

For the following question write the corresponding MongoDB queries

Link of JSON DATA

<https://github.com/rvsp/database/blob/master/mongodb/product.json>

Question1: Find all the information about each product.

```
> db.products.find({})
< {
  _id: ObjectId('6630fb12a32c65f273206d3a'),
  id: '1',
  product_name: 'Intelligent Fresh Chips',
  product_price: 655,
  product_material: 'Concrete',
  product_color: 'mint green'
}
{
  _id: ObjectId('6630fb12a32c65f273206d3b'),
  id: '2',
  product_name: 'Practical Fresh Sausages',
  product_price: 911,
```

Question2: Find the product price which are between 400 to 800

```
> db.products.find({product_price:{$gt:400,$lt:800}})
< {
  _id: ObjectId('6630fb12a32c65f273206d3a'),
  id: '1',
  product_name: 'Intelligent Fresh Chips',
  product_price: 655,
  product_material: 'Concrete',
  product_color: 'mint green'
}
{
  _id: ObjectId('6630fb12a32c65f273206d3c'),
  id: '3',
  product_name: 'Refined Steel Car',
```

Question 3: Find the product price which are not between 400 to 600.

```
> db.products.find({product_price:{$not:{$gt:400,$lt:800}}})
< {
  _id: ObjectId('6630fb12a32c65f273206d3b'),
  id: '2',
  product_name: 'Practical Fresh Sausages',
  product_price: 911,
  product_material: 'Cotton',
  product_color: 'indigo'
}
{
  _id: ObjectId('6630fb12a32c65f273206d3e'),
  id: '5',
  product_name: 'Sleek Cotton Chair',
  product_price: 33,
  product_material: 'Fresh'
```

Question 4: List the four product which are greater than 500 in price.

```
> db.products.find({product_price:{$gt:500}}).limit(4)
< {
  _id: ObjectId('6630fb12a32c65f273206d3a'),
  id: '1',
  product_name: 'Intelligent Fresh Chips',
  product_price: 655,
  product_material: 'Concrete',
  product_color: 'mint green'
}
{
  _id: ObjectId('6630fb12a32c65f273206d3b'),
  id: '2',
  product_name: 'Practical Fresh Sausages',
  product_price: 911,
  product_material: 'Cotton',
  product_color: 'indigo'
}
```

Question 5: Find the product name and product material of each products.

```
> db.products.find({}, {product_name:1, product_material:1})
< {
  _id: ObjectId('6630fb12a32c65f273206d3a'),
  product_name: 'Intelligent Fresh Chips',
  product_material: 'Concrete'
}
{
  _id: ObjectId('6630fb12a32c65f273206d3b'),
  product_name: 'Practical Fresh Sausages',
  product_material: 'Cotton'
}
{
  _id: ObjectId('6630fb12a32c65f273206d3c'),
  product_name: 'Refined Steel Car',
  product_material: 'Rubber'
}
```

Question 6: Find the product with a row id of 10.

```
> db.products.findOne({"id":"10"})
< {
  _id: ObjectId('6630fb12a32c65f273206d43'),
  id: '10',
  product_name: 'Generic Wooden Pizza',
  product_price: 84,
  product_material: 'Frozen',
  product_color: 'indigo'
}
```

Question 7: Find only the product name and product material

```
> db.products.find({}, {product_name:1,product_material:1,_id:0})
< {
  product_name: 'Intelligent Fresh Chips',
  product_material: 'Concrete'
}
{
  product_name: 'Practical Fresh Sausages',
  product_material: 'Cotton'
}
{
  product_name: 'Refined Steel Car',
  product_material: 'Rubber'
}
```

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

```
> db.products.find({"product_material":"/soft/i})
< {
  _id: ObjectId('6630fb12a32c65f273206d3d'),
  id: '4',
  product_name: 'Gorgeous Plastic Pants',
  product_price: 492,
  product_material: 'Soft',
  product_color: 'plum'
}
{
  _id: ObjectId('6630fb12a32c65f273206d42'),
  id: '9',
  product_name: 'Awesome Wooden Ball',
  product_price: 28,
  product_material: 'Soft',
  product_color: 'azure'
}
```

Question 9: Find products which contain product color indigo and product price 492.00

```
> db.products.find({ product_color: "indigo"})
< {
  _id: ObjectId('6630fb12a32c65f273206d3b'),
  id: '2',
  product_name: 'Practical Fresh Sausages',
  product_price: 911,
  product_material: 'Cotton',
  product_color: 'indigo'
}
{
  _id: ObjectId('6630fb12a32c65f273206d43'),
  id: '10',
  product_name: 'Generic Wooden Pizza',
  product_price: 84,
  product_material: 'Frozen',
  product_color: 'indigo'
}
```

```
> db.products.find({ product_price: 492.00 })
< {
  _id: ObjectId('6630fb12a32c65f273206d3d'),
  id: '4',
  product_name: 'Gorgeous Plastic Pants',
  product_price: 492,
  product_material: 'Soft',
  product_color: 'plum'
}
```

Question 10: Delete the products which product price value are same

Solution:

```
db.products.aggregate ([
  { $group: { _id: '$product_price', count: { $sum: 1 } } },
  { $match: { count : { $gt: 1 } } } ] )

.forEach((e) =>
  { db.products.deleteMany({ product_price: e._id }) });
```