Lab11. Ecommerce Product Catalog Design using MongoDB

In order to manage an ecommerce system, the first thing you need is a product catalog. Product catalogs must have the capacity to store many different types of objects with different sets of attributes. These kinds of data collections work quite well with MongoDB's flexible data model, making MongoDB a natural fit for this type of data.

Question1. Create a collection **Product** and insert the following document for Audio Album

At the beginning of the document, a document-based schema should contain general product information, to facilitate searches of the entire catalog. After the common fields, we'll add a details subdocument that contains fields that vary between product types such as Audio Album or Movies.

```
{
        _id: "p01",
        type: "Audio Album",
        title: "Bergmans volume15",
        description: "by Fr Bergmans",
        release_date:"2020-02-15"

        shipping: {
            weight: 6,
            dimensions: {
                width: 10, height: 10, depth: 1 },
        },

        pricing: {
                list: 1200, retail: 1100, savings: 100, percentage_savings: 8
        },

        details: {
                title: "Jeba Thotta Jeya Geethangal",
                artist: "Fr Bergmans",
                genre: [ "folk", "Spiritual" ],

                tracks: [
                        "Song1", "Song2, "Song3", Song4"
                ],
        },
}
```

Question2. Insert the following document for a Movie to **Product** collection

A movie item would have the same fields for general product information, shipping, and pricing, but different fields for the **details** subdocument.

```
{
        _id: "p02",
        type: "Movie",
        title: "Master",
        description: "Movie by Actor Vijay",
        release_date:"2020-03-14"


        shipping: {
            weight: 10,
            dimensions: {
                width: 20, height: 10, depth: 2  },
        },

        pricing: {
                list: 15000, retail: 14000, savings: 500, percentage_savings: 12
        },

        details: {
                title: "Master",
                director: [ "Rex", "Peter" ],
                writer: [ "Sam", "Jones" ],
                length: "3 Hours"
        },
}
```

Question3. Show all documents in the collection Product

```
> db.products.find().pretty()
```

Question4. all products with a discount greater than 10%, sorted by descending percentage discount

```
>db.products.find({"pricing.list":{$gt:10}}).sort({"pricing.
  percentage_savings":1});
```

Question5. Find documents for the audio albums of folk genre, sorted in reverse chronological order

```
> db.products.find({"details.genre":"folk"}).sort({_id:-1})
```

Question6. Find films that is acted by actor 'Vijay', sorted by release date

```
> db.products.find({description:"Movie by Actor Vijay"},
  {title:1, _id:0}).
```

Question7. Find movies with a particular word in the title

```
> db.product.find({title:"Master"})
```

Question8. Write atleast 3 queries for Product collection that are useful for Ecommerce website

1)
```
>db.products.find({},{pricing:1, type:1, _id:0})
```

2)
```
> db.products.find({"shipping.weight":{$not:{$gte:9}}},
  {pricing:1, type:1, _id:0})
```

3) > db.products.find({"pricing.savings": {$gte: 300}},
{pricing:1, type:1, details:1, _id:0})