  mascot: 'Lion'
},
{
  _id: ObjectId('68ec0e207acca404625104d5'),
  houseId: 2,
  name: 'Hufflepuff',
  mascot: 'Badger'
},
{
  _id: ObjectId('68ec0e207acca404625104d6'),
  houseId: 3,
  name: 'Slytherin',
  mascot: 'Serpent'
},
{
  _id: ObjectId('68ec0e207acca404625104d7'),
  houseId: 4,
  name: 'Ravenclaw',
  mascot: 'Eagle'
}
]
[Atlas atlas-n7wwok-shard-0 [primary] web335DB> db.students.find()
[
  {
    _id: ObjectId('68ec0e207acca404625104d8'),
    studentId: 1001,
    firstName: 'Harry',
    lastName: 'Potter',
    houseId: 1
  },
  {
    _id: ObjectId('68ec0e207acca404625104d9'),
    studentId: 1002,
    firstName: 'Hermione',
    lastName: 'Granger',
    houseId: 1
  },
  {
    _id: ObjectId('68ec0e207acca404625104da'),
    studentId: 1003,
    firstName: 'Ron',
    lastName: 'Weasley',
    houseId: 1
  },
  {
    _id: ObjectId('68ec0e207acca404625104db'),
    studentId: 1004,
    firstName: 'Luna',
    lastName: 'Lovegood',
    houseId: 4
  },
  {
    _id: ObjectId('68ec0e207acca404625104dc'),
    studentId: 1005,
    firstName: 'Cho',
    lastName: 'Chang',
    houseId: 4
  },
  {
    _id: ObjectId('68ec0e207acca404625104dd'),
    studentId: 1006,
    firstName: 'Draco',
    lastName: 'Malfoy',
    houseId: 3
  }
]
[Atlas atlas-n7wwok-shard-0 [primary] web335DB> ]
```

Figure 3: Execution of newell_assignment6.2.js script, performing CRUD and agg

```
web-335 — mongosh mongodb+srv://<credentials>@web335cluster.sbfdyc0.mongodb.net/?retryWrites=true&w=ma...

{
  _id: ObjectId('68ec0e207acca404625104d5'),
  houseId: 2,
  name: 'Hufflepuff',
  mascot: 'Badger'
},
{
  _id: ObjectId('68ec0e207acca404625104d6'),
  houseId: 3,
  name: 'Slytherin',
  mascot: 'Serpent'
},
{
  _id: ObjectId('68ec0e207acca404625104d7'),
  houseId: 4,
  name: 'Ravenclaw',
  mascot: 'Eagle'
}
]
[Atlas atlas-n7wwok-shard-0 [primary] web335DB> db.students.find()
[
  {
    _id: ObjectId('68ec0e207acca404625104d8'),
    studentId: 1001,
    firstName: 'Harry',
    lastName: 'Potter',
    houseId: 1
  },
  {
    _id: ObjectId('68ec0e207acca404625104d9'),
    studentId: 1002,
    firstName: 'Hermione',
    lastName: 'Granger',
    houseId: 1
  },
  {
    _id: ObjectId('68ec0e207acca404625104da'),
    studentId: 1003,
    firstName: 'Ron',
    lastName: 'Weasley',
    houseId: 1
  },
  {
    _id: ObjectId('68ec0e207acca404625104db'),
    studentId: 1004,
    firstName: 'Luna',
    lastName: 'Lovegood',
    houseId: 4
  },
  {
    _id: ObjectId('68ec0e207acca404625104dc'),
    studentId: 1005,
    firstName: 'Cho',
    lastName: 'Chang',
    houseId: 4
  },
  {
    _id: ObjectId('68ec0e207acca404625104dd'),
    studentId: 1006,
    firstName: 'Draco',
    lastName: 'Malfoy',
    houseId: 3
  }
]
[Atlas atlas-n7wwok-shard-0 [primary] web335DB> load("week-6/newell_assignment6.2.js")
true
[Atlas atlas-n7wwok-shard-0 [primary] web335DB> ]
```

Figure 4: Aggregation query results displaying students grouped by house, confi

```

web-335 — mongosh mongodb+srv://<credentials>@web335cluster.sbfdyc0.mongodb.net/?retryWrites=true&w=ma...

{
  _id: ObjectId('68ec0e207acca404625104d9'),
  studentId: 1002,
  firstName: 'Hermione',
  lastName: 'Granger',
  houseId: 1
},
{
  _id: ObjectId('68ec0e207acca404625104da'),
  studentId: 1003,
  firstName: 'Ron',
  lastName: 'Weasley',
  houseId: 1
},
{
  _id: ObjectId('68ec0e207acca404625104db'),
  studentId: 1004,
  firstName: 'Luna',
  lastName: 'Lovegood',
  houseId: 4
},
{
  _id: ObjectId('68ec0e207acca404625104dc'),
  studentId: 1005,
  firstName: 'Cho',
  lastName: 'Chang',
  houseId: 4
},
{
  _id: ObjectId('68ec0e207acca404625104dd'),
  studentId: 1006,
  firstName: 'Draco',
  lastName: 'Malfoy',
  houseId: 3
}
]
[Atlas atlas-n7wwok-shard-0 [primary] web335DB> load("week-6/newell_assignment6.2.js")
true
Atlas atlas-n7wwok-shard-0 [primary] web335DB> db.students.aggregate([
| { $lookup: { from: "houses", localField: "houseId", foreignField: "houseId", as: "house" } },
| { $unwind: "$house" },
| { $group: { _id: "$house.name", students: { $push: { firstName: "$firstName", lastName: "$lastName" } } } },
| { $project: { _id: 0, House: "$_id", Students: "$students" } },
| { $sort: { House: 1 } }
| ])
[
|
| {
|   House: 'Gryffindor',
|   Students: [
|     { firstName: 'Harry', lastName: 'Potter' },
|     { firstName: 'Hermione', lastName: 'Granger' },
|     { firstName: 'Ron', lastName: 'Weasley' }
|   ]
| },
| {
|   House: 'Ravenclaw',
|   Students: [
|     { firstName: 'Luna', lastName: 'Lovegood' },
|     { firstName: 'Cho', lastName: 'Chang' }
|   ]
| },
| {
|   House: 'Slytherin',
|   Students: [ { firstName: 'Draco', lastName: 'Malfoy' } ]
| }
| ]
Atlas atlas-n7wwok-shard-0 [primary] web335DB>

```

Conclusion

All queries and aggregation operations were executed successfully in MongoDB Atlas.

The following learning objectives were demonstrated:

- Creation and manipulation of MongoDB collections using insertMany()
- Execution of CRUD operations (Insert, Update, Delete)
- Implementation of aggregation pipelines using \$lookup, \$unwind, \$group, and \$project
- Verification of dataset relationships between houses and students

This confirms full completion of Assignment 6.2 requirements.