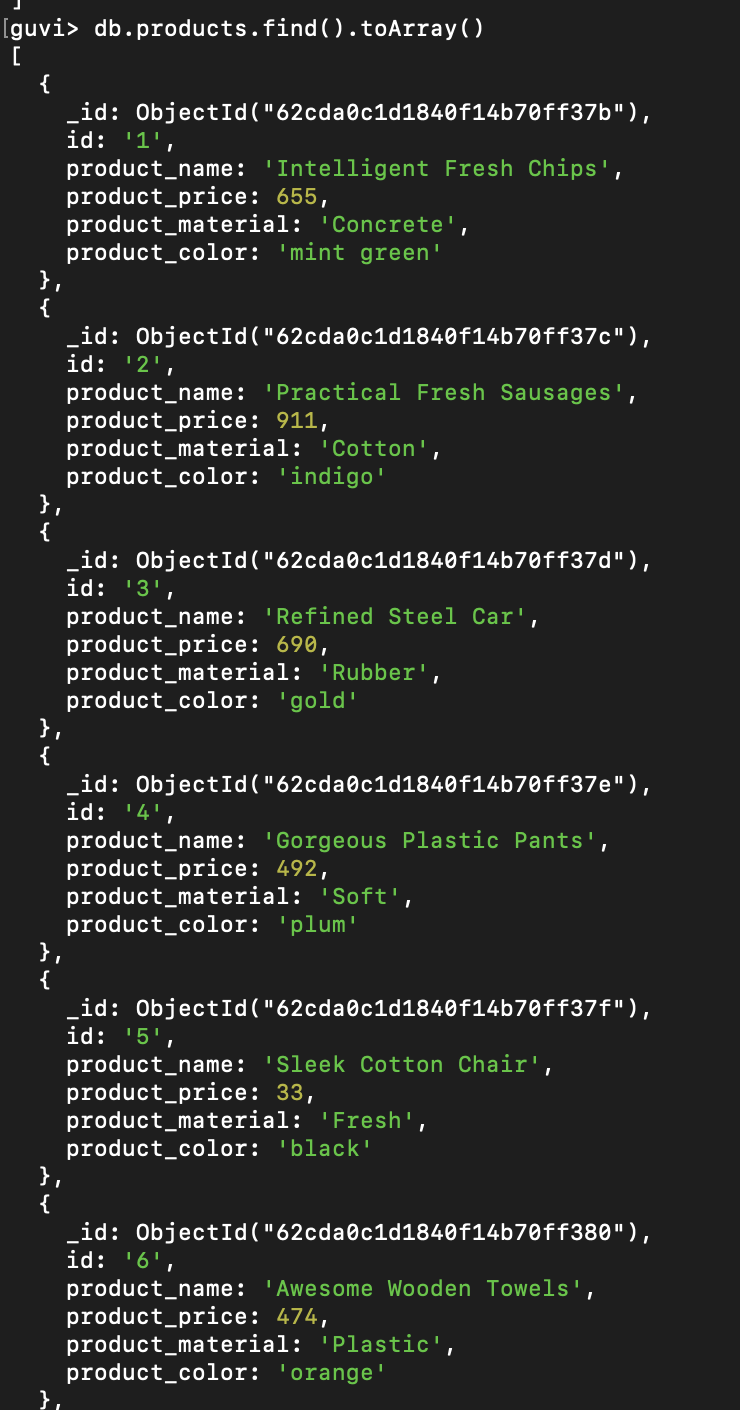
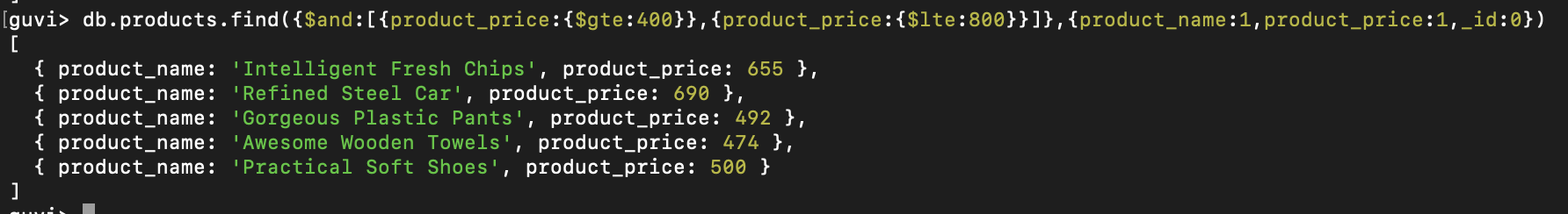
1. Find all the information about each products

**Query:-** db.products.find().toArray()



1. Find the product price which are between 400 to 800

**Query:**-db.products.find({$and:[{product\_price:{$gte:400}},{product\_price:{$lte:800}}]},{product\_name:1,product\_price:1,\_id:0})



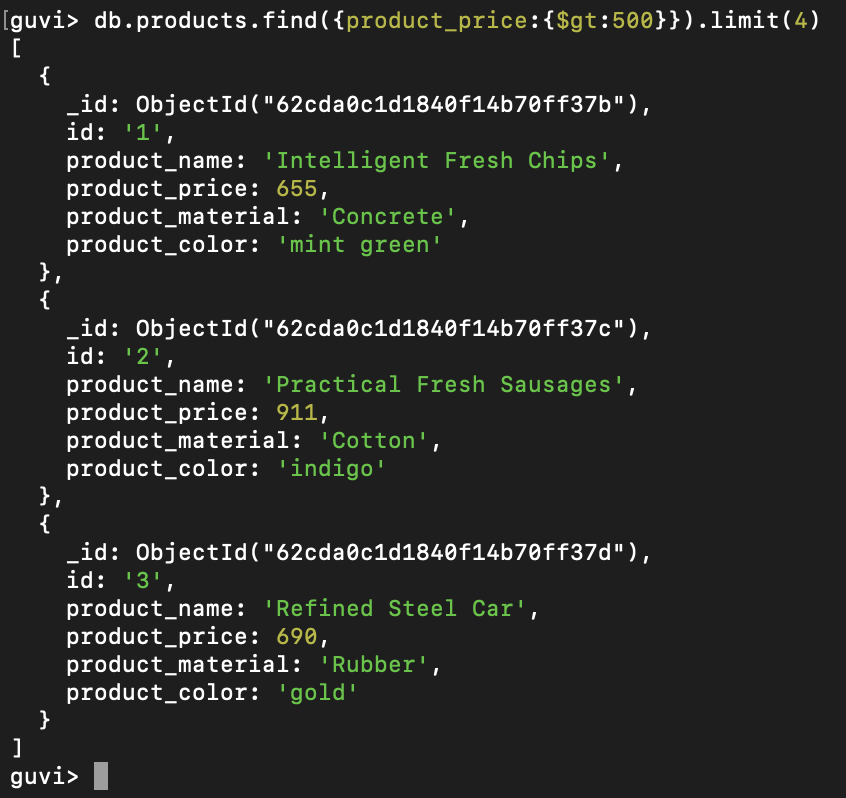
1. Find the product price which are not between 400 to 600

**Query:-** db.products.find({$or:[{product\_price:{$lte:400}},{product\_price:{$gte:600}}]},{product\_name:1,product\_price:1,\_id:0})



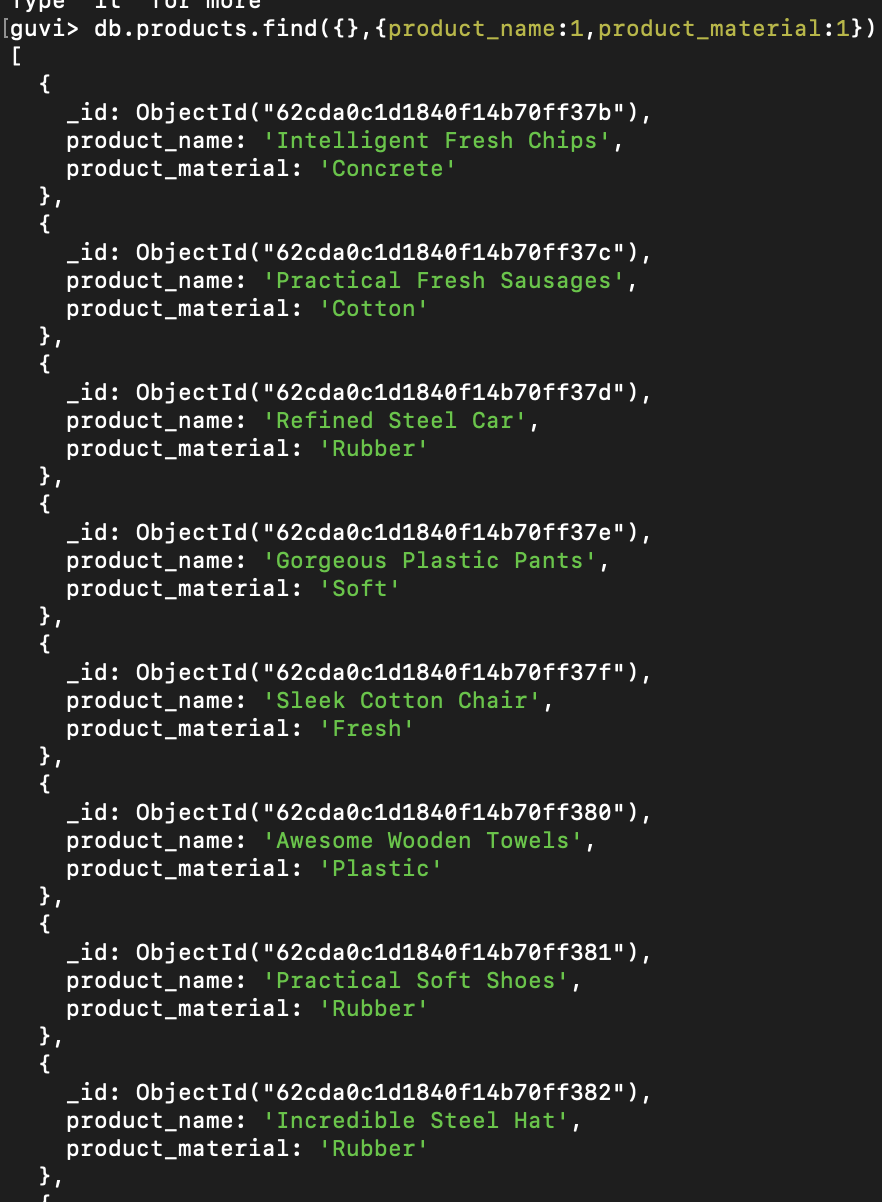
1. List the four product which are grater than 500 in price

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



1. Find the product name and product material of each products

**Query:-**db.products.find({},{product\_name:1,product\_material:1})



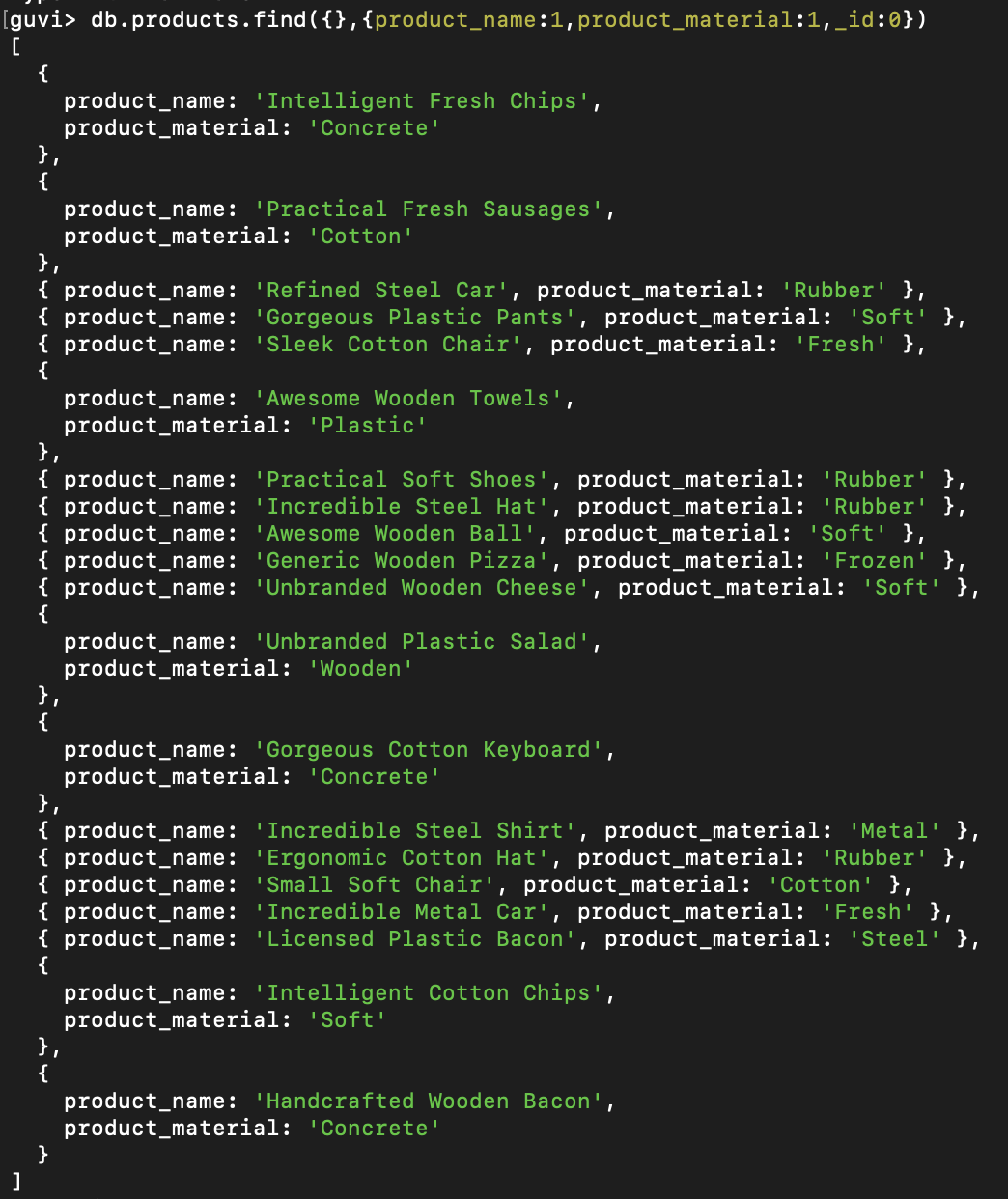
1. Find the product with a row id of 10

**Query:-**db.products.find({id:'10'})



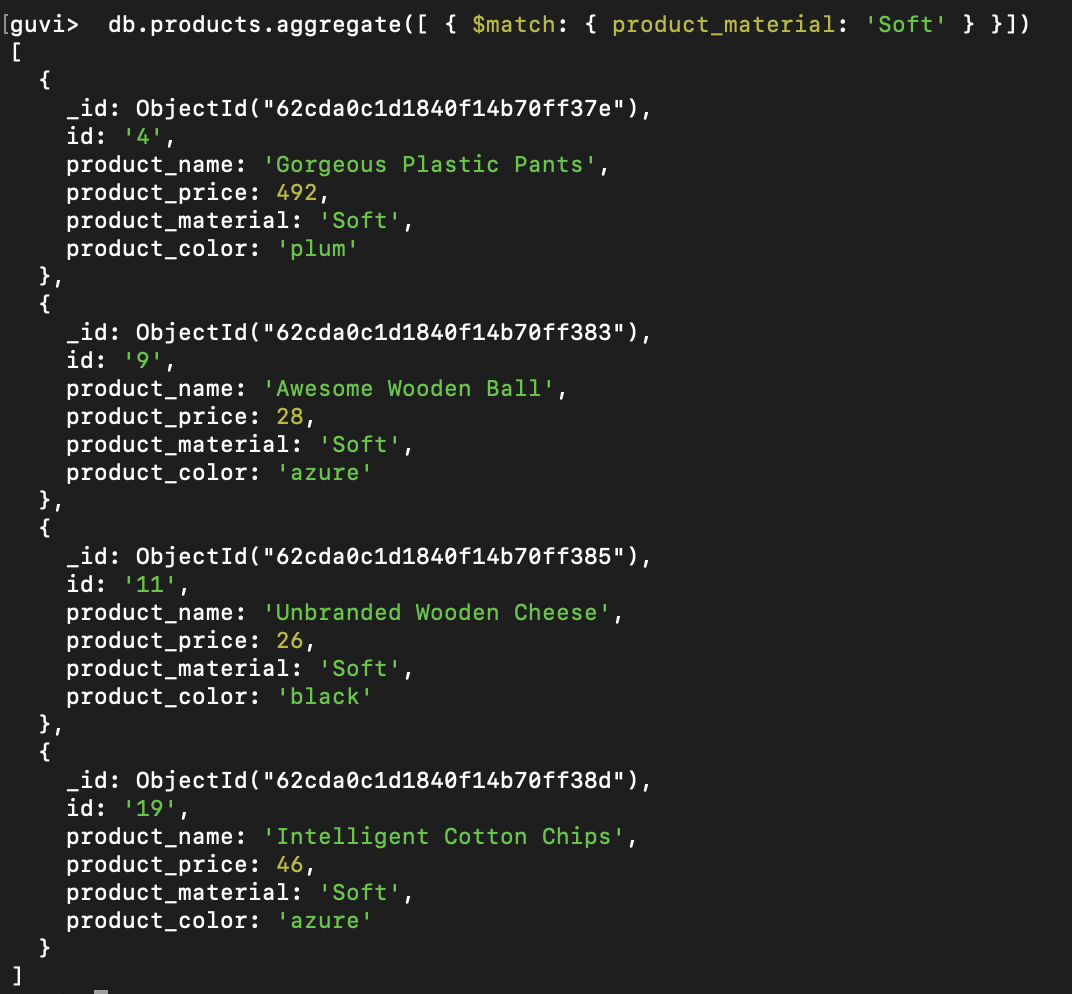
1. Find only the product name and product material

**Query:-** db.products.find({},{product\_name:1,product\_material:1,\_id:0})



1. Find all products which contain the value of soft in product material

**Query:-**db.products.aggregate([ { $match: { product\_material: 'Soft' } }])



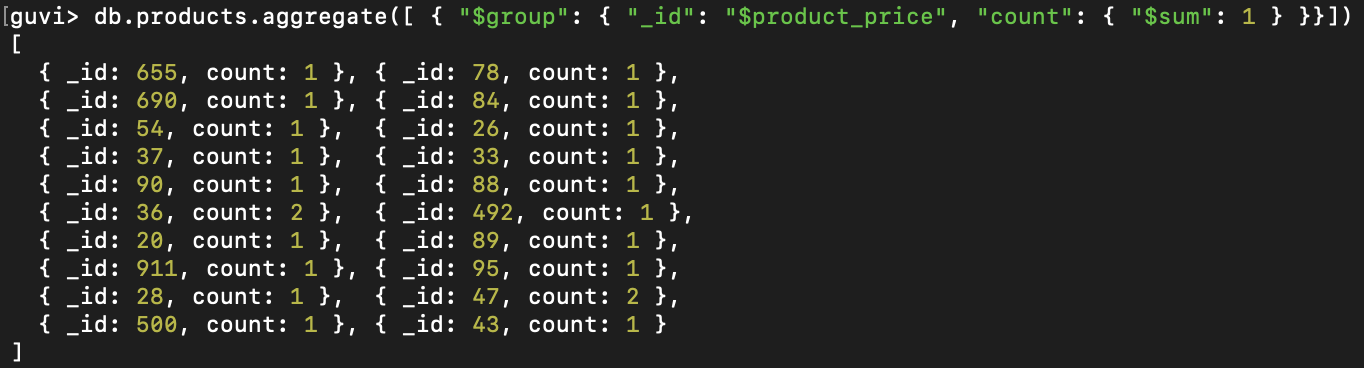
1. Find products which contain product color indigo and product price 492.00

**Query:-**db.products.aggregate([{$match:{$and:[{product\_price:492},{product\_color:"indigo"}]}}]

**There are no products that has color indigo and price 492**

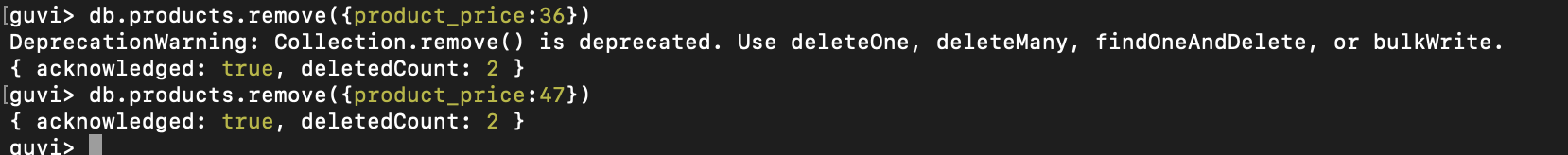
1. Delete the products which product price value are same

Code to find the repetitive prices:-



From the above snippet, its seen that 47 and 36 are the prices that are repeating.

To delete those document, we used remove function:-



Snippet of Collection post removing:-

