

Mohamed Sameh Elkholly

MongoDB – Lab 3

Restore inventory database

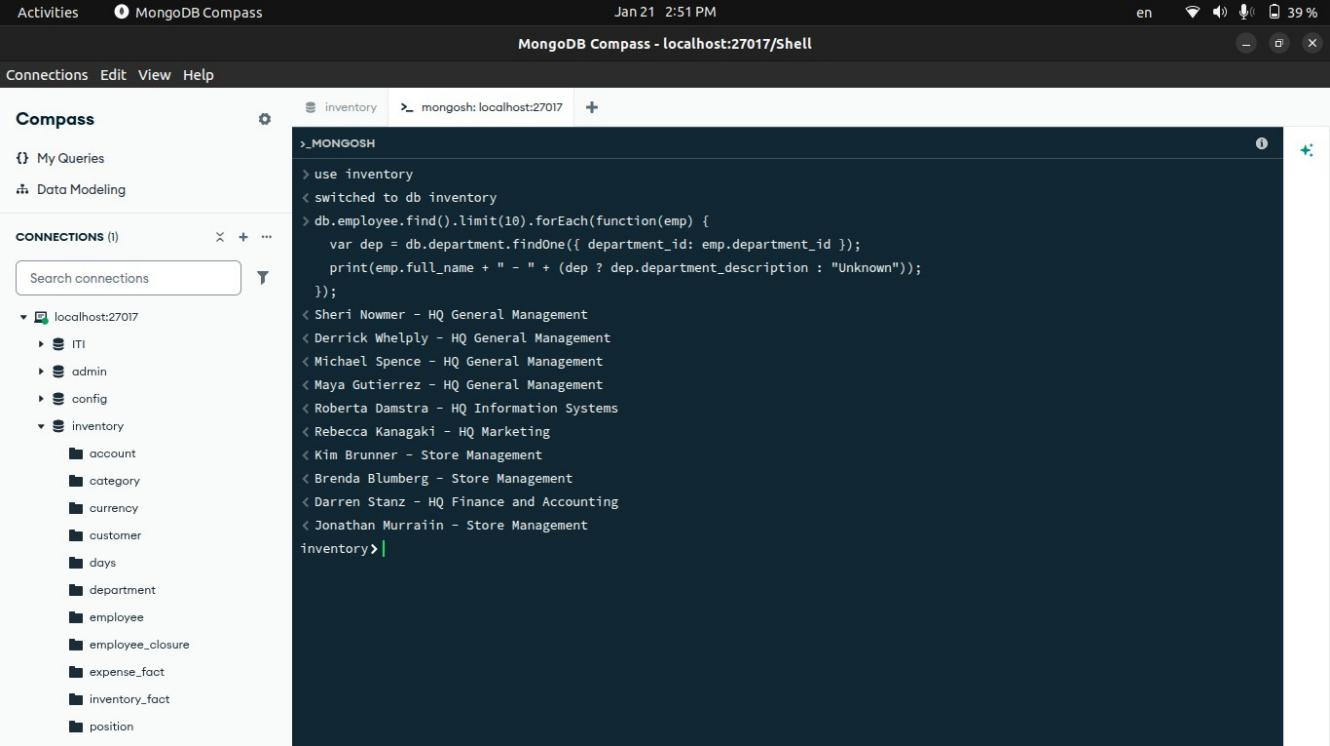
Write the following command on windows shell

mongorestore -d inventory “write folder path”

Ex: mongorestore -d inventory D:\intake39\MongoDB\Lab2\Inventory

Relations:

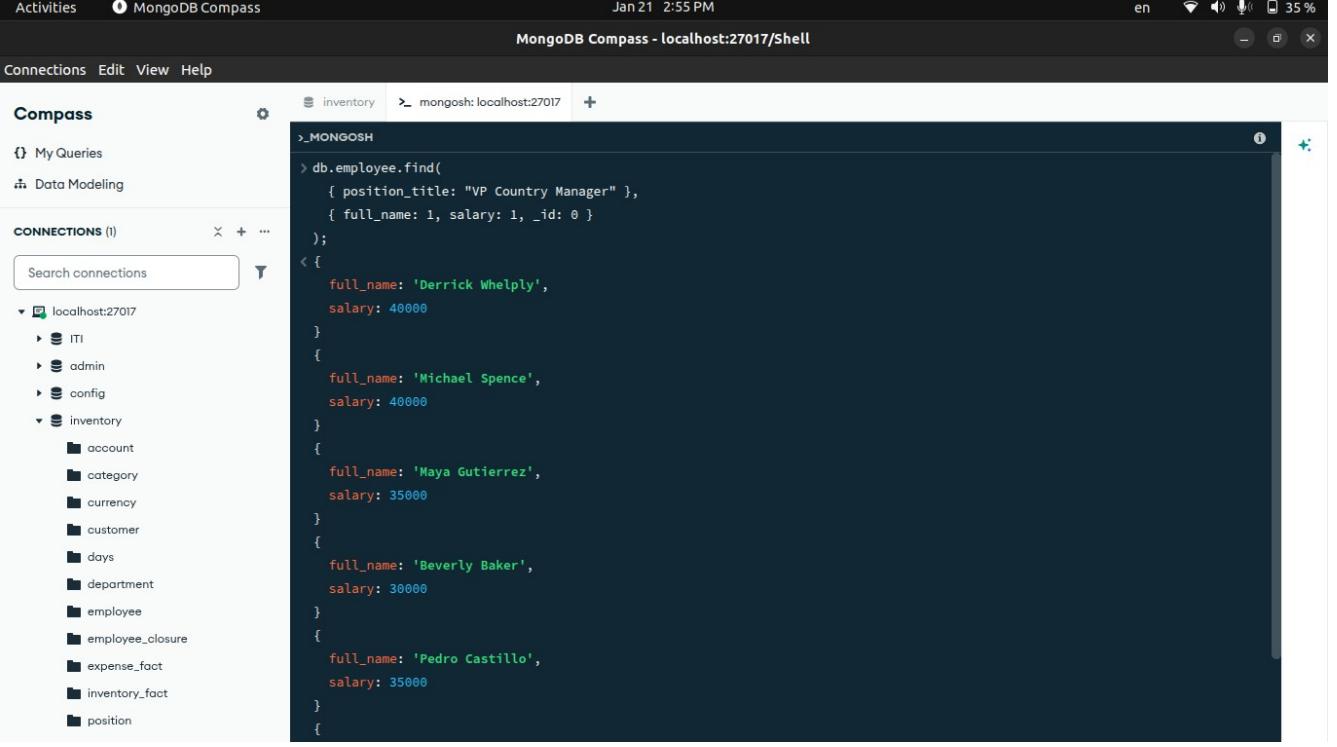
a- Display employees fullname and department name for all employees



The screenshot shows the MongoDB Compass interface. On the left, the sidebar displays the 'Compass' section with 'My Queries' and 'Data Modeling' options, and a 'CONNECTIONS (1)' section listing a connection to 'localhost:27017'. The main area shows the mongo shell interface with the following command and output:

```
> MONGOSH
> use inventory
< switched to db inventory
> db.employee.find().limit(10).forEach(function(emp) {
    var dep = db.department.findOne({ department_id: emp.department_id });
    print(emp.full_name + " - " + (dep ? dep.department_description : "Unknown"));
});
< Sheri Nowmer - HQ General Management
< Derrick Whelply - HQ General Management
< Michael Spence - HQ General Management
< Maya Gutierrez - HQ General Management
< Roberta Damstra - HQ Information Systems
< Rebecca Kanagaki - HQ Marketing
< Kim Brunner - Store Management
< Brenda Blumberg - Store Management
< Darren Stanz - HQ Finance and Accounting
< Jonathan Murrain - Store Management
inventory> |
```

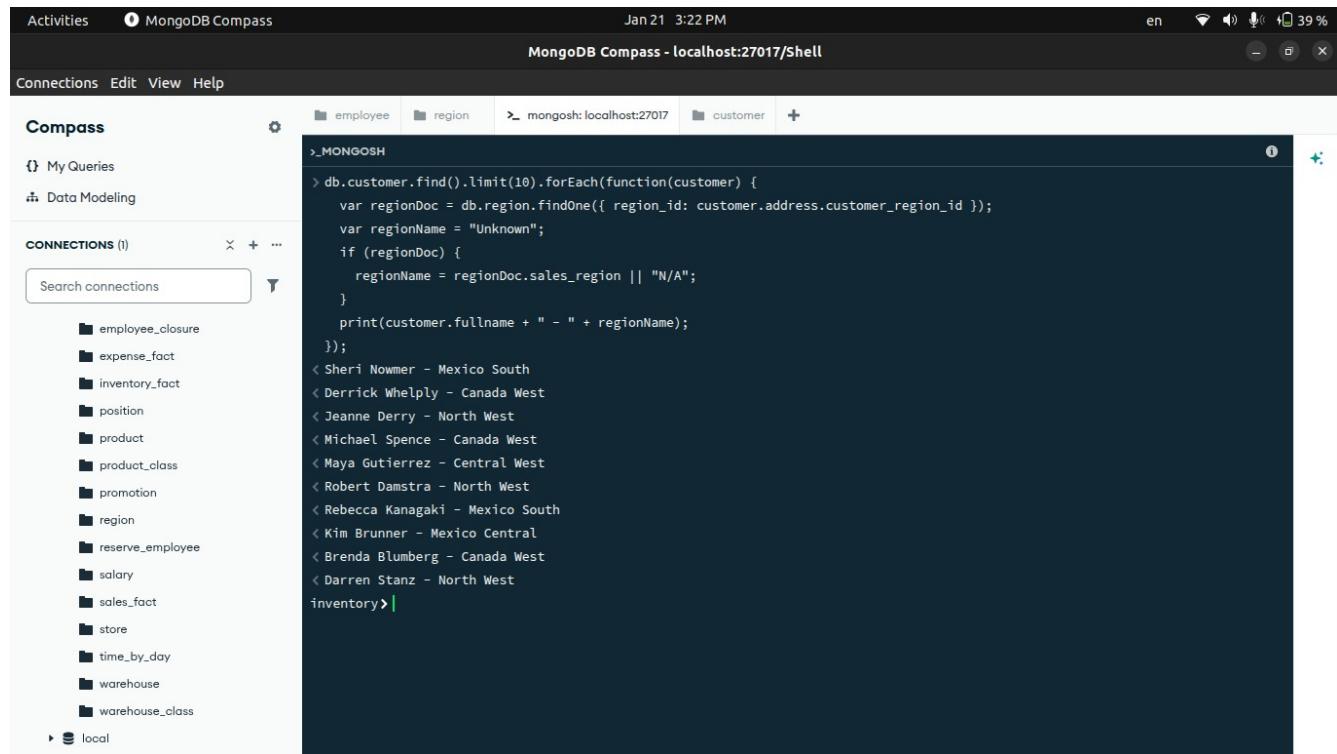
b- Display employees with position “VP Country Manager” (only display employee full name and salary).



The screenshot shows the MongoDB Compass interface. The left sidebar displays the 'Connections' section, which lists a single connection to 'localhost:27017'. The main area shows a query results pane titled 'mongosh: localhost:27017/Shell'. The query is:

```
>_MONGOSH
> db.employee.find(
  { position_title: "VP Country Manager" },
  { full_name: 1, salary: 1, _id: 0 }
);
< [
  {
    full_name: 'Derrick Whelply',
    salary: 40000
  },
  {
    full_name: 'Michael Spence',
    salary: 40000
  },
  {
    full_name: 'Maya Gutierrez',
    salary: 35000
  },
  {
    full_name: 'Beverly Baker',
    salary: 30000
  },
  {
    full_name: 'Pedro Castillo',
    salary: 35000
  }
]
```

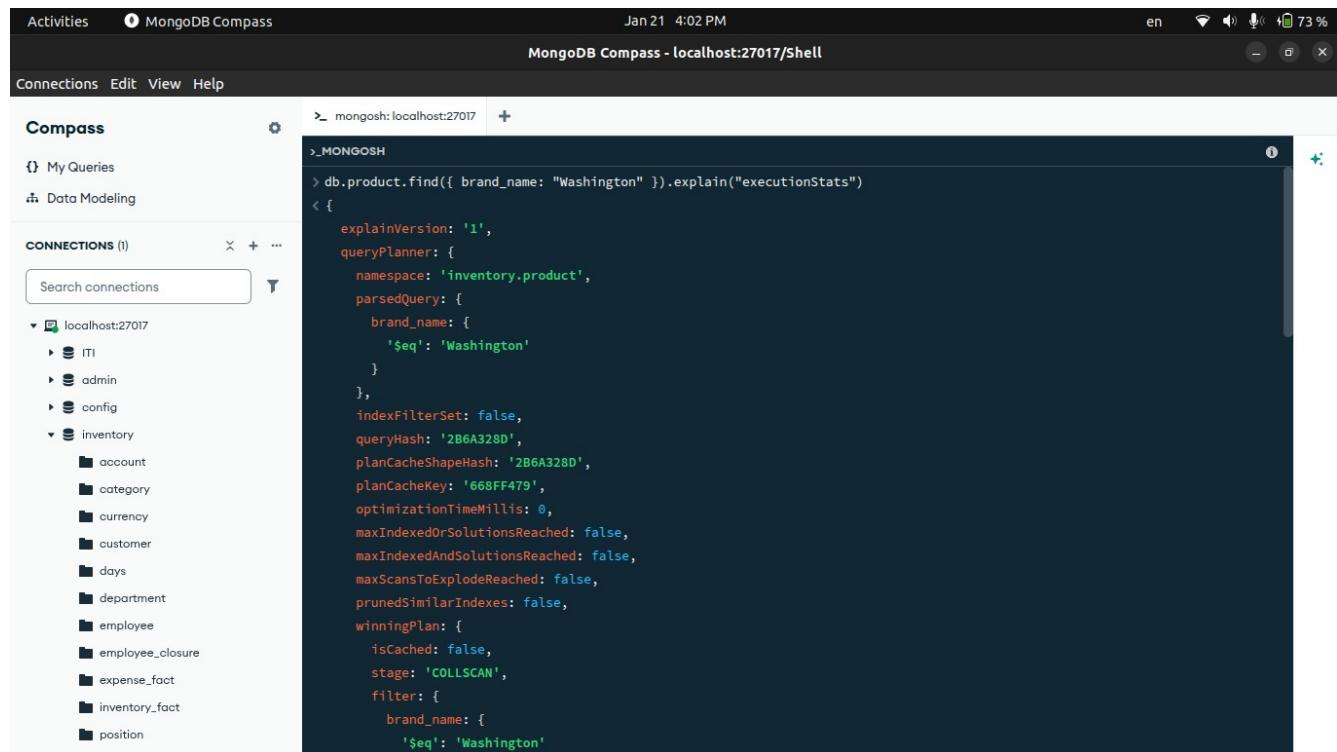
c- Display customers full names and their regions.



The screenshot shows the MongoDB Compass interface. The left sidebar displays a list of connections, including 'employee_closure', 'expense_fact', 'inventory_fact', 'position', 'product', 'product_class', 'promotion', 'region', 'reserve_employee', 'salary', 'sales_fact', 'store', 'time_by_day', 'warehouse', and 'warehouse_class'. The main panel shows a MongoDB shell command window with the following code:

```
>_MONGOSH
> db.customer.find().limit(10).forEach(function(customer) {
  var regionDoc = db.region.findOne({ region_id: customer.address.customer_region_id });
  var regionName = "Unknown";
  if (regionDoc) {
    regionName = regionDoc.sales_region || "N/A";
  }
  print(customer.fullname + " - " + regionName);
});
< Sheri Nomer - Mexico South
< Derrick Whelby - Canada West
< Jeanne Derry - North West
< Michael Spence - Canada West
< Maya Gutierrez - Central West
< Robert Damstra - North West
< Rebecca Kanagaki - Mexico South
< Kim Brunner - Mexico Central
< Brenda Blumberg - Canada West
< Darren Stanz - North West
inventory> |
```

d- In product find all products that was branded by " Washington"
(try to createIndex on brand_name and test your query speed)



The screenshot shows the MongoDB Compass interface. The left sidebar displays a list of connections, including 'localhost:27017' (selected), 'ITI', 'admin', 'config', and 'inventory'. Under 'inventory', there are sub-items: 'account', 'category', 'currency', 'customer', 'days', 'department', 'employee', 'employee_closure', 'expense_fact', 'inventory_fact', and 'position'. The main panel shows a MongoDB shell command window with the following code:

```
>_mongosh
> db.product.find({ brand_name: "Washington" }).explain("executionStats")
< {
  explainVersion: '1',
  queryPlanner: {
    namespace: 'inventory.product',
    parsedQuery: {
      brand_name: {
        '$eq': 'Washington'
      }
    },
    indexFilterSet: false,
    queryHash: '2B6A328D',
    planCacheShapeHash: '2B6A328D',
    planCacheKey: '668FF479',
    optimizationTimeMillis: 0,
    maxIndexedOrSolutionsReached: false,
    maxIndexedAndSolutionsReached: false,
    maxScansToExplodeReached: false,
    prunedSimilarIndexes: false,
    winningPlan: {
      isCached: false,
      stage: 'COLLSCAN',
      filter: {
        brand_name: {
          '$eq': 'Washington'
        }
      }
    }
  }
}
```

Activities MongoDB Compass Jan 21 4:02 PM en 75 %

MongoDB Compass - localhost:27017/Shell

Connections Edit View Help

Compass

- My Queries
- Data Modeling

CONNECTIONS (1)

localhost:27017

- IT
- admin
- config
- inventory**
 - account
 - category
 - currency
 - customer
 - days
 - department
 - employee
 - employee_closure
 - expense_fact
 - inventory_fact
 - position

```
>_ mongosh:localhost:27017 + 
>_MONGOSH
> db.product.createIndex({ brand_name: 1 })
< brand_name_1
> db.product.find({ brand_name: "Washington" }).explain("executionStats")
< {
  explainVersion: '1',
  queryPlanner: {
    namespace: 'inventory.product',
    parsedQuery: {
      brand_name: {
        '$eq': 'Washington'
      }
    },
    indexFilterSet: false,
    queryHash: '2B6A328D',
    planCacheShapeHash: '2B6A328D',
    planCacheKey: 'E5707192',
    optimizationTimeMillis: 0,
    maxIndexedOrSolutionsReached: false,
    maxIndexedAndSolutionsReached: false,
    maxScansToExplodeReached: false,
    prunedSimilarIndexes: false,
    winningPlan: {
      isCached: false,
      stage: 'FETCH',
      inputStage: {

```

Aggregate function:

a- Group products by brand name, count number and display the result as follow:

Activities MongoDB Compass Jan 21 3:29 PM en 44 %

MongoDB Compass - localhost:27017/Shell

Connections Edit View Help

Compass

- My Queries
- Data Modeling

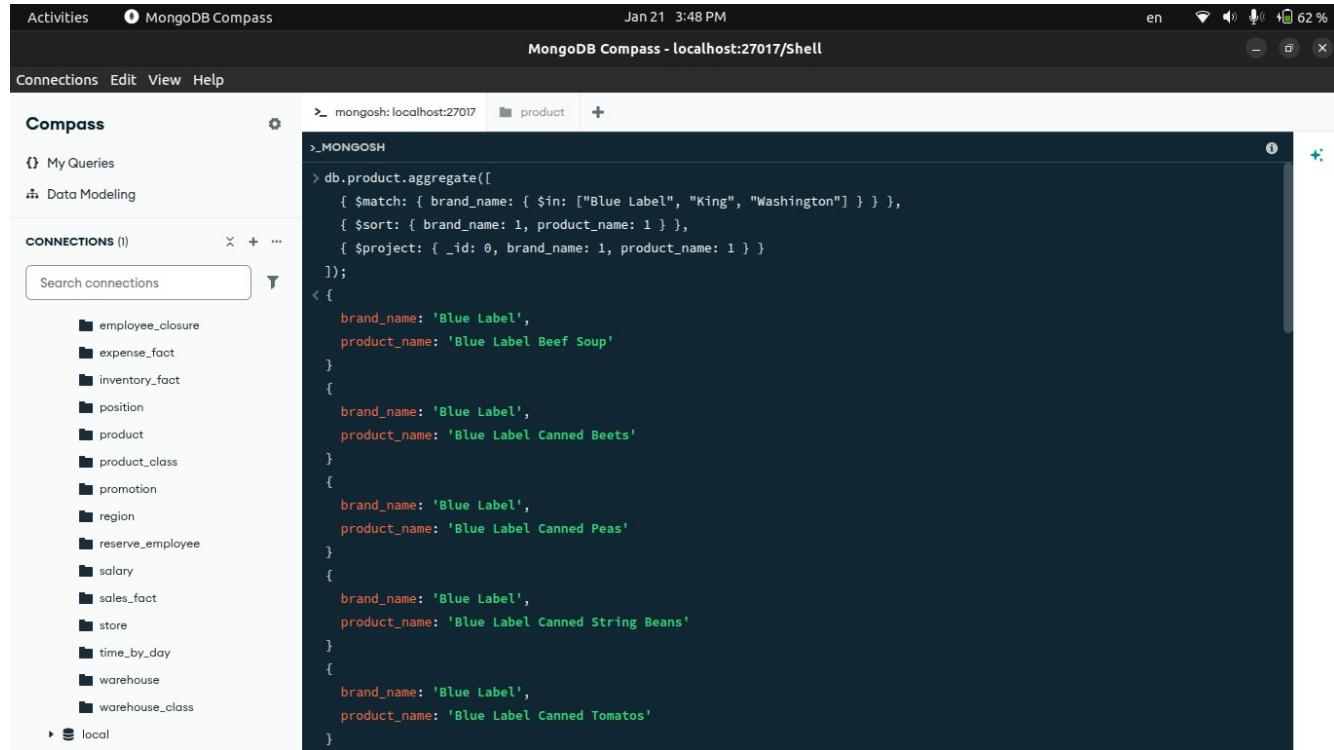
CONNECTIONS (1)

localhost:27017

- employee_closure
- expense_fact
- inventory_fact
- position
- product**
- product_class
- promotion
- region
- reserve_employee
- salary
- sales_fact
- store
- time_by_day
- warehouse
- warehouse_class

```
>_ mongosh:localhost:27017 product + 
>_MONGOSH
> db.product.aggregate([
  { $group: { _id: "$brand_name", count: { $sum: 1 } } }
]);
< [
  {
    _id: 'Jeffers',
    count: 4
  },
  {
    _id: 'High Quality',
    count: 32
  },
  {
    _id: 'Colossal',
    count: 5
  },
  {
    _id: 'Great',
    count: 9
  },
  {
    _id: 'Fabulous',
    count: 11
  },
  {
    _id: 'Big City',
    count: 1
  }
]
```

b- Group products by brand_name and product_name ,only select brand names ("Blue Label","King","Washington") then sort them by brand_name andproduct_name ascending and display result as follow



The screenshot shows the MongoDB Compass interface. On the left, the sidebar lists various database collections: employee_closure, expense_fact, inventory_fact, position, product, product_class, promotion, region, reserve_employee, salary, sales_fact, store, time_by_day, warehouse, and warehouse_class. The main panel displays a MongoDB shell command in the 'mongosh:localhost:27017' database:

```
>_MONGOSH
> db.product.aggregate([
  { $match: { brand_name: { $in: ["Blue Label", "King", "Washington"] } } },
  { $sort: { brand_name: 1, product_name: 1 } },
  { $project: { _id: 0, brand_name: 1, product_name: 1 } }
]);
< [
  {
    brand_name: 'Blue Label',
    product_name: 'Blue Label Beef Soup'
  },
  {
    brand_name: 'Blue Label',
    product_name: 'Blue Label Canned Beets'
  },
  {
    brand_name: 'Blue Label',
    product_name: 'Blue Label Canned Peas'
  },
  {
    brand_name: 'Blue Label',
    product_name: 'Blue Label Canned String Beans'
  },
  {
    brand_name: 'Blue Label',
    product_name: 'Blue Label Canned Tomatos'
  }
]
```

BONUS:Display maximum salary for each department (display department name and maximum salary)