Luis F Castro

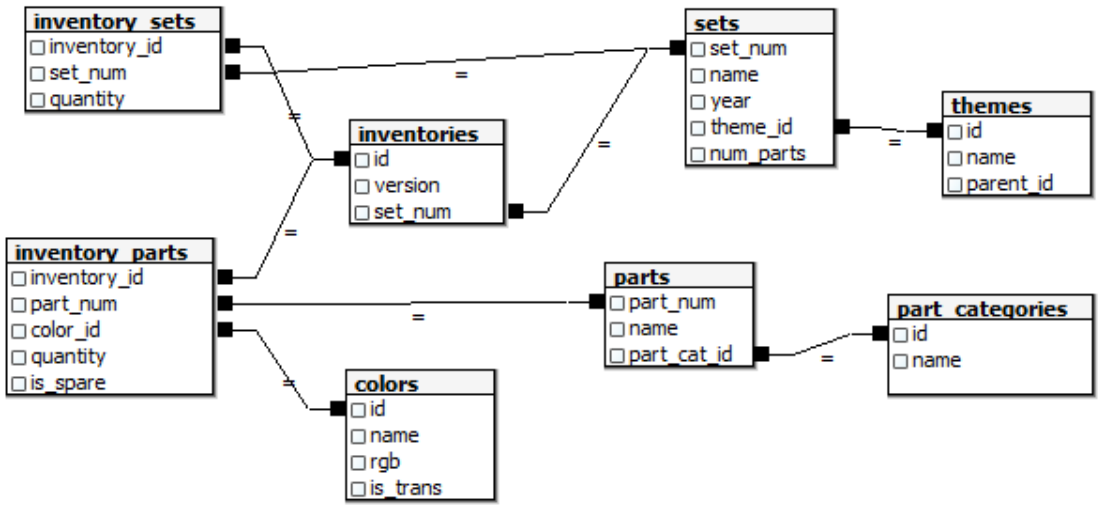
Proyecto “LEGO INC”

Objetivo:

De una base de datos proporcionada por la compañía LEGO se desea saber el estatus de la misma en cuanto a tema de inventarios, además de hacer un análisis de datos, se debe proporcionar una posible causa del problema derivado de dicho análisis, así como una solución.

Bases de Datos:

La información está separada en 8 tablas organizadas bajo el esquema que se muestra a continuación.



A continuación, se muestra el contenido de cada campo.

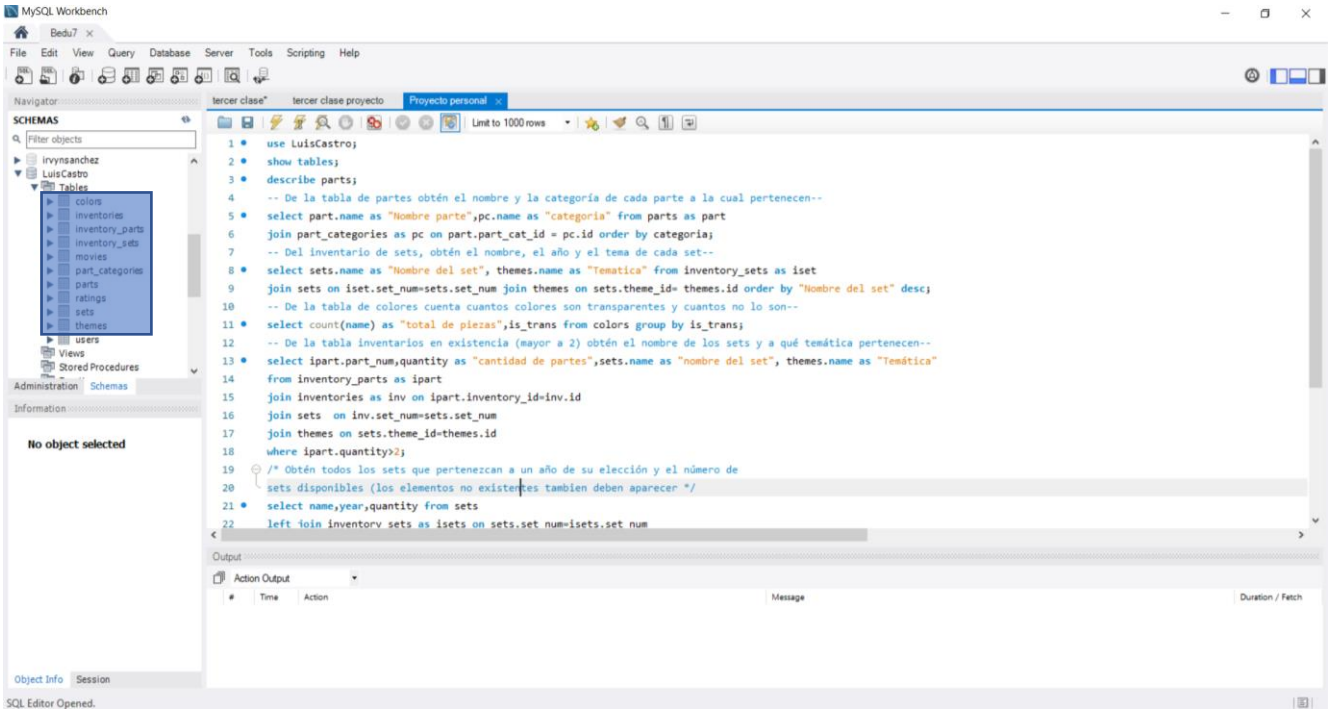
- COLORS
  - Id
    - Unique ID for this color.
  - Name
    - The human-readable name of the color.
  - Rgb
    - The approximate RGB color.
  - is_trans
    - Whether or not the given color is transparent/translucent.
- INVENTORIES
  - Id
    - Unique ID for this inventory entry.
  - Versión
    - Version number.
  - set_num
    - Set number (form `sets.csv`).
- INVENTORY_PARTS

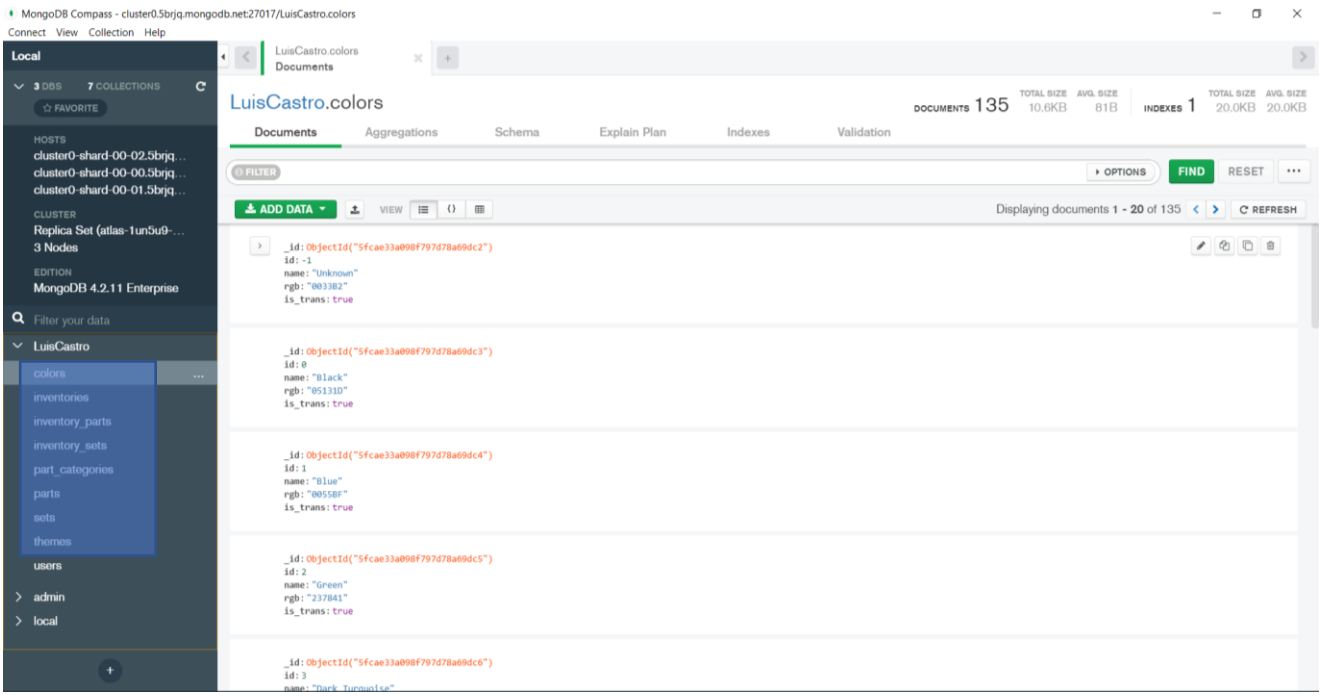
    - inventory_id
        - Unique ID for the inventory this part is appearing in. This is the same as the id value in `inventories.csv`.
    - part_num
        - Unique ID for the part.
    - color_id
        - Unique ID for the color, as per `colors.csv`.
    - Quantity
        - The number of copies of this part included in the set!
    - is_spare
        - Whether or not this is a spare part. Spare parts are additional parts not needed to finish the set.
- INVENTORY_SETS
    - inventory_id
        - Unique inventory ID from `inventories.csv`.
    - set_num
        - Unique set ID from `sets.csv`.
    - quantity
        - The quantity of the inventory included.
- PART_CATEGORIES
    - id
        - Unique ID for the part category.
    - name
        - The category of stuff the part is in.
- Parts
    - part_num
        - Unique ID for the part.
    - name
        - Name of the part
    - part_cat_id
        - Part category unique ID (from `part_categories.csv`).
- Sets
    - Set_num
        - Unique set ID.
    - Name
        - The name of the set.
    - Year
        - Year the set was published.
    - Theme_id
        - Unique ID for the theme used for the set (from `themes.csv`).
    - Num_parts
        - The number of parts included in the set.
- Themes
    - Id
        - Theme unique ID.
    - Name
        - Name of the theme.
    - Parent_id
        - Unique ID for the larger theme, if there is one.

Luis F Castro

Creación de Bases de datos en MYSQL y MongoDB

Por motivos de rendimiento y tiempo la base de datos que fue ingresada a MYSQL fue reducida en cantidad de datos, esto puede llegar a causar discrepancias en las consultas.

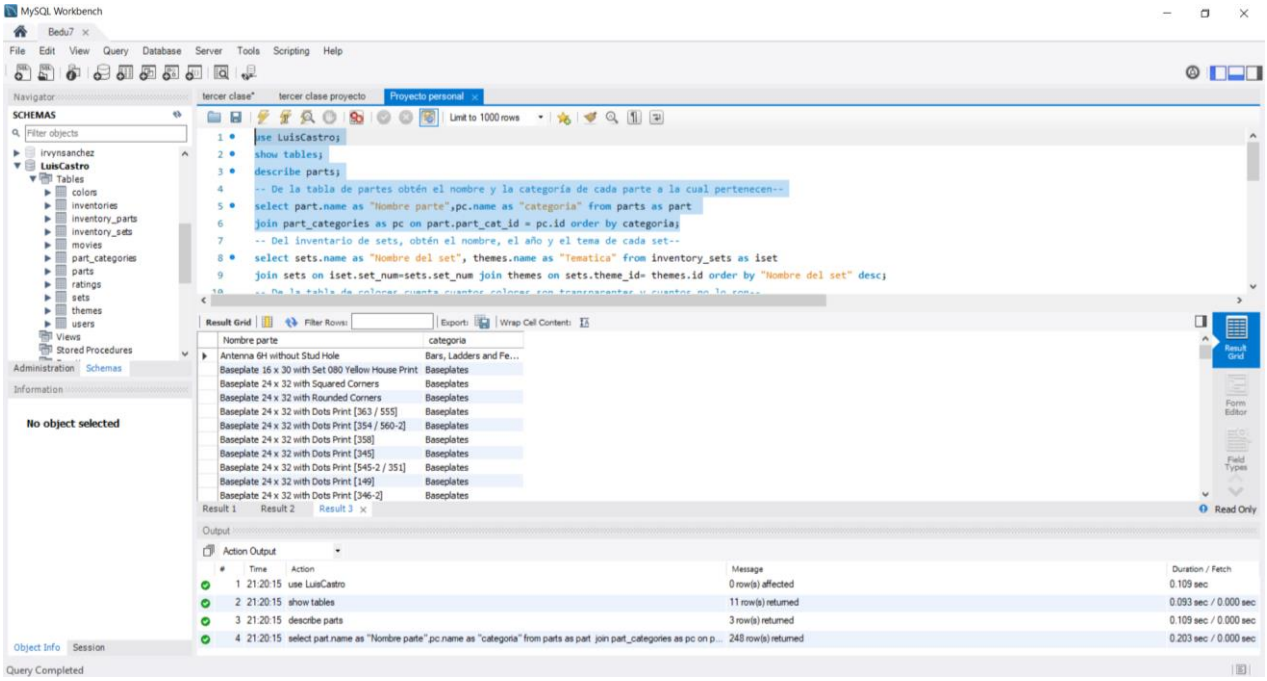


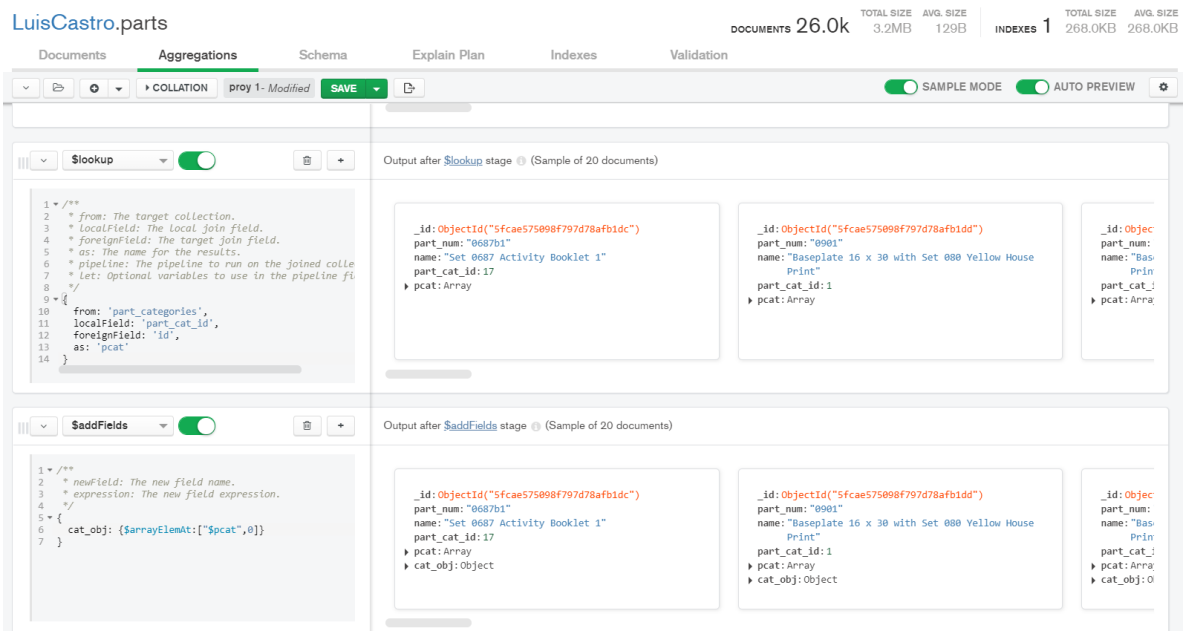


Luis F Castro

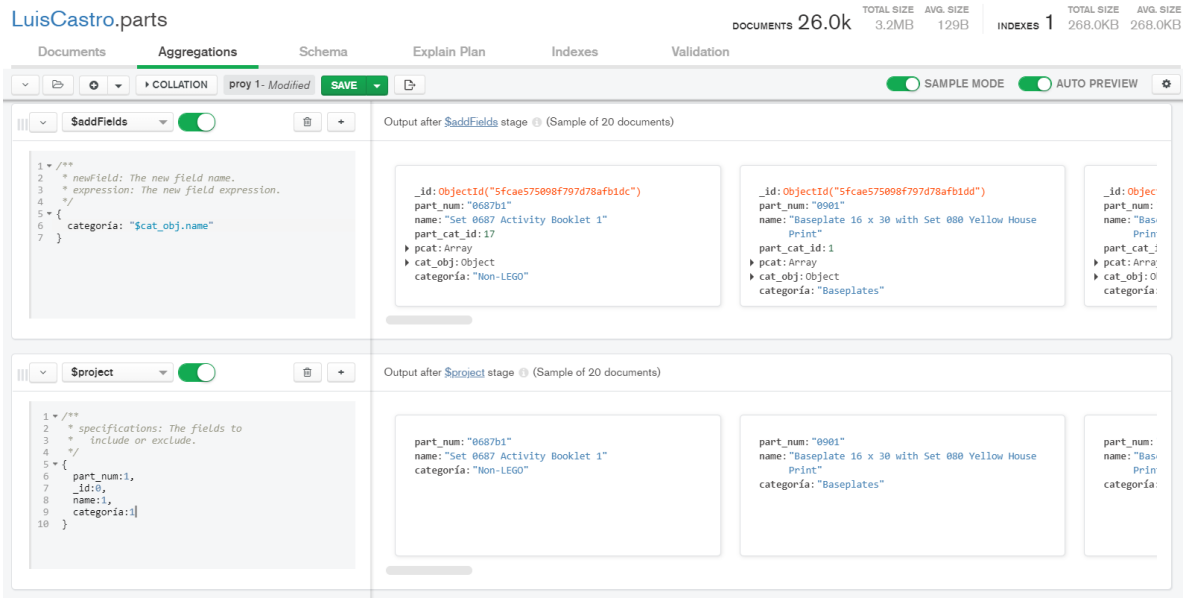
Consultas

1.-De la tabla de partes obtén el nombre y la categoría de cada parte a la cual pertenecen.

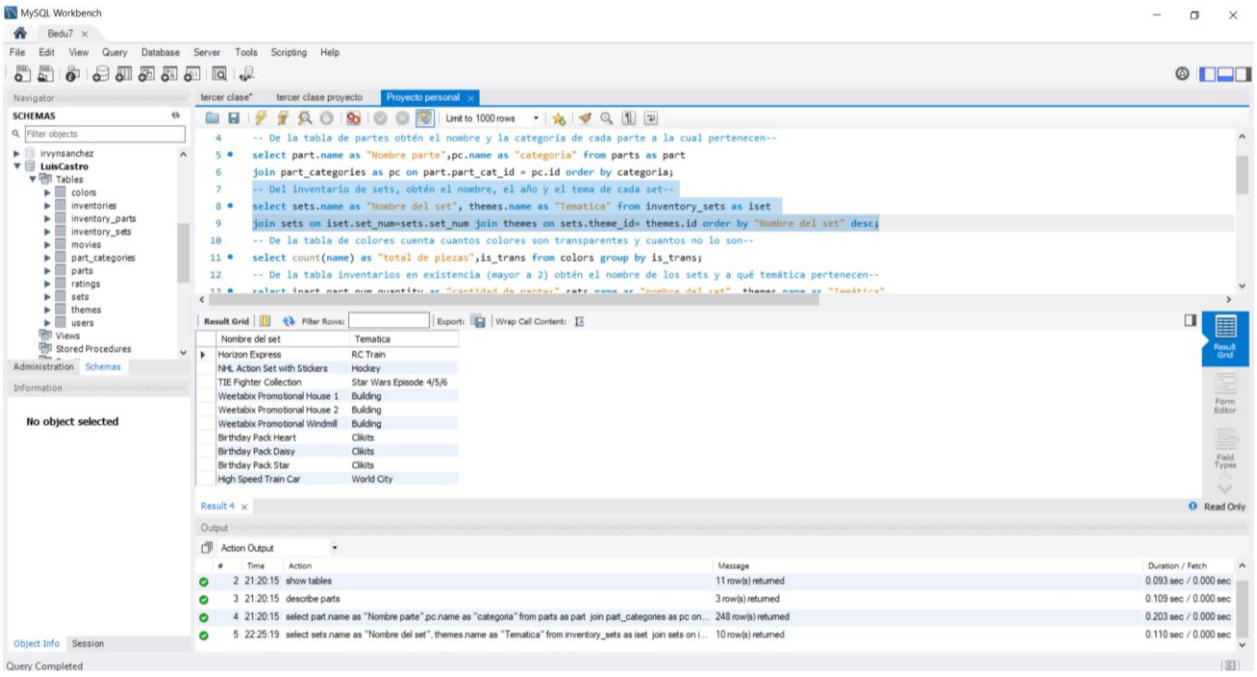




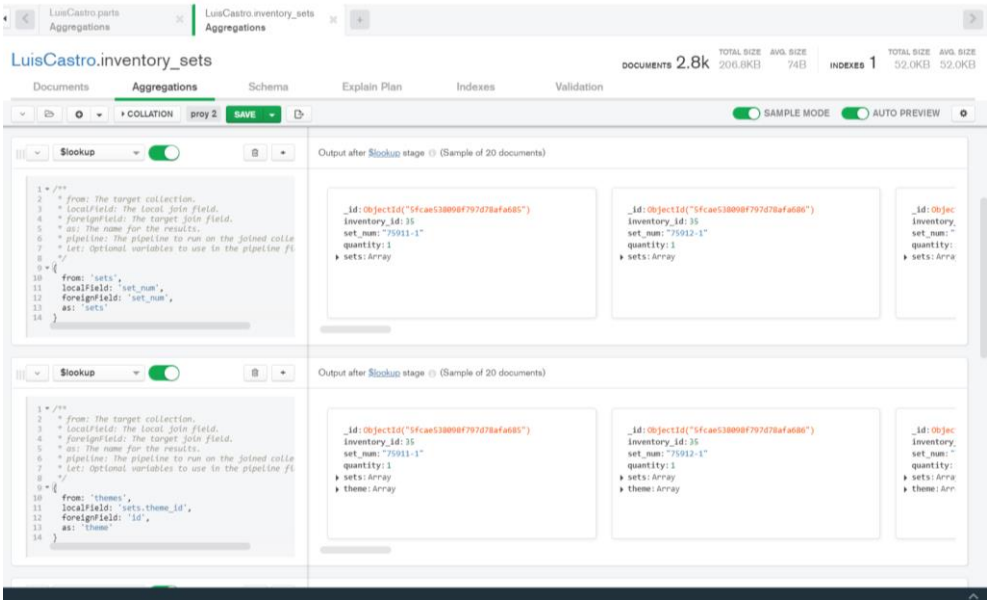
Luis F Castro

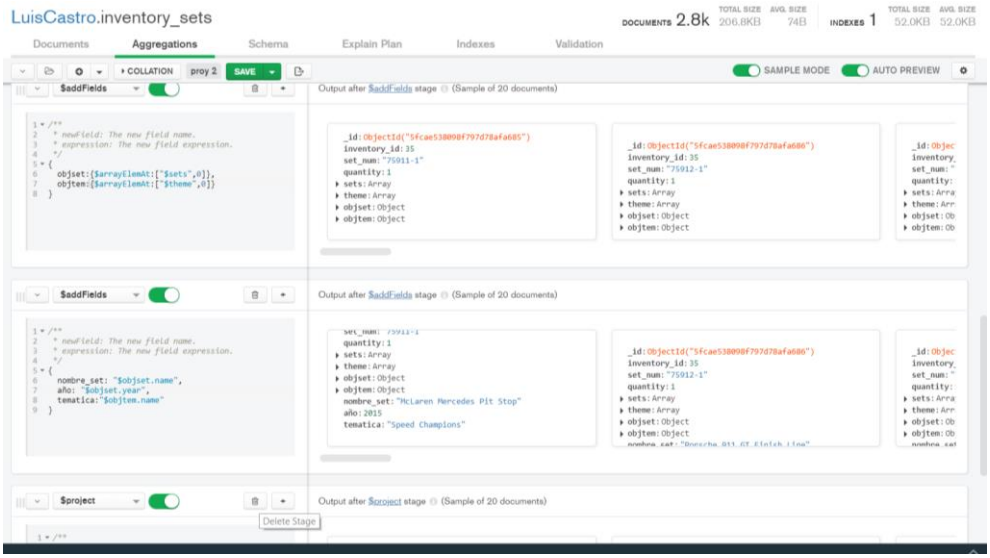


2.-Del inventario de sets, obtén el nombre, el año y el tema de cada set.

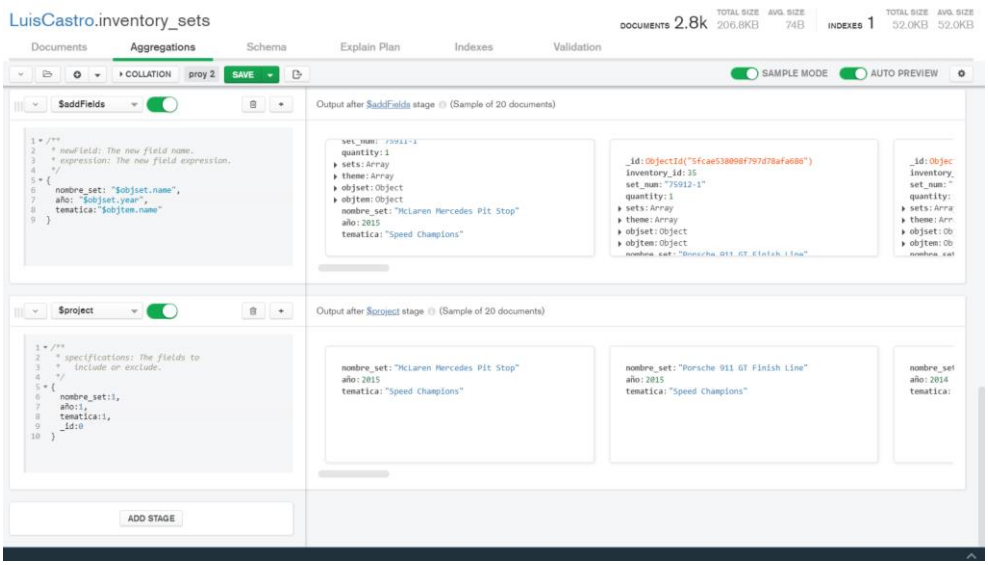


Luis F Castro

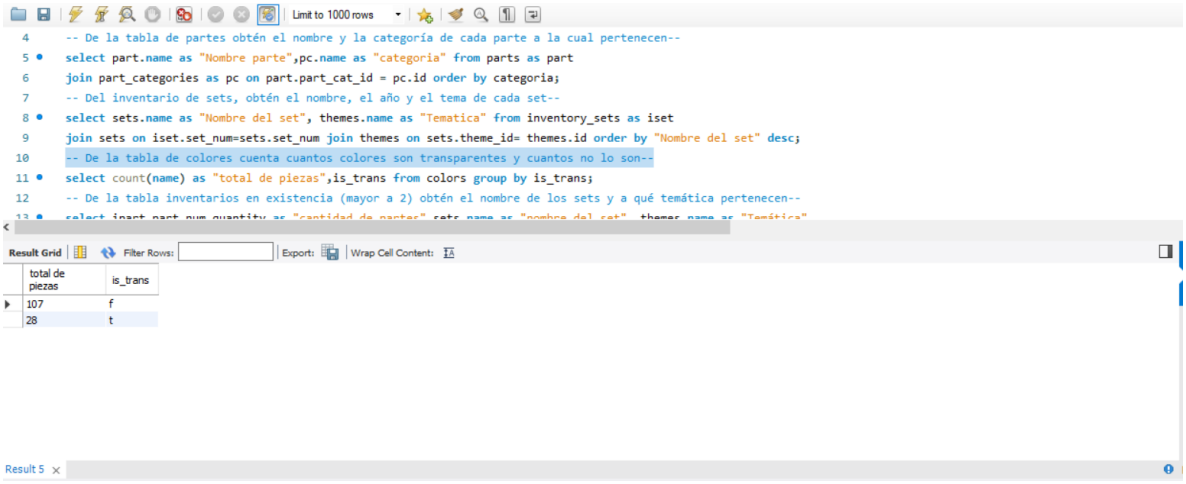




Luis F Castro



3.- De la tabla de colores cuenta cuantos colores son transparentes y cuantos no lo son.



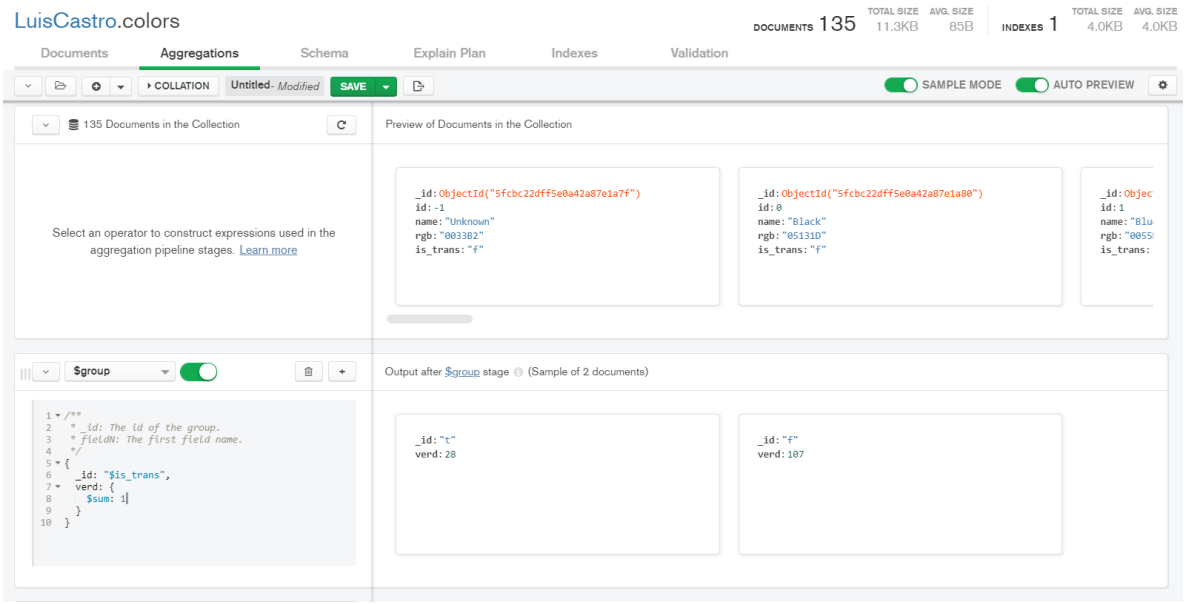
```
-- De la tabla de partes obtén el nombre y la categoría de cada parte a la cual pertenecen--
select part.name as "Nombre parte",pc.name as "categoria" from parts as part
join part_categories as pc on part.part_cat_id = pc.id order by categoria;
-- Del inventario de sets, obtén el nombre, el año y el tema de cada set--
select sets.name as "Nombre del set", themes.name as "Tematica" from inventory_sets as iset
join sets on iset.set_num=sets.set_num join themes on sets.theme_id= themes.id order by "Nombre del set" desc;
-- De la tabla de colores cuenta cuantos colores son transparentes y cuantos no lo son--
select count(name) as "total de piezas",is_trans from colors group by is_trans;
-- De la tabla inventarios en existencia (mayor a 2) obtén el nombre de los sets y a qué temática pertenecen--
```

Result Grid

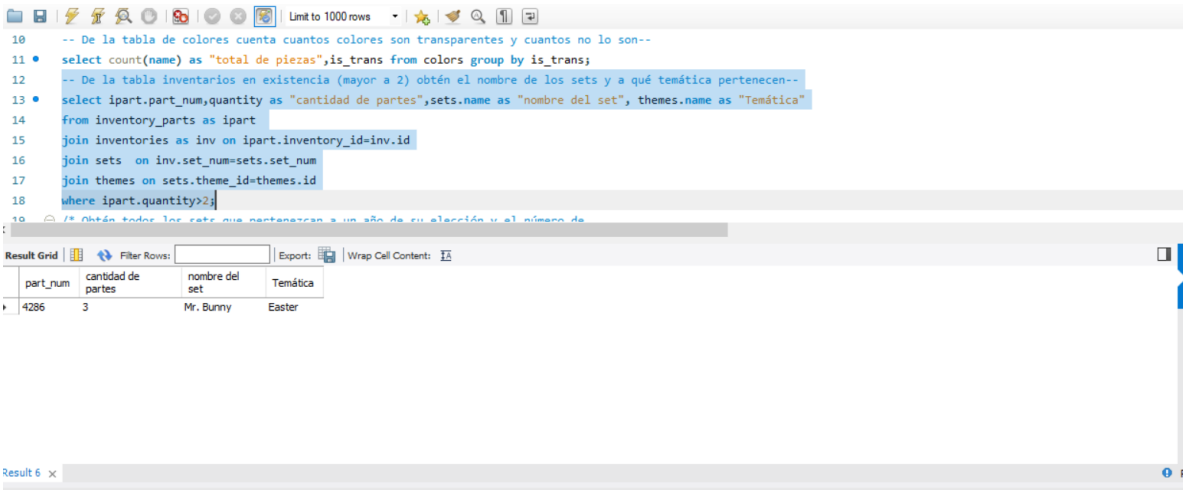
total de piezas	is_trans
107	f
28	t

Result 5

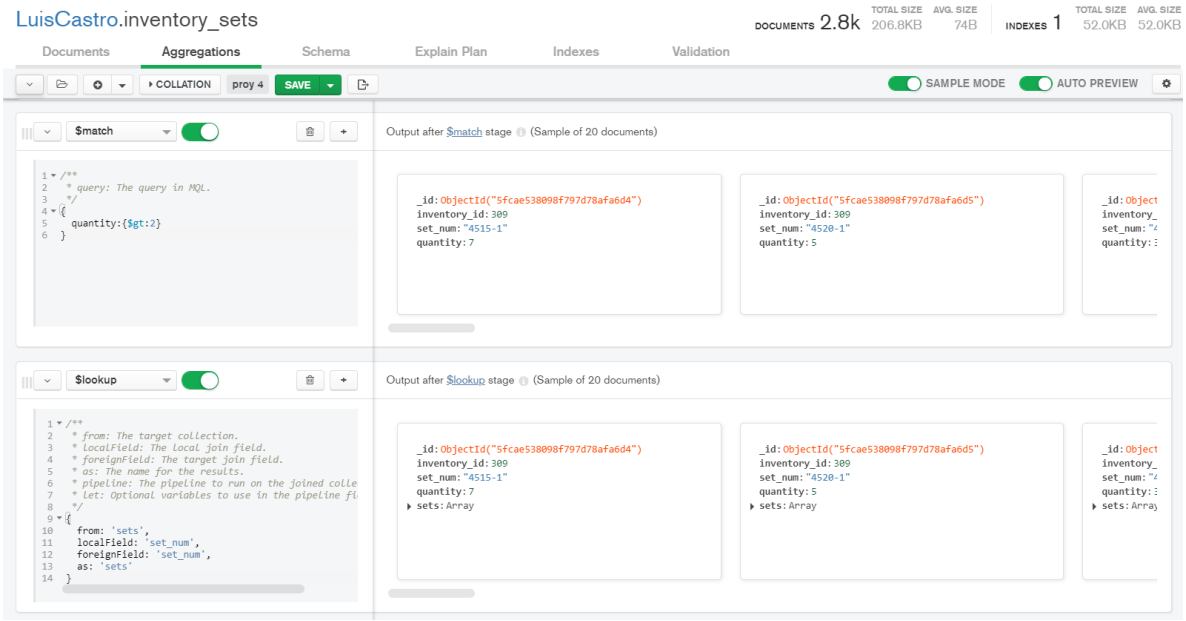
Luis F Castro

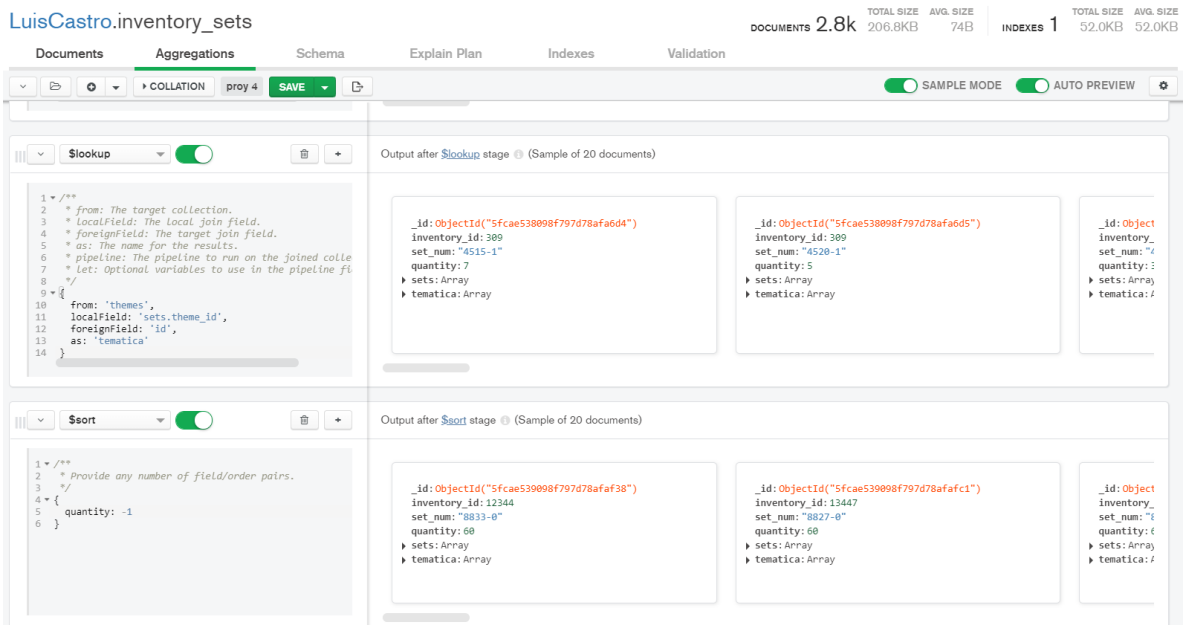


4.- De la tabla inventarios en existencia (mayor a 2) obtén el nombre de los sets y a qué temática pertenecen.

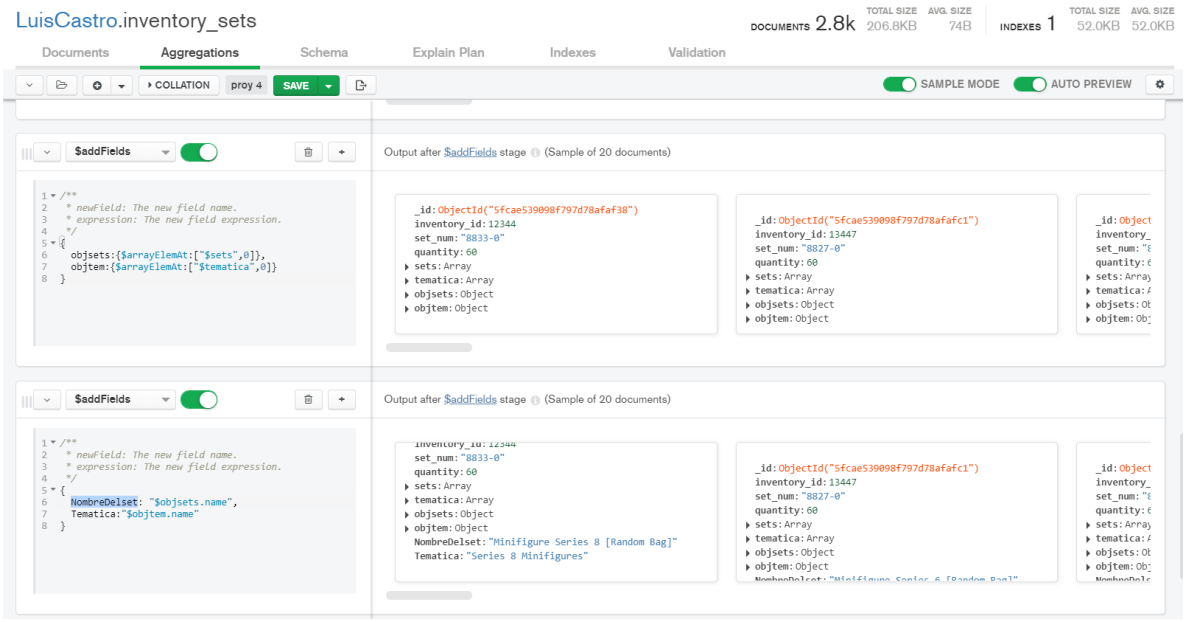


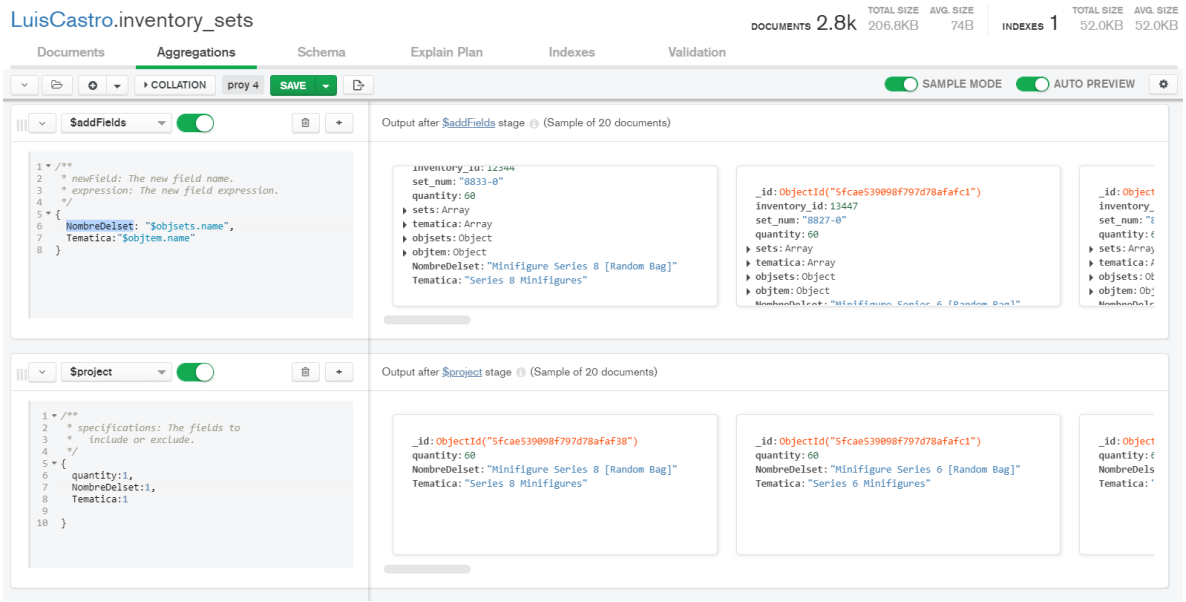
Luis F Castro





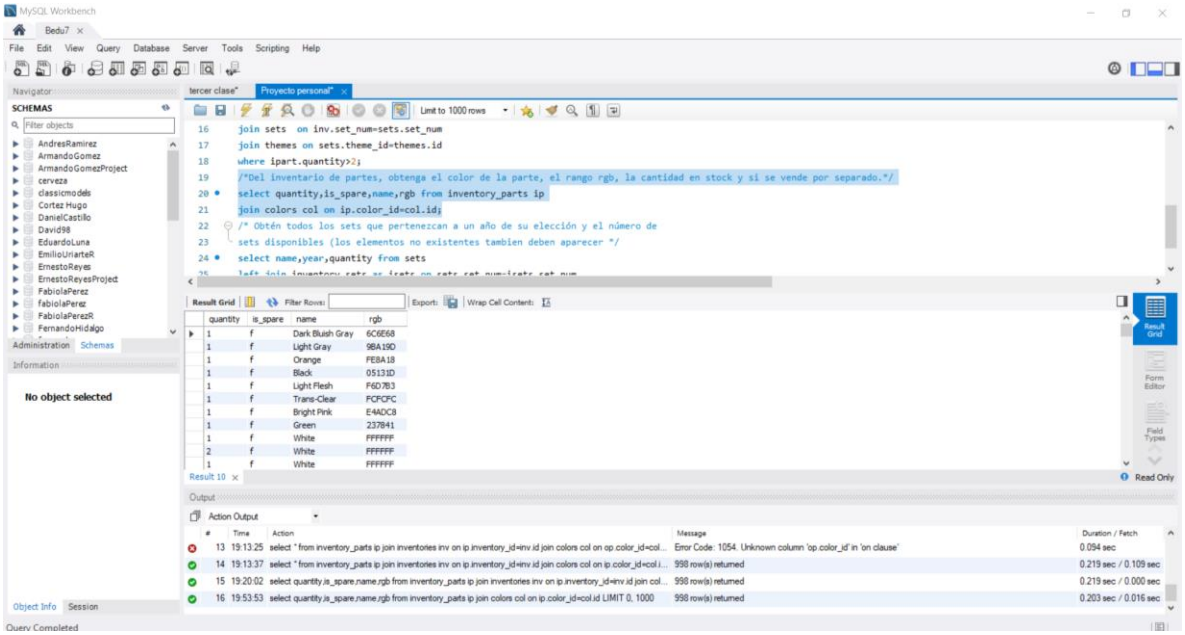
Luis F Castro

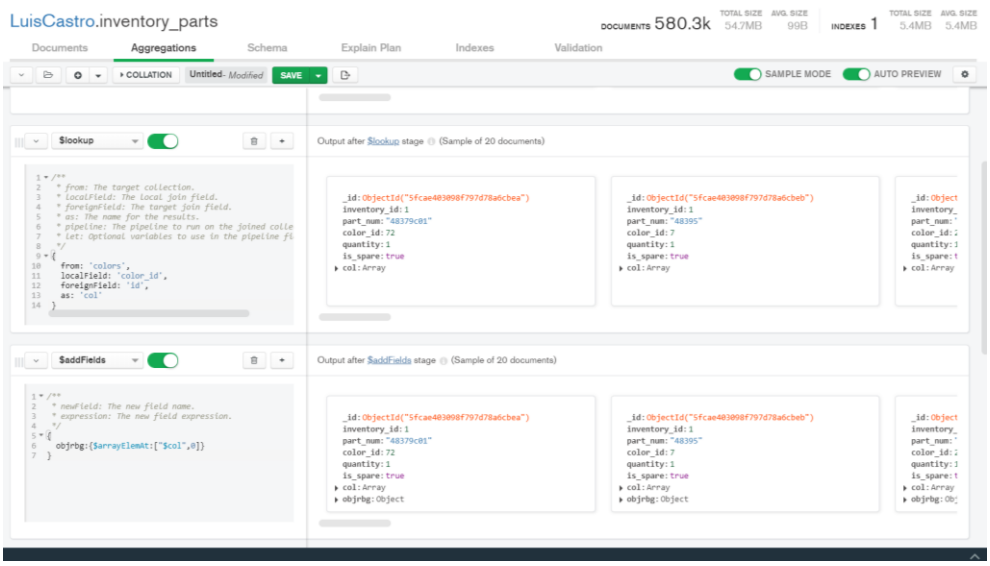




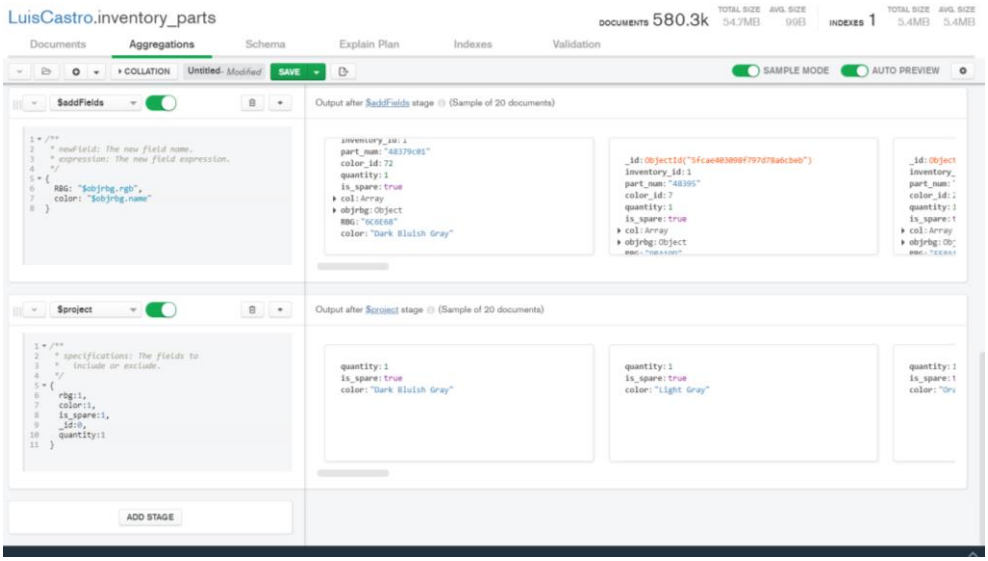
5.-Del inventario de partes, obtenga el color de la parte, el rango rgb, la cantidad en stock y si se vende por separado.

Luis F Castro

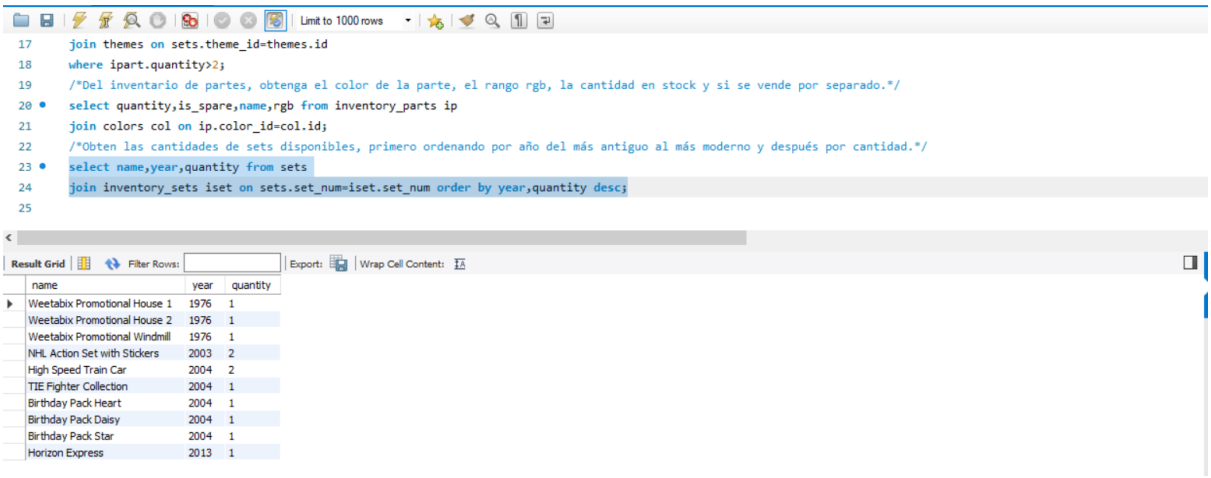




Luis F Castro



6.-Obten las cantidades de sets disponibles, primero ordenando por año del más antiguo al más moderno y después por cantidad.

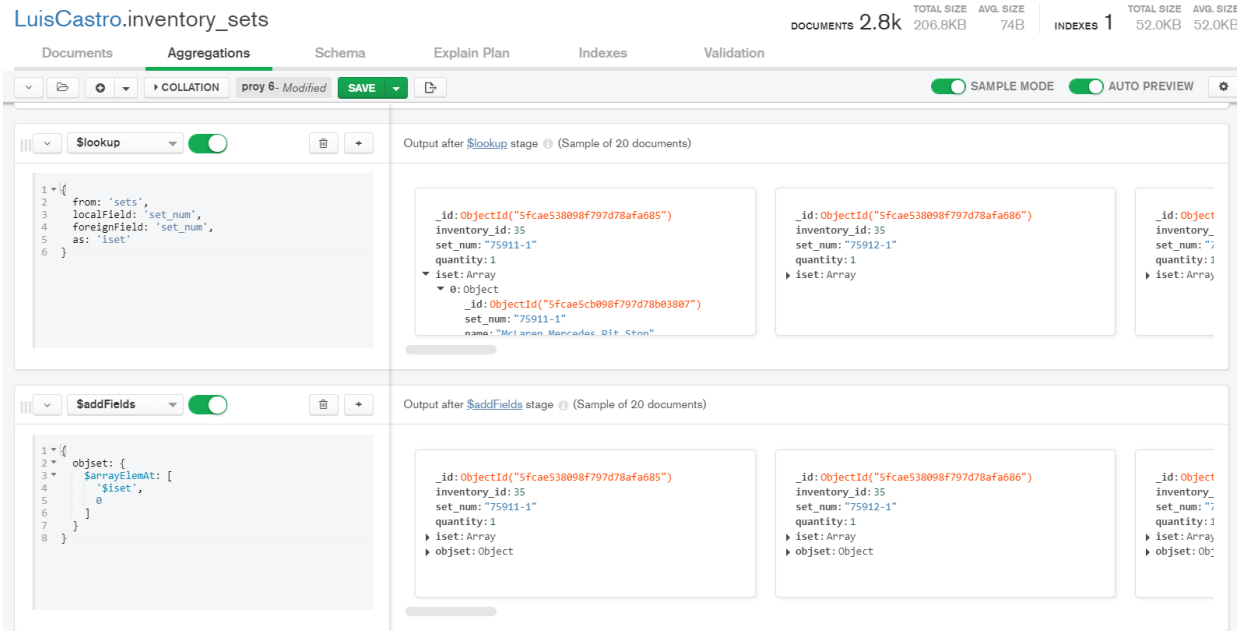


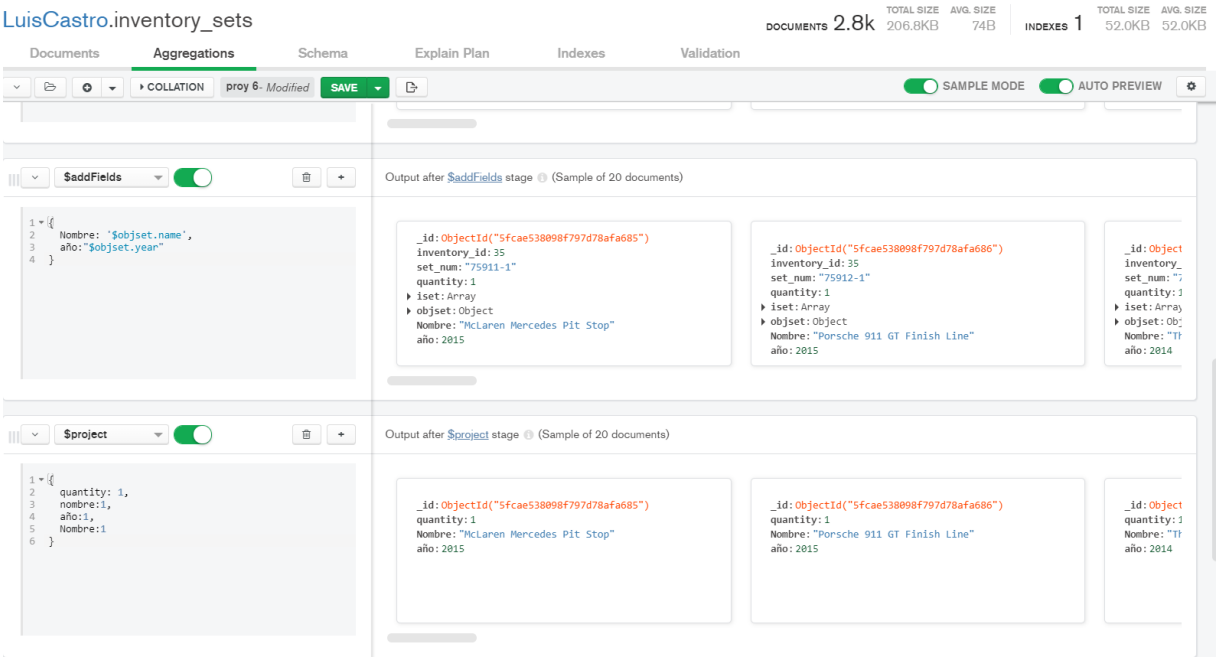
```
17    join themes on sets.theme_id=themes.id
18    where ipart.quantity>2;
19    /*Del inventario de partes, obtenga el color de la parte, el rango rgb, la cantidad en stock y si se vende por separado.*/
20    select quantity,is_spare,name,rgb from inventory_parts ip
21    join colors col on ip.color_id=col.id;
22    /*Obten las cantidades de sets disponibles, primero ordenando por año del más antiguo al más moderno y después por cantidad.*/
23    select name,year,quantity from sets
24    join inventory_sets iset on sets.set_num=iset.set_num order by year,quantity desc;
25
```

Result Grid

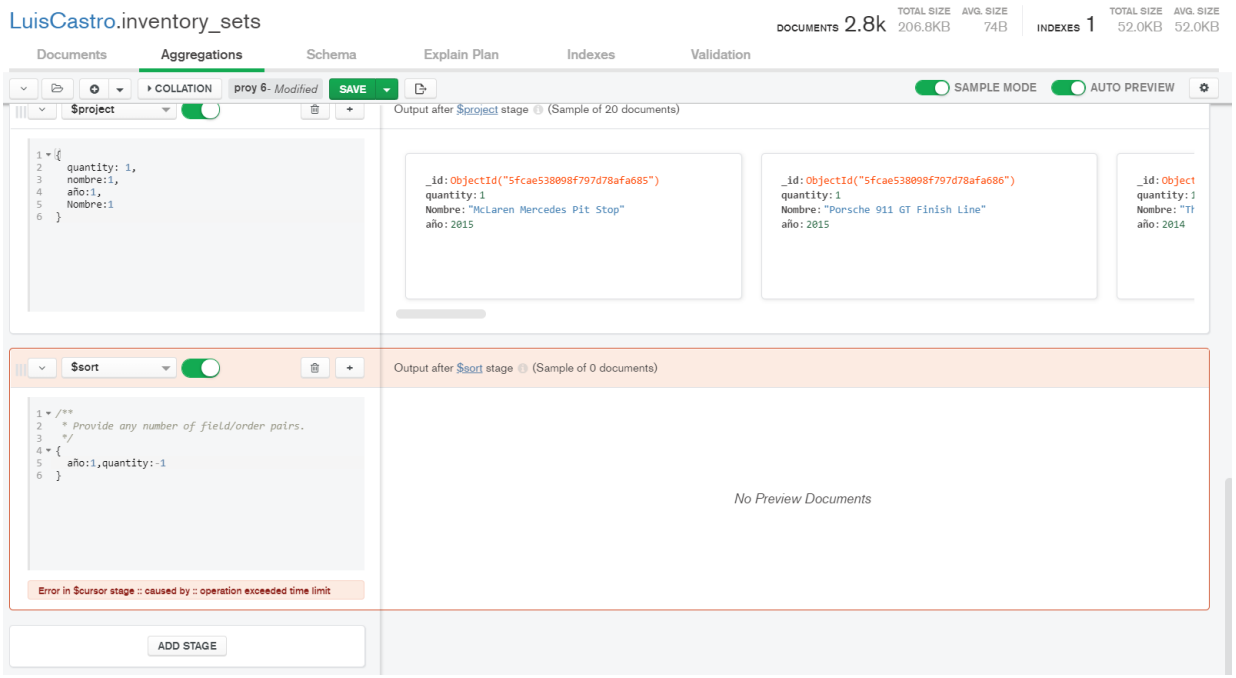
name	year	quantity
Weetabix Promotional House 1	1976	1
Weetabix Promotional House 2	1976	1
Weetabix Promotional Windmill	1976	1
NHL Action Set with Stickers	2003	2
High Speed Train Car	2004	2
TIE Fighter Collection	2004	1
Birthday Pack Heart	2004	1
Birthday Pack Daisy	2004	1
Birthday Pack Star	2004	1
Horizon Express	2013	1

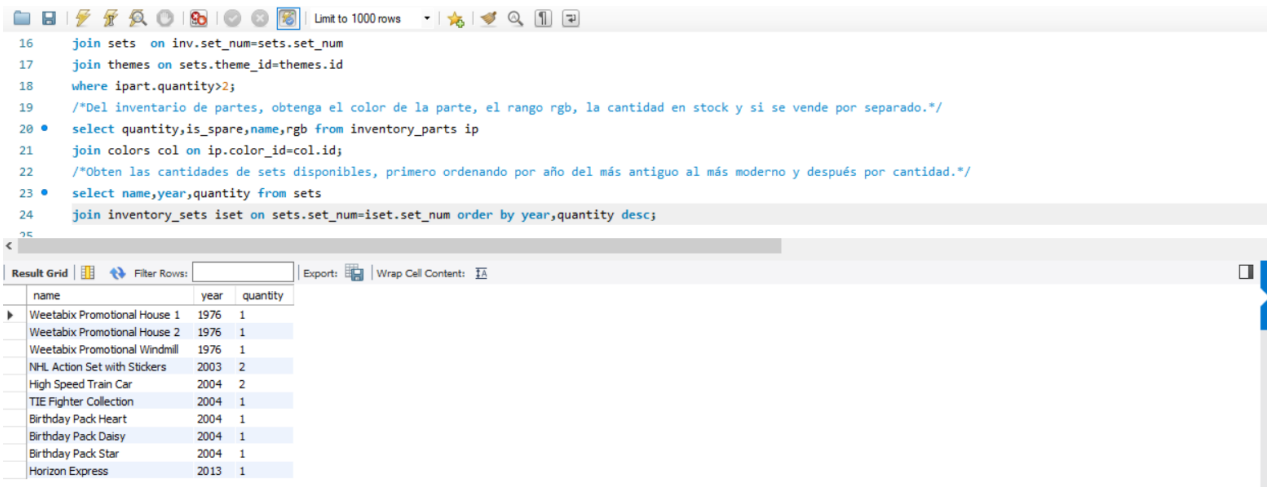
Luis F Castro



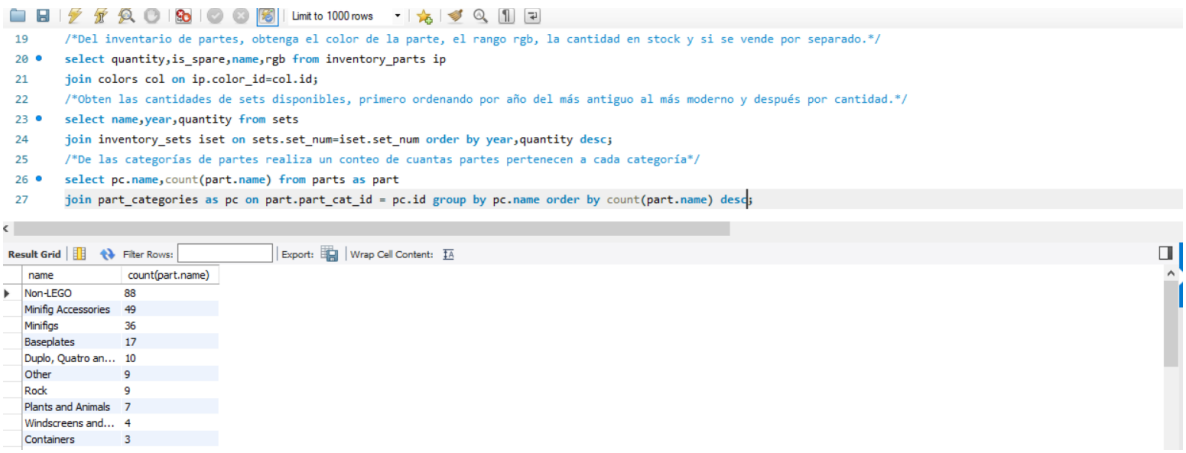


Luis F Castro

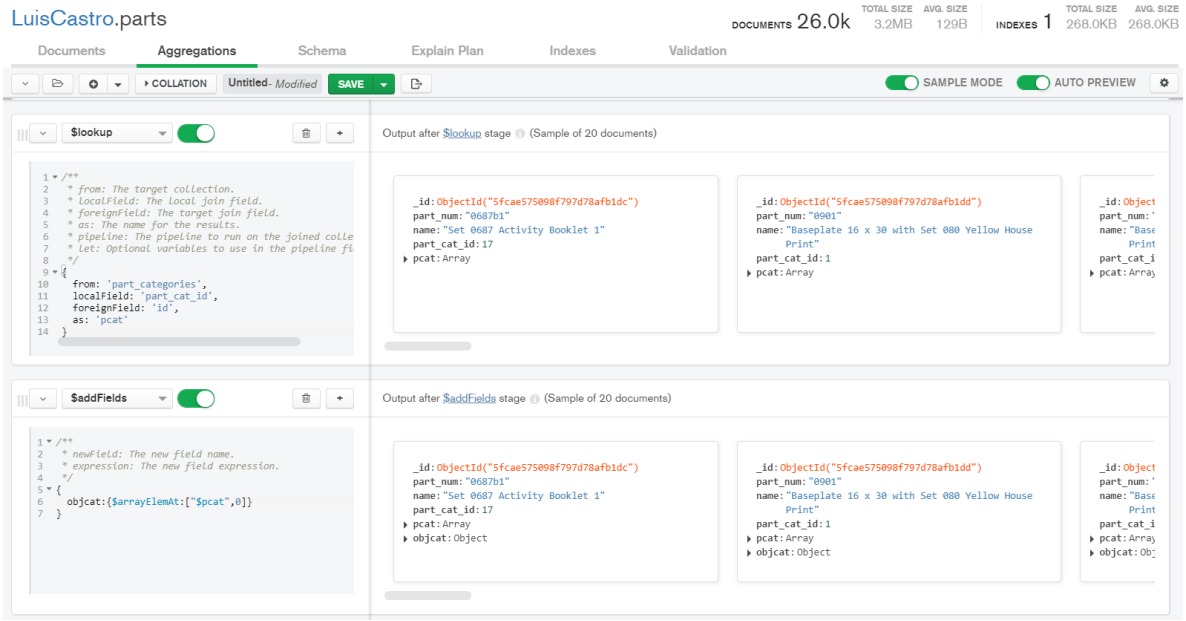


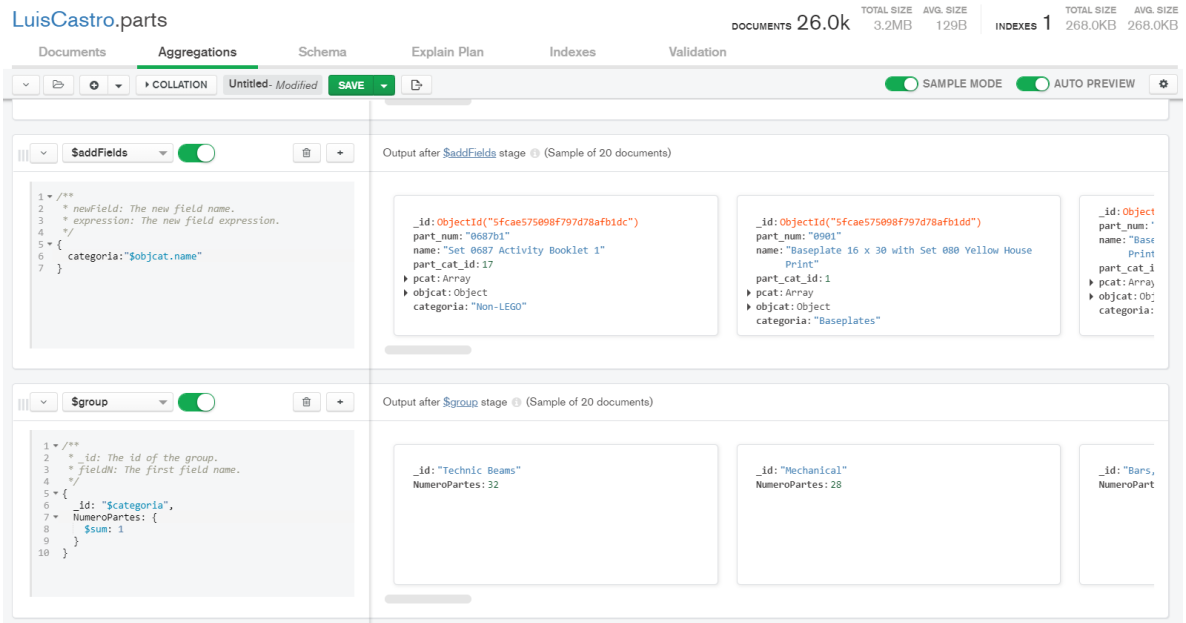


7.-Consulta cuantas partes hay por categoría y ordénalas de mayor a menor.

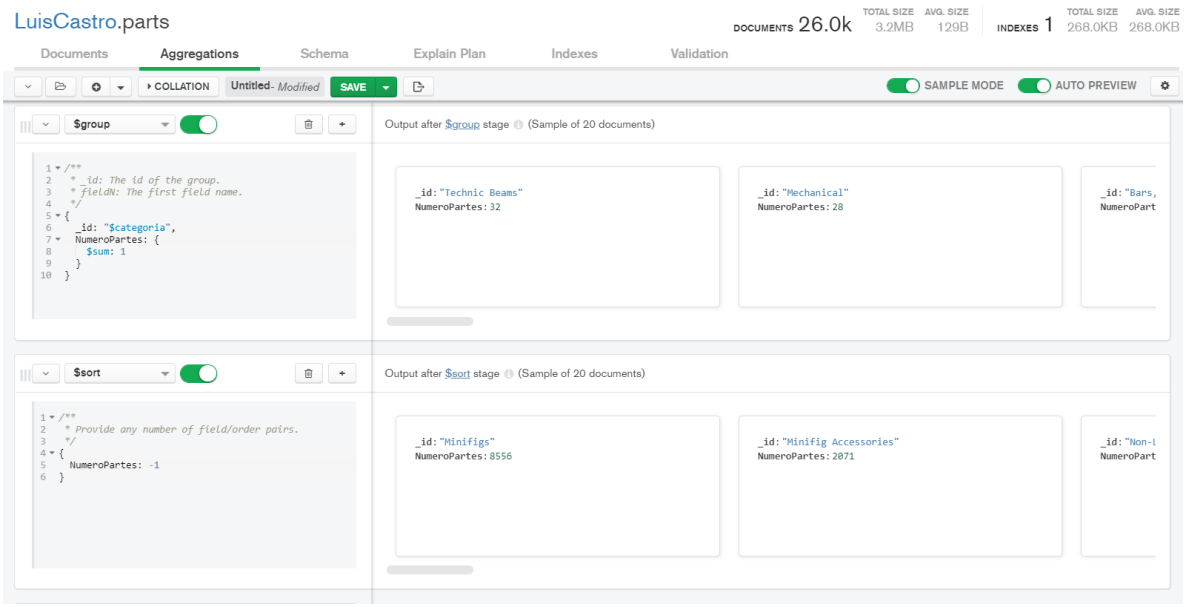


Luis F Castro

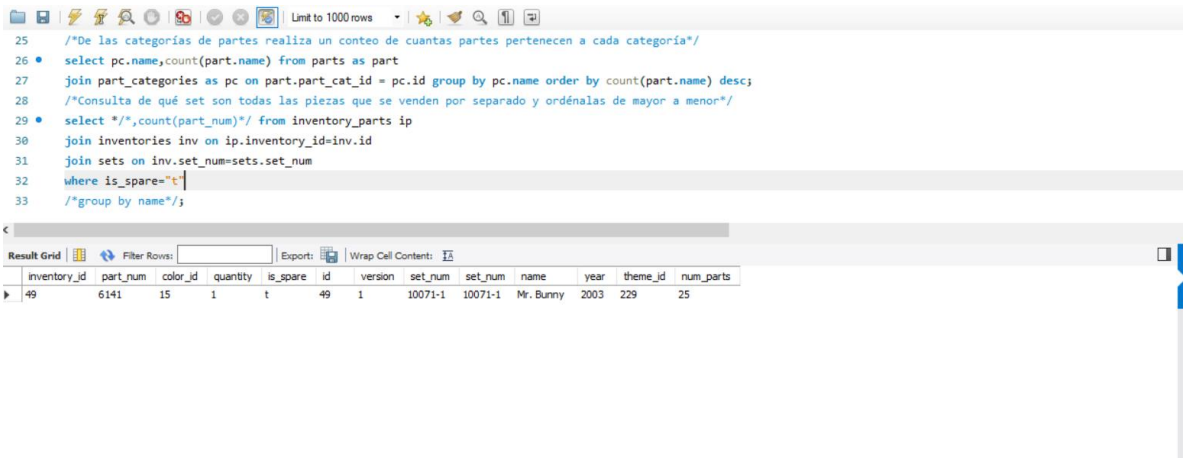




Luis F Castro

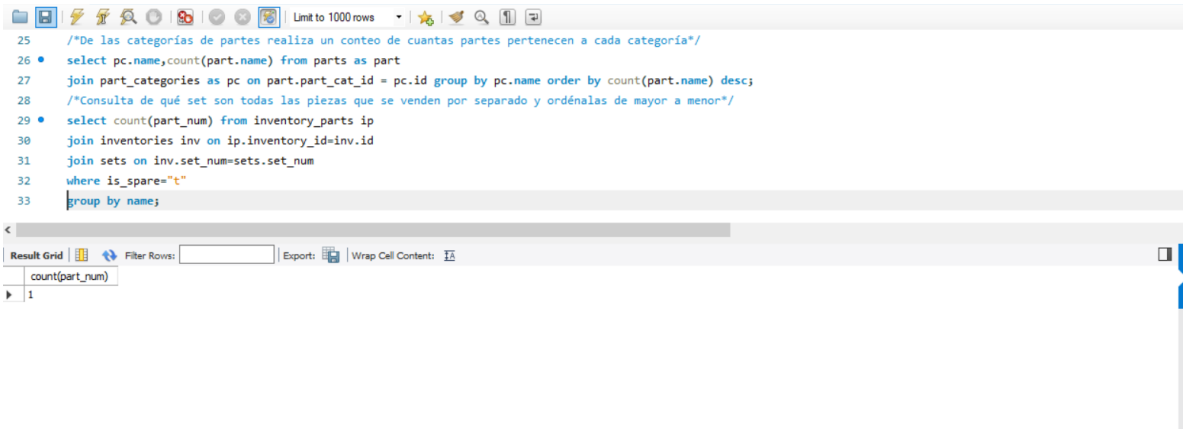


8.-Consulta de qué set son todas las piezas que se venden por separado y ordénalas de mayor a menor.



```
/*De las categorías de partes realiza un conteo de cuantas partes pertenecen a cada categoría*/
select pc.name,count(part.name) from parts as part
join part_categories as pc on part.part_cat_id = pc.id group by pc.name order by count(part.name) desc;
/*Consulta de qué set son todas las piezas que se venden por separado y ordénalas de mayor a menor*/
select */*,count(part_num)*/ from inventory_parts ip
join inventories inv on ip.inventory_id=inv.id
join sets on inv.set_num=sets.set_num
where is_spare="t"
/*group by name*/;
```

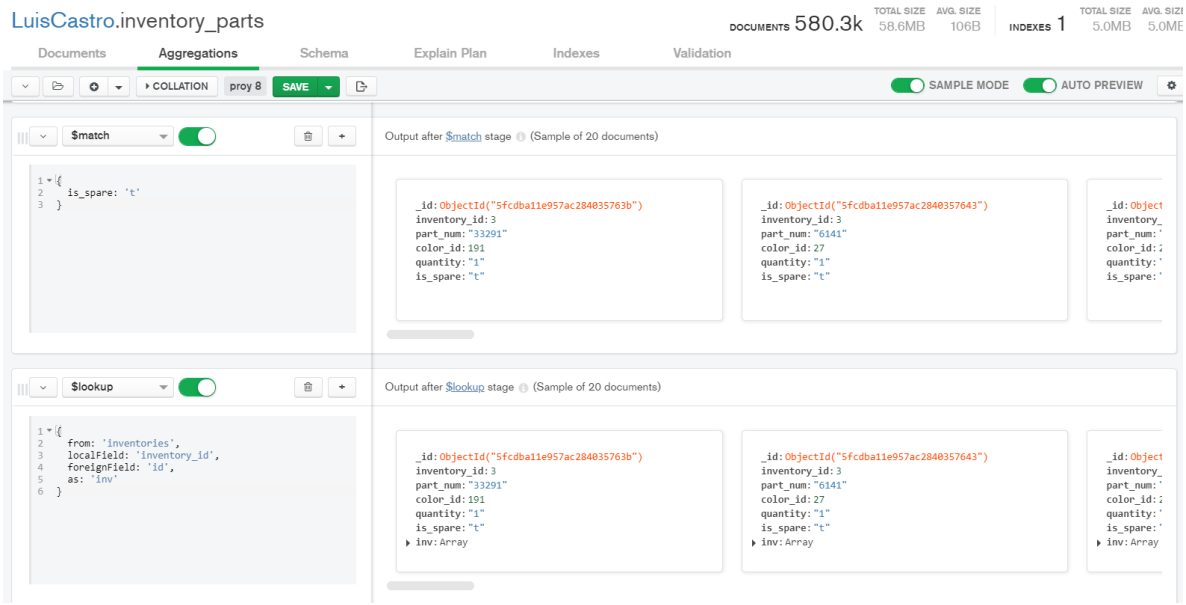
inventory_id	part_num	color_id	quantity	is_spare	id	version	set_num	set_num	name	year	theme_id	num_parts
49	6141	15	1	t	49	1	10071-1	10071-1	Mr. Bunny	2003	229	25

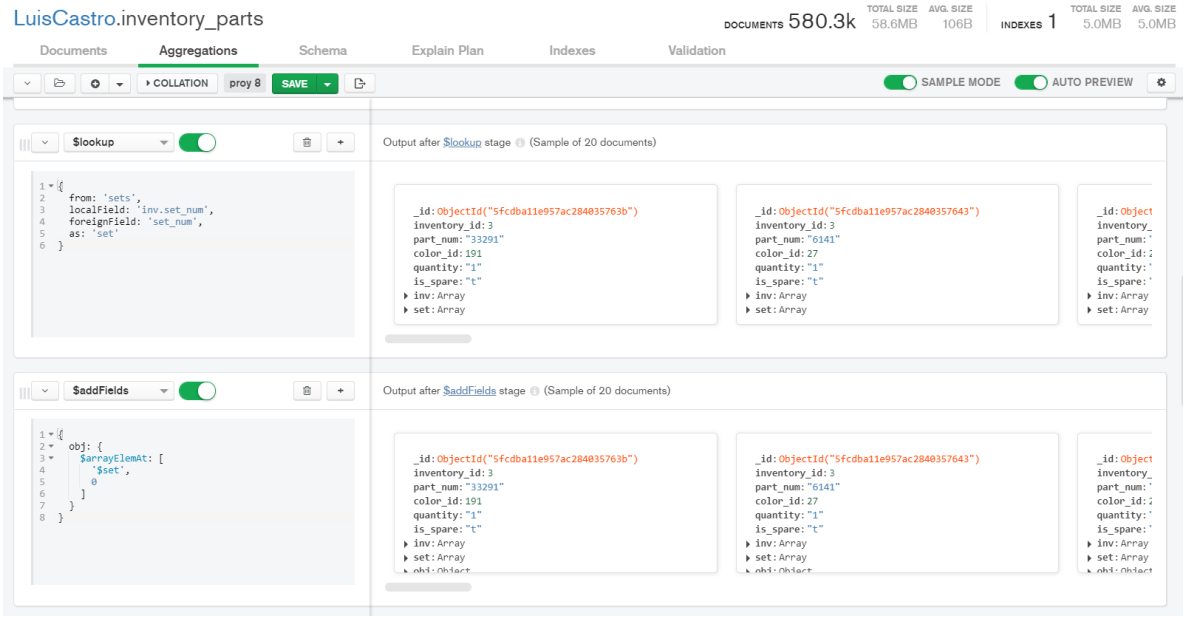


```
/*De las categorías de partes realiza un conteo de cuantas partes pertenecen a cada categoría*/
select pc.name,count(part.name) from parts as part
join part_categories as pc on part.part_cat_id = pc.id group by pc.name order by count(part.name) desc;
/*Consulta de qué set son todas las piezas que se venden por separado y ordénalas de mayor a menor*/
select count(part_num) from inventory_parts ip
join inventories inv on ip.inventory_id=inv.id
join sets on inv.set_num=sets.set_num
where is_spare="t"
group by name;
```

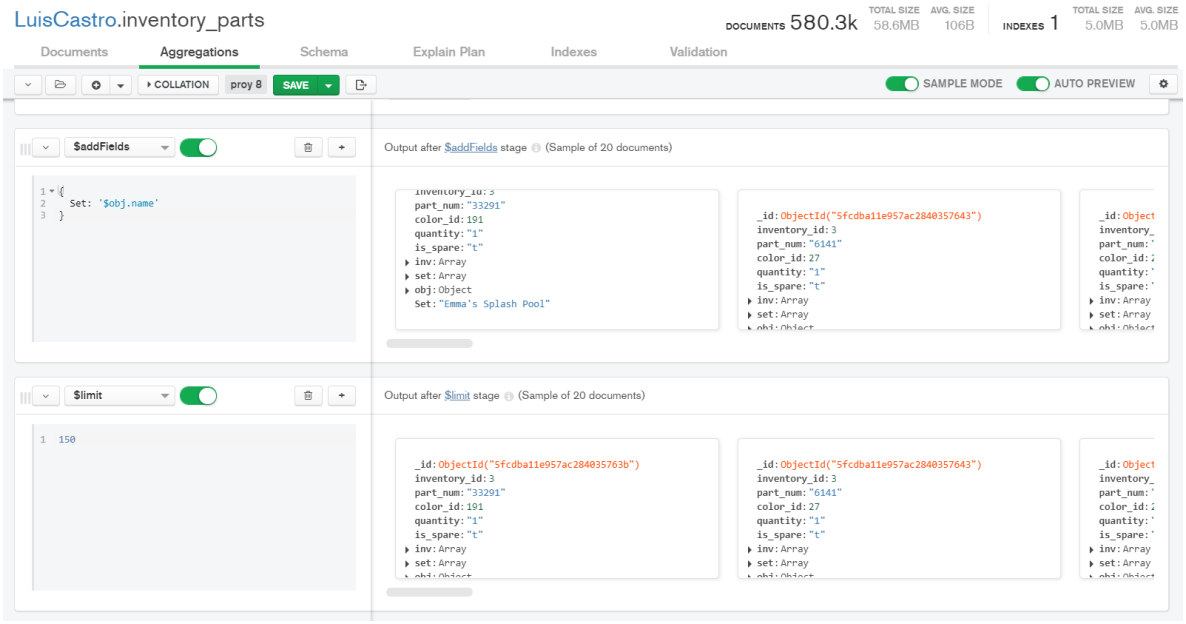
count(part_num)
1

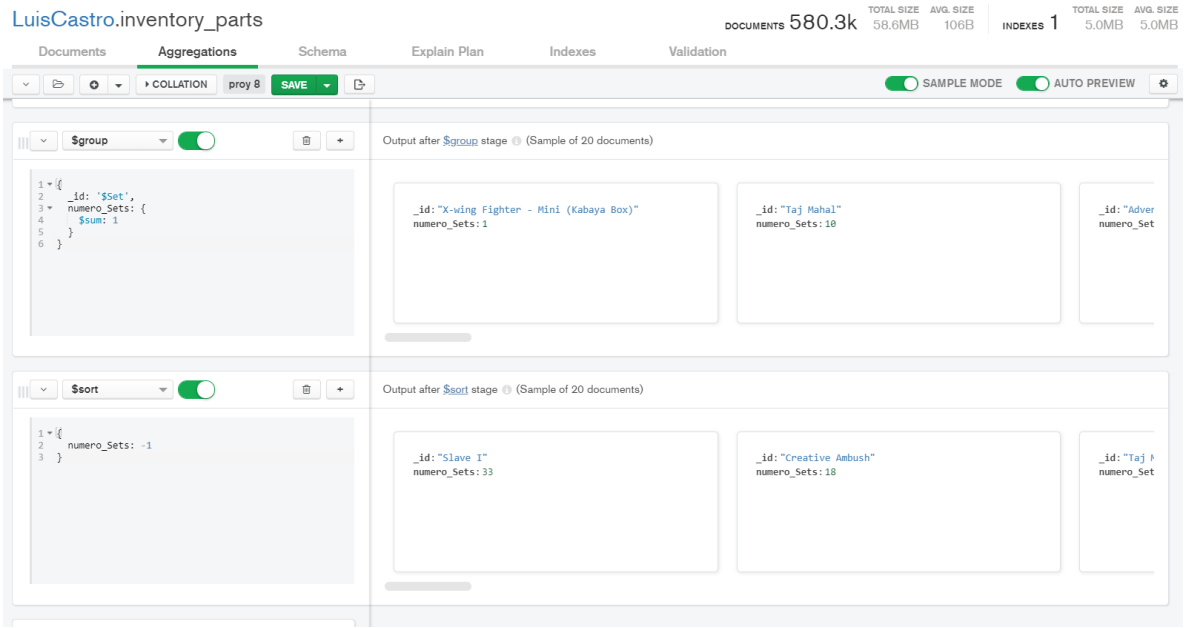
Luis F Castro





Luis F Castro





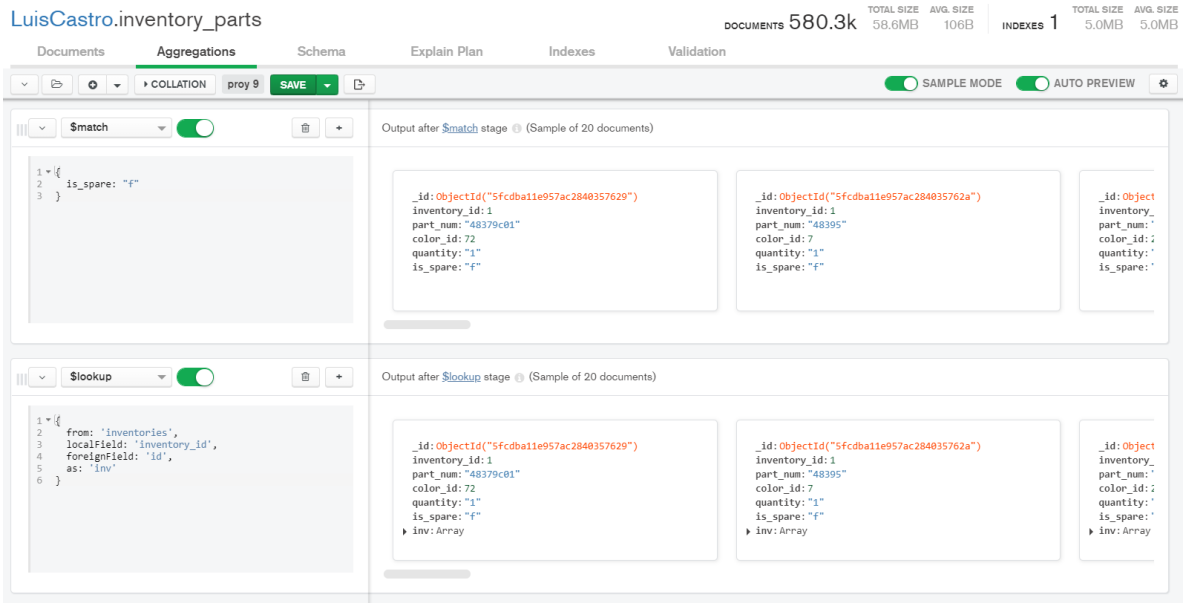
9.- Consulta de qué set son todas las piezas que no se venden por separado y ordénalas de mayor a menor.

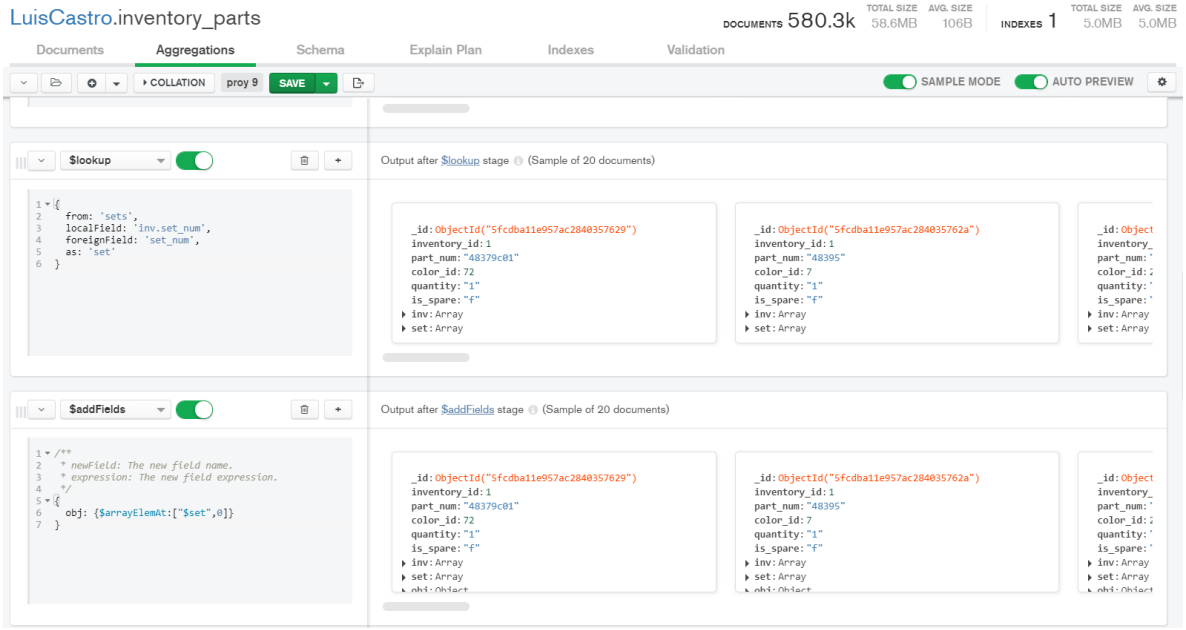
Luis F Castro

```
Limit to 1000 rows
31    join sets on inv.set_num=sets.set_num
32    where is_spare="t"
33    group by name;
34    /*Consulta de qué set son todas las piezas que no se venden por separado y ordénalas de mayor a menor*/
35 •  select count(part_num) from inventory_parts ip
36    join inventories inv on ip.inventory_id=inv.id
37    join sets on inv.set_num=sets.set_num
38    where is_spare="f"
39    group by name;
```

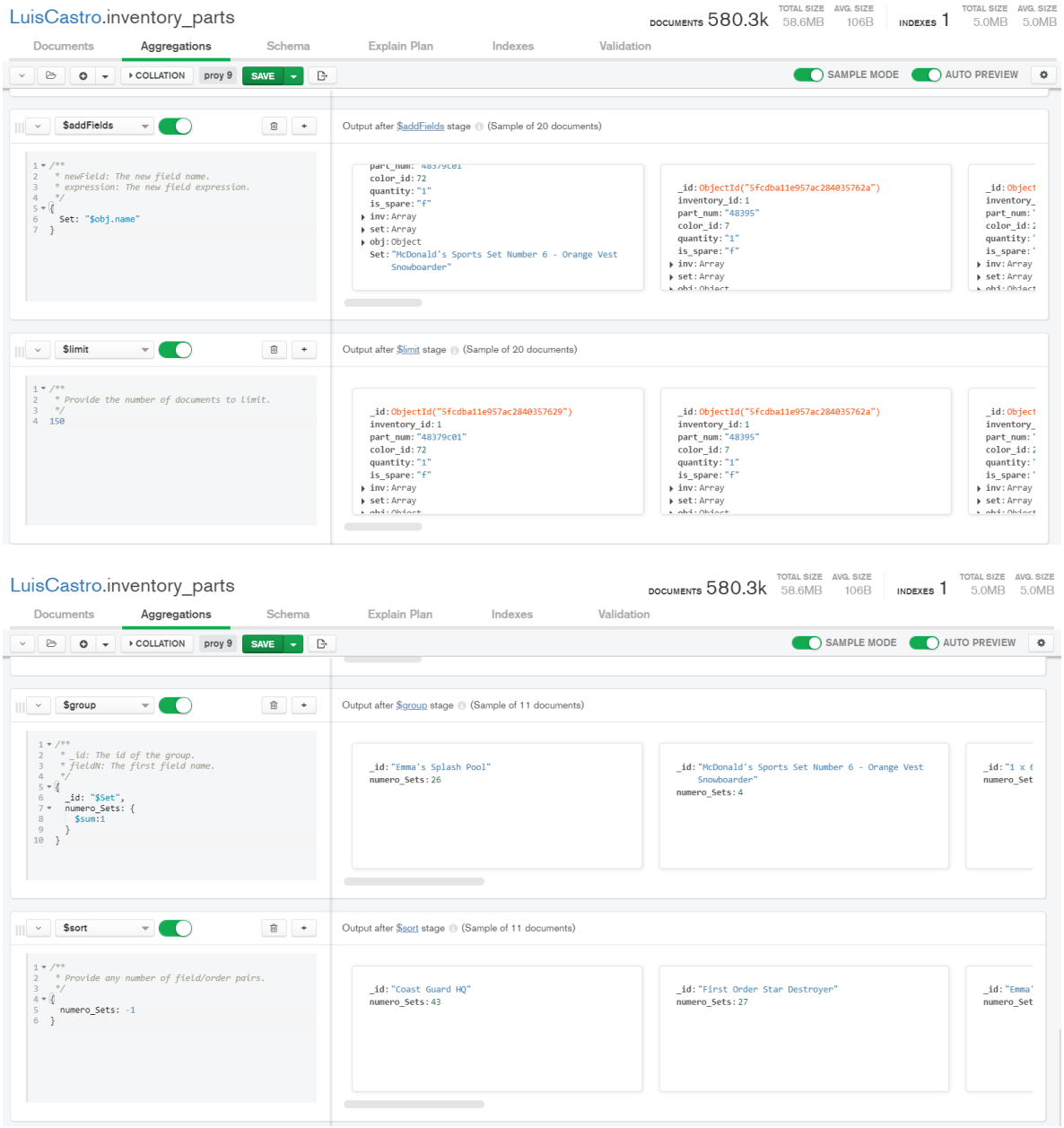
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

count(part_num)
17





Luis F Castro



10.- Crea una consulta empezando por la tabla partes de inventarios que incluya, el numero de parte, la cantidad, si se vende suelta, el color, el rgb, si es transparente, el set al que pertenece, el año del set y la temática.

Luis F Castro

```
38  where is_spare="f"
39  group by name;
40  /*Crea una consulta empezando por la tabla partes de inventarios que incluya, el numero de parte, la cantidad,
41  si se vende suelta, el color, el rgb, si es transparente, el set al que pertenece, el año del set y la temática.*/
42  select part_num,quantity,is_spare,version,sets.name,year,themes.name,col.name,rgb,is_trans from inventory_parts ip
43  join inventories inv on ip.inventory_id=inv.id
44  join sets on inv.set_num=sets.set_num
45  join themes on sets.theme_id= themes.id
46  join colors col on ip.color_id=col.id;
```

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

part_num	quantity	is_spare	version	name	year	name	name	rgb	is_trans
3665	2	f	1	Mr. Bunny	2003	Easter	Blue	0055BF	f
3040b	2	f	1	Mr. Bunny	2003	Easter	Blue	0055BF	f
3020	1	f	1	Mr. Bunny	2003	Easter	Blue	0055BF	f
3004	1	f	1	Mr. Bunny	2003	Easter	Blue	0055BF	f
3710	1	f	1	Mr. Bunny	2003	Easter	Red	C91A09	f
3039	1	f	1	Mr. Bunny	2003	Easter	Red	C91A09	f
3023	1	f	1	Mr. Bunny	2003	Easter	Red	C91A09	f
3022	1	f	1	Mr. Bunny	2003	Easter	Red	C91A09	f
4286	3	f	1	Mr. Bunny	2003	Easter	Yellow	F2CD37	f
3710	2	f	1	Mr. Bunny	2003	Easter	Yellow	F2CD37	f

LuisCastro.inventory_parts

DOCUMENTS 580.3k TOTAL SIZE 58.6MB AVG. SIZE 106B | INDEXES 1 TOTAL SIZE 5.0MB AVG. SIZE 5.0MB

Documents | Aggregations | Schema | Explain Plan | Indexes | Validation

COLLATION | proy 10 | SAVE | SAMPLE MODE | AUTO PREVIEW

$lookup

```
/**
 * from: The target collection.
 * localField: The local join field.
 * foreignField: The target join field.
 * as: The name for the results.
 * pipeline: The pipeline to run on the joined colle
 * let: Optional variables to use in the pipeline fi
 */
{
  from: 'colors',
  localField: 'color_id',
  foreignField: 'id',
  as: 'col'
}
```

Output after $lookup stage (Sample of 20 documents)

```
_id: ObjectId("5fcdba11e957ac2840357629")
inventory_id: 1
part_num: "48379c01"
color_id: 72
quantity: "1"
is_spare: "f"
col: Array
```

```
_id: ObjectId("5fcdba11e957ac284035762a")
inventory_id: 1
part_num: "48395"
color_id: 7
quantity: "1"
is_spare: "f"
col: Array
```

$lookup

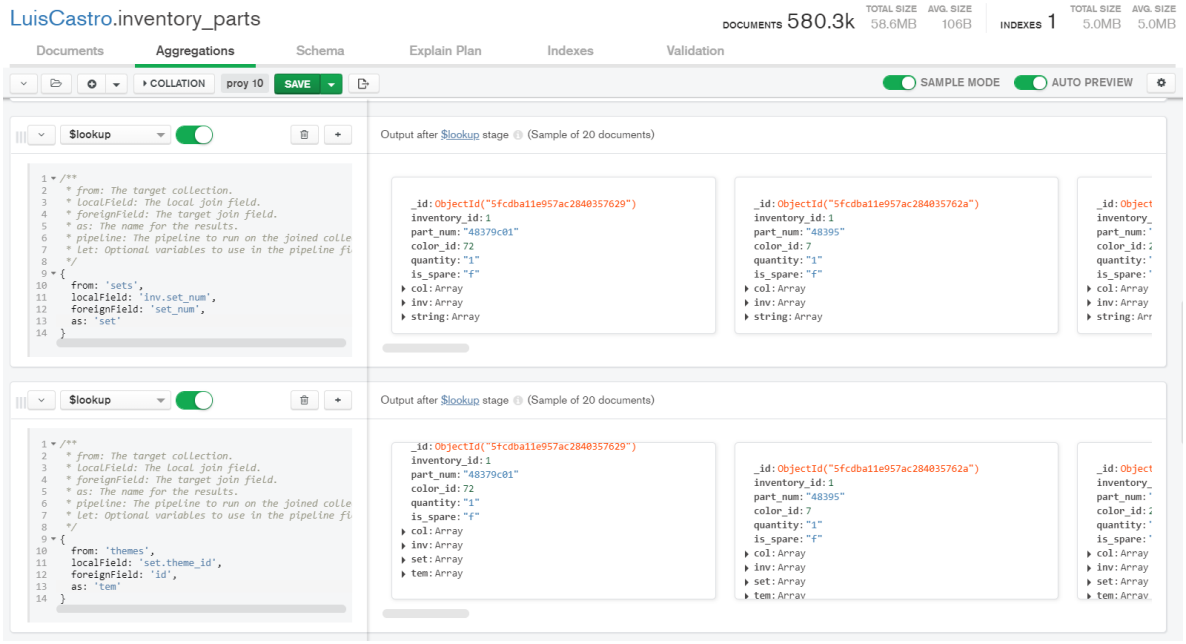
```
/**
 * from: The target collection.
 * localField: The local join field.
 * foreignField: The target join field.
 * as: The name for the results.
 * pipeline: The pipeline to run on the joined colle
 * let: Optional variables to use in the pipeline fi
 */
{
  from: 'inventories',
  localField: 'inventory_id',
  foreignField: 'id',
  as: 'inv'
}
```

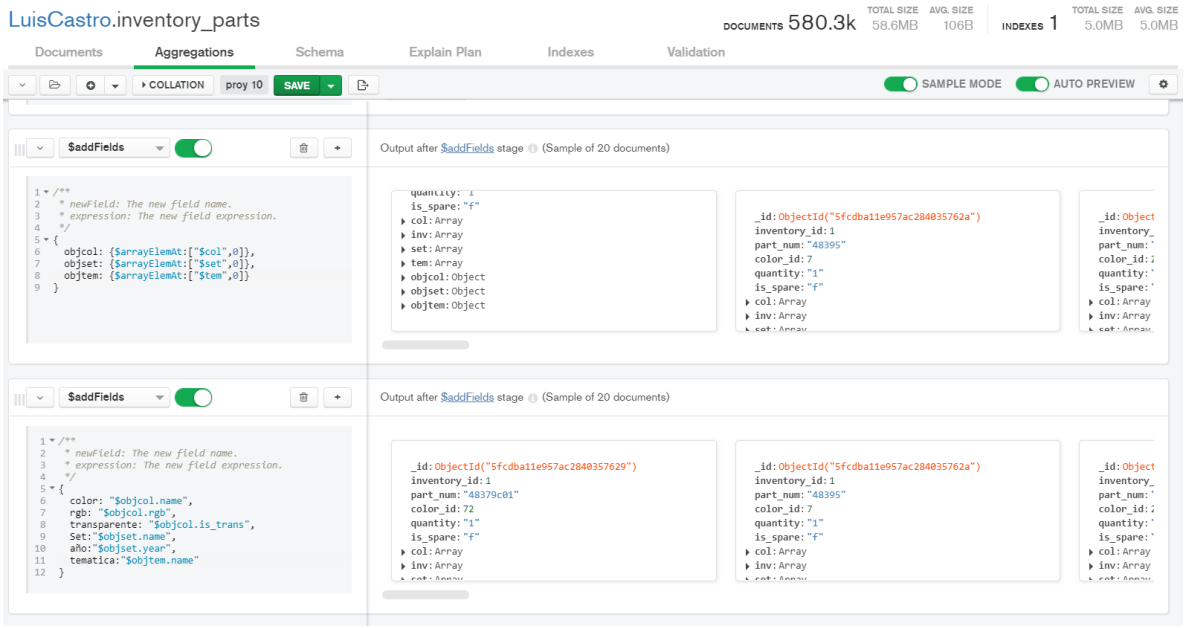
Output after $lookup stage (Sample of 20 documents)

```
_id: ObjectId("5fcdba11e957ac2840357629")
inventory_id: 1
part_num: "48379c01"
color_id: 72
quantity: "1"
is_spare: "f"
col: Array
string: Array
```

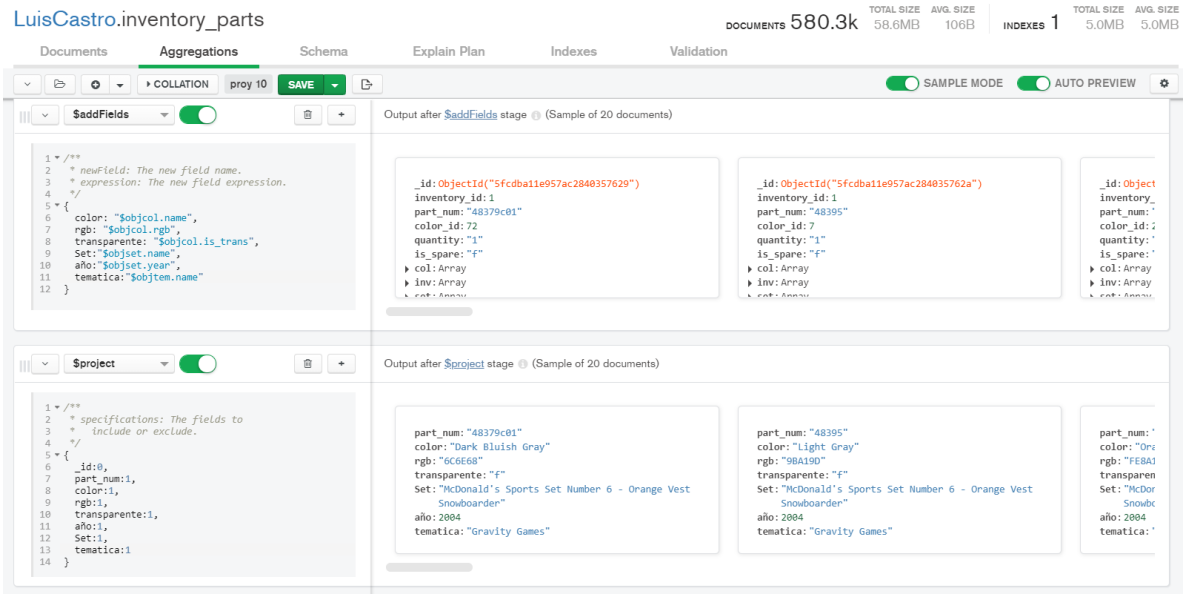
```
_id: ObjectId("5fcdba11e957ac284035762a")
inventory_id: 1
part_num: "48395"
color_id: 7
quantity: "1"
is_spare: "f"
col: Array
string: Array
```

Luis F Castro

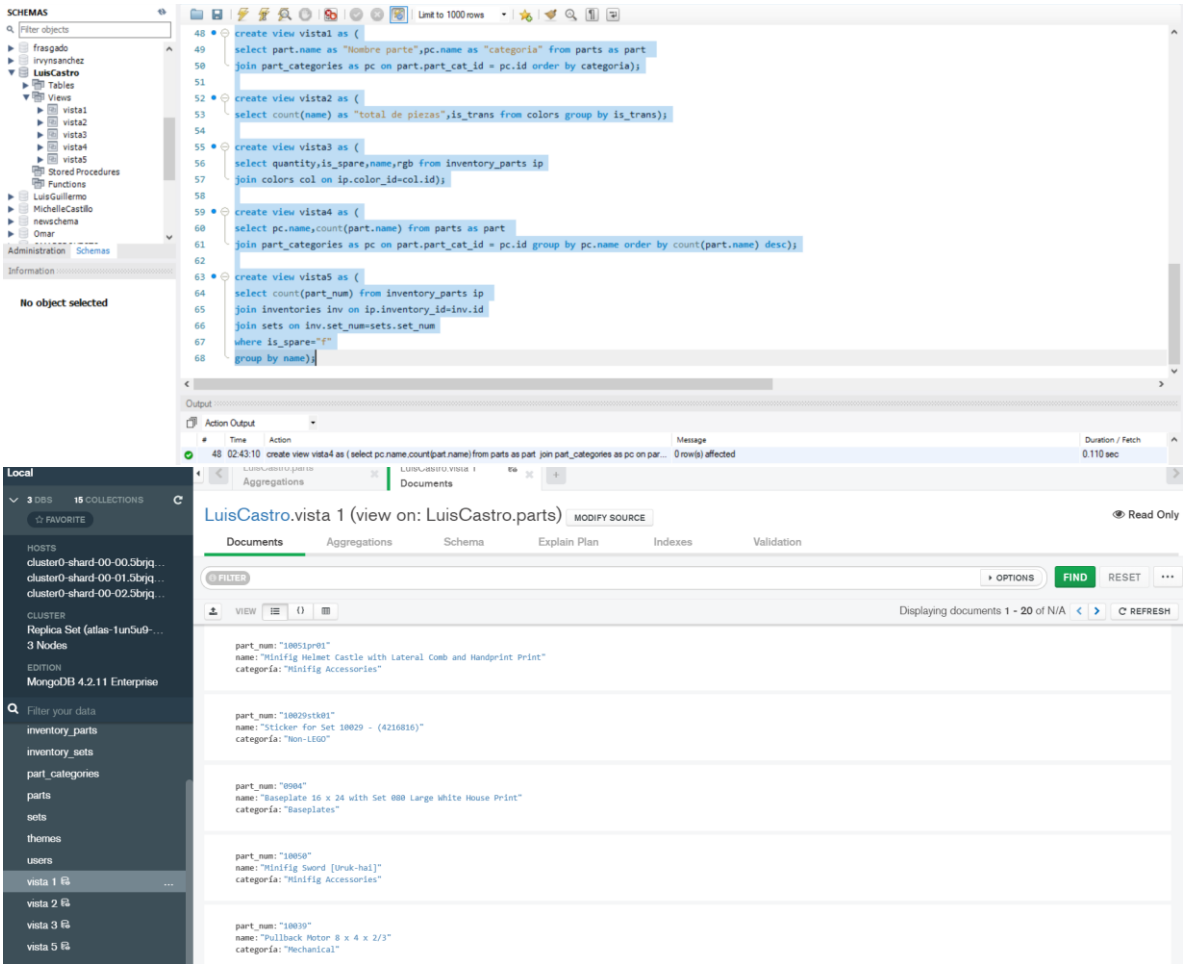




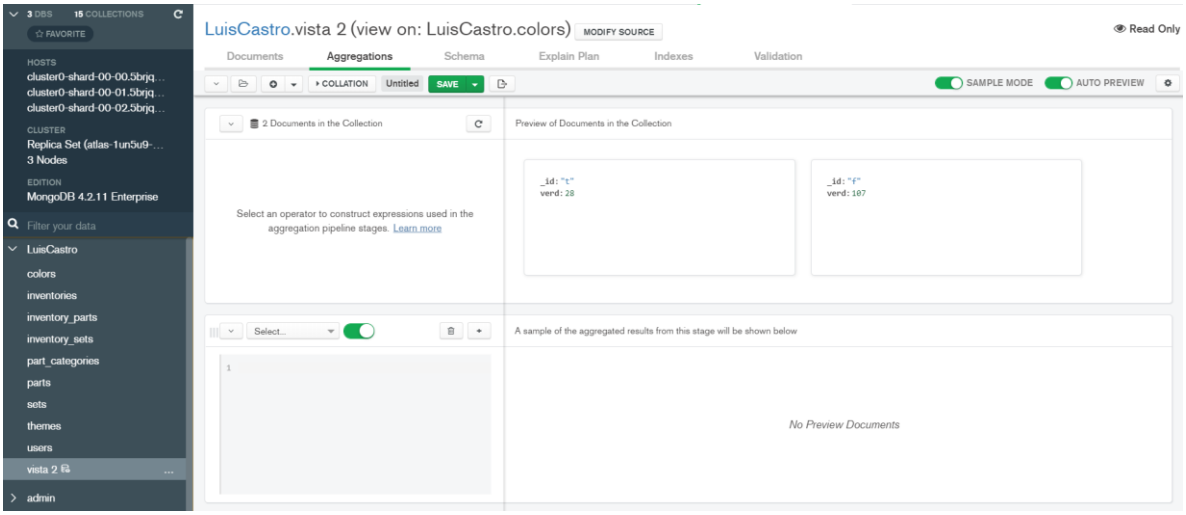
Luis F Castro

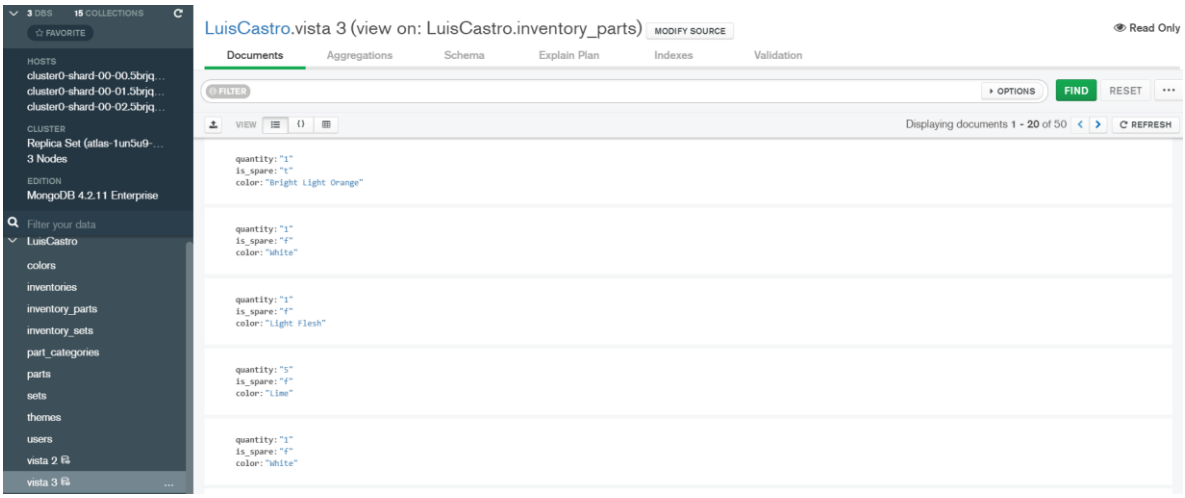


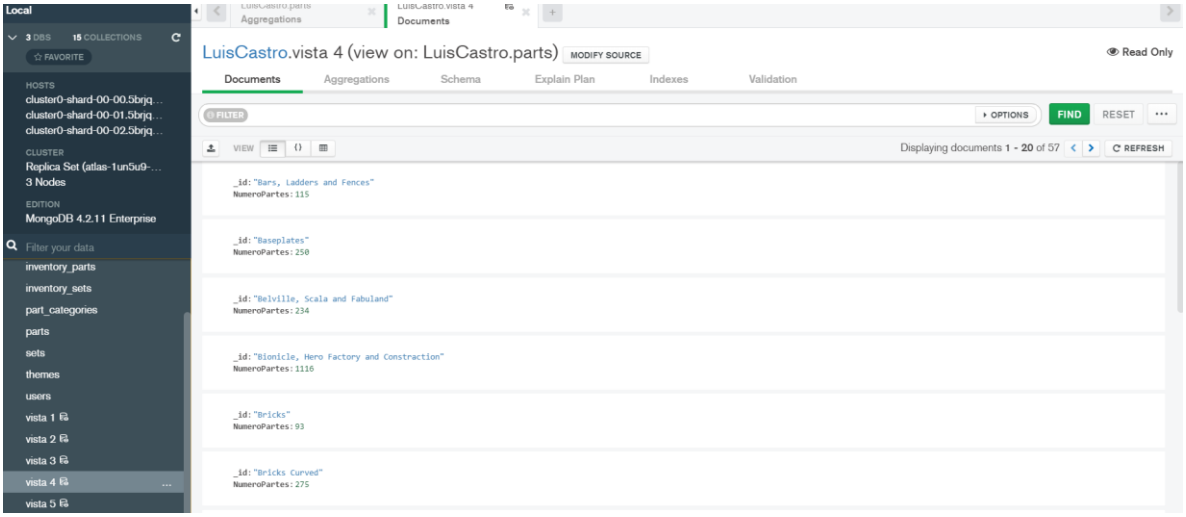
11-15 convierte 5 de las consultas anteriores en vistas.



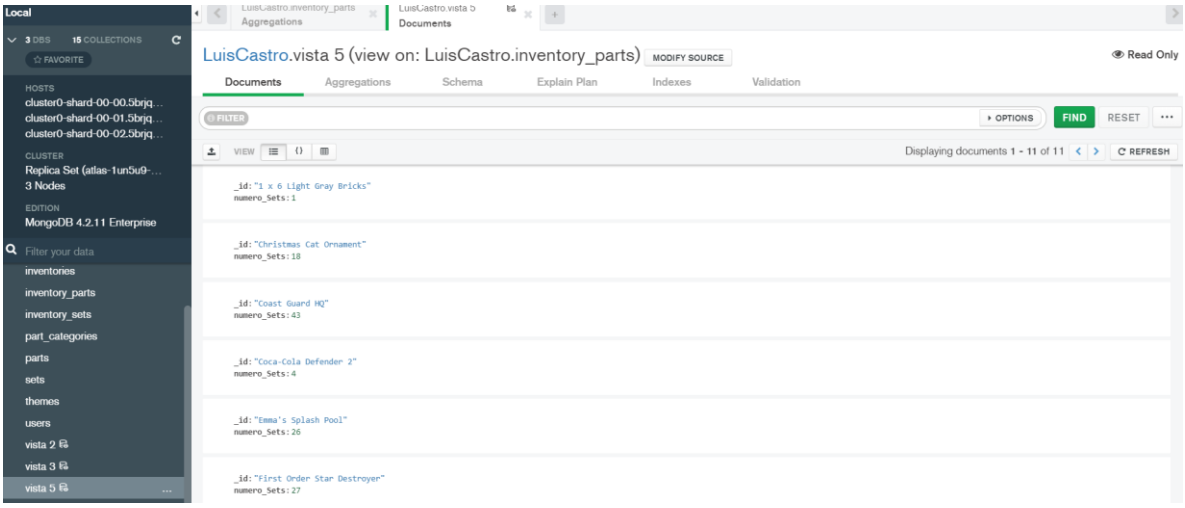
Luis F Castro







Luis F Castro



Luis F Castro

Conclusión

1.- Según las consultas realizadas de las bases de datos podemos afirmar que los inventarios que contienen pertenecen a una tienda, no perteneces a un gran almacén.

2.-El stock se mantiene con números promedio, esto indica que la oferta y la demanda están equilibradas para el tema de los “Sets” salvo por los paquetes que cuyo contenido es misterioso, es decir los sets que tienen piezas pertenecientes a algún tema en específico, pero el contenido fue hecho de piezas al azar, este modelo de negocios impulsa al comprador a seguir comprando los paquetes hasta obtener una pieza en especial.

3.-Algunos “Sets” comparten piezas por pertenecer a la misma temática, una propuesta para ventas puede ser juntar en una promoción los “Sets” de años pasados en conjunto con los “Sets” modernos, esta propuesta permite por ambas partes recibir una ganancia, por el lado del cliente recibe 2 productos con funcionalidad parecida y aprovechar las sinergias de ambos sets mientras que el vendedor comienza a vender “Sets” con poca probabilidad de venta.

4.-Los productos Lego tienen gran versatilidad en sus temáticas y sus propuestas de productos tienen muchas sinergias entre sí por esto mismo es buena idea sugerir un programa de lealtad.

5.-La Base de datos tiene mucha información para brindar pero al tratarse de un inventario es recomendable ampliar la base de datos agregando precios y sacar mucho valor del las bases de datos.

Bibliografía

https://www.kaggle.com/rtatman/lego-database