

1. Find all the information about each products ?

Ans: `db.taskProducts.find();`

example:

```
{
  _id: ObjectId('66e57cc28e34328df78cde5f'),
  id: '1',
  product_name: 'Intelligent Fresh Chips',
  product_price: 655,
  product_material: 'Concrete',
  product_color: 'mint green'
}
{
  _id: ObjectId('66e57cc28e34328df78cde60'),
  id: '2',
  product_name: 'Practical Fresh Sausages',
  product_price: 911,
  product_material: 'Cotton',
  product_color: 'indigo'
}
{
  _id: ObjectId('66e57cc28e34328df78cde61'),
  id: '3',
```

```
product_name: 'Refined Steel Car',  
product_price: 690,  
product_material: 'Rubber',  
product_color: 'gold'  
}.....
```

total 25 id in product collection

2.Find the product price which are between 400 to 800 ?

Ans:db.taskProducts.find({'product_price':{'\$gt:400,\$lt:800}});

example:

```
{  
  _id: ObjectId('66e57cc28e34328df78cde5f'),  
  id: '1',  
  product_name: 'Intelligent Fresh Chips',  
  product_price: 655,  
  product_material: 'Concrete',  
  product_color: 'mint green'  
}
```

```
{  
  _id: ObjectId('66e57cc28e34328df78cde61'),  
  id: '3',  
  product_name: 'Refined Steel Car',  
  product_price: 690,
```

```
    product_material: 'Rubber',
    product_color: 'gold'
}
{
    _id: ObjectId('66e57cc28e34328df78cde62'),
    id: '4',
    product_name: 'Gorgeous Plastic Pants',
    product_price: 492,
    product_material: 'Soft',
    product_color: 'plum'
}
{
    _id: ObjectId('66e57cc28e34328df78cde64'),
    id: '6',
    product_name: 'Awesome Wooden Towels',
    product_price: 474,
    product_material: 'Plastic',
    product_color: 'orange'
}
{
    _id: ObjectId('66e57cc28e34328df78cde65'),
    id: '7',
```

```
product_name: 'Practical Soft Shoes',  
product_price: 500,  
product_material: 'Rubber',  
product_color: 'pink'  
}
```

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

?Ans:db.taskProducts.find({'product_price':{\$not:{\$gte:400,\$lte:600}}});

example:

```
{  
  _id: ObjectId('66e57cc28e34328df78cde5f'),  
  id: '1',  
  product_name: 'Intelligent Fresh Chips',  
  product_price: 655,  
  product_material: 'Concrete',  
  product_color: 'mint green'  
}
```

```
{  
  _id: ObjectId('66e57cc28e34328df78cde60'),  
  id: '2',  
  product_name: 'Practical Fresh Sausages',  
  product_price: 911,  
  product_material: 'Cotton',  
}
```

```
    product_color: 'indigo'
  }
  {
    _id: ObjectId('66e57cc28e34328df78cde61'),
    id: '3',
    product_name: 'Refined Steel Car',
    product_price: 690,
    product_material: 'Rubber',
    product_color: 'gold'
  }.....etc
```

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

Ans:db.taskProducts.find({'product_price':{\$gte:500}}).limit(4);

example:

```
{
  _id: ObjectId('66e57cc28e34328df78cde5f'),
  id: '1',
  product_name: 'Intelligent Fresh Chips',
  product_price: 655,
  product_material: 'Concrete',
  product_color: 'mint green'
}
{
```

```

    _id: ObjectId('66e57cc28e34328df78cde60'),
    id: '2',
    product_name: 'Practical Fresh Sausages',
    product_price: 911,
    product_material: 'Cotton',
    product_color: 'indigo'
  }
  {
    _id: ObjectId('66e57cc28e34328df78cde61'),
    id: '3',
    product_name: 'Refined Steel Car',
    product_price: 690,
    product_material: 'Rubber',
    product_color: 'gold'
  } ..... etc

```

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

?Ans: db.taskProducts.find({}, {'product_name':1, 'product_material':1});

example:

```

{
  _id: ObjectId('66e57cc28e34328df78cde5f'),
  product_name: 'Intelligent Fresh Chips',
  product_material: 'Concrete'
}

```

```
}  
  
{  
  _id: ObjectId('66e57cc28e34328df78cde60'),  
  product_name: 'Practical Fresh Sausages',  
  product_material: 'Cotton'  
}.....etc
```

6.Find the product with a row id of 10 ?

Ans:db.taskProducts.findOne({'id':'10'});

example:

```
{  
  _id: ObjectId('66e57cc28e34328df78cde68'),  
  id: '10',  
  product_name: 'Generic Wooden Pizza',  
  product_price: 84,  
  product_material: 'Frozen',  
  product_color: 'indigo'  
}
```

7.Find only the product name and product material

Ans:db.taskProducts.find({},{'_id':0,'product_name':1,'product_material':1});

example:

```
{  
  
  product_name: 'Intelligent Fresh Chips',
```

```
product_material: 'Concrete'
```

```
}.....etc
```

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

```
Ans:db.taskProducts.find({'product_material':'Soft'});
```

example:

```
{  
  _id: ObjectId('66e57cc28e34328df78cde62'),  
  id: '4',  
  product_name: 'Gorgeous Plastic Pants',  
  product_price: 492,  
  product_material: 'Soft',  
  product_color: 'plum'  
}
```

```
{  
  _id: ObjectId('66e57cc28e34328df78cde67'),  
  id: '9',  
  product_name: 'Awesome Wooden Ball',  
  product_price: 28,  
  product_material: 'Soft',  
  product_color: 'azure'
```

```
}.....etc
```

9.Find products which contain product color indigo and product price 492.00 ?


```
Ans:db.taskProducts.find({"$or": [{'product_color': 'indigo'
},{'product_price':492.00}]});
```

example:

```
{
  _id: ObjectId('66e57cc28e34328df78cde60'),
  id: '2',
  product_name: 'Practical Fresh Sausages',
  product_price: 911,
  product_material: 'Cotton',
  product_color: 'indigo'
}
{
  _id: ObjectId('66e57cc28e34328df78cde62'),
  id: '4',
  product_name: 'Gorgeous Plastic Pants',
  product_price: 492,
  product_material: 'Soft',
  product_color: 'plum'
}
```

10.Delete the products which product price value are 28 ?

```
Ans:db.taskProducts.deleteOne({_id: ObjectId('66e57cc28e34328df78cde67')});
```

example:

```
{  
  acknowledged: true,  
  deletedCount: 1  
}
```