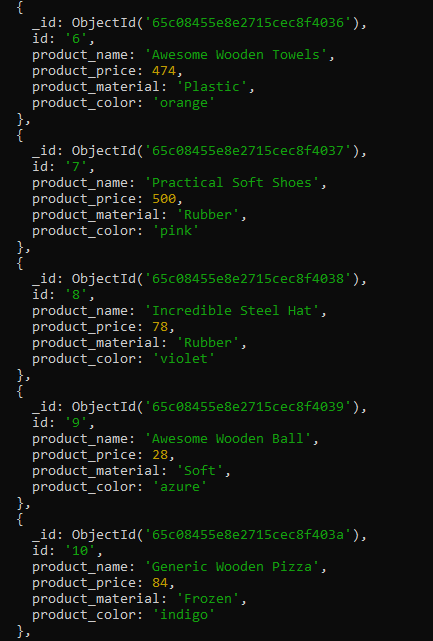
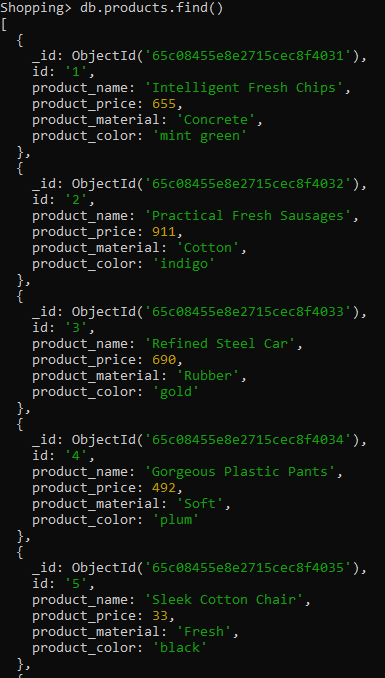
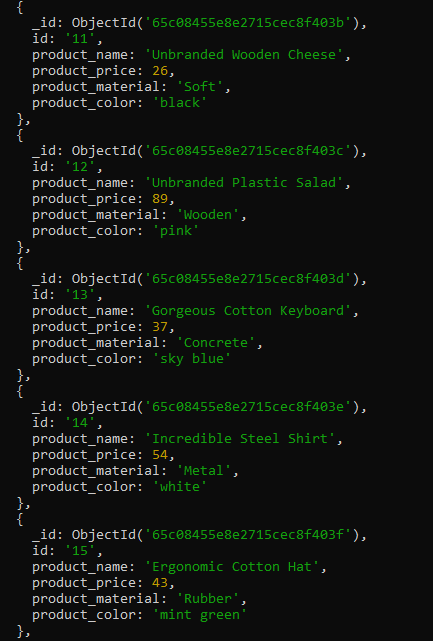
M**ongoDB Day 1**

//Find all the information about each products

//db.products.find() we can use this and to show all the data we can give db.products.find().toArray()





2)Find the product price which are between 400 to 800

Using find

//db.products.find({product\_price:{$gt:400,$lt:800}})

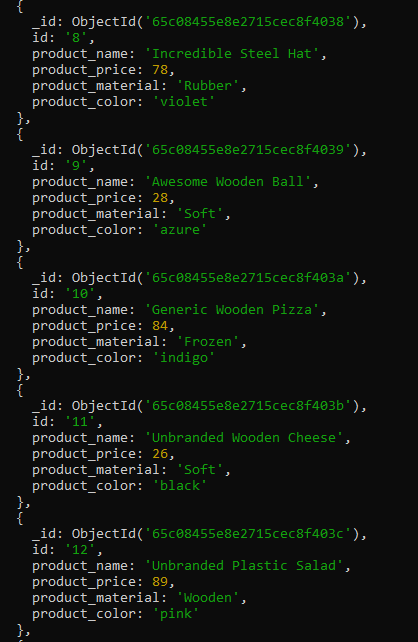


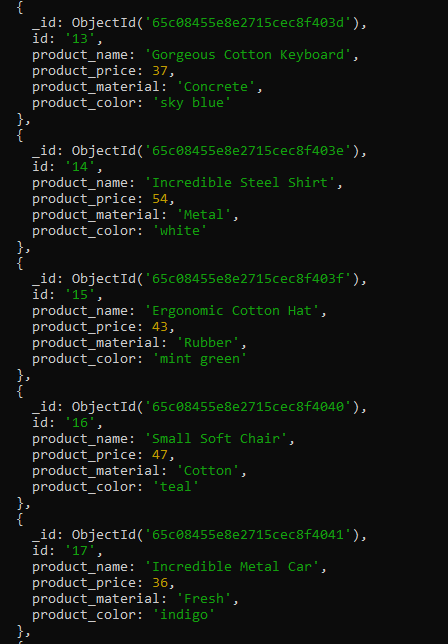
3) Find the product price which are not between 400 to 600

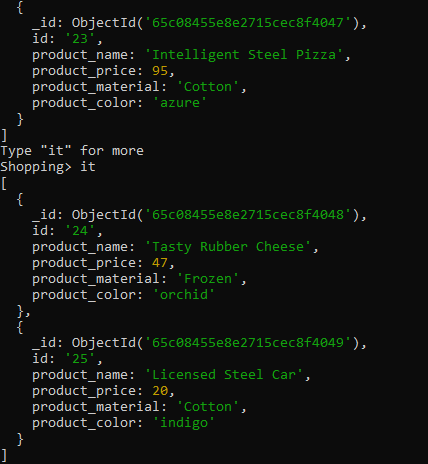
Using find

//db.products.find({product\_price:{$not:{$gt:400,$lt:600}}})







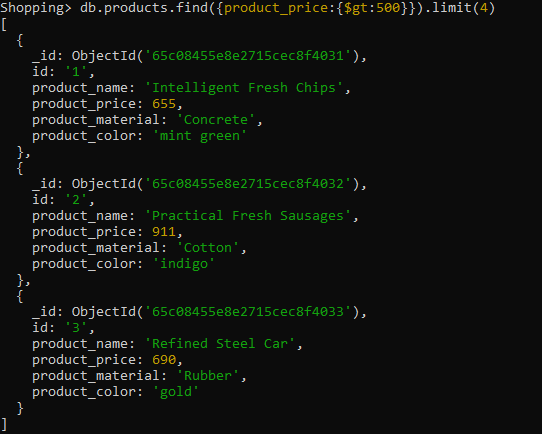


4) List the four product which are greater than 500 in price

Using find

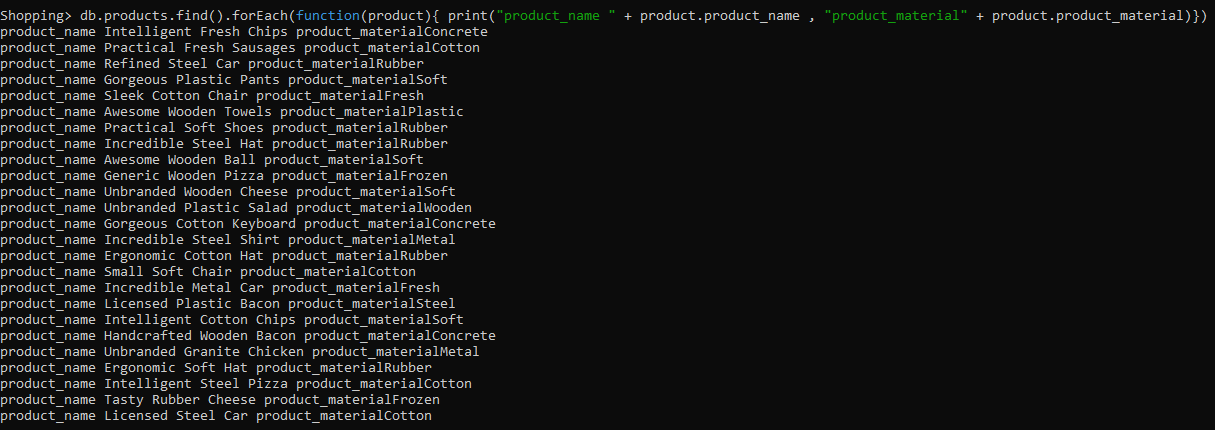
//db.products.find({product\_price:{$gt:500}}).limit(4)

//in this data there are just 3 products above price 500



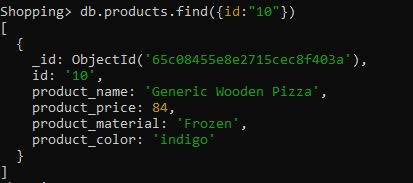
5) Find the product name and product material of each products

// db.products.find().forEach(function(product){ print( “product\_name” + product.product\_name , “product\_material” + product.product\_material)})



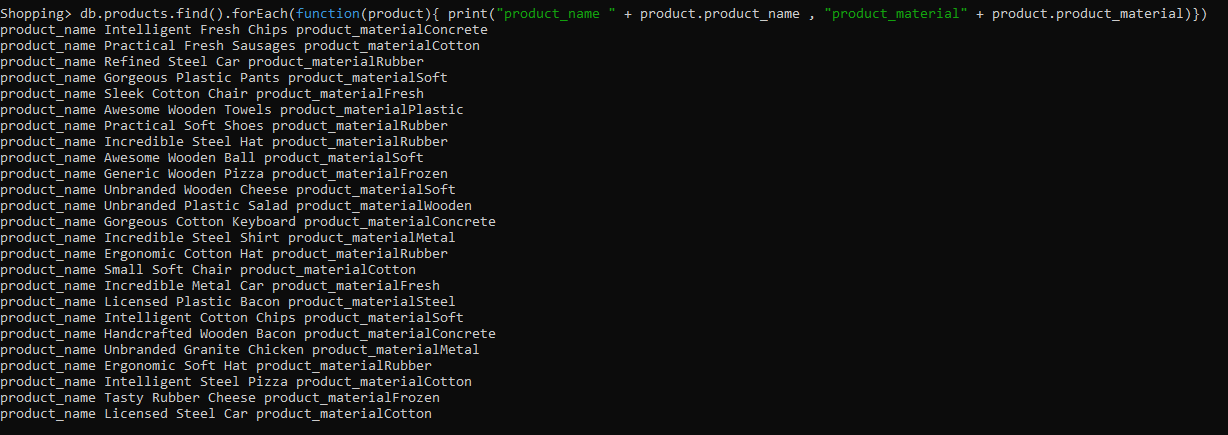
6) Find the product with a row id of 10

//db.products.find({id:”10”})



7) Find only the product name and product material

// db.products.find().forEach(function(product){ print( “product\_name” + product.product\_name , “product\_material” + product.product\_material)})



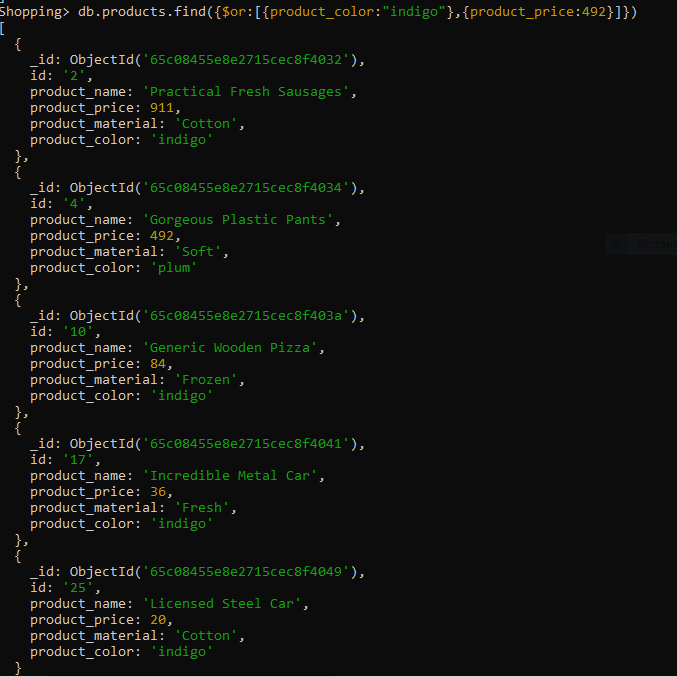
8) Find all products which contain the value of soft in product material

//db.products.find({product\_material:”Soft”})



9) Find products which contain product color indigo  and product price 492.00

db.products.find({$or:[{product\_price:492},{product\_color:”indigo”}]})



10) Delete the products which product price value are same

db.products.([{$group:{ \_id:”product\_price”,

duplicate:{$addToSet:”$\_id”},

TotalCount:{$sum:1}

}}, { $match:{TotalCount:{$gt:1}}]).forEach((val)=>{ val.duplicate.shift(); db.products.deleteMany({\_id:{ $in:val.duplicate}})})

\*this are duplicate product\_price in products



\*Removed product\_price duplicates from products



