



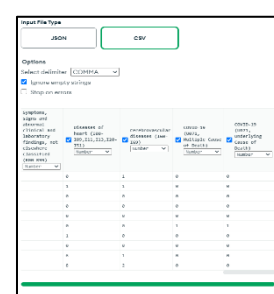
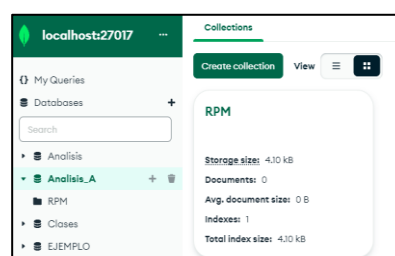
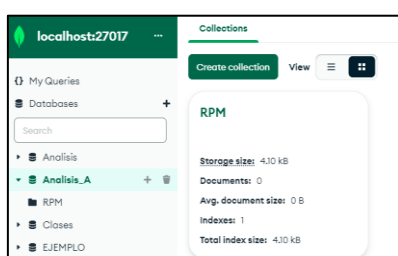
AH Recuentos provisionales mensuales de muertes por causas seleccionadas de muerte por sexo, edad, raza y origen hispano

Monitoreo de muertes provocadas por diversas enfermedades usando base de datos NoSQL en mongoDB

El objetivo del estudio es mediante inserción de un dataset a la base de datos Mongo DB, realizar un análisis mediante consultas utilizando un sistema propio de documento conocido con el nombre de BSON, pasando objetos JSON como parámetros. Estas consultas se realizarán a un recuento mensual de muertes causadas por diversas enfermedades, en variables como, sexo, edad, raza y origen. Con el fin de observar que enfermedad alrededor de los años 2019, 2020 y 2021 tuvo un mayor impacto, para este análisis se han elegido una serie de preguntas que filtran las causas de muerte causadas por cada enfermedad.

Este archivo ha sido extraído de la web DATA.GOV¹ en la sección de salud y los datos han sido suministrados por el Centro de prevención y control de enfermedades.

Desde mongo compass se realiza la inserción de nuestro fichero con el que se trabajará, primeramente, se crea una base de datos llamada Analisis_A, anexa a esta se crea una colección que será nuestro caso de estudio llamada RPM (Recuentos provisionales mensuales de muertes por causas seleccionadas de muerte por sexo, edad, raza y origen hispano), en nuestra colección anexamos nuestro fichero el cual será anexado en CSV para facilitar el manejo de los datos:



Los datos que componen el dataset son:

- ❖ AnalysisDate: Día del análisis del dato.
- ❖ Date Of Death Year: Día de año de muerte.
- ❖ Date Of Death Month: Día del mes de Muerte.
- ❖ Start Date: Comienzo del día del ingreso.
- ❖ End Date: Día de muerte de las personas.
- ❖ Jurisdiction of Occurrence: jurisdicción de ocurrencia (lugar de la muerte).
- ❖ Sex: Genero.
- ❖ Race/Ethnicity: Raza/Etnia.
- ❖ AgeGroup: grupo de edad.
- ❖ "AllCause" : todas las causas.
- ❖ "NaturalCause" : causa natural
- ❖ Septicemia (A40-A41): septicemia (A40-A41).
- ❖ Malignant neoplasms (C00-C97) : neoplasma maligno (C00-C97).
- ❖ Diabetes mellitus (E10-E14): diabetes mellitus (E10-E14).
- ❖ Alzheimer disease (G30): enfermedad de Alzheimer.
- ❖ Influenza and pneumonia (J09-J18): neumonía e influenza.
- ❖ Chronic lower respiratory diseases (J40-J47)": Enfermedades respiratorias crónicas bajas.
- ❖ Other diseases of respiratory system (J00-J06,J30-J39,J67,J70-J98)" : Otras enfermedades del sistema respiratorio.
- ❖ "Nephritis, nephrotic syndrome and nephrosis (N00-N07,N17-N19,N25-N27)" : Nefritis, síndrome nefrótico y nefrosis.
- ❖ Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified (R00-R99): Síntomas, signos y hallazgos clínicos y de laboratorio anormales, no clasificados en otra parte.
- ❖ "Diseases of heart (I00-I09,I11,I13,I20-I51): Enfermedades del corazón.
- ❖ "Cerebrovascular diseases (I60-I69): Enfermedades cerebrovasculares.
- ❖ "COVID-19 (U071, Multiple Cause of Death): Covid-19 (múltiples causas de muerte)
- ❖ "COVID-19 (U071, Underlying Cause of Death): COVID-19 (Causas de muertes subyacentes).

Se usa NoSQLBooster for Mongo DB

La cantidad de datos a analizar son **3960** comprobados mediante la siguiente consulta(contar):



Se utiliza el **find** para poder visualizar todos los documentos, así como el **count** para contar todos los documentos que hay en nuestra base de datos.

var query2 = se utiliza el **\$ne** para decir que seleccione todos los datos que no sean iguales a "13/10/2021" en el campo de "AnalysisDate"

var query3 = se utiliza el **\$ne** para decir que seleccione todos los datos que no sean iguales a "United States" en el campo de "Jurisdiction of Occurrence"

var logic = se utiliza el operador lógico **\$and** para que se cumplan las dos anteriores queries.

```
var query2 = {"AnalysisDate":{"$ne: "13/10/2021"}}
var query3 = {"Jurisdiction of Occurrence": {"$ne:"United States"}}
var logic = {"$and: [query2,query3]}
db.RPM.find(logic)
```

Como se observa en los documentos no hay ningún dato de *AnalysisDate* o de *Jurisdiction of Occurrence*. Por lo tanto, solo se dirá que todos los datos fueron analizados el día 13/10/2021 en la *Jurisdiction of Occurrence* de *United States*.

En esta selección se borrará la columna del día que fueron analizados:

var query4 = con el **\$exists: true** se seleccionan todos los documentos que existan en el campo "AnalysisDate"

var operacion1 = se borra el campo de "AnalysisDate" usando el operador **\$unset**

con el **"updateMany"** se actualizan los datos en la colección de nuestra base de dato dependiendo de la query y la operación asignada.

```
var query4 = {"AnalysisDate":{"$exists:true}}
var operacion1 = {"$unset":{"AnalysisDate":"13/10/2021"}}
db.RPM.updateMany(query4,operacion1)
db.RPM.find({}).limit(100)
```

```
1+ {
2   "acknowledged" : true,
3   "matchedCount" : 3960,
4   "modifiedCount" : 3960
5 }
```

En esta selección se borrará la columna jurisdicción de ocurrencia en la que ocurrieron las muertes:

var query5 = Se seleccionan todos los documentos

var operacion2 = se borra el campo de "Jurisdiction of Occurrence" usando el operador **\$unset**

con el **"updateMany"** se actualizan los datos en la colección de nuestra base de dato dependiendo de la query y la operación asignada.

```
var query5 = {}
var operacion2 = {"$unset":{"Jurisdiction of Occurrence":"United States"}}
db.RPM.updateMany(query5,operacion2)
db.RPM.find().limit(100)
```

```
1+ {
2   "acknowledged" : true,
3   "matchedCount" : 3960,
4   "modifiedCount" : 3960
5 }
```

Luego de esto se comienza a realizar un cambio de **"String"** a **"Int"** para los en todos los documentos que contengan datos de número, usando las siguientes consultas:

- el dato del año de muerte

\$set = con este operador se crea un nuevo campo sobre la ya existente "Date Of Death Year"

\$toInt = con este operador se cambia en este caso de **"String"** a un **"Int"** en todos los valores del campo de "Date Of Death Year"

con el **"updateMany"** se actualizan los datos en la colección de nuestra base de dato dependiendo de la query y la operación asignada.

```
db.RPM.updateMany({}, [{
  $set: {
    "Date Of Death Year": {
      $toInt: "$Date Of Death Year"
    }
  }
}])
```

```
1+ {
2   "acknowledged" : true,
3   "matchedCount" : 3960,
4   "modifiedCount" : 3960
5 }
```


Luego de esto se comienza a realizar un cambio de **"String"** a **"Date"** para las fechas usando las siguientes consultas:

Para el comienzo del dato.

\$set = con este operador se crea un nuevo campo sobre la ya existente "Start Date"

\$convert = con este operador se seleccionan todos los valores de **"input": "\$Start Date"** (en este caso **"String"**) y se ordena que cambie a **"to": "date"**

con el **"updateMany"** se actualizan los datos en la colección de nuestra base de dato dependiendo de la query y la operación asignada.

```
db.RPM.updateMany({}, [{
  $set: {
    "Start Date": {
      "$convert": {
        "input": "$Start Date",
        "to": "date"
      }
    }
  }
}])
db.RPM.find({}).limit(100)
```

Para el fin del dato.

\$set = con este operador se crea un nuevo campo sobre la ya existente "Start Date"

\$convert = con este operador se seleccionan todos los valores de **"input": "\$Start Date"** (en este caso **"String"**) y se ordena que cambie a **"to": "date"**

con el **"updateMany"** se actualizan los datos en la colección de nuestra base de dato dependiendo de la query y la operación asignada.

```
db.RPM.updateMany({}, [{
  $set: {
    "End Date": {
      "$convert": {
        "input": "$End Date",
        "to": "date"
      }
    }
  }
}])
db.RPM.find({}).limit(100)
```

*(Nota: todo el proceso anterior de cambio de **String** a **Int** y a **Date** se pueden realizar desde el mongo DB compass.)*

Finalmente terminamos teniendo un correspondiente tipo de input para cada campo, con el cual comenzaremos a realizar una serie de consultas para realizar un análisis estadístico de los datos, en base al dato de muerte, grupo de edad, sexo y diversa causalidad de muerte:

_id	Date Of Death Year	Date Of Death Month	Start Date	End Date	Sex	Race/Ethnicity	AgeGroup	AllCause	NaturalCause	Septicemia (A40-A41)
1	63a239c8d07976 2019	7	1/7/2019 2:00:00	31/7/2019 2:00:00	M	Other	0-4 years	61	52	0
2	63a239c8d07976 2019	9	1/9/2019 2:00:00	30/9/2019 2:00:00	F	Other	25-34 years	26	8	0
3	63a239c8d07976 2020	3	1/3/2020 1:00:00	31/3/2020 2:00:00	Female	Other	0-4 years	40	35	0
4	63a239c8d07976 2020	3	1/3/2020 1:00:00	31/3/2020 2:00:00	Female	Other	5-14 years	6	4	1
5	63a239c8d07976 2020	3	1/3/2020 1:00:00	31/3/2020 2:00:00	Female	Other	15-24 years	14	2	0
6	63a239c8d07976 2021	4	1/4/2021 2:00:00	30/4/2021 2:00:00	Male	Other	0-4 years	49	42	0
7	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	0-4 years	182	162	4
8	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	5-14 years	44	28	1
9	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	15-24 years	122	45	0
10	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	25-34 years	198	100	1
11	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	35-44 years	334	260	7
12	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	45-54 years	585	500	6
13	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	55-64 years	990	942	20
14	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	65-74 years	1355 (1.4K)	1311 (1.3K)	22
15	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	75-84 years	1951 (2.0K)	1908 (1.9K)	33
16	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Hispanic	85 years and over	2720 (2.7K)	2663 (2.7K)	28
17	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Non-Hispanic American Ind	0-4 years	17	15	0
18	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Non-Hispanic American Ind	5-14 years	3	1	0
19	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Non-Hispanic American Ind	15-24 years	12	3	0
20	63a239c8d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	F	Non-Hispanic American Ind	25-34 years	43	21	0

Según la imagen anterior podemos ver que para los datos del **2019** en el campo **"Sex"** se tienen valores de **F** para las **mujeres** y **M** para los **hombres**, por el contrario, en los años **2020-2021** se tienen valores de **Female** para las **mujeres** y **Male** para los **hombres**. Dicho de otra manera, una falta de homogeneidad en los datos lo cual podría ser un problema al momento de realizar consultas, un ejemplo podría ser al momento de realizar un conteo de las muertes referidas por el campo **"Sex"**, si no se tiene en cuenta lo anterior. Por lo tanto, se realizarán cambios para volver los valores de **F** a **Female** y los valores de **M** a **Male** en el campo **"Sex"**:

Aggregate = en nuestro caso estamos utilizando el operador de agregación para realizar diversas operaciones en los datos agrupados devolviendo un solo resultado

"\$addFields" = con este operador se agrega una nuevo campo llamado **"Sex"**

"\$switch" = con este operador remplazamos un cambio en un dato de un documento por otro, en nuestro caso se quiere cambiar específicos documentos (**\$eq**) como es en todos los valores que tengamos **"F"** del campo **"\$Sex"**, cambiarlos por **"Female"** y de la misma manera en **"M"** por **"Male"**

```
db.RPM.aggregate([
  {
    "$addFields": {
      "Sex": {
        "$switch": {
          "branches": [
            {"case": {"$eq": ["$Sex", "F"]}, "then": "Female"},
            {"case": {"$eq": ["$Sex", "M"]}, "then": "Male"}
          ],
          "default": "$Sex"
        }
      }
    }
  }
])
```

Primeramente, se usa el **agregate** para saber que el código no tiene ningún error y luego se cambia este de **aggregate** a **updateMany**:

_id	Date Of Death Year	Date Of Death Month	Start Date	End Date	Sex	Race/Ethnicity	AgeGroup	AIICause	NaturalCause	Sepsicemia (A40-A41)
1	63a239c9d07976 2019	7	1/7/2019 2:00:00	31/7/2019 2:00:00	Male	Other	0-4 years	61	52	0
2	63a239c9d07976 2019	9	1/9/2019 2:00:00	30/9/2019 2:00:00	Female	Other	25-34 years	26	8	0
3	63a239c9d07976 2020	3	1/3/2020 1:00:00	31/3/2020 2:00:00	Female	Other	0-4 years	49	35	0
4	63a239c9d07976 2020	3	1/3/2020 1:00:00	31/3/2020 2:00:00	Female	Other	5-14 years	6	4	1
5	63a239c9d07976 2020	3	1/3/2020 1:00:00	31/3/2020 2:00:00	Female	Other	15-24 years	14	2	0
6	63a239c9d07976 2021	4	1/4/2021 2:00:00	30/4/2021 2:00:00	Male	Other	0-4 years	49	42	0
7	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	0-4 years	182	162	4
8	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	5-14 years	44	28	1
9	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	15-24 years	122	45	0
10	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	25-34 years	198	100	1
11	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	35-44 years	334	260	7
12	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	45-54 years	585	500	6
13	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	55-64 years	990	942	29
14	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	65-74 years	1355 (1.4K)	1311 (1.3K)	22
15	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	75-84 years	1951 (2.0K)	1908 (1.9K)	33
16	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	85 years and over	2729 (2.7K)	2663 (2.7K)	28

Debido a que el cambio se realizó de manera óptima y no hubo otro tipo de cambio en los datos se realiza el cambio a **updateMany**:

```
db.RPM.updateMany({},[
  {
    "$addFields": {
      "Sex": {
        "$switch": {
          "branches": [
            {"case": {"$eq": ["$Sex", "F"]}, "then": "Female"},
            {"case": {"$eq": ["$Sex", "M"]}, "then": "Male"}
          ],
          "default": "$Sex"
        }
      }
    }
  }
])
```

```
{
  "acknowledged": true,
  "matchedCount": 3960,
  "modifiedCount": 1440
}
```

_id	Date Of Death Year	Date Of Death Month	Start Date	End Date	Sex	Race/Ethnicity	AgeGroup	AIICause	NaturalCause	Sepsicemia (A40-A41)
1	63a239c9d07976 2019	7	1/7/2019 2:00:00	31/7/2019 2:00:00	Male	Other	0-4 years	61	52	0
2	63a239c9d07976 2019	9	1/9/2019 2:00:00	30/9/2019 2:00:00	Female	Other	25-34 years	26	8	0
3	63a239c9d07976 2020	3	1/3/2020 1:00:00	31/3/2020 2:00:00	Female	Other	0-4 years	49	35	0
4	63a239c9d07976 2020	3	1/3/2020 1:00:00	31/3/2020 2:00:00	Female	Other	5-14 years	6	4	1
5	63a239c9d07976 2020	3	1/3/2020 1:00:00	31/3/2020 2:00:00	Female	Other	15-24 years	14	2	0
6	63a239c9d07976 2021	4	1/4/2021 2:00:00	30/4/2021 2:00:00	Male	Other	0-4 years	49	42	0
7	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	0-4 years	182	162	4
8	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	5-14 years	44	28	1
9	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	15-24 years	122	45	0
10	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	25-34 years	198	100	1
11	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	35-44 years	334	260	7
12	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	45-54 years	585	500	6
13	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	55-64 years	990	942	29
14	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	65-74 years	1355 (1.4K)	1311 (1.3K)	22
15	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	75-84 years	1951 (2.0K)	1908 (1.9K)	33
16	63a239c9d07976 2019	1	1/1/2019 1:00:00	31/1/2019 1:00:00	Female	Hispanic	85 years and over	2729 (2.7K)	2663 (2.7K)	28

Ya teniendo los datos completamente homogeneizados, se puede realizar consultas para realizar un análisis de los datos:

Contar el número de hombres que hay en nuestro estudio.

var query6 = Se filtra todos los documentos por el campo **"Sex"** que sean **"Male"** usando el operador **"\$match"**

var query7 = Se separa los documentos en grupos según una **"clave de grupo"** que tenga en este caso los documentos que se filtraron anteriormente con el operador **\$group** y con el operador **\$sum** se realiza una suma de todos estos documentos

var query8 = Se muestran los resultados sin proyectar el campo **"_id"**

con el **"Aggregate"** se para realizar diversas operaciones en los datos agrupados devolviendo un solo resultado.

```
var query6= { "$match" : { "Sex": "Male" }}
var query7= { "$group" : { "_id": null, "count death Male" : { "$sum" : 1 } } }
var query8= { "$project": { "_id" : 0 } }
var filtro=[query6,query7, query8]
db.RPM.aggregate(filtro)
```

	count death Male
1	1980 (2.0K)

En el estudio la cantidad de hombres muertos por cualquier tipo de enfermedad es de 1980, al comienzo del estudio se había observado que el dataset tiene 3960 documentos, teniendo el 50% de hombres y 50% de mujeres como objeto de estudio.

Contar el número de personas muertas en un rango de edad entre 25-34.

var query9 = Se filtra todos los documentos por el campo **"AgeGroup"** que estén en un rango de **"25-34 years"** usando el operador **"\$match"**

var query10 = Se separa los documentos en grupos según una **"clave de grupo"** que tenga en este caso los documentos que se filtraron anteriormente con el operador **\$group** y con el operador **\$sum** se realiza una suma de todos estos documentos

var query11 = Se muestran los resultados sin proyectar el campo **"_id"**

con el **"Aggregate"** se para realizar diversas operaciones en los datos agrupados devolviendo un solo resultado.

```
var query9= { "$match" : { "AgeGroup": "25-34 years" }}
var query10= { "$group" : { "_id": null, "count" : { "$sum" : 1 } } }
var query11= { "$project": { "_id" : 0 } }
var filtro2=[query9,query10, query11]
db.RPM.aggregate(filtro2).projection({}).sort({_id:-1}).limit(100)
```

	count death by age range
1	396

La cantidad de muertes de personas en un rango de 25-34 años es de 396.

media de muertes por diabetes mellitus.

Aggregate = en nuestro caso estamos utilizando el operador de agregación para realizar diversas operaciones en los datos agrupados devolviendo un solo resultado

"\$group" = con este operador se agrupa todos los documentos (muertes) del campo diabetes mellitus para realizar un promedio (**\$avg**) de todos los datos relacionados con las muertes causadas por esta enfermedad.

```
var query12= { $group: { "_id": "Conjunto de todos los datos",
"Media diabetes mellitus":{ "$avg" : "$Diabetes mellitus (E10-E14)" }}}
db.RPM.aggregate(query12)
```

	_id	Media diabetes mellitus
1	Conjunto de todos los datos	66.2227

El promedio de muertes causadas por la enfermedad de diabetes mellitus es de 66,2.

Calcular el documento exacto en el que hubo un índice de mayor y menor muertos causados por enfermedades relacionadas con Alzheimer

Query13 = se utiliza el operador **"\$sort"** para ordenar de forma ascendente (1) todos los datos del campo **"Alzheimer disease (G30)"**

Query14 = se agrupan (**\$group**) por un campo **"_id"** establecido a **null**, y se crea un campo llamado **docs** donde se añaden todos los documentos de la colección ordenados en la etapa previa. Para ello se ha usado la variable del sistema **\$\$CURRENT** que devuelve el documento que se está procesando.

Query15 = con la etapa **\$project**, se selecciona el primer y último elemento del Array de documentos, usando el operador **\$arrayElemAt**, quedando así sólo aquél cuyo campo **Alzheimer disease (G30)** es el máximo de la colección y el que tiene el valor mínimo.

```
var query13={"$sort": {"Alzheimer disease (G30)": 1}}
var query14={"$group": {"_id": null,"docs": {"$push": "$$CURRENT"}}}
var query15={"$project": {"max": {"$arrayElemAt": ["$docs",-1]},
                          "min": {"$arrayElemAt": ["$docs",0]}}}
var maxmin=query13,query14,query15
db.RPM.aggregate(maxmin)
```

_id	max	min
1 null	{23 fields}	{23 fields}

Al realizar la siguiente consulta se obtiene como resultados dos array con los datos agrupados, uno con el mayor número de muertes causadas por enfermedades relacionadas con Alzheimer el cual fueron 4.844 y otro con el menor número de muertes siendo 0. Como se observa en los dos siguientes recuadros:

```
{
  "_id": ObjectId("63a239c8d079766caf317c5a"),
  "Date Of Death Year": 2020,
  "Date Of Death Month": 12,
  "Start Date": ISODate("2020-12-01T01:00:00.000+01:00"),
  "End Date": ISODate("2020-12-31T01:00:00.000+01:00"),
  Sex: "Female",
  "Race/Ethnicity": "Non-Hispanic White",
  AgeGroup: "85 years and over",
  AllCause: 55227,
  NaturalCause: 53946,
  "Septicemia (A40-A41)": 393,
  "Malignant neoplasms (C00-C97)": 3644,
  "Diabetes mellitus (E10-E14)": 676,
  "Alzheimer disease (G30)": 4844,
  "Influenza and pneumonia (J09-J18)": 611,
  "Chronic lower respiratory diseases (J40-J47)": 1750,
  "Other diseases of respiratory system (J00-J06,J30-J39,J67,J70-J98)",
  "Nephritis, nephrotic syndrome and nephrosis (N00-N07,N17-N19,N25-N)",
  "Symptoms, signs and abnormal clinical and laboratory findings, not",
  "Diseases of heart (I00-I09,I11,I13,I20-I51)": 11093,
  "Cerebrovascular diseases (I60-I69)": 3483,
  "COVID-19 (U071, Multiple Cause of Death)": 15441,
  "COVID-19 (U071, Underlying Cause of Death)": 13510
}

(23 fields)

Press 'p' to show in JSON Viewer
```

```
{
  "_id": ObjectId("63a239c8d079766caf317160"),
  "Date Of Death Year": 2019,
  "Date Of Death Month": 7,
  "Start Date": ISODate("2019-07-01T02:00:00.000+02:00"),
  "End Date": ISODate("2019-07-31T02:00:00.000+02:00"),
  Sex: "Male",
  "Race/Ethnicity": "Other",
  AgeGroup: "0-4 years",
  AllCause: 61,
  NaturalCause: 52,
  "Septicemia (A40-A41)": 0,
  "Malignant neoplasms (C00-C97)": 1,
  "Diabetes mellitus (E10-E14)": 0,
  "Alzheimer disease (G30)": 0,
  "Influenza and pneumonia (J09-J18)": 1,
  "Chronic lower respiratory diseases (J40-J47)": 0,
  "Other diseases of respiratory system (J00-J06,J30-J39,J67,J70-J98)",
  "Nephritis, nephrotic syndrome and nephrosis (N00-N07,N17-N19,N25-N)",
  "Symptoms, signs and abnormal clinical and laboratory findings, not",
  "Diseases of heart (I00-I09,I11,I13,I20-I51)": 0,
  "Cerebrovascular diseases (I60-I69)": 1,
  "COVID-19 (U071, Multiple Cause of Death)": 0,
  "COVID-19 (U071, Underlying Cause of Death)": 0
}

(23 fields)

Press 'p' to show in JSON Viewer
```

En base a las consultas anteriores se centrar la investigación en las enfermedades con mayor índice de muerte en los años, además de esto, consultar de manera específica en cuanto a hombres y mujeres muertos. Luego se verá el año en el que hubo más muertes y se realizará un estudio mensual para ver el volumen de muerte por mes. Finalizado, con un análisis estadístico en gráficos de las variables usadas con MongoDB, para así obtener gráficos con una mayor potencia visual de nuestros datos. por lo tanto, se comenzará con una serie de preguntas:

¿Cuál fue la enfermedad que genero un mayor índice de muerte en los hombres?

En la **query16** se aplica un filtro usando el **\$match** en el que se observen todos los documentos relacionados con el año **2019** del campo **"Date Of Death Year"**, en la **query17** se aplica el mismo filtro, pero esta vez se quiere filtrar todos los documentos relacionados con **Male** en el campo **"Sex"**. Luego en **agrupa** se utiliza el operador **\$group** por un campo **"_id"** establecido a **Año 2019** y se crean un campo llamado **Total_Death_1** hasta **Total_Death_13** en el que se realiza la suma (**\$sum**) de cada campo de enfermedades causantes de muertes, donde se añaden todos los documentos de la colección ordenados

en la etapa previa. Se utiliza el operador *\$project* para no mostrar (0) el campo de "Total_Death_1" ya que se quieren descartar las muertes por causa natural. Para esto se realiza la siguiente consulta:

```
var query16 = { $match: {"Date Of Death Year":2019 }}
var query17 = { $match: {"Sex":"Male" }}
var agrupa = { $group: { _id:"Año 2019",Total_Death_1:{$sum:"$NaturalCause"},
Total_Death_2:{$sum:"$Septicemia (A40-A41)"},
Total_Death_3:{$sum:"$Diabetes mellitus (E10-E14)"},
Total_Death_4: { $sum: "$Alzheimer disease (G30)" },
Total_Death_5: { $sum: "$Influenza and pneumonia (J09-J18)" },
Total_Death_6: { $sum: "$Chronic lower respiratory diseases (J40-J47)" },
Total_Death_7: { $sum: "$Other diseases of respiratory system (J00-J06,J30-J39,J67,J70-J98)" },
Total_Death_8: { $sum: "$Nephritis, nephrotic syndrome and nephrosis (N00-N07,N17-N19,N25-N27)" },
Total_Death_9: { $sum: "$Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified (R00-R99)" },
Total_Death_10: { $sum: "$Diseases of heart (I00-I09,I11,I13,I20-I51)" },
Total_Death_11: { $sum: "$Cerebrovascular diseases (I60-I69)" },
Total_Death_12: { $sum: "$COVID-19 (U071, Multiple Cause of Death)" },
Total_Death_13: { $sum: "$COVID-19 (U071, Underlying Cause of Death)" }}}
var proyecta= { "$project": { "Total_Death_1": 0, } }
var etapas=[ query16,query17,agrupa,proyecta ]
db.RPM.aggregate(etapas)
```

Obteniendo así el siguiente resultado:

_id	Total_Death_2	Total_Death_3	Total_Death_4	Total_Death_5	Total_Death_6	Total_Death_7	Total_Death_8	Total_Death_9	Total_Death_10	Total_Death_11	Total_Death_12	Total_Death_13
1 Año 2019	18.866 (18.9K)	49.512 (49.5K)	37.983 (38.0K)	24.643 (24.6K)	73.721 (73.7K)	22.577 (22.6K)	26.669 (26.7K)	15.661 (15.7K)	357.729 (0.36M)	64.344 (64.3K)	0	0

Siendo el **Total_Death_10 (Enfermedades del corazón)** con un mayor índice de muertes (**357.729**) en el 2019 para los hombres.

Referente a la anterior consulta, se puede calcular el total de muertos que dejo cada enfermedad a medida de los años, si no usamos el filtro *\$match* en el año 2019 del campo "Date Of Death Year" y agrupamos utilizando el operador *\$group* por un campo "_id" establecido a en "\$Date Of Death Year" como se muestra en la siguiente consulta:

```
var query18 = { $match: {"Sex":"Male" }}
var agrupa1 = { $group: { _id:"$Date Of Death Year", Total_Death_1:{$sum:"$NaturalCause"},
Total_Death_2:{$sum:"$Septicemia (A40-A41)"},
Total_Death_3:{$sum:"$Diabetes mellitus (E10-E14)"},
Total_Death_4: { $sum: "$Alzheimer disease (G30)" },
Total_Death_5: { $sum: "$Influenza and pneumonia (J09-J18)" },
Total_Death_6: { $sum: "$Chronic lower respiratory diseases (J40-J47)" },
Total_Death_7: { $sum: "$Other diseases of respiratory system (J00-J06,J30-J39,J67,J70-J98)" },
Total_Death_8: { $sum: "$Nephritis, nephrotic syndrome and nephrosis (N00-N07,N17-N19,N25-N27)" },
Total_Death_9: { $sum: "$Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified (R00-R99)" },
Total_Death_10: { $sum: "$Diseases of heart (I00-I09,I11,I13,I20-I51)" },
Total_Death_11: { $sum: "$Cerebrovascular diseases (I60-I69)" },
Total_Death_12: { $sum: "$COVID-19 (U071, Multiple Cause of Death)" },
Total_Death_13: { $sum: "$COVID-19 (U071, Underlying Cause of Death)" }}}
var proyecta= { "$project": { "_id" : 1, "Total_Death_1": 0, } }
var etapas=[ query18,agrupa1, proyecta ]
db.RPM.aggregate(etapas).sort({_id:1})
```

Obtendremos que:

_id	Total_Death_2	Total_Death_3	Total_Death_4	Total_Death_5	Total_Death_6	Total_Death_7	Total_Death_8	Total_Death_9	Total_Death_10	Total_Death_11	Total_Death_12	Total_Death_13
1 2019	18.866 (18.9K)	49.512 (49.5K)	37.983 (38.0K)	24.643 (24.6K)	73.721 (73.7K)	22.577 (22.6K)	26.669 (26.7K)	15.661 (15.7K)	357.729 (0.36M)	64.344 (64.3K)	0	0
2 2020	19.626 (19.6K)	57.604 (57.6K)	41.290 (41.3K)	27.828 (27.8K)	73.018 (73.0K)	23.448 (23.4K)	27.328 (27.3K)	16.699 (16.7K)	383.666 (0.38M)	69.789 (69.8K)	209.252 (0.21M)	193.006 (0.19M)
3 2021	14.310 (14.3K)	40.819 (40.8K)	26.555 (26.6K)	15.302 (15.3K)	47.655 (47.7K)	17.040 (17.0K)	20.190 (20.2K)	39.648 (39.6K)	268.948 (0.27M)	50.557 (50.6K)	181.657 (0.18M)	165.549 (0.17M)

Para el año 2020 en **Total_Death_10 (enfermedades del corazón)** se obtuvo el mayor índice de muertes dando un valor total de **383.666**, obviando las muertes naturales.

Podemos aplicar la consulta para saber en qué año hubo la mayor y menor cantidad de mujeres muertas por enfermedades del corazón en su totalidad en un array de todas las enfermedades como se observa en la siguiente consulta:

```

var query19= { $match: { "Sex": "Female" } }
var agrupa2 = { $group: { _id: "$Date Of Death Year", Total_Death_1: { $sum: "$NaturalCause" },
Total_Death_2: { $sum: "$Septicemia (A40-A41)" },
Total_Death_3: { $sum: "$Diabetes mellitus (E10-E14)" },
Total_Death_4: { $sum: "$Alzheimer disease (G30)" },
Total_Death_5: { $sum: "$Influenza and pneumonia (J09-J18)" },
Total_Death_6: { $sum: "$Chronic lower respiratory diseases (J40-J47)" },
Total_Death_7: { $sum: "$Other diseases of respiratory system (J00-J06, J30-J39, J67, J70-J98)" },
Total_Death_8: { $sum: "$Nephritis, nephrotic syndrome and nephrosis (N00-N07, N17-N19, N25-N27)" },
Total_Death_9: { $sum: "$Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified (R00-R99)" },
Total_Death_10: { $sum: "$Diseases of heart (I00-I09, I11, I13, I20-I51)" },
Total_Death_11: { $sum: "$Cerebrovascular diseases (I60-I69)" },
Total_Death_12: { $sum: "$COVID-19 (U071, Multiple Cause of Death)" },
Total_Death_13: { $sum: "$COVID-19 (U071, Underlying Cause of Death)" } } }
var proyecta= { "$project": { _id: 1 } }
var query20= { "$sort": { "Total_Death_10": 1 } }
var query21= { "$group": { "_id": null, "docs": { "$push": "$$CURRENT" } } }
var query22= { "$project": { "max": { "$arrayElemAt": [ "$docs", -1 ] }, "min": { "$arrayElemAt": [ "$docs", 0 ] } } }
var maxmin=[query19, agrupa2, query20, query21, query22]
db.RPM.aggregate(maxmin)

```

En la **query19** se aplica un filtro usando el **\$match** en el que se observan todos los documentos relacionados con **Female** en el campo **"Sex"**. Luego en **agrupa** se utiliza el operador **\$group** por un campo **"_id"** establecido a **Años (2019-2020-2021)**, y se crean un campo llamado **Total_Death_1** hasta **Total_Death_13** realizando la suma (**\$sum**) de los datos de muertes en cada campo de donde se añaden todos los documentos de la colección que fueron filtrados en la etapa anterior, para saber cuál fue el total de mujeres muertas en relación con cada enfermedad en diversos años. Se utiliza el operador **\$project** para no mostrar (0) el campo de **"Total_Death_1"** ya que se quieren descartar las muertes por causa natural. *(Se sabe de antemano que la enfermedad con mayor índice de muertes para las mujeres son las enfermedades del corazón)*

En la **query20** se utiliza el operador **\$sort** para ordenar los documentos de forma ascendente en base a las muertes causadas por enfermedades del corazón, en conexión con la anterior **query** se han agrupado por un campo **_id** establecido a **null**, y se ha creado un campo llamado **docs** donde he añadido todos los documentos de la colección ordenados en la etapa previa. Para ello se ha usado la variable del sistema **\$\$CURRENT** que devuelve el documento que se está procesando. Por último, en la etapa **\$project**, se selecciona el primer y último elemento del Array de documentos, usando el operador **\$arrayElemAt**, quedando así sólo aquél cuyo campo **Total_Death_10** es el máximo de la colección y el que tiene el valor mínimo.

_id	max	min
1 null	{ "_id": 2020 } (14 fields)	{ "_id": 2021 } (14 fields)

Al realizar la siguiente consulta nos da como resultados dos array con los datos agrupados por el mayor número de muertes causadas por enfermedades relacionadas con enfermedades del corazón el cual fue **314.534** en el **2020** y el menor número de muertes siendo **218.919** para el año **2021**. Como se observa en los dos siguientes recuadros:

```

{
  _id: 2020,
  Total_Death_1: 1528961,
  Total_Death_2: 28497,
  Total_Death_3: 44691,
  Total_Death_4: 92991,
  Total_Death_5: 25835,
  Total_Death_6: 79758,
  Total_Death_7: 21729,
  Total_Death_8: 25271,
  Total_Death_9: 17524,
  Total_Death_10: 314534,
  Total_Death_11: 98725,
  Total_Death_12: 176885,
  Total_Death_13: 158493
}
(14 fields)

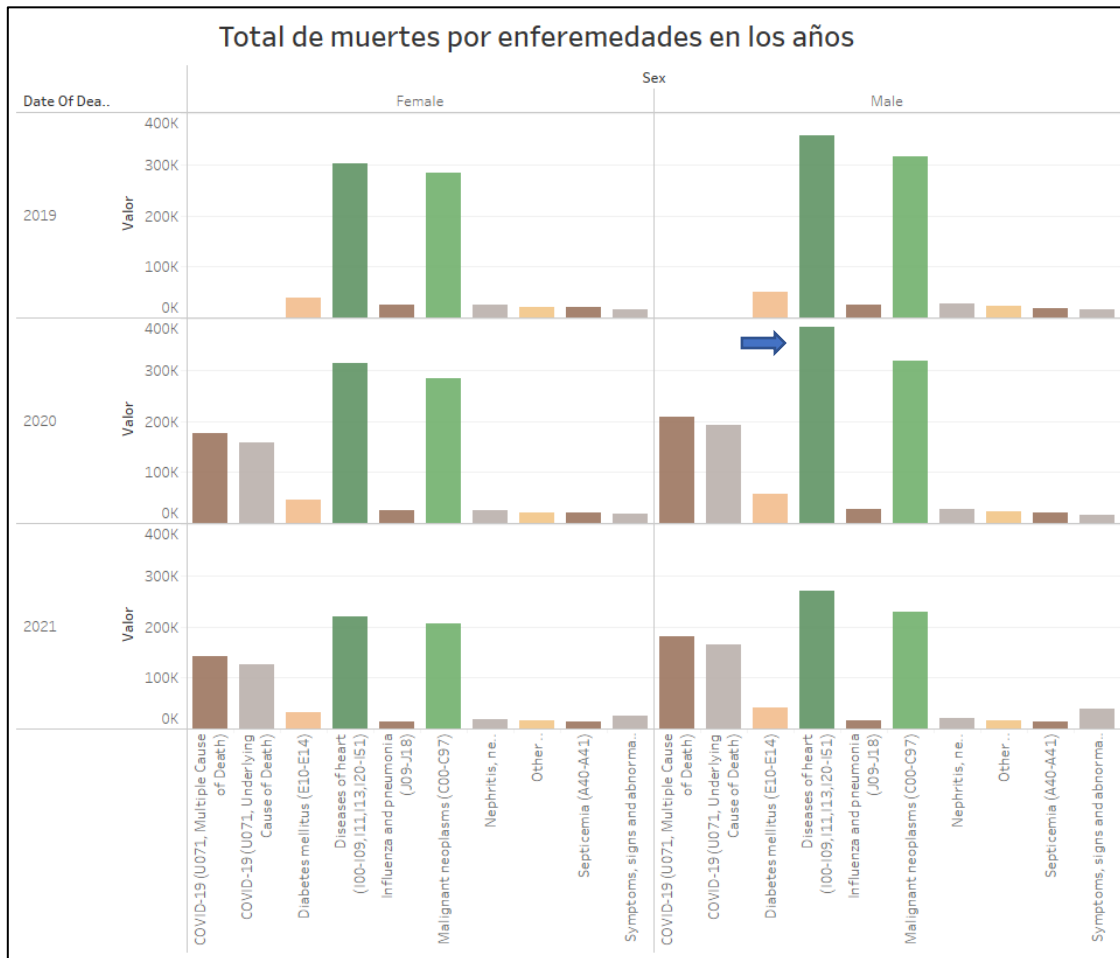
```

```

{
  _id: 2021,
  Total_Death_1: 1895554,
  Total_Death_2: 14349,
  Total_Death_3: 31484,
  Total_Death_4: 59198,
  Total_Death_5: 13315,
  Total_Death_6: 52881,
  Total_Death_7: 15434,
  Total_Death_8: 18374,
  Total_Death_9: 24693,
  Total_Death_10: 218919,
  Total_Death_11: 65431,
  Total_Death_12: 142648,
  Total_Death_13: 126388
}
(14 fields)

```

Para obtener un mejor análisis de las enfermedades que más muertos dejaron al pasar de los años, se exportan los datos a Tableau y se realiza un gráfico de barras:



Al observar las antiguas consultas realizadas en mongoDB y el grafico creado en conexión a estas en tableau, se decide encaminar las siguientes consultas en las muertes causadas por problemas del corazón a los hombres en el año 2020. Realizando la siguiente pregunta:

¿Cuál fue la etnia con más afectada por el grupo de enfermedades cardiacas en estados unidos en el año 2020?

var filtro = Se filtra todos los documentos por el campo **"Date Of Death Year"** que estén en el año 2020 usando el operador **"\$match"**

var fasegroup = Se separa los documentos en grupos según una **"clave de grupo"** que tenga en este caso **"Race/Ethnicity"** con el operador **\$group** y con el operador **\$sum** se realiza una suma de todos los documentos en base a **"\$Diseases of heart"** por cada raza o etnia.

con el **"Aggregate"** se para realizar diversas operaciones en los datos agrupados devolviendo un solo resultado. Se utiliza el **"sort"** para ordenar de mayor a menor y el **"limit"** para obtener solo un valor.

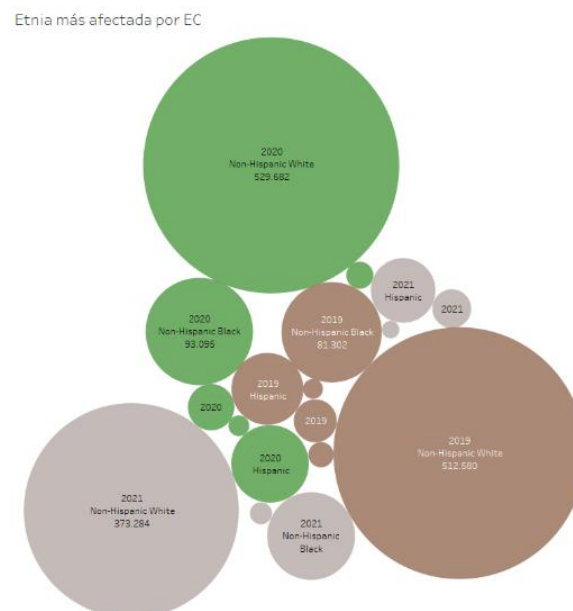
```
var filtro = {$match:{"Date Of Death Year": 2020}}
var grouprace= { "_id": "$Race/Ethnicity",
Total_Death_10:{$sum:"$Diseases of heart (I00-I09,I11,I13,I20-I51)" } }
var fasegroup= { $group: grouprace }
var etapas=[filtro, fasegroup ]
db.RPM.aggregate(etapas).sort({Total_Death_10:-1}).limit(1)
```

_id	Total_Death_10
1 Non-Hispanic White	529.682 (0.53M)

En los primeros pasos se elimino la casilla en la cual se sabe que todos los datos son de estados unidos, recordando esto se utiliza esta consulta para saber cuál fue la raza o etnia en la cual hubo más personas muertas por enfermedades cardiacas dando resultado de 529.682, no obstante, también podemos ordenar las razas de forma decreciente para saber que otros valores tienen las otras razas si quitamos el operador **“limit”** obteniendo los siguientes valores:

	_id	Total_Death_10
1	Non-Hispanic White	529.682 (0.53M)
2	Non-Hispanic Black	93.095 (93.1K)
3	Hispanic	48.607 (48.6K)
4	Non-Hispanic Asian	17.358 (17.4K)
5	Other	5945 (5.9K)
6	Non-Hispanic American Ind	3513 (3.5K)

Una forma visual de observar los resultados es uniéndolos con los de los años 2019 y 2021, exportando los datos a Tableau y se realizando un gráfico de círculos:



¿En qué rango de edad tuvo un mayor número de muertes por el grupo de enfermedades cardiacas en el año 2020?

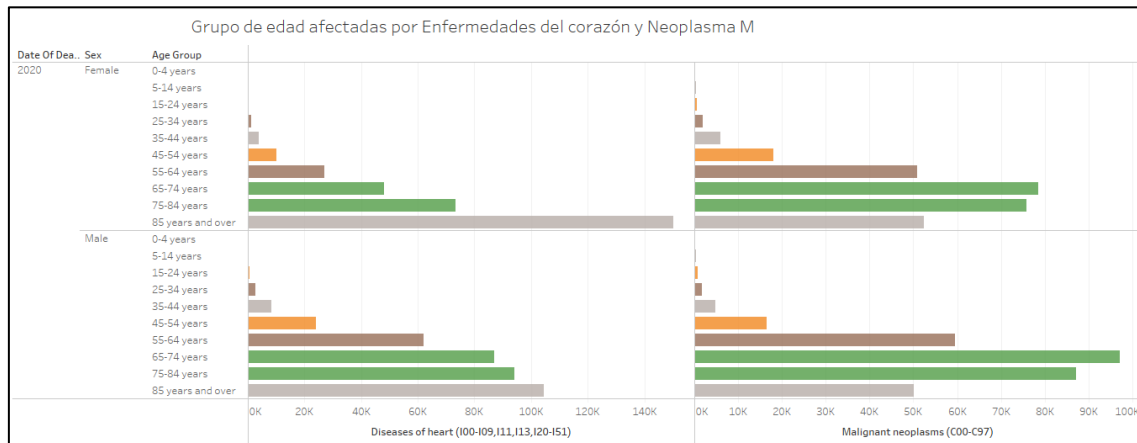
Si utilizamos el mismo operador anterior, pero cambiando la **clave de grupo**, **“\$Race/Ethnicity”** por **“\$AgeGroup”** obtendremos los siguientes resultados:

```
var filtro = {$match:{"Date Of Death Year": 2020}}
var groupce= { "_id": "$AgeGroup",
Total_Death_10:{$sum:{"$Diseases of heart (I00-I09,I11,I13,I20-I51)"}} }
var fasegroup= { $group: groupce }
var orden= {$sort:{"Total_Death_10":-1}}
var etapas=[filtro, fasegroup, orden ]
db.RPM.aggregate(etapas).limit(1)
```



	_id	Total_Death_10
1	85 years and over	254.650 (0.25M)
2	75-84 years	167.580 (0.17M)
3	65-74 years	135.191 (0.14M)
4	55-64 years	88.868 (88.9K)
5	45-54 years	34.299 (34.3K)
6	35-44 years	12.212 (12.2K)
7	25-34 years	4003 (4.0K)
8	15-24 years	873
9	0-4 years	356
10	5-14 years	168

Al graficar estos resultados y compararlos con la segunda enfermedad que genero mayor índice de muerte (Malignant neoplasms), se obtienen los siguientes resultados:



El grupo de edad con mayor índice de muertes en Diseases of heart son las personas en un rango de edad de 85 años o mayor. Por el contrario, en las muertes por Malignant neoplasms se observa que el rango de edad es entre 65-74 años.

En base a lo anterior, se quiere que actualizar la base de datos para los **hombres (Sex)** que tienen un problema del corazón de **85 años o más (AgeGroup)** y que sean **no-hispanicos blancos (Race/Ethnicity)** de todo el año **2020 (Date Of Death Year)**, añadiendo un campo en la cual se especifique una causa más exacta de muerte. Además, de saber si estas personas tenían o no una vida saludable, los datos se escogerán aplicando una serie de filtros con el **\$match** en base a los campos anteriormente mencionados:

```
var filtro = {$match:{"Date Of Death Year": 2020}}
var filtro1 = {$match:{"Sex":"Male"}}
var filtro2 = {$match:{"Race/Ethnicity": "Non-Hispanic White"}}
var filtro3 = {$match:{"AgeGroup": "85 years and over"}}
var proyecto = { "$project": { "_id": 1, "Start Date":0, "End Date":0 } }
var etapas=[filtro, filtro1,filtro2, filtro3, proyecto]
db.RPM.aggregate(etapas)
```



	_id	Date Of Di	Date Of D	Sex	Race/Ethnicity	AgeGroup	AllCause	NaturalCa	Septicemi	Malignant	Diabetes	Alzheimer	Influenza	Chronic lo	Other dise	Nephritis	Symptoms	Diseases	Cerebrova	COVID-19
1	63a239c8d07976caf3	2020	1	Male	Non-Hispanic White	85 years and over	26.452 (26.25, 26.521)	26.306	3718 (3.7, 513)	1633 (1.6, 764)	1519 (1.5, 496)	587	277	7982 (8.0, 1566)	1.6, 0	0				
2	63a239c8d07976caf3	2020	2	Male	Non-Hispanic White	85 years and over	24.619 (24.23, 24.992)	24.292	3358 (3.4, 521)	1489 (1.5, 751)	1314 (1.3, 427)	563	264	7606 (7.6, 1459)	1.5, 1	1				
3	63a239c8d07976caf3	2020	3	Male	Non-Hispanic White	85 years and over	26.641 (26.25, 27.03)	26.299	3594 (3.6, 529)	1663 (1.7, 880)	1388 (1.4, 445)	574	306	7897 (7.9, 1542)	1.5, 569	541				
4	63a239c8d07976caf3	2020	4	Male	Non-Hispanic White	85 years and over	30.815 (30.29, 31.34)	30.272	3292 (3.3, 562)	1692 (1.7, 656)	1304 (1.3, 431)	523	289	7776 (7.8, 1512)	1.5, 5851	5.9, 5513	5.5, 5			
5	63a239c8d07976caf3	2020	5	Male	Non-Hispanic White	85 years and over	26.023 (26.25, 25.798)	25.195	3279 (3.3, 473)	1531 (1.5, 448)	1116 (1.1, 344)	495	256	7082 (7.1, 1369)	1.4, 3605	3.6, 3281	3.3, 3			
6	63a239c8d07976caf3	2020	6	Male	Non-Hispanic White	85 years and over	22.805 (22.21, 23.39)	22.240	3169 (3.2, 477)	1434 (1.4, 384)	1018 (1.0, 327)	447	208	6629 (6.6, 1335)	1.3, 1441	1.4, 1241	1.2, 1			
7	63a239c8d07976caf3	2020	7	Male	Non-Hispanic White	85 years and over	24.704 (24.23, 25.17)	24.237	3455 (3.5, 478)	1549 (1.5, 394)	1079 (1.1, 316)	462	273	6918 (6.9, 1363)	1.4, 2184	2.2, 1929	1.9, 1			
8	63a239c8d07976caf3	2020	8	Male	Non-Hispanic White	85 years and over	24.886 (24.23, 25.13)	24.248	3534 (3.5, 523)	1502 (1.5, 372)	1019 (1.0, 339)	459	255	6837 (6.8, 1438)	1.4, 2300	2.3, 2051	2.1, 1			
9	63a239c8d07976caf3	2020	9	Male	Non-Hispanic White	85 years and over	24.238 (24.23, 24.24)	24.245	3538 (3.5, 529)	1499 (1.5, 398)	1082 (1.1, 326)	455	242	6731 (6.7, 1442)	1.4, 1732	1.7, 1487	1.5, 1			
10	63a239c8d07976caf3	2020	10	Male	Non-Hispanic White	85 years and over	26.766 (26.25, 27.27)	26.265	3621 (3.6, 531)	1553 (1.6, 402)	1102 (1.1, 362)	505	298	7180 (7.2, 1441)	1.4, 2939	2.9, 2603	2.6, 1			
11	63a239c8d07976caf3	2020	11	Male	Non-Hispanic White	85 years and over	31.230 (31.30, 31.16)	31.280	3653 (3.7, 589)	1576 (1.6, 475)	1165 (1.2, 379)	479	288	7632 (7.6, 1480)	1.5, 6825	6.8, 6151	6.2, 1			
12	63a239c8d07976caf3	2020	12	Male	Non-Hispanic White	85 years and over	37.815 (37.36, 38.27)	37.278	3655 (3.7, 634)	1877 (1.9, 581)	1243 (1.2, 439)	533	295	8442 (8.4, 1749)	1.7, 11,550	11, 10,444	11, 10			

Se utilizo el Project para ocultar los campos que no utilizaremos y tener una mejor visión de los documentos, con los resultados de la anterior tabla podemos saber a cuál “_id” le añadiremos la información para crear un campo nuevo el cual se llamará “HeartL” este contendrá una numeración específica del 1 al 10, lo cual nos servirá luego para adjuntar valores desde otra colección.


```

db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf317771")},{ $set:{"HeartL":7}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf3177e9")},{ $set:{"HeartL":6}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf31785e")},{ $set:{"HeartL":8}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf3178d6")},{ $set:{"HeartL":9}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf31794e")},{ $set:{"HeartL":4}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf3179c6")},{ $set:{"HeartL":1}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf317a3e")},{ $set:{"HeartL":3}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf317ab6")},{ $set:{"HeartL":2}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf317b2e")},{ $set:{"HeartL":5}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf317ba6")},{ $set:{"HeartL":5}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf317c1e")},{ $set:{"HeartL":10}})
db.RPM.updateMany({"_id" : ObjectId("63a239c8d079766caf317c96")},{ $set:{"HeartL":6}})

```



```

1- {
2  "acknowledged" : true,
3  "matchedCount" : 1,
4  "modifiedCount" : 1
5 }

```

Se modifica cada elemento uno a uno en el cual se actualiza (**\$set**) el documento añadiendo el campo **"HeartL"** en documentos específicos como se había explicado anteriormente con los filtros (**\$match**).

Se comprueba los documentos que fueron modificados:



_id	Date Of Di	Date Of Di	Sex	Race/Ethnicity	Age Group	AlcCause	NaturalCa	Septicmi	Malignant	Diabetes	r Alzheimer	Influenza	Chronic lo	Other dise	Nephritis	Symptoms	Diseases	Cerebrova	COVID-19	COVID-19	HeartL
63a239c8d079766caf3	2020	1	Male	Non-Hispanic White	85 years and over	26.452 (26.25.521 (21.306	3718 (3.79.513	1633 (1.69.764	1519 (1.59.496	587	277	7982 (8.09.1566 (1.69.9	9	7							
63a239c8d079766caf3	2020	2	Male	Non-Hispanic White	85 years and over	24.619 (24.23.699 (21.292	3359 (3.49.521	1499 (1.59.751	1314 (1.39.427	563	264	7606 (7.69.1459 (1.59.1	1	6							
63a239c8d079766caf3	2020	3	Male	Non-Hispanic White	85 years and over	26.941 (26.25.713 (21.299	3594 (3.69.529	1663 (1.79.880	1399 (1.49.445	574	306	7897 (7.99.1542 (1.59.569	541	8							
63a239c8d079766caf3	2020	4	Male	Non-Hispanic White	85 years and over	30.815 (30.29.993 (30.272	3292 (3.39.562	1692 (1.79.656	1304 (1.39.431	523	289	7776 (7.89.1512 (1.59.5851	5.99.5513 (5.59.9								
63a239c8d079766caf3	2020	5	Male	Non-Hispanic White	85 years and over	26.023 (26.25.098 (21.195	3279 (3.39.473	1531 (1.59.448	1116 (1.19.344	495	256	7062 (7.19.1369 (1.49.3605	3.99.3281 (3.39.4								
63a239c8d079766caf3	2020	6	Male	Non-Hispanic White	85 years and over	22.805 (22.21.949 (21.240	3169 (3.29.477	1434 (1.49.384	1018 (1.09.327	447	208	6629 (6.69.1335 (1.39.1441	1.49.1241 (1.29.1								
63a239c8d079766caf3	2020	7	Male	Non-Hispanic White	85 years and over	24.704 (24.23.800 (21.237	3455 (3.59.478	1549 (1.59.394	1079 (1.19.316	492	273	6918 (6.99.1363 (1.49.2184	2.29.1929 (1.99.3								
63a239c8d079766caf3	2020	8	Male	Non-Hispanic White	85 years and over	24.966 (24.23.938 (21.248	3534 (3.59.523	1502 (1.59.372	1019 (1.09.339	459	255	6837 (6.89.1438 (1.49.2300	2.39.2051 (2.19.2								
63a239c8d079766caf3	2020	9	Male	Non-Hispanic White	85 years and over	24.238 (24.23.345 (21.245	3538 (3.59.529	1499 (1.59.398	1082 (1.19.326	455	242	6731 (6.79.1442 (1.49.1732	1.79.1487 (1.59.5								
63a239c8d079766caf3	2020	10	Male	Non-Hispanic White	85 years and over	26.766 (26.25.772 (21.265	3621 (3.69.531	1553 (1.69.402	1102 (1.19.362	505	298	7180 (7.29.1441 (1.49.2939	2.99.2603 (2.69.5								
63a239c8d079766caf3	2020	11	Male	Non-Hispanic White	85 years and over	31.230 (31.30.241 (30.280	3653 (3.79.589	1576 (1.69.475	1165 (1.29.379	479	288	7632 (7.69.1480 (1.59.6825	6.89.6151 (6.29.10								
63a239c8d079766caf3	2020	12	Male	Non-Hispanic White	85 years and over	37.819 (37.36.743 (36.278	3655 (3.79.634	1877 (1.99.581	1243 (1.29.439	533	295	8442 (8.49.1749 (1.79.11.5591	11.10.444 (11.6								

se crea la siguiente colección agregando las causas específica de muertes por enfermedades del corazón insertando un array de documentos:

var mv = en el array se describe el **"_id"** del 1 al 10 el cual esta clasificado por causas de muerte relacionadas con afecciones del corazón.

Al insertar (**"insert(mv)"**) los anteriores documentos automáticamente se crea una nueva colección la cual se llama **"otherdisease"**, con el operador **find** se muestra los datos introducidos.

```

var mv = [
  {"_id" : 1,"cause" : "heart failure","lifestyle":"active"},
  {"_id" : 2,"cause" : "arterial hypertension","lifestyle":"active"},
  {"_id" : 3,"cause" : "excess cholesterol", "lifestyle":"active"},
  {"_id" : 4,"cause" : "heart attack", "lifestyle":"active"},
  {"_id" : 5,"cause" : "angina pectoris", "lifestyle":"active"},
  {"_id" : 6,"cause" : "heart failure","lifestyle":"low"},
  {"_id" : 7,"cause" : "arterial hypertension","lifestyle":"low"},
  {"_id" : 8,"cause" : "excess cholesterol", "lifestyle":"low"},
  {"_id" : 9,"cause" : "heart attack", "lifestyle":"low"},
  {"_id" : 10,"cause" : "angina pectoris", "lifestyle":"low"}]

db.otherdisease.insert(mv)
db.otherdisease.find().limit(100)

```

Al insertar los siguientes documentos se crean dos campos en una nueva colección, el primer campo se llama **"cause"** el cual nos dice una de las causas específicas dentro de las muertes por enfermedades del corazón, el segundo campo **"lifestyle"** nos habla del tipo del estilo de vida que tenía la persona si esta tenía una vida activa (buena actividad física) o una actividad baja (baja actividad física):

otherdisease 0.049 s 10 Docs		
_id	cause	lifestyle
1	heart failure	active
2	arterial hypertension	active
3	excess cholesterol	active
4	heart attack	active
5	angina pectoris	active
6	heart failure	low
7	arterial hypertension	low
8	excess cholesterol	low
9	heart attack	low
10	angina pectoris	low

Para añadir una colección dentro de otra se hará uso de operador **\$lookup** los filtros que se aplican son los anteriormente usados en los cuales se filtra por año de muerte 2020, género masculino, etnia de blanco no hispano y personas con una edad de 85 o más:

var union = se creará una carpeta la que se ha llamado **Mix Heart** en la cual los parámetros del 1 al 10 se introducen arrays de documentos en **mix heart** que dependen de la conexión que tengan las dos colecciones (*es una forma de hacer un inner join*).

var desuni = se utiliza un **\$unwind** para separar cada array y dejarlo como dos campos separados y no unidos por un array.

var proyección = utilizamos un **\$project** para que de esta forma no se muestren los **_id** tanto de la colección **RPM** como los de **"Mix Heart_id"** además de mostrar los datos que realmente queremos analizar, la final aplicamos un **sort** para que nos ordene de manera decreciente **(-1)** las muertes causadas por enfermedades del corazón y así poder saber el mes de muerte, la causa y el estilo de vida que llevan las personas con mayor índice de muerte. Con el **limit** sabremos un límite de las tres enfermedades

```
var filtro1 = { $match: { "Date Of Death Year": 2020 } }
var filtro2 = { $match: { "Sex": "Male" } }
var filtro3 = { $match: { "Race/Ethnicity": "Non-Hispanic White" } }
var filtro4 = { $match: { "AgeGroup": "85 years and over" } }
var union = { $lookup: { from: "otherdisease", localField: "HeartL",
foreignField: "_id", as: "Mix Heart" } }
var desuni = { $unwind: "$Mix Heart" }
var proyeccion = { $project: { "_id": 0, "Date Of Death Year" : 1,
"Date Of Death Month" : 1, "Sex" : 1, "Race/Ethnicity" : 1, "AgeGroup" : 1, "AllCause" : 1,
"Diseases of heart (I00-I09,I11,I13,I20-I51)": 1, "Mix Heart" : { "cause" : 1, "lifestyle" : 1 } } }
var etapas = [ filtro1, filtro2, filtro3, filtro4, union, desuni, proyeccion ]
db.RPM.aggregate(etapas).sort({ "Diseases of heart (I00-I09,I11,I13,I20-I51)": -1 }).limit(3)
```



	Date Of D:	Date Of D:	Sex	Race/Ethnicity	AgeGroup	AllCause	Diseases	Mix Heart
							cause	lifestyle
1	2020	12	Male	Non-Hispanic White	85 years and over	37.815 (3)	8442 (8.4k heart failure	low
2	2020	1	Male	Non-Hispanic White	85 years and over	26.452 (2)	7982 (8.0k arterial hypertension	low
3	2020	3	Male	Non-Hispanic White	85 years and over	26.641 (2)	7897 (7.9k excess cholesterol	low

Con esto se observa que en el mes de diciembre del **2020** se obtuvo un total de muertes por fallos del corazón de **8442**, en el mes 1 un total de muertes por hipertensión arterial de **7982** y en el mes de marzo un total de **7897**. Para dejar el **dataset** tal cual como estaba se mediante el operador (**\$unset**) eliminamos la columna de **HeartL** anteriormente creada:

```
var query5 = {}
var operacion2 = { $unset: { "HeartL": {} } }
db.RPM.updateMany(query5, operacion2)
db.RPM.find()
```



```
1 {
2   "acknowledged" : true,
3   "matchedCount" : 3960,
4   "modifiedCount" : 12
5 }
```

Conclusiones

En el estudio todos los campos de "Sex", "Race/Ethnicity", "Day Of The Death" están distribuidos (por campo) de manera equivalente por cada variable. El documento exacto en el que hubo mayores muertos por Alzheimer está en el año 2020, en el mes de diciembre para las personas de género femenino con una edad mayor o igual a 85, de etnia blanca no hispano teniendo una cantidad de 4.844 muertes. La enfermedad que genero un índice de muertos en los hombres (dejando de lado las causas por muerte natural) fueron las enfermedades cardiacas en el año 2020 con una cantidad de 383.666, la etnia más afectada por este mismo grupo de enfermedades fueron los blancos no hispanos con 529.682 muertos. además, el grupo de edad que fueron los mayores o iguales a 85, con un valore de 254.650 muertes. Al conectar la colección **otherdiese** con la colección **RPM** se observo que la causa mayor de muertes fue para las personas con un fallo en el corazón y un estilo de vida sedentario, siendo 8.442.