



# Demo Day M1 “Grocery store”

## Definición del proyecto

Fuentes BD

Problema a resolver

## Definición de la base de datos

Diagrama entidad relación

Glosario

Creación de la base de datos

## Solución de los objetivos

## Fundamentos SQL y MongoDB

1. ¿Cuáles productos se vendieron del 01 enero al 31 de enero del 2018?
2. ¿Cuáles han sido los 5 días con las mayores ventas?
3. ¿Cuáles son los productos que se vendieron de la categoría vegetales?
4. ¿Cuáles productos han vendido más de 10 unidades?
5. ¿Cuál fue el día que más impuestos se cobraron?

## Agrupaciones y sub consultas SQL. Consultas MongoDB

1. ¿Cuáles presentaciones se cuenta del producto “Bambino”?
2. ¿Cuántas variedades de producto tiene cada categoría y cuántas unidades promedio tiene cada producto?
3. ¿Cuáles son los productos en presentación de 200ml?
4. ¿Cuáles son los productos hechos de maíz?
5. Presenta una tabla donde se aprecie el nombre del producto, su categoría y las unidades vendidas en total del mismo.

## Join SQL. Agregaciones MongoDB

1. ¿Cuál es la venta neta, utilidad neta y utilidad sobre la venta mensual de cada categoría?
2. ¿Cuáles son las 5 categorías de productos que aportaron menor utilidad neta en el año?
3. ¿Cuál es la categoría, la utilidad y rotación de cada producto?
4. ¿Cuáles son las 5 categorías más costosas en el flujo de venta? Muestra su costo y monto de venta.
5. ¿Cuál es el costo, ingreso, utilidad y rotación de cada categoría de productos?

## Vistas

Producto\_categoria\_unidadesvendidas

Utilidad\_rotacion

## Conclusiones

## Definición del proyecto

### Fuentes BD

Los datos corresponden a una tienda de conveniencia (grocery\_sotre) en Polonia tomados de:

<https://www.kaggle.com/agatii/total-sale-2018-yearly-data-of-grocery-shop>. Se conforma de tres tablas:

1. Day\_sell: Información sobre la venta total por día del 2018.
2. SELL\_1: Reporte mensual de la venta de productos durante 2018.
3. ROTATION: La rotación de los productos en aparador desde enero a septiembre del 2018.

### Problema a resolver

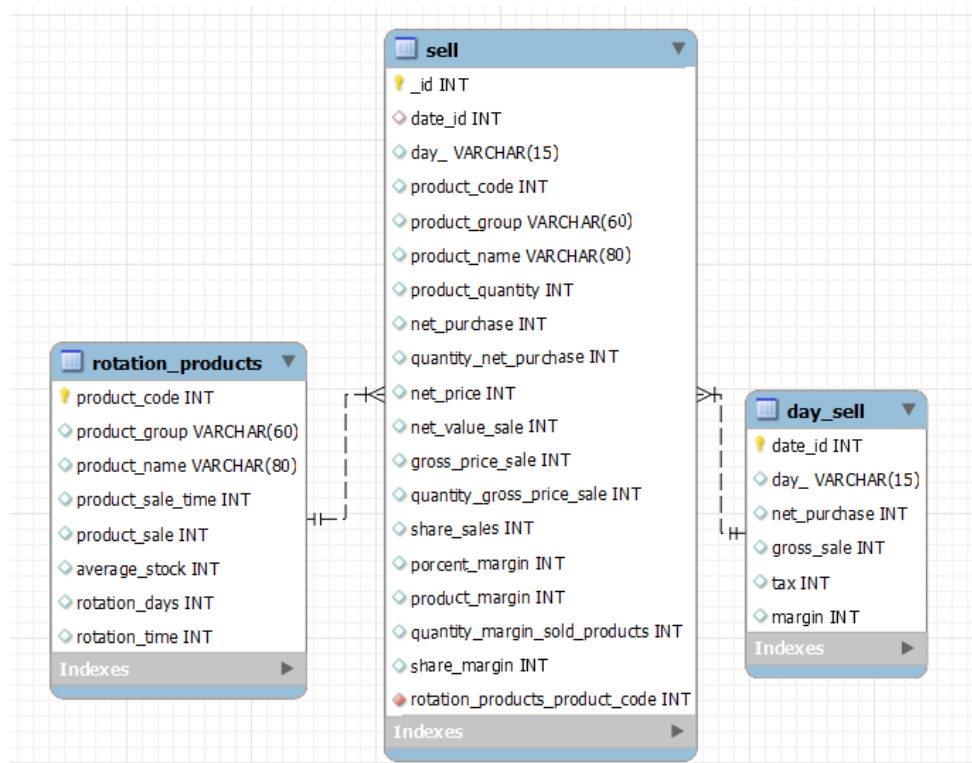
La tienda de conveniencia sufre un decremento en el ingreso del flujo de efectivo. Dicho problema se pretende resolver en el presente proyecto de dos maneras: primero al identificar y eliminar del inventario las categorías de productos que generan merma en la operación y, segundo al detectar fechas de baja operación en la que se puedan ejecutar actividades promocionales para aumentar la venta promedio.

Objetivos:

1. Identificar la categoría de productos con bajas ventas y con menor contribución marginal.
2. Identificar la categoría de productos con altas ventas y contribución marginal.
3. Identificar el mes con menos venta así como el comportamiento de venta de las categorías de productos dentro de dicho mes.

## Definición de la base de datos

### Diagrama entidad relación



### Glosario

Variable	Definición
Date	Día de venta
zn	Costo de venta por día
sb	Venta neta por día (con impuestos)
tax	Impuestos
marza	Utilidad por día
PKod	Número identificador de producto

Variable	Definición
Pgroup	Categoría de producto
Pname	Nombre del producto
Pquantity	Unidades del producto vendidas
pce_zn	Costo de venta por unidad
pwa_zn	Costo de venta por unidades vendidas
pce_sn	Precio de venta por unidad
pwa_sn	Precio de venta por unidades vendidas
pce_sb	Total de venta (con impuestos) por unidad
pwa_sb	Total de venta (con impuestos) por unidades vendidas
pudzs	Participación en las ventas (%)
pmarza	Margen de utilidad (%)
p_sales_in_time	Ventas del producto en el año
Psale	Cantidad de unidades vendidas
Pavarage_stock	Unidades promedio en inventario
Rotation_in_days	Rotación del producto por día

## Creación de la base de datos

### 1. SQL

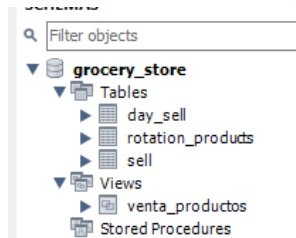
La tabla “day\_sell” y “sell” fueron cargadas en su totalidad, sin embargo, en la tabla “rotation\_productos” se eliminaron los valores cuya venta era de cero unidades por no ser representativos para el objetivo del proyecto.

```
CREATE DATABASE IF NOT EXISTS grocery_store;
use grocery_store;
create table if not exists day_sell (
  date_id int primary key,
  day_ varchar (15),
  net_purchase int,
  gross_sale int,
  tax int,
  margin int
);

create table if not exists sell (
  _id int primary key,
  date_id int,
  day_ varchar (15),
  product_code int,
  product_group varchar (60),
  product_name varchar (80),
  product_quantity int,
  net_purchase int,
  quantity_net_purchase int,
  net_price int,
  net_value_sale int,
  gross_price_sale int,
  quantity_gross_price_sale int,
  share_sales int,
  percent_margin int,
  product_margin int,
  quantity_margin_sold_products int,
  share_margin int
);

create table if not exists rotation_products (
  product_code int primary key,
  product_group varchar (60),
  product_name varchar (80),
  product_sale_time int,
  product_sale int,
```

```
average_stock int,  
rotation_days int,  
rotation_time int  
);
```



## 1. MongoDB

Los datos fueron cargados de manera completa en MongoDB

Collections				
<div> <div>Create collection</div> <div>View</div> <div> <div></div> <div></div> </div> </div> <div>Sort by Collection Name</div> <div></div>				
day_sell				
Storage size: 32.77 kB	Documents: 361	Avg. document size: 102.00 B	Indexes: 1	Total index size: 36.86 kB
product_sell				
Storage size: 2.48 MB	Documents: 21 K	Avg. document size: 325.00 B	Indexes: 1	Total index size: 548.86 kB
<div>producto_categoria_unidades</div> <div> <div>VIEW</div> <div>READ-ONLY</div> </div> <div>View on: product_sell</div>				
rotation_products				
Storage size: 737.28 kB	Documents: 16 K	Avg. document size: 193.00 B	Indexes: 1	Total index size: 368.64 kB

## Solución de los objetivos

La tienda cuenta con treinta y seis categorías de producto, de las cuales la que más ventas genera son los cigarros y en último lugar los lácteos.

costo_venta_utilidad				
	_id String	costo Double	venta Double	utilidad_venta Double
1	"CIGARETTES"	177809.09	189457.62	11648.53
2	"DAIRY_CHEESE"	87358.55	107821.18	20462.63
3	"BREAD"	85886.42	105849.8	19963.38
4	"BEER"	72820.22	93350.49	20530.27
5	"VEGETABLES"	62141.04	82969.79	20828.75
6	"VODKA_ALCOHOL"	59027.25	75380.11	16352.86
7	"DRINK_JUICE"	35134	51331.090000000004	16197.09
8	"SWEETS"	30482.06	43660.24	13178.18
9	"GENERAL_FOOD"	25749.23	34882.43	9133.2
10	"ICE_CREAMS_FROZEN"	23505.53	33758.48	10252.95
11	"KETCH_CONCENTRATE_MUSTARD_MAJO_"	17216.22	23712.46	6496.24
12	"CHEMISTRY"	15664.5	22484.46	6819.96

costo_venta_utilidad				
	_id String	costo Double	venta Double	utilidad_venta Double
21	"FISH"	4250.61	5843.56	1592.95
22	"WINE_ALCOHOL 18%"	4202.99	5665.72	1462.73
23	"EGGS"	4584.82	5638.5599999999995	1053.74
24	"SPICES"	3362.6	4918.82	1556.22
25	"CHEWING_GUM_LOLIPOPS"	2999.15	4661.81	1662.66
26	"COOKIES_BULK"	3357.83	4640.59	1282.76
27	"TABLETS"	2808.12	3712.12	904
28	"OCCASIONAL"	2016.71	3385.06	1368.35
29	"SOCKS_THIGHTS"	968.85	1405.13	436.28
30	"ART/_HYGIENIC"	620.39	920.43	300.04
31	"DISHES_FOR_CHILDREN"	595.96	821.47	225.51
32	"PET'S FOOD"	465.36	698.17	232.81

33	"LUNCH DINING DISHES"	270.65	381.78999999999996	111.14
34	"PACKAGES"	202.26	219.10999999999999	16.85
35	"ARTICLE_OF_HOUSEHOLD"	32.81	48.67	15.860000000000001
36	"MILK"	12.12	16.5	4.38

- Identificar la categoría de productos con bajas ventas y con menor contribución marginal.

Podemos notar que los productos con baja utilidad también presentan una baja rotación de inventario

```
[{$addFields: {
  date_: {
    $dateFromString: {
      dateString: '$Date',
      format: '%d/%m/%Y'
    }
  }
}}, {$lookup: {
```

```

from: 'rotation_products',
localField: 'PKod',
foreignField: 'PKod',
as: 'rotation_p'
}}, {$addFields: {
  rotation_p: {
    $reduce: {
      input: '$rotation_p',
      initialValue: {
        $arrayElemAt: [
          '$rotation_p',
          0
        ]
      },
      'in': {
        $cond: {
          'if': {
            $gt: [
              '$$this.Pgroup',
              '$$value.Pgroup'
            ]
          },
          then: '$$this',
          else: '$$value'
        }
      }
    }
  }
}}, {$addFields: {
  rotation_product: '$rotation_p.Rotation_in_days'
}}, {$group: {
  _id: '$Pgroup',
  costo: {
    $sum: '$pwa_zn'
  },
  venta: {
    $sum: '$pwa_sn'
  },
  utilidad_venta: {
    $sum: '$pkwmarza'
  },
  'rotación_promedio': {
    $avg: '$rotation_product'
  }
}}, {$sort: {
  'rotación_promedio': 1
}}

```

Output after [\\$sort](#) stage ⓘ (Sample of 20 documents)

```

_id: "LUNCH DINING DISHES"
costo: 270.65
venta: 381.78999999999996
utilidad_venta: 111.14
rotación_promedio: 0

```

```

_id: "MILK"
costo: 12.12
venta: 16.5
utilidad_venta: 4.38
rotación_promedio: 0

```

Output after [\\$group](#) stage (sample of 20 documents)

```
_id: "EGGS"
costo: 4584.82
venta: 5638.5599999999995
utilidad_venta: 1053.74
rotación_promedio: 0
```

```
_id: "PACKAGES"
costo: 202.26
venta: 219.10999999999999
utilidad_venta: 16.85
rotación_promedio: 0
```

- Identificar el mes con menor venta y el comportamiento de venta de las categorías de productos dentro de dicho mes.

Encontramos que el mes con menores ventas tiene es agosto, siendo la categoría con menor venta "Artículos of household"

```
[{$addFields: {
  date_: {
    $dateFromString: {
      dateString: '$Date',
      format: '%d/%m/%Y'
    }
  }
}}, {$addFields: {
  mes: {
    $month: '$date_'
  }
}}, {$group: {
  _id: '$mes',
  costo: {
    $sum: '$pwa_zn'
  },
  venta: {
    $sum: '$pwa_sn'
  },
  utilidad_venta: {
    $sum: '$pkwmarza'
  }
}}, {$sort: {
  venta: -1
}}]
```

```
_id: 8
costo: 83723.21
venta: 105377.96
utilidad_venta: 21654.75
```

ventas agosto				
	categoria String	venta_neta Double	utilidad_neta Double	utilidad_venta Double
1	"ARTICLE_OF_HOUSEHOLD"	1.06	0.32	30.19
2	"LUNCH DINING DISHES"	37.69	12.24	31.45000000000003
3	"PACKAGES"	41.18	2.42	21.72
4	"DISHES_FOR_CHILDREN"	45.97	11.29	23.8425
5	"SOCKS_THIGHTS"	73.15	23.62	32.053636363636365
6	"PET'S FOOD"	74.9	30.150000000000002	36.75857142857143
7	"ART/_HYGIENIC"	81.71000000000001	28.27	34.90571428571429
8	"OCCASIONAL"	192.87	119.35	46.150909090909096
9	"TABLETS"	220.47	54.11	26.025555555555556
10	"SPICES"	436.11	142.51	31.917462686567163

Categorías con menores ventas durante Agosto

## Fundamentos SQL y MongoDB

### 1. ¿Cuáles productos se vendieron del 01 enero al 31 de enero del 2018?

#### SQL

```
select day_, product_name from sell where date_id >=7 and date_id <=37;
```

day_	product_name
01/01/2018	PATYCZKI BELLA 200
01/01/2018	ACE 1 L
01/01/2018	ALMETTE CZOSNEK NIEDZWIEDZI 150GR
01/01/2018	ALMETTE JOGURTOWY 150G HOCHLAND
01/01/2018	ALMETTE OGOREK/ZIOLA 150G
01/01/2018	ALMETTE SER POMIDOR 150 G
01/01/2018	ALMETTE SZCZYPIOREK CEBULA 150G
01/01/2018	ALMETTE SZPINAK 150 G

#### MongoDB

```
[{$addFields: {
  date_: {
    $dateFromString: {
      dateString: '$Date',
      format: '%d/%m/%Y'
    }
  }
}}, {$match: {
  $and: [
    {
      date_: {
        $gte: ISODate('2018-01-01T00:00:00.000Z')
      }
    }
  ]
}
```



```

date_: {
  $lte: ISODate('2018-01-31T00:00:00.000Z')
}
}
}
}}

```

## grocery\_store.product\_sell

DOCUMENTS 21.3k STORAGE SIZE 2.5MB AVG. SIZE 325B INDEXES 1 TOTAL SIZE 548.9KB AVG. 548.9

Documents Aggregations Schema Explain Plan Indexes Validation

COLLATION Untitled - Modified SAVE SAMPLE MODE AUTO PREVIEW

**\$addFields** Output after \$addFields stage (Sample of 20 documents)

```

1 /**
2  * newField: The new field name.
3  * expression: The new field expression.
4  */
5 {
6   date_: {
7     $dateFromString: {
8       dateString: "$date",
9       format: "%d/%m/%Y"
10    }
11  }
12 }

```

```

pwa_sn: 2.57
pce_sb: 2.78
pwa_sb: 2.78
pudzb: 0
pmarza: 31.13
pmarzajedn: 0.8
pkwmarza: 0.8
pudzmarza: 0.01
date_: 2018-01-01T00:00:00.000+00:00

```

```

_id: ObjectId("61ee031e8281e78e240f5cda")
id_Date: 7
Date: "01/01/2018"
PKod: 5986
Pgroup: "CHEMISTRY"
Pname: "ACE 1 L"
Pquantity: 1
pce_zn: 3.2

```

**\$match** Output after \$match stage (Sample of 20 documents)

```

1 /**
2  * query: The query in MQL.
3  */
4 {
5   $and: [
6     {date_: {$gte: new Date("2018-01-01")}},
7     {date_: {$lte: new Date("2018-01-31")}}
8   ]
9 }

```

```

_id: ObjectId("61ee031e8281e78e240f5cda")
id_Date: 7
Date: "01/01/2018"
PKod: 5987
Pgroup: "CHEMISTRY"
Pname: "PATYCZKI BELLA 200"
Pquantity: 1
pce_zn: 1.77

```

```

Pgroup: "CHEMISTRY"
Pname: "ACE 1 L"
Pquantity: 1
pce_zn: 3.2
pwa_zn: 3.2
pce_sn: 5.68
pwa_sn: 5.68
pce_sb: 6.99
pwa_sb: 6.99
pudzb: 0.01

```

## 2. ¿Cuáles han sido los 5 días con las mayores ventas?

### SQL

```

select day_, gross_sale from day_sell
order by gross_sale desc limit 5;

```

Result Grid	Filter Rows:
day_	gross_sale
22/12/2018	6427
24/12/2018	5519
26/04/2018	5400
26/05/2018	5377
10/11/2018	5364

### MongoDB

```

[{$sort: {
  sb: -1
}}, {$limit: 5}]

```

Documents Aggregations Schema Explain Plan Indexes Validation

COLLATION Untitled- Modified SAVE SAMPLE MODE AUTO PREVIEW

**\$sort** Output after \$sort stage (Sample of 20 documents)

```
1 /**
2  * Provide any number of field/order pairs.
3  */
4 {
5   sb: -1
6 }
```

Output after \$sort stage (Sample of 20 documents)

```
{ "_id": "61ee02028281e78e240eb66e", "ID": 359, "Date": "22/12/2018", "zn": 4416.64, "sb": 6426.81, "tax": 734.23, "marza": 1275.94 }
```

```
{ "_id": "61ee02028281e78e240eb670", "ID": 361, "Date": "24/12/2018", "zn": 3807.9, "sb": 5518.87, "tax": 701.45, "marza": 1009.52 }
```

**\$limit** Output after \$limit stage (Sample of 5 documents)

```
1 /**
2  * Provide the number of documents to limit.
3  */
4 5
```

Output after \$limit stage (Sample of 5 documents)

```
{ "_id": "61ee02028281e78e240eb66e", "ID": 359, "Date": "22/12/2018", "zn": 4416.64, "sb": 6426.81, "tax": 734.23, "marza": 1275.94 }
```

```
{ "_id": "61ee02028281e78e240eb670", "ID": 361, "Date": "24/12/2018", "zn": 3807.9, "sb": 5518.87, "tax": 701.45, "marza": 1009.52 }
```

### 3. ¿Cuáles son los productos que se vendieron de la categoría vegetales?

#### SQL

```
select product_group, product_name from sell where product_group = "vegetables";
```

```
47 • select product_group, product_name from sell where product_group = "vegetables";
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

product_group	product_name
VEGETABLES	AMINO BLYSK ZUREK
VEGETABLES	DR POLEWA MLECZNA TWARDA
VEGETABLES	KT ANANAS
VEGETABLES	KT BANAN
VEGETABLES	KT BROKUL
VEGETABLES	KT BURACZEK CZERWONY
VEGETABLES	KT CEBUL CZERWONNA MODRA
VEGETABLES	KT CEBUL BIAŁA

#### MongoDB

```
[{$match: {
  Pgroup: 'VEGETABLES'
}}]
```

grocery\_store.product\_sell

DOCUMENTS 21.3k STORAGE SIZE 2.5MB AVG. SIZE 325B INDEXES 1 TOTAL SIZE 548.9KB AVG. 548.1

Documents Aggregations Schema Explain Plan Indexes Validation

COLLATION Untitled - Modified SAVE SAMPLE MODE AUTO PREVIEW

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

`{ $match: {`  
 `_id: ObjectId("61ee031e8281e78e240f5cda")`  
 `id_Date: 7`  
 `Date: "01/01/2018"`  
 `PKod: 5307`  
 `Pgroup: "CHEMISTRY"`  
 `Pname: "PATYCZKI BELLA 200"`  
 `Pquantity: 1`  
 `pce_zn: 1.77`  
`}`

`{ $match: {`  
 `_id: ObjectId("61ee031e8281e78e240f5cda")`  
 `id_Date: 7`  
 `Date: "01/01/2018"`  
 `PKod: 5986`  
 `Pgroup: "CHEMISTRY"`  
 `Pname: "ACE 1 L"`  
 `Pquantity: 1`  
 `pce_zn: 3.2`  
`}`

Output after \$match stage (Sample of 20 documents)

`1 /**`  
`2 * query: The query in MQL.`  
`3 */`  
`4 {`  
`5 Pgroup: "VEGETABLES"`  
`6 }`

`{ $match: {`  
 `_id: ObjectId("61ee031e8281e78e240f5cda")`  
 `id_Date: 7`  
 `Date: "01/01/2018"`  
 `PKod: 8404`  
 `Pgroup: "VEGETABLES"`  
 `Pname: "AMINO BLYSK ZUREK"`  
 `Pquantity: 6`  
 `pce_zn: 0.82`  
`}`

`{ $match: {`  
 `_id: ObjectId("61ee031e8281e78e240f5cda")`  
 `id_Date: 7`  
 `Date: "01/01/2018"`  
 `PKod: 6483`  
 `Pgroup: "VEGETABLES"`  
 `Pname: "DR POLENA MLECZNA TWARDA"`  
 `Pquantity: 1`  
 `pce_zn: 1.94`  
`}`

#### 4. ¿Cuáles productos han vendido más de 10 unidades?

##### SQL

```
select product_name, product_sale from rotation_products where product_sale >10;
```

```
--
61 • select product_name, product_sale from rotation_products where product_sale >10;
62
63
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [A](#)

product_name	product_sale
BELLA PODPASKI NOVA MAXI 10	18
BELLA PODPASKI NORMAL MAXI 10	24
RECZNIK PAPIER SZAST PRAST	11
NIVEA KREM 50ML	38
WKLADKI BELLA PANTY 20SZT	17
WKLADKI BELLA PANTY	20
PLATKI BELLA 80+30%	16
OPPI AMTAC7 VANISH 100ML POWN SASZETKA	22

rotation\_products 15 x

##### MongoDB

```
[{$match: {
  Psale: {
    $gt: 10
  }
}}]
```

COLLATION Untitled- Modified SAVE

SAMPLE MODE AUTO PREVIEW

16293 Documents in the Collection Preview of Documents in the Collection

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

```

_id: ObjectId("61ee036f8281e78e240fb010")
Pgroup: "ART._HYGIENIC"
Lp: 1
PKod: 156
Pname: "WATA 100G BELLA"
p_sale_in_time: 0.021390374
Psale: 8
Pavarage_stock: 0
Rotation_in_days: 0
    
```

```

_id: ObjectId("61ee036f8281e78e240fb010")
Pgroup: "ART._HYGIENIC"
Lp: 1
PKod: 155
Pname: "WATA BAWELNIANA BELLA 100G"
p_sale_in_time: 0.005347594
Psale: 2
Pavarage_stock: 0
Rotation_in_days: 0
    
```

\$match

```

1 // **
2 * query: The query in MQL.
3 //
4 {
5   Psale: {$gt: 10}
6 }
    
```

Output after \$match stage (Sample of 20 documents)

```

_id: ObjectId("61ee036f8281e78e240fb072")
Pgroup: "CHEMISTRY"
Lp: 1
PKod: 11597
Pname: "BREF 4-FORMULA AKTIV"
p_sale_in_time: 0.104278075
Psale: 39
Pavarage_stock: 0
    
```

## 5. ¿Cuál fue el día que más impuestos se cobraron?

### SQL

```
select day_, tax from day_sell order by tax desc limit 1;
```

62 • select day\_, tax from day\_sell order by tax desc limit 1;

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

day_	tax
01/09/2018	737

### MongoDB

```
[{$sort: {
  tax: -1
}}, {$limit: 1}]
```

grocery\_store.day\_sell DOCUMENTS 361

Documents Aggregations Schema Explain Plan Indexes Validation

COLLATION Untitled- Modified SAVE

```

1 /**
2  * Provide any number of field/order pairs.
3  */
4 {
5   tax: -1
6 }

```

```

_id: ObjectId("61ee02028281e78e240eb5ff")
ID: 248
Date: "01/09/2018"
zn: 3634.82
sb: 5325.8
tax: 736.87
marza: 954.11

```

\$limit Output after \$limit stage (Sample of 1 document)

```

1 /**
2  * Provide the number of documents to limit.
3  */
4 1

```

```

_id: ObjectId("61ee02028281e78e240eb5ff")
ID: 248
Date: "01/09/2018"
zn: 3634.82
sb: 5325.8
tax: 736.87
marza: 954.11

```

## Agrupaciones y sub consultas SQL. Consultas MongoDB

### 1. ¿Cuáles presentaciones se cuenta del producto “Bambino”?

SQL

```
select product_name from sell where product_name like "Bambino%";
```

```

63
64 • select product_name from sell where product_name like "Bambino%";
65

```

Result Grid Filter Rows: Export: Wrap Cell Content:

product_name
BAMBINO OLIWKA DLA DZIECI 150G
BAMBINO SZAMPON 400 ML
BAMBINO SZAMPON 400 ML
BAMBINO OLIWKA DLA DZIECI 150G
BAMBINO OLIWKA DLA DZIECI 150G
BAMBINO SZAMPON 400 ML
BAMBINO CHUSTECZKI NASACZ/ 63 SZT
BAMBINO SZAMPON 400 ML




## MongoDB

```
{
  filter: {
    Pname: RegExp('^BAMBINO')
  }
}
```

groce...rotation\_pr... DOCUMENTS 16.3k STORAGE 340.

Documents Aggregations Schema Explain Plan

**FILTER** {Pname: /^BAMBINO/}

**ADD DATA** VIEW    Display

```
{
  "_id": ObjectId("61ee036f8281e78e240fb00d"),
  "Pgroup": "ART_HYGIENIC",
  "Lp": 1,
  "PKod": 8478,
  "Pname": "BAMBINO SZAMPON 400 ML",
  "p_sale_in_time": 0.013368984,
  "Psale": 5,
  "Pavarage_stock": 0,
  "Rotation_in_days": 0
}
```

```
{
  "_id": ObjectId("61ee036f8281e78e240fb024"),
  "Pgroup": "ART_HYGIENIC",
  "Lp": 1,
  "PKod": 10138,
  "Pname": "BAMBINO CHUSTECZKI NAS?CZ.63",
  "p_sale_in_time": 0,
  "Psale": 0,
  "Pavarage_stock": 0,
  "Rotation_in_days": NaN,
  "Rotation in times": NaN
}
```

## 2. ¿Cuántas variedades de producto tiene cada categoría y cuántas unidades promedio tiene cada producto?

### SQL

```
select product_code,
count(product_name) as total_products,
sum(product_quantity) as products_quantity,
(sum(product_quantity) / count(product_name)) as avg_products
from sell group by product_code;
```

```

66 • select product
67 count(product_name) as total_products,
68 sum(product_quantity) as products_quantity,
69 (sum(product_quantity) / count(product_name)) as avg_products
70 from sell group by product_code;

```

result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

product_code	total_products	products_quantity	avg_products
5307	7	16	2.2857
5986	3	4	1.3333
399	6	22	3.6667
2246	7	27	3.8571
2247	9	38	4.2222
1305	6	22	3.6667
1142	7	22	3.1429
17316	8	33	4.1250
3818	4	15	3.7500

result 27 x

Output

Action Output

## MongoDB

```

[{$group: {
  _id: '$Pgroup',
  Variedades: {
    $sum: 1
  },
  unidades: {
    $sum: '$Pquantity'
  }
}}, {$addFields: {
  Promedio_unidades: {
    $divide: [
      '$unidades',
      '$Variedades'
    ]
  }
}}]

```

COLLATION: Untitled- Modified SAVE

```

3  * fieldN: The first field name.
4  */
5  {
6    _id: "$Pgroup",
7    Variedades: {
8      $sum: 1
9    },
10   unidades: {
11     $sum: "$Pquantity"
12   }
13 }

```

Output after \$addFields stage (Sam)

```

1  /**
2   * newField: The new field name.
3   * expression: The new field expression.
4   */
5  {
6    Promedio_unidades: {$divide:[
7      "$unidades", "$variedades "
8    ]}
9  }

```

Output after \$addFields stage (Sam)

```

_id: "DRINK_JUICE"
Variedades: 1856
unidades: 25456

```

Output after \$addFields stage (Sam)

```

_id: "CANNED_PATE"
Variedades: 336
unidades: 2470.723
Promedio_unidades: null

```

### 3. ¿Cuáles son los productos en presentación de 200ml?

#### SQL

```
select product_name from sell where product_name like "%200 ML%";
```

72 • select product\_name from sell where product\_name like "%200 ML%";

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

product_name
LODY COFFE LATTE 200 ML
LODY ICE FRESH 200 ML
NAP NAPI MULTIWITAMINA 200 ML
OLEJ UNIAN 200 ML
LODY COFFE LATTE 200 ML
NAP NAPI MULTIWITAMINA 200 ML
NAP TYMB/MULTIW/200 ML
OLEJ KOKOSOWY 200 ML

#### MongoDB

```

{
  filter: {
    Pname: RegExp('.*200 ML')
  }
}

```



grocery\_store.product\_sell DOC

**Documents** Aggregations Schema Explain Plan Indexes Validati

**FILTER** {Pname: /. \*200 ML/ }

**ADD DATA** **VIEW**

```

_id: ObjectId("61ee031e8281e78e240f5fcd")
id_Date: 7
Date: "01/01/2018"
PKod: 3006
Pgroup: "ICE_CREAMS_FROZEN"
Pname: "LODY COFFE LATTE 200 ML"
Pquantity: 4
pce_zn: 2.31
pwa_zn: 9.24
pce_sn: 3.23
pwa_sn: 12.92
pce_sb: 3.39
pwa_sb: 13.56
pudzs: 0.02
pmarza: 28.48
pmarzajedn: 0.92
pkwmarza: 3.68
pudzmarza: 0.04

```

#### 4. ¿Cuáles son los productos hechos de maíz?

##### SQL

```
select product_name from rotation_products where product_name like "%CORN%";
```

```

71
72 • select product_name from rotation_products where product_name like "%CORN%";
73

```

**Result Grid** Filter Rows:  | Export: | Wrap Cell Content:

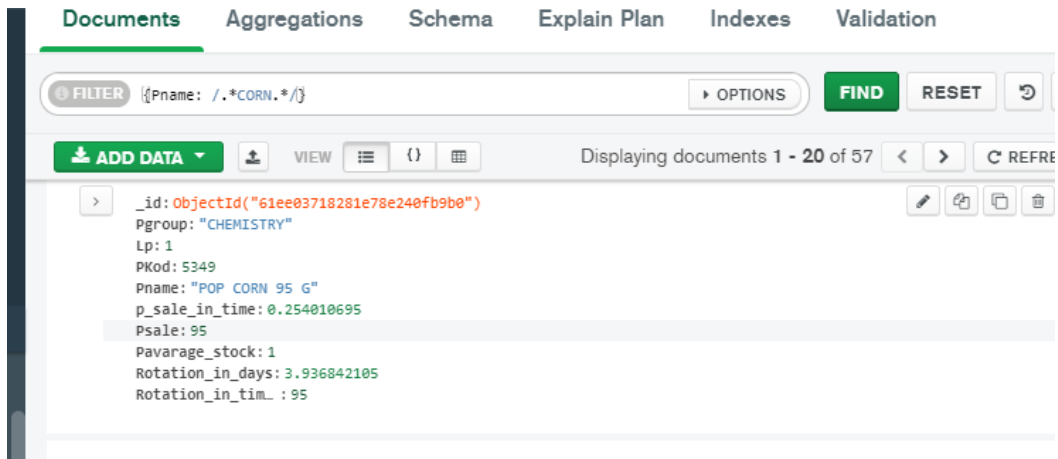
product_name
POP CORN 95 G

##### MongoDB

```

{
  filter: {
    Pname: RegExp('.*CORN.*')
  }
}

```



5. Presenta una tabla donde se aprecie el nombre del producto, su categoría y las unidades vendidas en total del mismo.

SQL

```
select product_name, product_group, sum(product_quantity) from sell group by product_name;
```

74 • `select product_name, product_group, sum(product_quantity) from sell group by product_name;`

Result Grid

product_name	product_group	sum(product_quantity)
PATYCZKI BELLA 200	CHEMISTRY	16
ACE 1L	CHEMISTRY	4
ALMETTE CZOSNEK NIEDZWIEDZI 150GR	DAIRY_CHESSE	22
ALMETTE JOGURTOWY 150G HOCHLAND	DAIRY_CHESSE	27
ALMETTE OGOREK/ZIOLA 150G	DAIRY_CHESSE	38
ALMETTE SER POMIDOR 150 G	DAIRY_CHESSE	22
ALMETTE SZCZYPIOREK CEBULA 150G	DAIRY_CHESSE	22

MongoDB

```
[{$group: {
  _id: {
    categoria: '$Pgroup',
    producto: '$Pname'
  },
  Unidades_vendidas: {
    $sum: '$Pquantity'
  }
}}, {$addFields: {
  Categoria: '$_id.categoria',
  producto: '$_id.producto'
}}, {$project: {
  _id: 0,
  Unidades_vendidas: 1,
  Categoria: 1,
  producto: 1
}}]
```

grocery\_store.producto\_categoria\_unidades READ-ONLY VIEW

Documents Aggregations Schema Explain Plan Indexes Validation

**FILTER** { field: 'value' }

VIEW VIEW { } VIEW

	producto_categoria_unidades		
	producto String	Unidades_vendidas Int32	Categoria String
1	"WEDEL CZEKOLADA NADZ TOFFI 100	5	"SWEETS"
2	"ZNICZ DUZY"	2	"GENERAL"
3	"BREF KULKI WC 50 G"	2	"CHEMISTRY"
4	"MUSZTARDA MIODOWA 270 G"	8	"DRINK_JUICE"
5	"PAP LUCKY STRIKE 22 RED"	69	"CIGARETTES"
6	"SOS BOLONSKI SLOIK DAWT 550G"	6	"GENERAL_FOOD"
7	"ZUREK BUTELKA 0_5 L"	7	"GENERAL"
8	"BELLA CHUSTECZKI ODSWIEZAJACE	7	"CHEMISTRY"

## Join SQL. Agregaciones MongoDB

1. ¿Cuál es la venta neta, utilidad neta y utilidad sobre la venta mensual de cada categoría?

SQL

```
select month(str_to_date(day_, "%d/%m/%Y")) AS MES,
product_group, (sum(net_value_sale)) as venta_total,
(sum(quantity_margin_sold_products)) as utilidad_neta,
(avg(porcent_margin)) as margen_utilidad
from sell group by product_group;
```

SQL File 1\* x

Limit to 1000 rows

```
29 • select month(str_to_date(day_, "%d/%m/%Y")) AS MES,
30     product_group, (sum(net_value_sale)) as venta_total,
31     (sum(quantity_margin_sold_products)) as utilidad_neta,
32     (avg(porcent_margin)) as margen_utilidad
33 from sell group by product_group;
34
```

Result Grid Filter Rows: Export: Wrap Cell Content: [A](#)

	MES	product_group	venta_total	utilidad_neta	margen_utilidad
▶	1	CHEMISTRY	22527	6806	32.1028
	1	DAIRY_CHESSE	107895	20480	24.3236
	1	GENERAL_FOOD	34910	9153	27.7878
	1	SWEETS	43650	13178	32.1685
	1	VEGETABLES	82976	20830	23.3175
	1	OILS	6598	1667	24.8994
	1	SPICES	4919	1552	30.8192
	1	TABLETS	3711	898	25.2683

## MongoDB

```
[{$addFields: {
  date_: {
    $dateFromString: {
      dateString: '$Date',
      format: '%d/%m/%Y'
    }
  }
}}, {$addFields: {
  mes: {
    $month: '$date_'
  }
}}, {$group: {
  _id: {
    categoria: '$Pgroup',
    mes: '$mes'
  },
  venta_neta: {
    $sum: '$pwa_sn'
  },
  utilidad_neta: {
    $sum: '$pkwmarza'
  },
  utilidad_venta: {
    $avg: '$pmarza'
  }
}}, {$addFields: {
  categoria: '$_id.categoria',
  mes: '$_id.mes'
}}, {$project: {
  _id: 0,
  venta_neta: 1,
  utilidad_neta: 1,
  utilidad_venta: 1,
  categoria: 1,
  mes: 1
}}, {$sort: {
  mes: 1
}}
```

Documents

grocery\_store.utilidad\_categoria\_mes READ-ONLY VIEW view on: grocery\_store.product\_sell

Documents Aggregations Schema Explain Plan Indexes Validation

FILTER { field: 'value' } OPTIONS FIND RESET

VIEW {}

Displaying documents 1 - 20 of 382

	categoria String	mes Int32	venta_neta Double	utilidad_neta Double	utilid
1	"ARTICLE_OF_HOUSEHOLD"	1	2.96	0.97	34.655
2	"VODKA_ALCOHOL"	1	6753.45	1455.5	21.636
3	"ICE_CREAMS_FROZEN"	2	1031.28	274.44	26.482
4	"SPICES"	3	495.9	151.54	30.279
5	"OILS"	7	735.79	198.05	27.58
6	"DRINK_JUICE"	9	6033.12	1946.71	32.675
7	"VEGETABLES"	9	7308.4	1243.1299999999999	23.743

## 2. ¿Cuáles son las 5 categorías de productos que aportaron menor utilidad neta en el año?

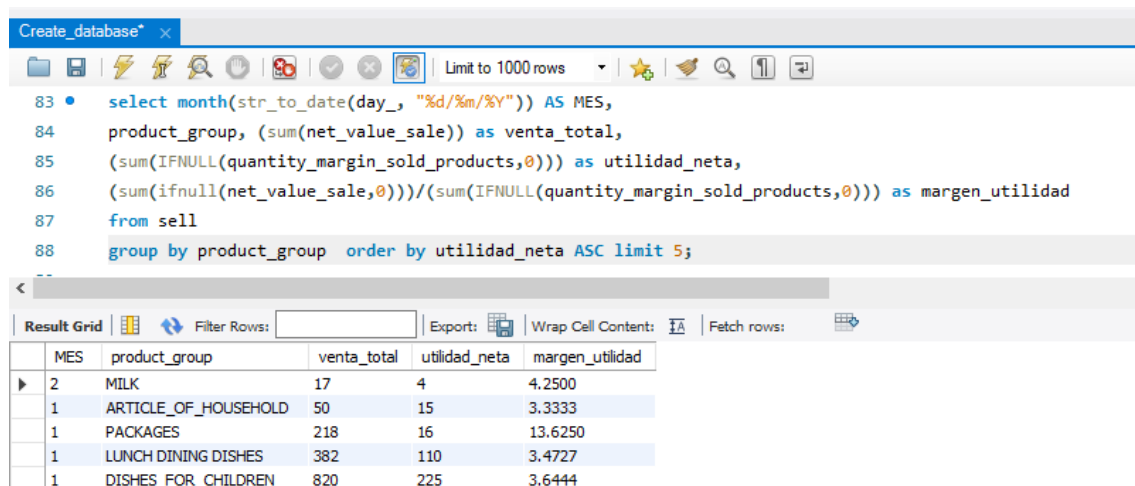
### SQL

```
-- Opción 1
select month(str_to_date(day_, "%d/%m/%Y")) AS MES,
product_group,
(sum(net_value_sale)) as venta_total,
(sum(IFNULL(quantity_margin_sold_products,0))) as utilidad_neta,
(sum(ifnull(net_value_sale,0))/(sum(IFNULL(quantity_margin_sold_products,0)))) as margen_utilidad
from sell
group by product_group order by utilidad_neta ASC limit 5;

-----

-- Opción 2
create view venta_margen_utilidad as
select month(str_to_date(day_, "%d/%m/%Y")) AS MES,
product_group, (sum(net_value_sale)) as venta_total,
(sum(quantity_margin_sold_products)) as utilidad_neta,
(avg(porcent_margin)) as margen_utilidad
from sell;

select MES, product_group, venta_total, utilidad_neta, margen_utilidad
from venta_margen_utilidad
group by product_group
order by utilidad_neta asc limit 5;
```



The screenshot shows a SQL IDE window titled "Create\_database\*". The query editor contains the following SQL code:

```
83 • select month(str_to_date(day_, "%d/%m/%Y")) AS MES,
84 product_group, (sum(net_value_sale)) as venta_total,
85 (sum(IFNULL(quantity_margin_sold_products,0))) as utilidad_neta,
86 (sum(ifnull(net_value_sale,0))/(sum(IFNULL(quantity_margin_sold_products,0)))) as margen_utilidad
87 from sell
88 group by product_group order by utilidad_neta ASC limit 5;
```

Below the query editor, the "Result Grid" tab is active, displaying the following data:

	MES	product_group	venta_total	utilidad_neta	margen_utilidad
2		MILK	17	4	4.2500
1		ARTICLE_OF_HOUSEHOLD	50	15	3.3333
1		PACKAGES	218	16	13.6250
1		LUNCH DINING DISHES	382	110	3.4727
1		DISHES_FOR_CHILDREN	820	225	3.6444

### MongoDB

```
[{$addFields: {
  date_: {
    $dateFromString: {
      dateString: '$Date',
      format: '%d/%m/%Y'
    }
  }
}}, {$addFields: {
  mes: {
    $month: '$date_'
  }
}}, {$group: {
  _id: {
    categoria: '$Pgroup'
  },
```

```

venta_neta: {
  $sum: '$pwa_sn'
},
utilidad_neta: {
  $sum: '$pkwmarza'
},
utilidad_venta: {
  $avg: '$pmarza'
}
}}, {$addFields: {
  categoria: '$_id.categoria'
}}, {$project: {
  _id: 0,
  venta_neta: 1,
  utilidad_neta: 1,
  utilidad_venta: 1,
  categoria: 1
}}, {$sort: {
  utilidad_neta: 1
}}, {$limit: 5]}

```

grocery\_store.5 productos poca utilidad READ-ONLY VIEW view on: grocery\_store.utilidad\_categoria

[Documents](#) [Aggregations](#) [Schema](#) [Explain Plan](#) [Indexes](#) [Validation](#)

**FILTER** { field: 'value' } OPTIONS FIND RESET

VIEW LIST JSON GRID Displaying documents 1 - 5 of 5 < >

	categoria String	venta_neta Double	utilidad_neta Double	utilidad_venta Double
1	"MILK"	16.5	4.38	26.55
2	"ARTICLE_OF_HOUSEHOLD"	48.67	15.860000000000001	32.96095238095238
3	"PACKAGES"	219.10999999999999	16.85	10.606111111111112
4	"LUNCH DINING DISHES"	381.78999999999996	111.14	29.20151515151515
5	"DISHES_FOR_CHILDREN"	821.47	225.51	26.823529411764707

### 3. ¿Cuál es la categoría, la utilidad y rotación de cada producto?

SQL

```

select s.product_name, s.product_group, quantity_margin_sold_products, rotation_time
from sell as s
join rotation_products as r
on s.product_code = r.product_code;

```

Create\_database\* x

Limit to 1000 rows

```

90 • select s.product_name, s.product_group, quantity_margin_sold_products, rotation
91 from sell as s
92 join rotation_products as r
93 on s.product_code = r.product_code;
94
95

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

	product_name	product_group	quantity_margin_sold_products	rotation_time
▶	ACE 1 L	CHEMISTRY	2	2
	BELLA HERBS 12 Z WERBENA	ART/_HYGIENIC	1	4
	BELLA PODPASKI NORMAL MAXI 10	ART/_HYGIENIC	4	24
	BELLA PODPASKI NOVA MAXI 10	ART/_HYGIENIC	-1	3
	BREF 4-FORMULA LEMON	CHEMISTRY	13	6
	BREF KULKI BLU AKTIV	CHEMISTRY	1	0
	CHUSTECZKI DO KOLOROW 10 SZT	CHEMISTRY	3	1
	CLIN ZAPAS 500 ML	CHEMISTRY	2	20
	CUKIERKI KROWKA WAWEL/SOLIDARNOSC	COOKIES_BULK	2	1
	CUKIERKI TRUFLE ODRA LUZ	COOKIES_BULK	5	13

## MongoDB

```

[{$addFields: {
  date_: {
    $dateFromString: {
      dateString: '$Date',
      format: '%d/%m/%Y'
    }
  }
}}, {$lookup: {
  from: 'rotation_products',
  localField: 'PKod',
  foreignField: 'PKod',
  as: 'rotation_p'
}}, {$addFields: {
  rotation_p: {
    $reduce: {
      input: '$rotation_p',
      initialValue: {
        $arrayElemAt: [
          '$rotation_p',
          0
        ]
      },
      'in': {
        $cond: {
          'if': {
            $gt: [
              '$$this.Pgroup',
              '$$value.Pgroup'
            ]
          },
          then: '$$this',
          else: '$$value'
        }
      }
    }
  }
}}, {$addFields: {
  rotation_product: '$rotation_p.Rotation_in_days'
}}, {$project: {
  _id: 0,
  Pname: 1,
  Pgroup: 1,
  pmarzajedn: 1,

```

```
rotation_product: 1
}}}
```

grocery\_store.product\_sell DOCUMENTS 21.3k STORAGE SIZE 2.7MB AVG. 3/

Documents Aggregations Schema Explain Plan Indexes Validation

COLLATION rotation\_productsell\_utilidad SAVE SAMP

21

\$addFields Output after \$addFields stage (Sample of 20 documents)

```
1 /**
2  * newField: The new field name.
3  * expression: The new field expression.
4  */
5  {
6    rotation_product: "$rotation_p.Rotation_in_days"
7  }
```

pwa\_sb: 15.96  
pudzb: 0.02  
pmarza: 29.47  
pmarzajedn: 1.12  
pkwmarza: 4.48  
pudzmarza: 0.05  
date\_: 2018-01-01T00:00:00.000+00:00  
rotation\_p: Object  
rotation\_product: 0

\_id: 0  
id\_Da:  
Date:  
PKod:  
Pgrou:  
Pname:  
Pquan:  
pce\_zi

\$project Output after \$project stage (Sample of 20 documents)

```
1 /**
2  * specifications: The fields to
3  * include or exclude.
4  */
5  {
6    _id: 0,
7    Pname: 1,
8    Pgrou: 1,
9    pmarzajedn: 1,
10   rotation_product: 1
11 }
```

Pgroup: "DAIRY\_CHESSE"  
Pname: "ALMETTE CZOSNEK NIEDZWIEDZI 150GR"  
pmarzajedn: 1.12  
rotation\_product: 0

Pgrou:  
Pname:  
pmarz:  
rotat:

#### 4. ¿Cuáles son las 5 categorías más costosas en el flujo de venta? Muestra su costo y monto de venta.

SQL

```
create view sell_detail as
select (str_to_date(day_, "%d/%m/%Y")) AS día,
product_group, product_name, product_quantity, quantity_net_purchase, net_value_sale, percent_margin, quantity_margin_sold_products
from sell;

select product_group,
sum(product_quantity) as cantidad,
sum(quantity_net_purchase) as costo, sum(net_value_sale) as venta
from sell_detail
group by product_group
order by sum(quantity_net_purchase) desc limit 5;
```



0	from sell;
1	
2	• select * from sell_detail;
3	
4	• select product_group, sum(product_quantity) as cantidad, sum(quantity_net_purchase) as costo, sum(net_value_sale) as
5	from sell_detail group by product_group order by sum(quantity_net_purchase) desc limit 5;
6	

product_group	cantidad	costo	venta
CIGARETTES	21192	177826	189456
DAIRY_CHEESE	32852	87374	107895
BREAD	65136	85891	105859
BEER	46440	72839	93332
VEGETABLES	32948	62143	82976

## MongoDB

```
[{$addFields: {
  date_: {
    $dateFromString: {
      dateString: '$Date',
      format: '%d/%m/%Y'
    }
  }
}}, {$addFields: {
  mes: {
    $month: '$date_'
  }
}}, {$group: {
  _id: {
    categoria: '$Pgroup'
  },
  costo: {
    $sum: '$pwa_zn'
  },
  venta: {
    $sum: '$pwa_sn'
  },
  utilidad_venta: {
    $avg: '$pmarza'
  }
}}, {$addFields: {
  categoria: '$_id.categoria'
}}, {$sort: {
  costo: -1
}}, {$limit: 5}, {$project: {
  _id: 0,
  venta: 1,
  costo: 1,
  utilidad_venta: 1,
  categoria: 1
}}]
```

grocery\_store.product\_sell

DOCUMENTS 21.3k STORAGE SIZE 2.7MB AVG. SIZE 325B

Documents Aggregations Schema Explain Plan Indexes Validation

COLLATION costo\_venta\_utilidad SAVE SAMPLE M

1 5

```
{
  "_id": 0,
  "venta": 1,
  "costo": 1,
  "utilidad_venta": 1,
  "categoria": 1
}
```

Output after \$project stage (Sample of 5 documents)

```
costo: 177809.09
venta: 189457.62
utilidad_venta: 9.616571798188874
categoria: "CIGARETTES"
```

## 5. ¿Cuál es el costo, ingreso, utilidad y rotación de cada categoría de productos?

SQL

```
select s.product_group as categoria,
sum(s.quantity_net_purchase) as costo,
sum(s.net_value_sale) as venta,
sum(s.quantity_margin_sold_products) as utilidad, avg(rotation_time) as rotacion
from sell as s
join rotation_products as r
on s.product_code = r.product_code
group by s.product_group;
```

```
62 sum(s.net_value_sale) as venta,
63 sum(s.quantity_margin_sold_products) as utilidad, avg(rotation_time) as rotacion
64 from sell as s
65 join rotation_products as r
66 on s.product_code = r.product_code
67 group by s.product_group;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: I A

categoria	costo	venta	utilidad	rotacion
ART/_HYGIENIC	157	244	78	8.4545
GROATS_RICE_PASTA	612	889	272	6.7564
CHEMISTRY	3676	5564	1859	6.2102
COOKIES_BULK	297	450	153	3.7222
CHIPS_FLAKES	571	874	302	27.1731
DISHES_FOR_CHILDREN	90	120	30	14.0000
ARTICLE_OF_HOUSEHOLD	6	11	3	19.0000
PET'S FOOD	20	29	11	2.0000

```

[{$addFields: {
  date_: {
    $dateFromString: {
      dateString: '$Date',
      format: '%d/%m/%Y'
    }
  }
}}, {$lookup: {
  from: 'rotation_products',
  localField: 'PKod',
  foreignField: 'PKod',
  as: 'rotation_p'
}}, {$addFields: {
  rotation_p: {
    $reduce: {
      input: '$rotation_p',
      initialValue: {
        $arrayElemAt: [
          '$rotation_p',
          0
        ]
      },
      'in': {
        $cond: {
          'if': {
            $gt: [
              '$$this.Pgroup',
              '$$value.Pgroup'
            ]
          },
          then: '$$this',
          else: '$$value'
        }
      }
    }
  }
}}, {$addFields: {
  rotation_product: '$rotation_p.Rotation_in_days'
}}, {$group: {
  _id: '$Pgroup',
  costo: {
    $sum: '$pwa_zn'
  },
  venta: {
    $sum: '$pwa_sn'
  },
  utilidad_venta: {
    $sum: '$pkwmarza'
  },
  'rotación_promedio': {
    $avg: '$rotation_product'
  }
}}]

```

grocery\_store.product\_sell DOCUMENTS 21.3k STORAGE SIZE 2.7MB AVG. SIZE 325B INDEXES 1 TOTAL 274.4

Documents Aggregations Schema Explain Plan Indexes Validation

COLLATION costo\_venta\_utilidad\_rotación SAVE SAMPLE MODE AUTO PRE

Output after \$group stage (Sample of 20 documents)

```

1 /**
2  * _id: The id of the group.
3  * fieldN: The first field name.
4  */
5 {
6   _id: "$Pgroup",
7   costo: {
8     $sum: "$pwa_zn"
9   },
10  venta: {
11    $sum: "$pwa_sn"
12  },
13  utilidad_venta: {
14    $sum: "$pkwmarza"
15  },
16  rotación_promedio: {
17    $avg: "$rotation_product"
18  }
19 }

```

Sample output documents:

```

_id: "EGGS"
costo: 4584.82
venta: 5638.5599999999995
utilidad_venta: 1053.74
rotación_promedio: 0

_id: "PACKAGES"
costo: 202.26
venta: 219.10995
utilidad_venta:
rotación_promedio:

```

## Vistas

### Producto\_categoria\_unidadesvendidas

#### SQL

Navigator SCHEMAS Filter objects

grocery\_store

- Tables
  - day\_sell
  - rotation\_products
  - sell
- Views
  - venta\_productos
    - product\_name
    - product\_group
    - sum(product\_quantity)

Administration Schemas Information

View: venta\_productos

Columns:

Column	Type
product_name	varchar
product_group	varchar
sum(product_quantity)	decimal

Create\_database

```

64 • select product_name from sell where product_name like "Bambino%";
65 • select product_code from sell;
66 • select product_code,
67   count(product_name) as total_products,
68   sum(product_quantity) as products_quantity,
69   (sum(product_quantity) / count(product_name)) as avg_products
70   from sell group by product_code;
71
72 • select product_name from rotation_products where product_name like "%CORN%";
73
74 • select product_name, product_group, sum(product_quantity) from sell group by product_name;
75
76 • create view venta_productos as
77   select product_name, product_group, sum(product_quantity) from sell group by product_name;
78
79

```

#### MongoDB

## grocery\_store.producto\_categoria\_unidades

READ-ONLY

VIEW

Documents

Aggregations

Schema

Explain Plan

Indexes

Validation

FILTER { field: 'value' }

VIEW

producto\_categoria\_unidades

	producto String	Unidades_vendidas Int32	Categoria String
1	"WEDEL CZEKOLADA NADZ TOFFI 100	5	"SWEETS"
2	"ZNICZ DUZY"	2	"GENERAL"
3	"BREF KULKI WC 50 G"	2	"CHEMISTRY"
4	"MUSZTARDA MIODOWA 270 G"	8	"DRINK_JUICE"
5	"PAP LUCKY STRIKE 22 RED"	69	"CIGARETTES"
6	"SOS BOLONSKI SLOIK DAWT 550G"	6	"GENERAL_FOOD"
7	"ZUREK BUTELKA 0_5 L"	7	"GENERAL"
8	"BELLA CHUSTECZKI ODSWIEZAJACE	7	"CHEMISTRY"

## Utilidad\_rotacion

### SQL

Navigator

SCHEMAS

Filter objects

- Tables
  - day\_sell
  - rotation\_products
  - sell
- Views
  - utilidad\_rotacion
  - venta\_productos
- Stored Procedures
- Functions
- hollywood\_market
- proyecto\_zapateria

Administration Schemas

Information

Table: rotation\_products

Columns:

product_code	int PK
product_group	varchar(60)
product name	varchar(80)

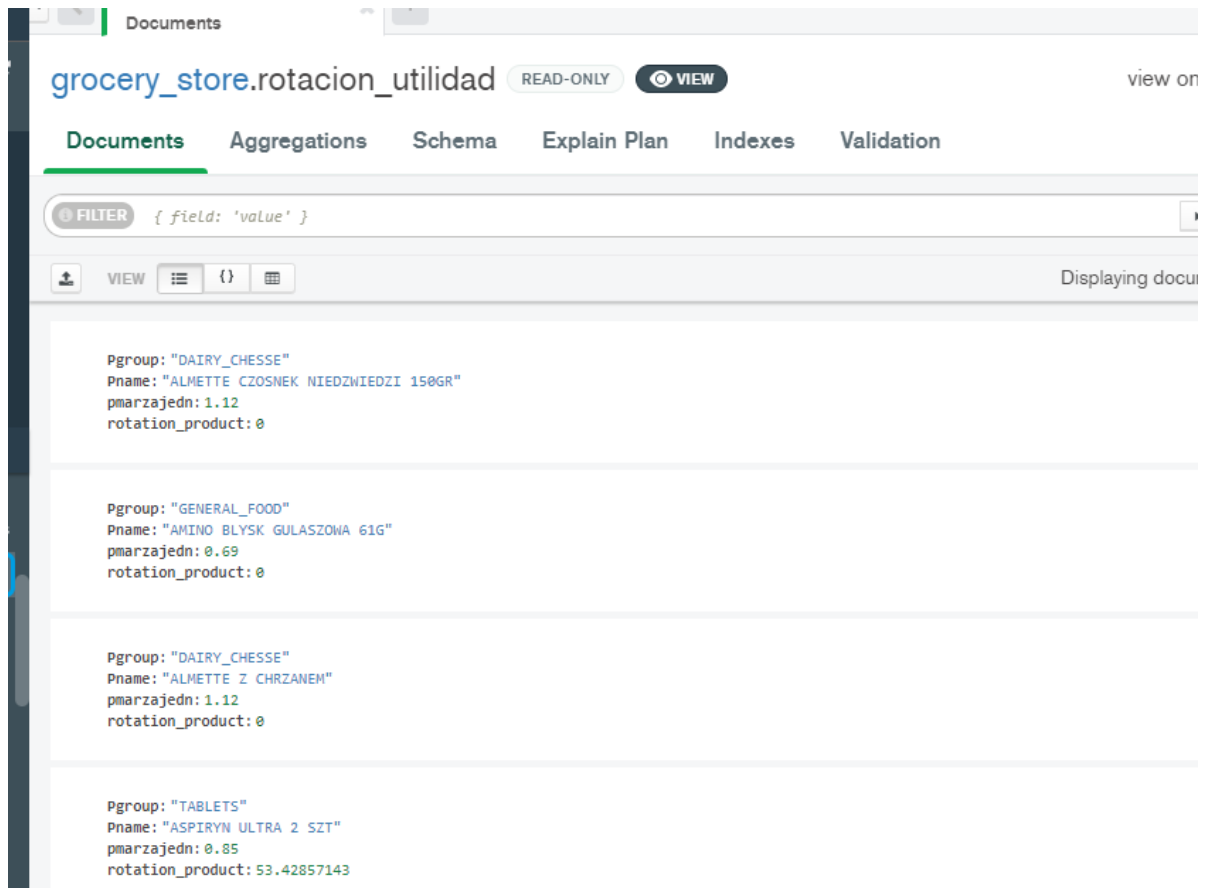
Create\_database\*

Limit to 1000 rows

```

80
81 • select day_ from day_sell;
82
83 • select month(str_to_date(day_, "%d/%m/%Y")) AS MES,
84 product_group, (sum(net_value_sale)) as venta_total,
85 (sum(IFNULL(quantity_margin_sold_products,0))) as utilidad_neta,
86 (sum(ifnull(net_value_sale,0)))/(sum(IFNULL(quantity_margin_sold_products,0))) as margen_utilidad
87 from sell
88 group by product_group order by utilidad_neta ASC limit 5;
89
90 • create view utilidad_rotacion as
91 select s.product_name, s.product_group, quantity_margin_sold_products, rotation_time
92 from sell as s
93 join rotation_products as r
94 on s.product_code = r.product_code;
95
96
  
```

### MongoDB



## Conclusiones

1. Se sugiere eliminar las categorías de:
  - a. Leche
  - b. Artículos para la casa
  - c. Paquetes
  - d. Lunch
  - e. Comida para animalesa través de promociones que les dé salida al inventario
2. Durante el mes de agosto es conveniente realizar promociones para aumentar el flujo de venta.
3. Es importante mantener los índices de venta de los productos que más compran los clientes, que representan mayor utilidad y tienen alta rotación en nuestro anaquel. Dichas categorías son:
  - a. Cigarros
  - b. Queso
  - c. Pan
  - d. Cerveza
  - e. Vegetales

Siendo los vegetales, cerveza, queso y pan (en ese orden) los que mayor utilidad proporcionan; por lo que se puede considerar adaptar el modelo de negocio y la distribución de la tienda para incrementar el flujo de la venta en esos productos.