Name: Russell Schlup

This is a final exam for ADB. The final will cover MongoDB and will be worth 35% of your total grade. Each task is worth 10 points. Partial credit will be awarded.

Please read the entire questions. If you are asked for multiple parts in an answer (Statement & Results), each part is worth 50% of the total question value.

In the examples below, text in ALL CAPS needs to be replaced with a value.

IE: Replace VENDOR NAME with Apple, Samsung, GNC or Acme in the statement:

```
_id: VENDOR NAME
```

Text is ALL CAPITALS needs to be replaced with some data of your choice.

Do not put spaces in the _id fields.

Start Exam

1. Start Mongo and create a new database called “adb1403f” with a collection called ‘products’.

We will be storing inventory for a store. Their inventory is purchased from multiple vendors located across the country.

2. Create documents for vendors in the ADB collection. (Vendors can be fictional.)

Four vendor documents total. **Create the object first** in the format below, **then save the object** to the database.(2 steps). There should be no spaces in the _id or name fields. We’re using a natural key for the _id field. The phone number type should be “main”, “customer service”, or “fax”. Two phone numbers are required per vendor. Also, remember MongoDB is case sensitive.

Paste all 8 statements below.

```
_id: VENDOR NAME
name: VENDOR NAME
type:”vendor”
address: ADDRESS
city: CITY
state: STATE
ll: [LONGITUDE,LATTITUDE]        (Must be an array of numeric values)
phone: [
            {type:TYPE, number:NUMBER},
            {type:TYPE, number:NUMBER}
      ]
```

Use the data below to create your vendors.

Address	Longitude	Latitude
4092 Eastgate Drive, Orlando, FL	-79.441833	44.012893
451 E Altamonte Dr, Altamonte Springs	-81.375883	28.667207
One Microsoft Way, Redmond, WA	-122.131378	47.638197
381 Brea Canyon, Road Walnut, CA	-117.844840	34.013444

```
ikea = {_id: 'IKEA',name: 'IKEA',type:"vendor",address: "4092 Eastgate Drive",city: "Orlando",state: "FL",ll: [-79.441833,44.012893],phone: [{type:'main', number:'(888) 888-4532'},{type:'fax', number:'(888) 888-4532'}]}
go = {_id: 'Go! Calendars', name: 'Go! Calendars',type:"vendor",address: "451 E Altamonte Dr",city: "Altamonte Springs",state: "FL",ll: [-81.375883,28.667207],phone: [{type:'main', number:'(321) 207-0021'},{type:'fax', number:'(321) 207-0021'}]}
microsoft = {_id: 'Microsoft Corporation',name: 'Microsoft Corporation',type:"vendor",address:
```

"One Microsoft Way",city: "Redmond",state: "WA",ll: [-122.131378,47.638197],phone: [{type:'main', number:'(425) 882-8080'},{type:'fax', number:'(425) 882-8080'}]}

viewsonic = {_id: 'ViewSonic Corporation',name: 'ViewSonic Corporation',type:"vendor",address: "381 Brea Canyon, Road",city: "Walnut",state: "CA",ll: [-117.844840,34.013444],phone: [{type:'main', number:'(425) 882-8080'},{type:'fax', number:'(425) 882-8080'}]}

db.adb.insert(ikea)

db.adb.insert(go)

db.adb.insert(microsoft)

db.adb.insert(viewsonic)

3. Create documents for products in the ADB collection.

5 products for 1 vendor

3 products for 1 vendor

2 products for 1 vendor

2 products for 1 vendor

Twelve products over 4 vendors. Save the data to the database using a **single statement** for each document. Use the format below. Ex: Vendor=Apple, Product=IPad

Features should be an array of features (strings). Each array should be different and contain 1-3 features.

Feature Ex: [“bluetooth”,”WiFi”,”Retina Display”,”Shock Resistant”]

Paste the 12 statements below.

_id: PRODUCT NAME

name: PRODUCT NAME

type: “product”

vendor: VENDOR NAME

category: CATEGORY (string - Electronic, TV, Clothing, Health, Furniture…)

features []

db.adb.insert({_id: 'xbox360',name: 'Xbox 360',type: "product",vendor: "Microsoft Corporation",category: "Electronic",features: ["RGBY Compatible", "Wi-Fi", "Wireless Controllers"]})

db.adb.insert({_id: 'zune',name: 'Zune',type: "product",vendor: "Microsoft Corporation",category: "Electronic",features: ["Mp3", "Mp4", "Wi-Fi"]})

db.adb.insert({_id: 'microsoftsurface',name: 'Microsoft Surface',type: "product",vendor: "Microsoft Corporation",category: "Electronic",features: ["USB", "Stand Up", "Wi-Fi"]})

db.adb.insert({_id: 'xboxone',name: 'Xbox One',type: "product",vendor: "Microsoft

```
Corporation",category: "Electronic",features: ["HDMI", "1080 p", "Wi-Fi"]})
db.adb.insert({_id: 'microsoftmouse',name: 'Microsoft Mouse',type: "product",vendor: "Microsoft Corporation",category: "Electronic",features: ["USB", "Blue Tooth", "Rechargeable"]})
db.adb.insert({_id: 'arholma',name: 'Arholma',type: "product",vendor: "IKEA",category: "Furniture",features: ["Modular", "Combination"]})
db.adb.insert({_id: 'falster',name: 'Falster',type: "product",vendor: "IKEA",category: "Furniture",features: ["Wood", "Patio", 'Paintable']})
db.adb.insert({_id: 'applaro',name: 'Applaro',type: "product",vendor: "IKEA",category: "Furniture",features: ["Drop-Leaf", "Table", 'Wood']})
db.adb.insert({_id: 'borisvallejo',name: "Boris Vallejo & Julie Bell's Fantasy 2014 Wall Calendar",type: "product",vendor: "Go! Calendars",category: "Electronic",features: ["12-month span", "mediumsquare"]})
db.adb.insert({_id: 'badcat2014wallcalendar',name: 'Bad Cat 2014 Wall Calendar',type: "product",vendor: "Go! Calendars",category: "Electronic",features ["12-month span", "mediumsquare"]})
db.adb.insert({_id: 'vt4200l',name: 'VT4200-L',type: "product",vendor: "ViewSonic Corporation",category: "Electronic",features: ["HDMI", "RGB", 'Wi-Fi']})
db.adb.insert({_id: 'pro9000',name: 'Pro9000',type: "product",vendor: "ViewSonic Corporation",category: "Electronic",features: ["HDMI", "RGB", 'Wi-Fi']})
```

4. Create an index (ascending) on the name field. Then run the command to list all of your indexes. Paste both commands and results.

```
> db.adb.ensureIndex({"name":1})
> db.adb.getIndexes()
[
        {
                "v" : 1,
                "key" : {
                        "_id" : 1
                },
                "ns" : "adb1403f.adb",
                "name" : "_id_"
        },
        {
                "v" : 1,
                "key" : {
                        "name" : 1
```

```
                },
                "ns" : "adb1403f.adb",
                "name" : "name_1"
        }
]
```

5. Write a command to return one product document (type="product") by querying the vendor field (vendor = ???). Do not use findOne. Only one document should be returned. Paste command and results.

```
> db.adb.find({type:'product',vendor:'Microsoft Corporation'}).limit(1)
{ "_id" : "Xbox 360", "name" : "Xbox 360", "type" : "product", "vendor" : "Microsoft Corporation", "category" : "Electronic", "features" : [  "RGBY Compatible",  "Wi-Fi",  "Wireless Controllers" ] }
```

6. Write a command to return the products of 2 different vendors.

Paste command and results.

```
>  db.adb.find({type:'product',vendor:{$in:['ViewSonic Corporation','Go! Calendars']}})
{ "_id" : "badcat2014wallcalendar", "name" : "Boris Vallejo & Julie Bell's Fantasy 2014 Wall Calendar", "type" : "product", "vendor" : "Go! Calendars", "category" : "Electronic", "features" : [ "12-month span",  "mediumsquare" ] }
{ "_id" : "vt4200l", "name" : "VT4200-L", "type" : "product", "vendor" : "ViewSonic Corporation", "category" : "Electronic", "features" : [  "HDMI",  "RGB",  "Wi-Fi" ] }
{ "_id" : "pro9000", "name" : "Pro9000", "type" : "product", "vendor" : "ViewSonic Corporation", "category" : "Electronic", "features" : [  "HDMI",  "RGB",  "Wi-Fi" ] }
{ "_id" : "borisvallejo", "name" : "Boris Vallejo & Julie Bell's Fantasy 2014 Wall Calendar", "type" : "product", "vendor" : "Go! Calendars", "category" : "Electronic", "features" : [  "12-month span", "mediumsquare" ] }
```

7. Write a command to return vendors, by querying a specific phone number.

(Choose a number that exist in one of your vendor documents)

Your logic should be: Where type = vendor and phone number = NUMBER

Paste command and results.

```
> db.adb.find({type:'vendor',phone:{$elemMatch:{number:'(888) 888-4532'}}})
{ "_id" : "ikea", "name" : "IKEA", "type" : "vendor", "address" : "4092 Eastgate Drive", "city" : "Orlando", "state" : "FL", "ll" : [  -79.441833,  44.012893 ], "phone" : [    {           "type" : "main",           "number" : "(888) 888-4532" }, {     "type" : "fax",  "number" : "(888) 888-4532" } ] }
```

8. Add a new field called “rating” to 8 product documents using the update command with $set. These values should not repeat (all ratings cannot be 7) and must be a numeric. All vendors should have a rating on at least 1 product. Paste ALL commands below.

rating: Number between 1-10 (This is the consumer rating.)

```
db.adb.update({_id:'xbox360'},{$set:{rating:5}})
db.adb.update({_id:'zune'},{$set:{rating:3}})
db.adb.update({_id:'xboxone'},{$set:{rating:1}})
db.adb.update({_id:'arholma'},{$set:{rating:7}})
db.adb.update({_id:'falster'},{$set:{rating:8}})
db.adb.update({_id:'vt4200l'},{$set:{rating:9}})
db.adb.update({_id:'badcat2014wallcalendar'},{$set:{rating:6}})
db.adb.update({_id:'borisvallejo'},{$set:{rating:4}})
```

9. Write a command to add “EMP Resistant” to the features array on one of the product documents. Paste the command.

```
db.adb.update({_id:'zune'},{$push:{features:'EMP Resistant'}})
```

10. Write a command to create an index on the ll array. Remember ll contains the longitude and latitude, use the appropriate type of index. Paste the command.

```
db.adb.ensureIndex({ll:'2d'}, {type:1})
```

11. Write a command to return the closest vendor using the ll array. For your current location use Full Sail 3300 University Boulevard, Winter Park, Fl, 32792, Long:-81.30151, Lat: 28.59716

Paste the command and results.

```
> db.adb.findOne({ll:{$near:[-81.30151,28.59716]}})
{
        "_id" : "gocalendars",
        "name" : "Go! Calendars",
        "type" : "vendor",
        "address" : "451 E Altamonte Dr",
        "city" : "Altamonte Springs",
        "state" : "FL",
        "ll" : [
                -81.375883,
                28.667207
```

```
        ],
        "phone" : [
                {
                        "type" : "main",
                        "number" : "(321) 207-0021"
                },
                {
                        "type" : "fax",
                        "number" : "(321) 207-0021"
                }
        ]
}
```

12. Write a command to return all documents in the database ordered by name. Paste the command and results.

```
> db.adb.find().sort({name:1})
{ "_id" : "applaro", "name" : "Applaro", "type" : "product", "vendor" : "IKEA", "category" : "Furniture", "features" : [ "Drop-Leaf", "Table", "Wood" ] }
{ "_id" : "arholma", "category" : "Furniture", "features" : [ "Modular", "Combination" ], "name" : "Arholma", "rating" : 7, "type" : "product", "vendor" : "IKEA" }
{ "_id" : "borisvallejo", "category" : "Electronic", "features" : [ "12-month span", "mediumsquare" ], "name" : "Boris Vallejo & Julie Bell's Fantasy 2014 Wall Calendar", "rating" : 4, "type" : "product", "vendor" : "Go! Calendars" }
{ "_id" : "badcat2014wallcalendar", "category" : "Electronic", "features" : [ "12-month span", "mediumsquare" ], "name" : "Boris Vallejo & Julie Bell's Fantasy 2014 Wall Calendar", "rating" : 6, "type" : "product", "vendor" : "Go! Calendars" }
{ "_id" : "falster", "category" : "Furniture", "features" : [ "Wood", "Patio", "Paintable" ], "name" : "Falster", "rating" : 8, "type" : "product", "vendor" : "IKEA" }
{ "_id" : "gocalendars", "name" : "Go! Calendars", "type" : "vendor", "address" : "451 E Altamonte Dr", "city" : "Altamonte Springs", "state" : "FL", "ll" : [ -81.375883, 28.667207 ], "phone" : [ { "type" : "main", "number" : "(321) 207-0021" }, { "type" : "fax", "number" : "(321) 207-0021" } ] }
{ "_id" : "ikea", "name" : "IKEA", "type" : "vendor", "address" : "4092 Eastgate Drive", "city" : "Orlando", "state" : "FL", "ll" : [ -79.441833, 44.012893 ], "phone" : [ { "type" : "main", "number" : "(888) 888-4532" }, { "type" : "fax", "number" : "(888) 888-4532" } ] }
{ "_id" : "microsoftcorporation", "name" : "Microsoft Corporation", "type" : "vendor", "address" : "One Microsoft Way", "city" : "Redmond", "state" : "WA", "ll" : [ -122.131378, 47.638197 ], "phone" : [ { "type" : "main", "number" : "(425) 882-8080" }, { "type" : "fax",
```

"number" : "(425) 882-8080" }] }
{ "_id" : "microsoftmouse", "name" : "Microsoft Mouse", "type" : "product", "vendor" : "Microsoft Corporation", "category" : "Electronic", "features" : ["USB", "Blue Tooth", "Rechargeable"] }
{ "_id" : "microsoftsurface", "name" : "Microsoft Surface", "type" : "product", "vendor" : "Microsoft Corporation", "category" : "Electronic", "features" : ["USB", "Stand Up", "Wi-Fi"] }
{ "_id" : "pro9000", "name" : "Pro9000", "type" : "product", "vendor" : "ViewSonic Corporation", "category" : "Electronic", "features" : ["HDMI", "RGB", "Wi-Fi"] }
{ "_id" : "vt4200l", "category" : "Electronic", "features" : ["HDMI", "RGB", "Wi-Fi"], "name" : "VT4200-L", "rating" : 9, "type" : "product", "vendor" : "ViewSonic Corporation" }
{ "_id" : "viewsoniccorporation", "name" : "ViewSonic Corporation", "type" : "vendor", "address" : "381 Brea Canyon, Road", "city" : "Walnut", "state" : "CA", "ll" : [-117.84484, 34.013444], "phone" : [{ "type" : "main", "number" : "(425) 882-8080" }, { "type" : "fax", "number" : "(425) 882-8080" }] }
{ "_id" : "xbox360", "category" : "Electronic", "features" : ["RGBY Compatible", "Wi-Fi", "Wireless Controllers"], "name" : "Xbox 360", "rating" : 5, "type" : "product", "vendor" : "Microsoft Corporation" }
{ "_id" : "xboxone", "category" : "Electronic", "features" : ["HDMI", "1080 p", "Wi-Fi"], "name" : "Xbox One", "rating" : 1, "type" : "product", "vendor" : "Microsoft Corporation" }
{ "_id" : "zune", "category" : "Electronic", "features" : ["Mp3", "Mp4", "Wi-Fi", "EMP Resistant"], "name" : "Zune", "rating" : 3, "type" : "product", "vendor" : "Microsoft Corporation" }

13. Write a command to return the product with the highest rating (type= product). Only one document should be returned.

Paste the command and results.

```
> db.adb.findOne({$query:{type:'product',rating:{$exists:true}}, $orderby:{rating:-1}})
{
        "_id" : "vt4200l",
        "category" : "Electronic",
        "features" : [
                "HDMI",
                "RGB",
                "Wi-Fi"
        ],
        "name" : "VT4200-L",
        "rating" : 9,
        "type" : "product",
        "vendor" : "ViewSonic Corporation"
}
```

14. Write a command to return the product with the fourth highest rating. Only one document should be returned.

Paste the command and results.

```
> db.adb.find({$query:{type:'product',rating:{$exists:true}}, $orderby:{rating:-1}}).skip(3).limit(1)
{ "_id" : "badcat2014wallcalendar", "category" : "Electronic", "features" : [  "12-month span", "mediumsquare" ], "name" : "Boris Vallejo & Julie Bell's Fantasy 2014 Wall Calendar", "rating" : 6, "type" : "product", "vendor" : "Go! Calendars" }
```

15. Write a command to return the average "rating" for all products (type=product) in the database by vendor. Only include documents with an "rating" field. IE: if there is not an "rating" field, do not count it as a 0. (Hint: Use the group command.)

Paste the command and results.

```
> db.adb.group({
... cond: {rating:{$exists:true},type:'product'},
... key: {vendor: true},
... initial: {ratingAvg: 0, totalRating:0, count:0},
... reduce: function(obj,prev){
... prev.totalRating += obj.rating
... prev.count ++
... },
... finalize: function(out){
... out.ratingAvg = out.totalRating / out.count;
... }
... })
[
        {
                "vendor" : "Microsoft Corporation",
                "ratingAvg" : 3,
                "totalRating" : 9,
                "count" : 3
        },
        {
                "vendor" : "IKEA",
                "ratingAvg" : 7.5,
```

```
                "totalRating" : 15,
                "count" : 2
        },
        {
                "vendor" : "ViewSonic Corporation",
                "ratingAvg" : 9,
                "totalRating" : 9,
                "count" : 1
        },
        {
                "vendor" : "Go! Calendars",
                "ratingAvg" : 5,
                "totalRating" : 10,
                "count" : 2
        }
]
```

16. Write a command to return 1 vendor by the type and name field.
Your logic should be: where type = TYPE and name = NAME
Paste the command and results.

```
> db.adb.findOne({type:'vendor', name:'Microsoft Corporation'})
{
        "_id" : "microsoftcorporation",
        "name" : "Microsoft Corporation",
        "type" : "vendor",
        "address" : "One Microsoft Way",
        "city" : "Redmond",
        "state" : "WA",
        "ll" : [
                -122.131378,
                47.638197
        ],
        "phone" : [
                {
                        "type" : "main",
```

```
                        "number" : "(425) 882-8080"
                },
                {
                        "type" : "fax",
                        "number" : "(425) 882-8080"
                }
        ]
}
```

17. Write a command to delete a single product. Use the document's unique id (_id) in your statement.

Paste the command below.

```
db.adb.remove({_id:'arholma'})
```

18. Write a command to return the count of all product documents in your database.

Paste the command and results below.

```
> db.adb.find({type:'product'}).count()
11
```

19. Write a command to remove all products for one vendor from your database. Just the products.

Paste the command.

```
db.adb.remove({vendor:'Go! Calendars'})
```

20. Export your collection to a csv format using the mongoexport command. Only export the numeric and string fields for both products and vendors. (Do not export array fields)

Paste the command below.

```
./mongoexport -db adb1403f --collection adb --csv --out
~/Desktop/ADB/lab/final/adb1403f_3_26_2014.csv --fields
_id,address,city,name,state,type,category,vendor
```

FINISHING YOUR EXAM

1. Save your files. Convert this word doc to a PDF.
   Name your files (pdf & csv) FIRSTINITIAL LASTNAME.XXX
2. Raise your hand and inform the Instructor that you are finished.

Submit the PDF & CSV.