**Spring 2020**

**Final Project**

**IS 680 Database Management Systems**



Alex George

Professor Li

May 13, 2020

**Table of Contents**

JSON Schema 3

Inserting Data 5

Shipping Order Update 10

Show.js 11

SalesReport.js 12

Aging.js 13

**Creating JSON Schema and New Collection**:





Mongo Shell Output:

db.createCollection("p\_orders", {

validator: {

$jsonSchema: {

bsonType: "object",

required: ["\_id", "title", "line\_items", "orderStatus", "orderDate" ],

properties: {

\_id: {bsonType: "int",

minimum: 1,

maximum: 9999},

title: {bsonType: "string"},

line\_items: {bsonType: "object",

required: ["itemId", "itemDescription", "quantity", "price", "total"],

properties: {

itemId: {

enum: [ "p1", "p2", "p4", "p5", "p6" ],

description: "can be only p1, p2, p3, p4, p5, or p6"},

itemDescription: {bsonType: "string"},

quantity: {bsonType: "int",

minimum: 0

},

price: {bsonType: "double",

minimum: 0

},

total: {bsonType: "double"}

}},

warehouseContact: {bsonType: "object",

required: ["warehouseId", "warehousePhone", "warehouseStreet",

"warehouseCity", "warehouseState", "warehouseZip"],

properties: {

warehouseId: {

enum: [ "w1", "w2", "w4", "w5", "w6" ],

description: "can only be w1, w2, w3, w4, w5, or w6"},

warehouseName:{bsonType: "string"},

warehousePhone: {bsonType: "string"},

warehouseStreet: {bsonType: "string"},

warehouseCity: {bsonType: "string"},

warehouseState: {bsonType: "string"},

warehouseZip: {bsonType: "int"}}

},

customerContact: {bsonType: "object",

required: ["customerId", "firstName", "lastName", "customerPhone",

"customerStreet", "customerCity", "customerState", "CustomerZip"],

properties: {

customerId: {bsonType: "int",

minimum: 100,

maximum: 1000

},

firstName: {bsonType: "string"},

lastName: {bsonType: "string"},

customerPhone: {bsonType: "string"},

customerStreet: {bsonType: "string"},

customerCity: {bsonType: "string"},

customerState: {bsonType: "string"},

CustomerZip: {bsonType: "int"}}

},

orderDate: {bsonType: "date"},

shipDate: {bsonType: "date"},

orderStatus: {

enum: ["Shipped", "Not Shipped"],

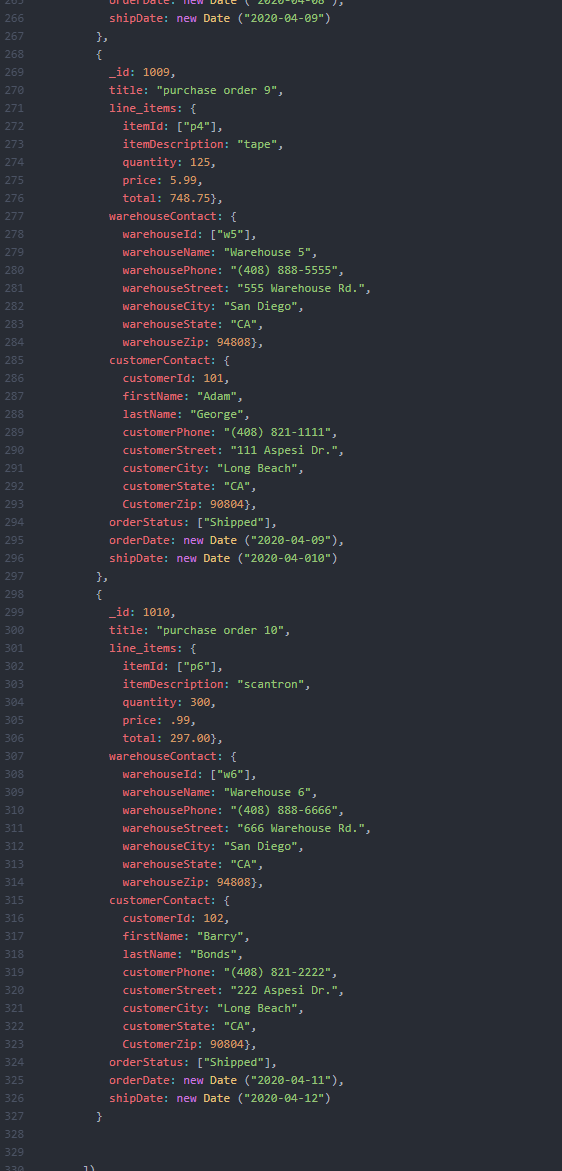
description: "can only be Shipped or Not Shipped"}

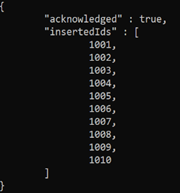
}

}

}})

**Inserting into data into the Collection:**

****

****Mongo Shell Output:

db.p\_orders.**insertOne**({

\_id: 1001,

title: "purchase order 1",

line\_items: {

itemId: ["p1"],

itemDescription: "pencil",

quantity: 150,

price: 2.99,

total: 448.50},

warehouseContact: {

warehouseId: ["w2"],

warehouseName: "Warehouse 2",

warehousePhone: "(408) 888-2222",

warehouseStreet: "222 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 101,

firstName: "Adam",

lastName: "George",

customerPhone: "(408) 821-1111",

customerStreet: "111 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

orderDate: new Date ("2020-04-01"),

shipDate: new Date ("2020-04-02")

});

db.p\_orders.**insertOne**(

{

\_id: 1002,

title: "purchase order 2",

line\_items: {

itemId: ["p2"],

itemDescription: "pen",

quantity: 200,

price: 3.99,

total: 798.00},

warehouseContact: {

warehouseId: ["w1"],

warehouseName: "Warehouse 1",

warehousePhone: "(408) 888-1111",

warehouseStreet: "111 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 104,

firstName: "David",

lastName: "Smith",

customerPhone: "(408) 821-4444",

customerStreet: "444 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

orderDate: new Date ("2020-04-01"),

shipDate: new Date ("2020-04-03")

})

db.p\_orders.**insertOne**(

{

\_id: 1003,

title: "purchase order 3",

line\_items: {

itemId: ["p1"],

itemDescription: "pencil",

quantity: 200,

price: 2.99,

total: 598.00},

warehouseContact: {

warehouseId: ["w2"],

warehouseName: "Warehouse 2",

warehousePhone: "(408) 888-2222",

warehouseStreet: "222 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 101,

firstName: "Adam",

lastName: "George",

customerPhone: "(408) 821-1111",

customerStreet: "111 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

orderDate: new Date ("2020-04-03"),

shipDate: new Date ("2020-04-04")

})

db.p\_orders.**insertMany**([

{

\_id: 1004,

title: "purchase order 4",

line\_items: {

itemId: ["p5"],

itemDescription: "eraser",

quantity: 265,

price: 3.99,

total: 1057.35},

warehouseContact: {

warehouseId: ["w6"],

warehouseName: "Warehouse 6",

warehousePhone: "(408) 888-6666",

warehouseStreet: "666 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 106,

firstName: "Frank",

lastName: "Thomas",

customerPhone: "(408) 821-6666",

customerStreet: "666 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

orderDate: new Date ("2020-05-03"),

shipDate: new Date ("2020-05-04")

},

{

\_id: 1005,

title: "purchase order 5",

line\_items: {

itemId: ["p1"],

itemDescription: "pencil",

quantity: 120,

price: 2.99,

total: 358.80},

warehouseContact: {

warehouseId: ["w5"],

warehouseName: "Warehouse 5",

warehousePhone: "(408) 888-5555",

warehouseStreet: "555 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 103,

firstName: "Chris",

lastName: "Paul",

customerPhone: "(408) 821-3333",

customerStreet: "333 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

orderDate: new Date ("2020-05-03"),

shipDate: new Date ("2020-05-08")

},

{

\_id: 1006,

title: "purchase order 6",

line\_items: {

itemId: ["p6"],

itemDescription: "scantron",

quantity: 600,

price: .99,

total: 594.00},

warehouseContact: {

warehouseId: ["w2"],

warehouseName: "Warehouse 2",

warehousePhone: "(408) 888-2222",

warehouseStreet: "222 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 1005,

firstName: "Eva",

lastName: "Doe",

customerPhone: "(408) 821-5555",

customerStreet: "555 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

orderDate: new Date ("2020-04-04"),

shipDate: new Date ("2020-04-05")

},

{

\_id: 1007,

title: "purchase order 7",

line\_items: {

itemId: ["p2"],

itemDescription: "pen",

quantity: 100,

price: 3.99,

total: 399.00},

warehouseContact: {

warehouseId: ["w1"],

warehouseName: "Warehouse 1",

warehousePhone: "(408) 888-1111",

warehouseStreet: "222 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 104,

firstName: "David",

lastName: "Smith",

customerPhone: "(408) 821-4444",

customerStreet: "444 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

orderDate: new Date ("2020-04-06"),

shipDate: new Date ("2020-04-07")

},

{

\_id: 1008,

title: "purchase order 8",

line\_items: {

itemId: ["p1"],

itemDescription: "pencil",

quantity: 400,

price: 2.99,

total: 1196.00},

warehouseContact: {

warehouseId: ["w4"],

warehouseName: "Warehouse 4",

warehousePhone: "(408) 888-4444",

warehouseStreet: "444 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 106,

firstName: "Frank",

lastName: "Thompson",

customerPhone: "(408) 821-6666",

customerStreet: "666 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

orderDate: new Date ("2020-04-08"),

shipDate: new Date ("2020-04-09")

},

{

\_id: 1009,

title: "purchase order 9",

line\_items: {

itemId: ["p4"],

itemDescription: "tape",

quantity: 125,

price: 5.99,

total: 748.75},

warehouseContact: {

warehouseId: ["w5"],

warehouseName: "Warehouse 5",

warehousePhone: "(408) 888-5555",

warehouseStreet: "555 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 101,

firstName: "Adam",

lastName: "George",

customerPhone: "(408) 821-1111",

customerStreet: "111 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

orderDate: new Date ("2020-04-09"),

shipDate: new Date ("2020-04-010")

},

{

\_id: 1010,

title: "purchase order 10",

line\_items: {

itemId: ["p6"],

itemDescription: "scantron",

quantity: 300,

price: .99,

total: 297.00},

warehouseContact: {

warehouseId: ["w6"],

warehouseName: "Warehouse 6",

warehousePhone: "(408) 888-6666",

warehouseStreet: "666 Warehouse Rd.",

warehouseCity: "San Diego",

warehouseState: "CA",

warehouseZip: 94808},

customerContact: {

customerId: 102,

firstName: "Barry",

lastName: "Bonds",

customerPhone: "(408) 821-2222",

customerStreet: "222 Aspesi Dr.",

customerCity: "Long Beach",

customerState: "CA",

CustomerZip: 90804},

orderStatus: ["Shipped"],

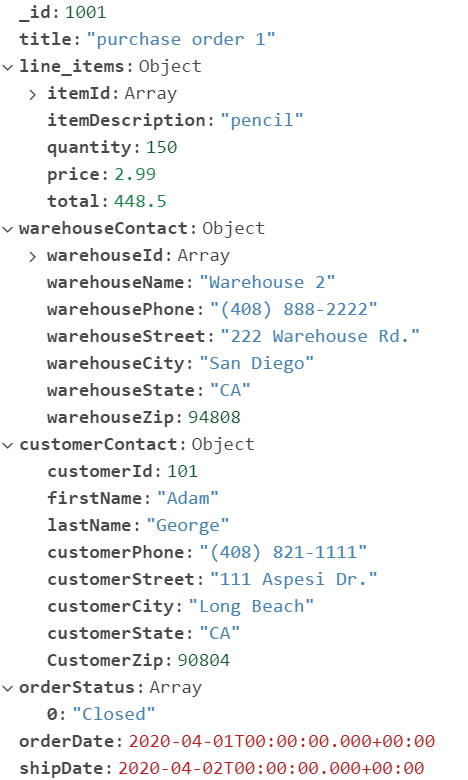
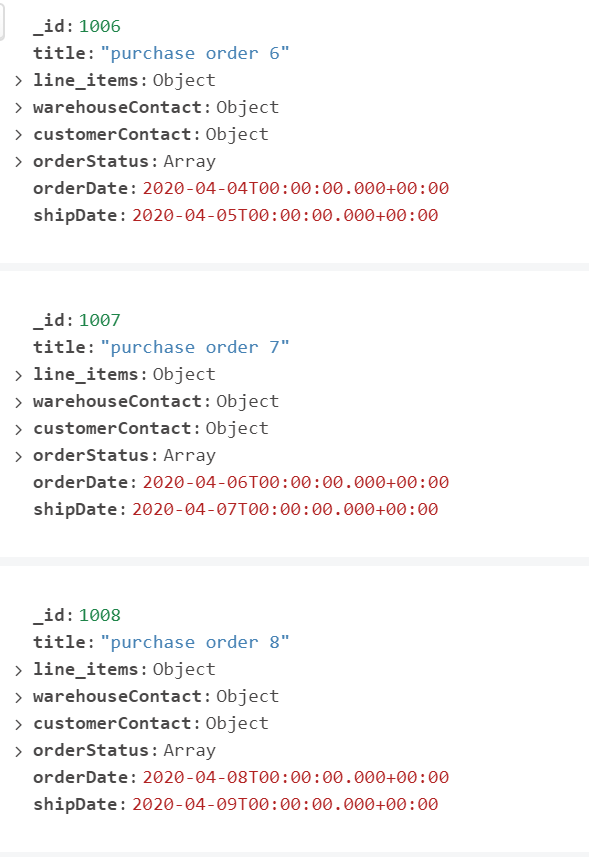
orderDate: new Date ("2020-04-11"),

shipDate: new Date ("2020-04-12")

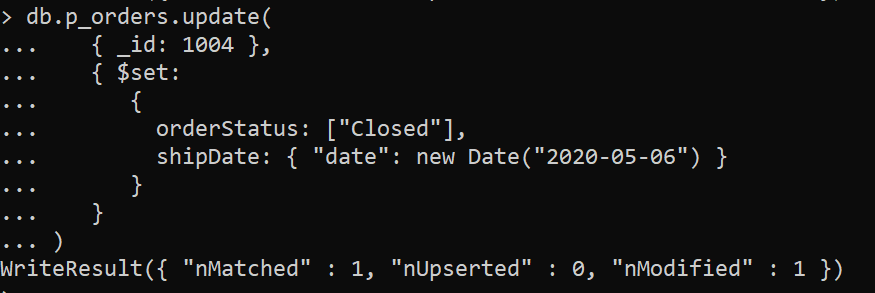
}

])

**The Graphic User Interface View:**

**Project Scripts:**

**Shipping Updates:**

db.p\_orders.update(

{ \_id: 1004 },

{ $set:

{

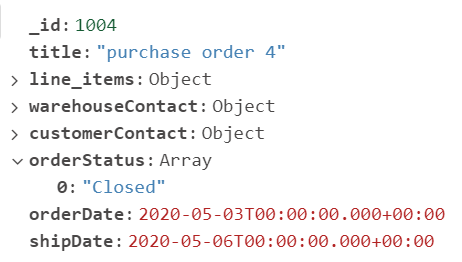
orderStatus: ["Closed"],

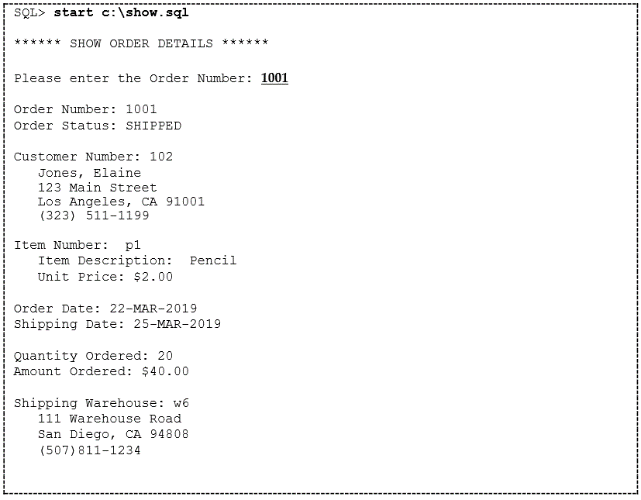
shipDate: { "date": new Date("2020-05-06") }

}

}

)

 After update: 

**Show.js** – All details of the order is displayed



**Salesreport.js -** This displays number of orders, total units and amount according to each product and the month/year of the order open date

db.p\_orders.aggregate([

{$project: {

\_id: {

itemId: "$line\_items.itemId",

itemDescription: "$line\_items.itemDescription",

orderDate: {$dateToString: {format: "%m-%G",date: "$orderDate"}},

totalQty:{$sum: "$line\_items.quantity"},

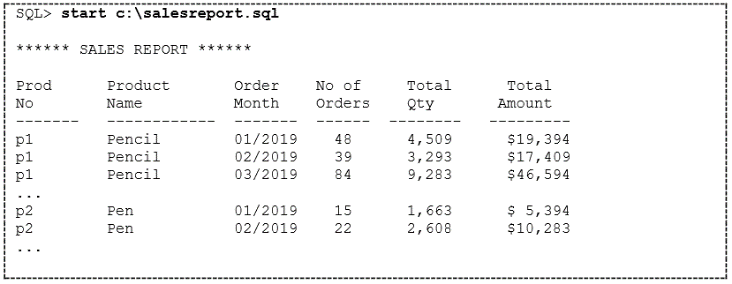
totalAmt:{$sum: "$line\_items.total"}}

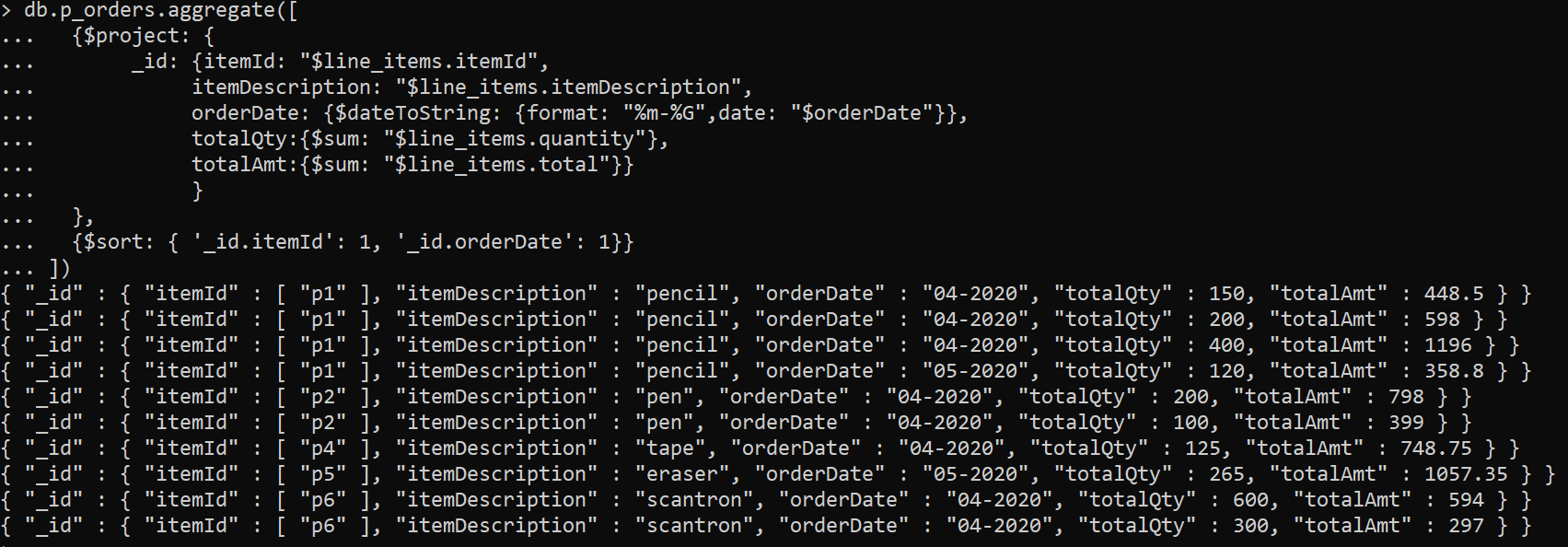
}

},

{$sort: { '\_id.itemId': 1, '\_id.orderDate': 1}}

])





**Aging.js -** This report displays Open orders and sort by the longest DaysOpen. DaysOpen is number of days between date open and today

db.p\_orders.aggregate([

{$match: {orderStatus: "Open", shipdate: null}},

{$project: {

\_id: {

oNum: "$\_id",

oDate: {$dateToString: {format: "%m-%d-%G",date: "$orderDate"}},

oStatus: "$orderStatus",

pNum: "$line\_items.itemId",

item: "$line\_items.itemDescription",

oQty: "$line\_items.quantity",

price: "$line\_items.price",

oAmt:{$round: { $multiply: [ "$line\_items.price", "$line\_items.quantity" ] }},

daysOpen: {$trunc: {$divide: [{ $subtract: [new Date(), '$orderDate'] }, 1000 \* 60 \* 60 \* 24]}}

}

}

},

{$sort: {'\_id.oNum': 1, '\_id.oDate': 1}}

])

