

Question 1: THE NUMBER OF PRODUCTS IN THE COLLECTION

```
> db.products.countDocuments({})
```

Output:

```
> 4802
```

Question 2: THE PRODUCT WHICH HAS “SHARON’S, SORBET, DUTCH CHOCOLATE” AS THE NAME.

```
> db.products.find({product_name:"Sharon's, sorbet, dutch chocolate"},{product_name:true, _id:true})
```

Output:

```
> { _id: '0009073102079',  
  product_name: 'Sharon\'s, sorbet, dutch chocolate' }
```

Question 3: How many times the product having 0009073102079 as _id has been modified. Pay attention: how do you match the value of the _id? Think about how to do and how you should have done this match in another context.

In the database, the number of times the product in a given collection was updated is given by the key “rev” for each product. Hence, in order to look the number of modifications, we just need to call out the “rev” key.

```
> db.products.find({_id:"0009073102079"},{rev:true})
```

Output:

```
> { _id: '0009073102079', rev: 2 }
```

Question 4: The products that have sodium in the nutriments list.

```
> db.products.find({"nutriments.sodium" : { $exists : true }},{_id:true})
```

Output:

```
>
```

Question 5: The products that have the nutriscore_grade equal c.

```
> db.products.find({nutrition_grades_tags: ["c"]}).count()
```

Output:

```
> 287
```

Question 6: **How many different creators participated in the product creation**

```
> db.products.distinct('creator').length
```

Output:

```
> 131
```

Question 7:

Question 8: **The product(s) the most recently modified**

```
>
```

```
db.products.find({}, {last_modified_t: true, product_name: true}).sort({last_modified_t: -1}).limit(5)
```

Output:

```
> { _id: '000123456789', last_modified_t: 1636317917 }
```

```
{ _id: '0003128296',  
  product_name: 'Lasagne Bolognese',  
  last_modified_t: 1636314906 }
```

```
{ _id: '00053334',  
  last_modified_t: 1636301807,  
  product_name: 'Mixed fruit' }
```

```
{ _id: '00043298',  
  last_modified_t: 1636301694,  
  product_name: 'Vanilla Cola' }
```

```
{ _id: '0008577002892',  
  product_name: 'Mapple syrup',  
  last_modified_t: 1636292107 }
```

Question 9: The products that have exactly 1 ingredient.

```
>
```

Output:

```
> { _id: '000', product_name: 'hemp seeds' }
```

```
{ _id: '0000000016117',
```

```

    product_name: 'Organic Long Grain White Rice' }
{ _id: '0000000016094', product_name: 'Organic Polenta' }
{ _id: '0000000016612', product_name: 'Organic Adzuki Beans' }
{ _id: '0000000016650', product_name: 'Organic Penne Pasta' }
{ _id: '0000000016933',
  product_name: 'Organic Golden Flax Seeds' }

```

> Number of products: 239

Question 10: The products that have 20 or more ingredients.

```
> db.products.find({"ingredients_n":{$gt:20}},{product_name:true})
```

Output:

```

> { _id: '000000000001', product_name: 'hyde icon' }
{ _id: '0000000001199', product_name: 'Solène céréales poulet' }
{ _id: '0000000003827', product_name: 'Suedois saumon' }
{ _id: '0000000005272',
  product_name: 'Sandwich solene céréales sicilien' }
{ _id: '0000000005470', product_name: 'BAguette bressan' }
{ _id: '0000000016124', product_name: 'Organic Muesli' }
{ _id: '0000000016872', product_name: 'Zen Party Mix' }
{ _id: '0000000018289',
  product_name: 'Antioxidant Mix - Berries & Chocolate' }

```

Question 11: How many products are characterized as "desserts".

```
> db.products.find({"_keywords":"dessert"}).count()
```

Output:

```
> 47
```

Question 12: How many products are characterized by "chocolate".

```
> db.products.find({"_keywords":"chocolate"}).count()
```

Output:

```
> 536
```

Question 13: How many products are characterized by chocolate and dessert.

```
>
db.products.find({$and:[{"_keywords":"chocolate"}, {"_keywords":"dessert"}]}).
count()
Output:
> 3
```

Question 14: How many products are characterized by chocolate or dessert.

```
>
db.products.find({$or:[{"_keywords":"chocolate"}, {"_keywords":"dessert"}]}).c
ount()
Output:
> 580
```

Question 15: For the documents inside the collection, provide a query that converts the type of field categories from a String to an array and moves data into the new attribute called new_att_category

```
> db.products.updateMany({}, {'$set':{'new_att_category':{'$split':
['$categories', ' ', ' ' ]}}}, false, true)
```

Question 16: How many products have nutriscore_grade equal to F and contain ingredients with palm oil?

```
>
db.products.find({$and:[{nutriscore_grade:["f"]}, {"ingredients_analysis_tags"
:"palm-oil"}, {"ingredients":"palm-oil"}]}).count()

Output:
> 0
```

