

Database Systems

Lab 13

Shazain

1) 1

```
>_MONGOSH

> db.restaurants.aggregate([
  {
    $addFields: {
      avgScore: { $avg: "$grades.score" }
    }
  },

  {
    $project: {
      "name": 1,
      "avgScore": 1
    }
  }
])
< {
  _id: ObjectId('673ec62f3a635a3395746f3c'),
  name: 'Morris Park Bake Shop',
  avgScore: 8.2
}
{
  _id: ObjectId('673ec62f3a635a3395746f3d'),
  name: 'Wendy'S',
  avgScore: 13.75
}
{
  _id: ObjectId('673ec62f3a635a3395746f3e'),
  name: 'Dj Reynolds Pub And Restaurant',
  avgScore: 9.25
}
{
  _id: ObjectId('673ec62f3a635a3395746f3f'),
  name: 'Riviera Caterer',
  avgScore: 9
}
{
  _id: ObjectId('673ec62f3a635a3395746f40'),
  name: 'Tov Kosher Kitchen',

```

2) 2

```
> _MONGOSH
> db.restaurants.aggregate(
  [
    {$unwind: "$grades"},
    {$match: {} },
    {$group: {_id: "$cuisine",
              max: {$max: "$grades.score"},
              avg: {$avg: "$grades.score"},
              min: {$min: "$grades.score"}
            }}
  ])
< {
  _id: 'Pancakes/Waffles',
  max: 37,
  avg: 11.210526315789474,
  min: 3
}
{
  _id: 'Mediterranean',
  max: 27,
  avg: 11.764705882352942,
  min: 2
}
{
  _id: 'Chinese/Cuban',
  max: 29,
  avg: 12.44,
  min: 3
}
{
  _id: 'Brazilian',
  max: 51,
  avg: 12.064516129032258,
  min: 2
}
{
  _id: 'German',
  max: 27,
  avg: 11.076923076923077,
  min: 3
}
```

3) 3

```
>_MONGOSH ⓘ  
  
> db.restaurants.aggregate([  
  {  
    $group: {  
      _id: "$borough",  
      count: {$sum: 1}  
    }  
  },  
  {}  
)  
< {  
  _id: 'Manhattan',  
  count: 1883  
}  
{  
  _id: 'Bronx',  
  count: 309  
}  
{  
  _id: 'Queens',  
  count: 738  
}  
{  
  _id: 'Brooklyn',  
  count: 684  
}  
{  
  _id: 'Staten Island',  
  count: 158  
}  
Atlas atlas-gcwai0-shard-0 [primary] iba_db>
```

4) 4

```
>_MONGOSH
> db.restaurants.aggregate([
  {
    $group: {
      _id: {cuisine: "$cuisine",
        borough: "$borough"},
      restCount: {$sum: 1}
    }
  }
])
< {
  _id: {
    cuisine: 'Bakery',
    borough: 'Brooklyn'
  },
  restCount: 29
}
{
  _id: {
    cuisine: 'Armenian',
    borough: 'Manhattan'
  },
  restCount: 1
}
{
  _id: {
    cuisine: 'Mediterranean',
    borough: 'Manhattan'
  },
  restCount: 11
}
{
  _id: {
    cuisine: 'Chicken',
    borough: 'Bronx'
  },
  restCount: 7
}
```

5) 5

```
>_MONGOSH ⓘ
> db.restaurants.aggregate([
  {
    $match: {borough: "Brooklyn"}
  },
  {
    $unwind: "$grades"
  },
  {
    $group: {_id: "$cuisine",
             "average": {$avg: "$grades.score"}}
  },
  {
    $sort: {average: -1}
  },
  {
    $limit: 2
  },
  {
    $project: {"borough": 1, "cuisine": 1, "average": 1}
  }
])
< {
  _id: 'Asian',
  average: 16.4
}
{
  _id: 'Japanese',
  average: 15.888888888888889
}
Atlas atlas-gcwai0-shard-0 [primary] iba_db>
```

6) 6

```
>_MONGOSH ⓘ

> db.restaurants.insertOne({
  "address": {
    "building": "1007",
    "coord": [-73.856077, 40.848447],
    "street": "IBA st.",
    "zipcode": "10001"
  },
  "borough": "Karachi",
  "cuisine": "All",
  "grades": [
    {
      "date": new Date("2014-03-03T00:00:00Z"),
      "grade": "A",
      "score": 10
    },
    {
      "date": new Date("2011-03-10T00:00:00Z"),
      "grade": "B",
      "score": 14
    }
  ],
  "name": "IBA Students Canteen",
  "restaurant_id": "12340000"
})
< {
  acknowledged: true,
  insertedId: ObjectId('6748080692ac2b1afd7312b3')
}
Atlas atlas-gcwai0-shard-0 [primary] iba_db> |
```

7) 7

```
>_MONGOSH
{
  date: 2012-12-05T00:00:00.000Z,
  grade: 'A',
  score: 13
},
{
  date: 2012-05-17T00:00:00.000Z,
  grade: 'A',
  score: 11
}
],
name: 'The Movable Feast',
restaurant_id: '40361606'
}
Type "it" for more
> db.restaurants.updateMany(
  {
    borough: "Brooklyn",
    cuisine: "Bakery"
  },
  {
    $set: {cuisine: "Bakery and Icecream"}
  },
  {
    multi: true
  }
)
< {
  acknowledged: true,
  insertedId: null,
  matchedCount: 29,
  modifiedCount: 29,
  upsertedCount: 0
}
>
>
>
Atlas atlas-gcwa10-shard-0 [primary] iba_db>
```

8) 8

9) 9

```
>_MONGOSH
> db.restaurants.find(
  {
    "grades": {
      $not: {
        $elemMatch: {"score": {$lte: 5}}
      }
    }
  },
  {
    _id: 1,
    restaurantid: 1,
    grades: 1
  }
)
< {
  _id: ObjectId('674809ce48cc6d44f0c06843'),
  grades: [
    {
      date: 2014-12-30T00:00:00.000Z,
      grade: 'A',
      score: 8
    },
    {
      date: 2014-07-01T00:00:00.000Z,
      grade: 'B',
      score: 23
    },
    {
      date: 2013-04-30T00:00:00.000Z,
      grade: 'A',
      score: 12
    },
    {
      date: 2012-05-08T00:00:00.000Z,
      grade: 'A',
      score: 12
    }
  ]
}
```

10)

```
>_MONGOSH
> db.createCollection("Orders")
< { ok: 1 }
```

A

```
>_MONGOSH
> db.orders.aggregate([
  {
    $group: {_id: "$Rep",
             sum: {$sum: "$Total"}}
  }
])
< {
  _id: 'Jardine',
  sum: 2812.19
}
{
  _id: 'Jones',
  sum: 2363.04
}
{
  _id: 'Gill',
  sum: 1749.8700000000001
}
{
  _id: 'Morgan',
  sum: 1387.77
}
{
  _id: 'Thompson',
  sum: 1203.1100000000001
}
{
  _id: 'Smith',
  sum: 1641.43
}
{
  _id: 'Sorvino',
  sum: 1283.6100000000001
}
,
  _id: 'Jardine',
```

B

```
>_MONGOSH ❗  
  
> db.orders.distinct("Item")  
< [ 'Binder', 'Desk', 'Pen', 'Pen Set', 'Pencil' ]  
Atlas atlas-gcwai0-shard-0 [primary] iba_db>
```

C

```
>_MONGOSH ❗  
  
> db.orders.aggregate([  
  {  
    $group: {_id: "$Item", items: {$sum: "$Units"}}  
  }  
  
  ])  
< {  
  _id: 'Binder',  
  items: 722  
}  
{  
  _id: 'Desk',  
  items: 10  
}  
{  
  _id: 'Pen',  
  items: 278  
}  
{  
  _id: 'Pen Set',  
  items: 395  
}  
{  
  _id: 'Pencil',  
  items: 716  
}  
Atlas atlas-gcwai0-shard-0 [primary] iba_db>
```

D

```
>_MONGOSH ⓘ  
  
> db.orders.aggregate([  
  {  
    $group: {_id: "$Region", count: {$sum: 1}}  
  }  
  
  ])  
< {  
  _id: 'East',  
  count: 13  
}  
{  
  _id: 'West',  
  count: 6  
}  
{  
  _id: 'Central',  
  count: 24  
}  
Atlas atlas-gcwa10-shard-0 [primary] 1ba_db>
```

E

```
>_MONGOSH ⓘ  
  
> db.orders.aggregate([  
  {  
    $match: {Region: "Central"}  
  },  
  {  
    $group: {_id: "$Region", orders: {$sum: 1}}  
  }  
  ])  
< {  
  _id: 'Central',  
  orders: 24  
}  
Atlas atlas-gcwa10-shard-0 [primary] 1ba_db>
```

F

```
>_MONGOSH ⓘ

> db.orders.aggregate([
  {
    $group: {
      _id: "$Item",
      totalunits: {$sum: "$Units"}
    }
  },
  {
    $sort: {totalunits: -1}
  }
])
< {
  _id: 'Binder',
  totalunits: 722
}
{
  _id: 'Pencil',
  totalunits: 716
}
{
  _id: 'Pen Set',
  totalunits: 395
}
{
  _id: 'Pen',
  totalunits: 278
}
{
  _id: 'Desk',
  totalunits: 10
}
Atlas atlas-gcwa10-shard-0 [primary] 1ba_db>|
```

SCENARIO

```
>_MONGOSH ❏
> db.createCollection("SalesRep", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["s_name", "s_email", "s_performance_score", "customers"],
      properties: {
        s_name: {
          bsonType: "string",
          description: "SalesRep name is required and must be a string."
        },
        s_email: {
          bsonType: "string",
          pattern: "^.+@.+\\.\\.+$",
          description: "SalesRep email is required, must be a valid email format."
        },
        s_performance_score: {
          bsonType: "int",
          minimum: 1,
          maximum: 10,
          description: "Performance score is required and must be between 1 and 10."
        },
        customers: {
          bsonType: "array",
          maxItems: 10,
          items: {
            bsonType: "object",
            required: ["cust_name", "cust_email"],
            properties: {
              cust_name: {
                bsonType: "string",
                description: "Customer name is required and must be a string."
              },
              cust_email: {
                bsonType: "string",
                pattern: "^.+@.+\\.\\.+$",
                description: "Customer email is required and must be a valid email format."
              }
            }
          }
        },
        description: "A SalesRep can deal with a maximum of 10 customers."
      }
    }
  }
})
< { ok: 1 }
Atlas atlas-gcwai0-shard-0 [primary] iba_db>
```

>_MONGOSH



```
> db.createCollection("Customer", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["cust_name", "cust_add", "cust_email", "salesReps"],
      properties: {
        cust_name: {
          bsonType: "string",
          description: "Customer name is required and must be a string."
        },
        cust_add: {
          bsonType: "string",
          description: "Customer address is required and must be a string."
        },
        cust_email: {
          bsonType: "string",
          pattern: "^.+@.+\\.\\.+$",
          description: "Customer email is required and must be a valid email format."
        },
        salesReps: {
          bsonType: "array",
          maxItems: 5,
          items: {
            bsonType: "object",
            required: ["s_name", "s_email"],
            properties: {
              s_name: {
                bsonType: "string",
                description: "SalesRep name is required and must be a string."
              },
              s_email: {
                bsonType: "string",
                pattern: "^.+@.+\\.\\.+$",
                description: "SalesRep email is required and must be a valid email format."
              }
            }
          }
        },
        description: "A Customer can be dealt with by a maximum of 5 SalesReps."
      }
    }
  }
})
```

```
< { ok: 1 }
```

```
Atlas atlas-gcwai0-shard-0 [primary] iba_db>
```

```
> db.createCollection("Department", {
  validator: {
    $jsonSchema: {
      bsonType: "object",
      required: ["dept_name", "dept_region", "grade", "manager"],
      properties: {
        dept_name: {
          bsonType: "string",
          description: "Department name is required and must be a string."
        },
        dept_region: {
          enum: ["East", "West", "Central"],
          description: "Region must be East, West, or Central."
        },
        grade: {
          bsonType: "string",
          minLength: 1,
          maxLength: 1,
          description: "Grade is required and must be a single character string."
        },
        manager: {
          bsonType: "object",
          required: ["s_name", "s_email"],
          properties: {
            s_name: {
              bsonType: "string",
              description: "Manager name is required and must be a string."
            },
            s_email: {
              bsonType: "string",
              pattern: "^.+@.+\.\.+$",
              description: "Manager email is required and must be a valid email format."
            }
          }
        }
      },
      description: "Each department must have a manager who is a SalesRep."
    }
  }
})
< { ok: 1 }
```

Atlas atlas-gcwai0-shard-0 [primary] iba_db> |