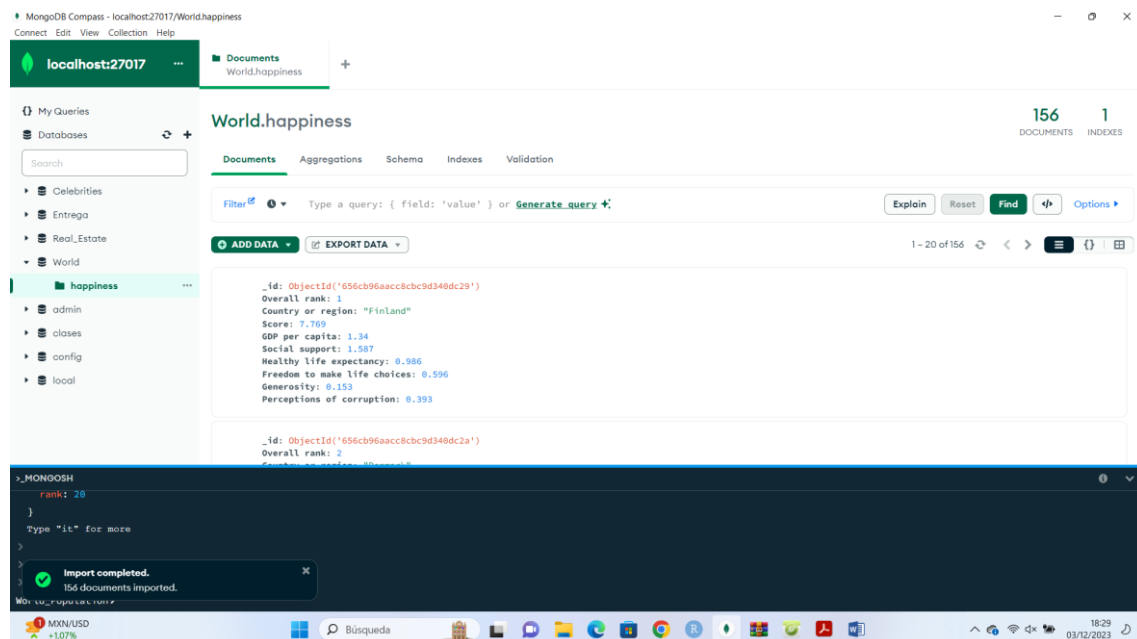


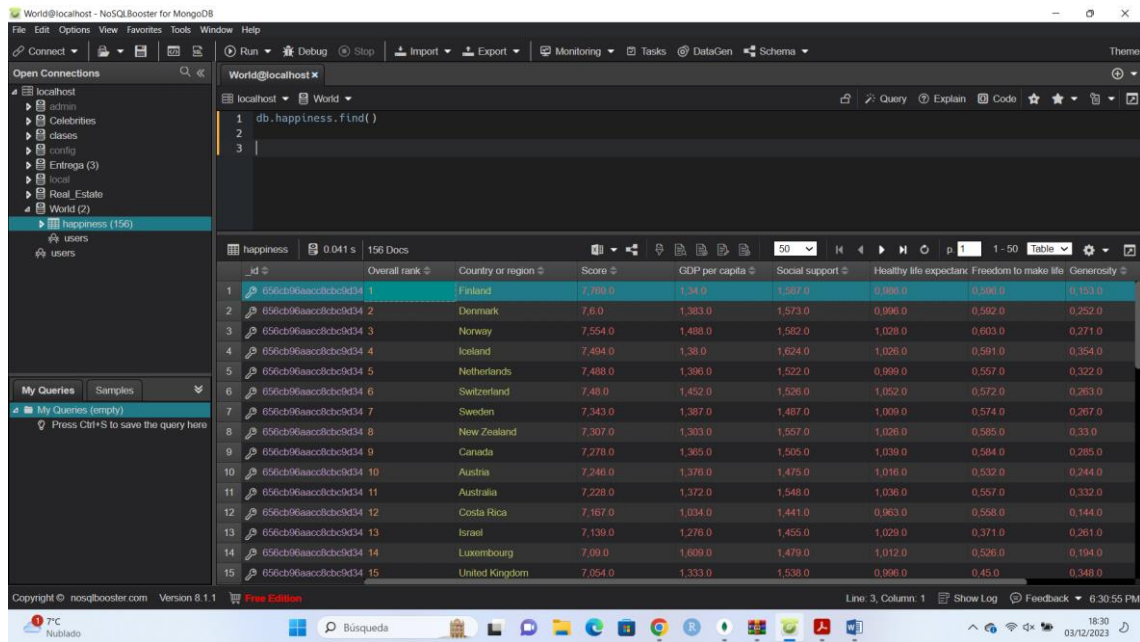
Mi dataset se centra en el ranking de los países más felices del mundo, evaluando diversos factores que contribuyen a la felicidad. La estructura de cada entrada en mi dataset es un documento JSON con varios campos relevantes.

Por ejemplo, cada país tiene un 'Overall rank' que indica su posición en el ranking, un nombre de país ('Country or region'), y puntajes específicos como 'Score' para la felicidad, 'GDP per capita' para el Producto Interno Bruto per cápita, 'Social support' para el apoyo social, 'Healthy life expectancy' para la esperanza de vida saludable, 'Freedom to make life choices' para la libertad de elección, 'Generosity' para la generosidad, y 'Perceptions of corruption' para la percepción de corrupción.

Cada uno de estos campos proporciona información valiosa para entender los factores que influyen en la felicidad de un país, y así se estructuran y representan los datos en mi dataset.

He cargado e importado el dataset en MongoDB siguiendo los pasos que a continuación muestro. La base de datos y la colección han sido creadas con éxito.





db.Felicidad.find()

Primero obtenemos el número de documentos para Felicidad

```
var numero_documentos = db.Felicidad.countDocuments();
```

```
print("Numero de documentos en la coleccion Felicidad: " + numero_documentos);
```

EJERCICIOS SOBRE PROYECCION:

PROYECTAMOS SOLO LOS CAMPOS “COUNTRY, REGION Y SCORE”

```
9 // PROYECTAMOS solo los campos "Country or region" y "Score"
10
11 db.Felicidad.find({}, { "Country or region": 1, "Score": 1, "_id": 0 });
12
13 // Proyectamos países con un puntaje mayor a 7, mostrando solo "Country or region" y "Score"
14 db.Felicidad.find({ "Score": { $gt: 7 } }, { "Country or region": 1, "Score": 1, "_id": 0 });
15
16
```

```
db.Felicidad.find({}, { "Country or region": 1, "Score": 1, "_id": 0 });
```

PROYECTAMOS PAISES CON UN PUNTAJE MAYOR A 7, MOSTRANDO SOLO COUNTRY, REGION Y SCORE.

```
db.Felicidad.find({ "Score": { $gt: 7 } }, { "Country or region": 1, "Score": 1, "_id": 0 });
```

EJERCICIOS SOBRE INSERCIÓN Y ACTUALIZACIÓN:

INSERTAMOS ALGUNOS DATOS FICTICIOS COMO INVENTADO E INVENTADO2

```
17
18
19 // INSERTAR (Insertamos algunos datos ficticios)
20
21 var inserts = [
22   { "Country": "Inventado", "Score": 7.5 },
23   { "Country": "Inventado2", "Score": 6.8, "GDP": 1.2, "SocialSupport": 1.4 }
24 ];
25 db.Felicidad.insertMany(inserts);
26
```

```
var inserts = [
  { "Country": "Inventado", "Score": 7.5 },
  { "Country": "Inventado2", "Score": 6.8, "GDP": 1.2, "SocialSupport": 1.4 }
];
db.Felicidad.insertMany(inserts);
```

ACTUALIZAMOS EL INDICE DE FELICIDAD Y EL GDP DE LOS PAISES INVENTADOS

```
28
29 // ACTUALIZAR (Actualizamos el indice de felicidad y el GDP de los paises inventados)
30
31 db.Felicidad.update({ "Country": "Inventado" }, { $set: { "Score": 6.0 } })
32 db.Felicidad.update({ "Country": "Inventado2" }, { $set: { "Score": 5, "GDP": 0.2 } }
33
34
```

```
db.Felicidad.update({ "Country": "Inventado" }, { $set: { "Score": 6.0 } })
db.Felicidad.update({ "Country": "Inventado2" }, { $set: { "Score": 5, "GDP": 0.2 } }
```

EJERCICIOS SOBRE AGREGACION, FILTRADO Y FECHAS:

ESTADISTICAS GLOBALES (calculamos la media, el mínimo y el máximo de Score y GDP)

```
34
35 // ESTADISTICAS GLOBALES (calculamos la media, el minimo y el maximo de Score y GDP)
36
37 var estadisticas_globales = [
38   {
39     $group: {
40       _id: null,
41       avgScore: { $avg: "$Score" },
42       minScore: { $min: "$Score" },
43       maxScore: { $max: "$Score" },
44       avgGDP: { $avg: "$GDP" },
45       minGDP: { $min: "$GDP" },
46       maxGDP: { $max: "$GDP" },
47     }
48   }
49 ];
50
51 db.Felicidad.aggregate(estadisticas_globales);
52
```

```
var estadisticas_globales = [
  {
    $group: {
      _id: null,
      avgScore: { $avg: "$Score" },
      minScore: { $min: "$Score" },
      maxScore: { $max: "$Score" },
      avgGDP: { $avg: "$GDP" },
      minGDP: { $min: "$GDP" },
      maxGDP: { $max: "$GDP" },
    }
  }
];
```

```
db.Felicidad.aggregate(estadisticas_globales);
```

OBTENEMOS EL PAIS MAS FELIZ

```
74
75 // Pais más feliz
76 db.Felicidad.find().sort({ "Score": -1 }).limit(1)
77
78 // Pais menos feliz
79 db.Felicidad.find().sort({ "Score": 1 }).limit(1)
80
```

```
db.Felicidad.find().sort({ "Score": -1 }).limit(1)
```

OBTENEMOS EL PAIS MENOS FELIZ

```
74
75 // País más feliz
76 db.Felicidad.find().sort({ "Score": -1 }).limit(1)
77
78 // País menos feliz
79 db.Felicidad.find().sort({ "Score": 1 }).limit(1)
80
```

db.Felicidad.find().sort({ "Score": 1 }).limit(1)

MOSTRAMOS LOS 5 PAISES CON MAYOR INDICE PER CAPITA GDP

```
59
60 // MOSTRAMOS LOS 5 PAISES CON MAYOR INDICE PER CAPITA GDP
61
62 var top5GDPPerCapita = [{ $sort: { "GDP per capita": -1 } }, { $limit: 5 }, { $project: { _id: 0, "Country or region": 1, "GDP per capita": 1 } }];
63 db.Felicidad.aggregate(top5GDPPerCapita);
64 |
65
```

var top5GDPPerCapita = [{ \$sort: { "GDP per capita": -1 } }, { \$limit: 5 }, { \$project: { _id: 0, "Country or region": 1, "GDP per capita": 1 } }];

db.Felicidad.aggregate(top5GDPPerCapita);

COMPARAMOS QATAR QUE ES EL PAIS QUE MAS GDP TIENE CON OTROS PAISES UTILIZANDO LA FUNCION FACET

```
85
86 // COMPARAMOS QATAR QUE ES EL QUE MAS GDP TIENE CON OTROS PAISES UTILIZANDO LA FUNCION FACET
87
88 var comparar_qatar = [
89   { $sort: { "GDP per capita": -1 } },
90   {
91     $facet: {
92       highestGDP: [{ $limit: 1 }, { $project: { _id: 0, country: "$Country or region", GDP: "$GDP per capita" } }],
93       otherCountries: [{ $skip: 1 }, { $project: { _id: 0, country: "$Country or region", GDP: "$GDP per capita" } }],
94     },
95   },
96 ];
97
98 db.Felicidad.aggregate(comparar_qatar);
99 |
100
```

var comparar_qatar = [

{ \$sort: { "GDP per capita": -1 } },

{

\$facet: {

highestGDP: [{ \$limit: 1 }, { \$project: { _id: 0, country: "\$Country or region", GDP: "\$GDP per capita" } }],

otherCountries: [{ \$skip: 1 }, { \$project: { _id: 0, country: "\$Country or region", GDP: "\$GDP per capita" } }],

},

```
},  
];
```

```
db.Felicidad.aggregate(comparar_qatar);
```

DISTRIBUIAMOS GENEROSIDAD ORDENADA DE PAIS MAS GENEROSO A MENOS GENEROSO,
CON SU MEDIA, MINIMO Y MAXIMO

```
81  
82 // DISTRIBUIAMOS GENEROSIDAD ORDENADA DE PAIS MAS GENEROSO A MENOS GENEROSO, CON SU MEDIA, MINIMO Y MAXIMO  
83  
84 var distribucion_generosidad = [  
85   {  
86     $group: {  
87       _id: "$Country or region",  
88       averageGenerosity: { $avg: "$Generosity" },  
89       minGenerosity: { $min: "$Generosity" },  
90       maxGenerosity: { $max: "$Generosity" },  
91       count: { $sum: 1 },  
92     },  
93   },  
94   {  
95     $sort: { averageGenerosity: -1 },  
96   },  
97 ];  
98  
99 db.Felicidad.aggregate(distribucion_generosidad);  
100
```

```
var distribucion_generosidad = [  
  
  {  
  
    $group: {  
  
      _id: "$Country or region",  
  
      averageGenerosity: { $avg: "$Generosity" },  
  
      minGenerosity: { $min: "$Generosity" },  
  
      maxGenerosity: { $max: "$Generosity" },  
  
      count: { $sum: 1 },  
  
    },  
  
  },  
  
  { $sort: { averageGenerosity: -1 } },  
  
];
```

```
db.Felicidad.aggregate(distribucion_generosidad);
```

ORDENAMOS LOS 5 PAISES CON MEJOR APOYO SOCIAL

```
100
101 // ORDENAMOS LOS 5 PAISES CON MEJOR APOYO SOCIAL
102
103
104 var Mejor_apoyo_social = [{ $sort: { "Social support": -1 } }, { $limit: 5 }];
105
106 db.Felicidad.aggregate(Mejor_apoyo_social);
107
108
```

```
var Mejor_apoyo_social = [{ $sort: { "Social support": -1 } }, { $limit: 5 }];
```

```
db.Felicidad.aggregate(Mejor_apoyo_social);
```

A CONTINUACION, REMOVEMOS DOS DECIMALES DE NUESTRO DATASET

```
109 // A continuación, hemos removido dos decimales de nuestro dataset
110
111 var remover_decimales = [
112   {
113     $set: {
114       "Score": { $round: ["$Score", 2] },
115       "GDP per capita": { $round: ["$GDP per capita", 2] },
116       "Healthy life expectancy": { $round: ["$Healthy life expectancy", 2] },
117       "Freedom to make life choices": { $round: ["$Freedom to make life choices", 2] },
118       "Generosity": { $round: ["$Generosity", 2] },
119       "Perceptions of corruption": { $round: ["$Perceptions of corruption", 2] },
120     },
121   },
122 ];
123
124 db.Felicidad.aggregate(remover_decimales);
125
```

```
var remover_decimales = [
{
  $set: {
    "Score": { $round: ["$Score", 2] },
    "GDP per capita": { $round: ["$GDP per capita", 2] },
    "Healthy life expectancy": { $round: ["$Healthy life expectancy", 2] },
    "Freedom to make life choices": { $round: ["$Freedom to make life choices", 2] },
    "Generosity": { $round: ["$Generosity", 2] },
    "Perceptions of corruption": { $round: ["$Perceptions of corruption", 2] },
  },
},
];
```

```
db.Felicidad.aggregate(remover_decimales);
```

CONTAMOS LA CANTIDAD DE PAISES EN CADA RANGO DE PUNTAJE DE FELICIDAD

```
126 //Contamos la cantidad de paises en cada rango de puntaje de felicidad:
127
128 var rango_felicidad = [
129   {
130     $bucket: {
131       groupBy: "$Score",
132       boundaries: [0, 3, 5, 7, 10],
133       default: "Other",
134       output: {
135         count: { $sum: 1 }
136       }
137     }
138   }
139 ];
140
141 db.Felicidad.aggregate(rango_felicidad);
142
```

```
var rango_felicidad = [
  {
    $bucket: {
      groupBy: "$Score",
      boundaries: [0, 3, 5, 7, 10],
      default: "Other",
      output: {
        count: { $sum: 1 }
      }
    }
  }
];

db.Felicidad.aggregate(rango_felicidad);
```

MOSTRAMOS LOS 5 PAISES DONDE LA PERCEPCION DE CORRUPCION ES MAS BAJA

```
143
144 // MOSTRAMOS LOS 5 PAISES DONDE LA PERCEPCION DE CORRUPCION ES MAS BAJA
145
146 var percepcion_corrupcion = [
147   { $sort: { "Perceptions of corruption": -1 } },
148   { $limit: 5 },
149   { $project: { _id: 0, "Country or region": 1, "Perceptions of corruption": 1 } },
150 ];
151 db.Felicidad.aggregate(percepcion_corrupcion);
152
```

```
var percepcion_corrupcion = [
  { $sort: { "Perceptions of corruption": -1 } },
  { $limit: 5 },
  { $project: { _id: 0, "Country or region": 1, "Perceptions of corruption": 1 } },
];

db.Felicidad.aggregate(percepcion_corrupcion);
```


OBTENEMOS EL MAXIMO Y EL MINIMO TOTAL

```
154
155 // OBTENEMOS EL MAXIMO Y EL MINIMO TOTAL
156
157 var Masfelices_minymax = [
158   { $group: { _id: null, maxScore: { $max: "$Score" }, minScore: { $min: "$Score" } } }
159 ];
160
161 var result = db.Felicidad.aggregate(Masfelices_minymax);
162
163 printjson(result.toArray());
164
```

var Masfelices_minymax = [

{ \$group: { _id: null, maxScore: { \$max: "\$Score" }, minScore: { \$min: "\$Score" } } }

];

var result = db.Felicidad.aggregate(Masfelices_minymax);

printjson(result.toArray());

CALCULAMOS EL TOTAL GDP DE TODOS LOS PAISES

```
165
166
167 // Calculamos el total GDP de todos los paises
168
169 var sumStage = { $group: { _id: null, totalGDP: { $sum: "$GDP per capita" } } };
170 db.Felicidad.aggregate([sumStage]);
171
```

var sumStage = { \$group: { _id: null, totalGDP: { \$sum: "\$GDP per capita" } } };

db.Felicidad.aggregate([sumStage]);

OBTENEMOS EL PAIS MENOS CORRUPTO

```
172
173 // Obtenemos el pais menos corrupto
174
175 db.Felicidad.find().sort({ "Perceptions of corruption": -1 }).limit(1)
176 |
```

db.Felicidad.find().sort({ "Perceptions of corruption": -1 }).limit(1)

AÑADIMOS UN CAMPO DE FECHA FICTICIO A DOS DOCUMENTOS, FINLAND Y DENMARK

```
176
177
178 // Añadimos un campo de fecha ficticio a dos documentos
179
180 db.Felicidad.updateOne(
181   { "Country or region": "Finland" },
182   { $set: { "fechaCreacion": ISODate("2022-01-10T12:00:00Z") } }
183 );
184
185 db.Felicidad.updateOne(
186   { "Country or region": "Denmark" },
187   { $set: { "fechaCreacion": ISODate("2022-01-12T15:30:00Z") } }
188 );
189
```

```
db.Felicidad.updateOne(
  { "Country or region": "Finland" },
  { $set: { "fechaCreacion": ISODate("2022-01-10T12:00:00Z") } }
);
```

```
db.Felicidad.updateOne(
  { "Country or region": "Denmark" },
  { $set: { "fechaCreacion": ISODate("2022-01-12T15:30:00Z") } }
);
```

MOSTRAMOS LOS DOCUMENTOS CREADOS DESPUES DE LA FECHA ESPECIFICA

```
190 // Encontramos documentos creados después de la fecha específica
191
192 var fechaLimite = ISODate("2022-01-11T00:00:00Z");
193 db.Felicidad.find( { "fechaCreacion": { $gt: fechaLimite } } );
194
```

```
var fechaLimite = ISODate("2022-01-11T00:00:00Z");
db.Felicidad.find( { "fechaCreacion": { $gt: fechaLimite } } );
```

MOSTRAMOS LOS DOCUMENTOS CREADOS ENTRE LAS DOS FECHAS

```
194 // Encontramos documentos creados entre dos fechas
195
196 var fechaInicio = ISODate("2022-01-09T00:00:00Z");
197 var fechaFin = ISODate("2022-01-14T23:59:59Z");
198 db.Felicidad.find( { "fechaCreacion": { $gte: fechaInicio, $lte: fechaFin } } );
199
200
```

```
var fechaInicio = ISODate("2022-01-09T00:00:00Z");
var fechaFin = ISODate("2022-01-14T23:59:59Z");
db.Felicidad.find( { "fechaCreacion": { $gte: fechaInicio, $lte: fechaFin } } );
```

RESTAMOS 3 DIAS A LA FECHA ACTUAL

```
201
202 // Restamos 3 días a la fecha actual
203
204 var diasParaRestar = 3;
205 var aggPipeline = [{ $addFields: { "fechaCreacion": { $subtract: [{ $toDate: "$fechaCreacion" }, { $multiply: [diasParaRestar * 24 * 60
206 db.Felicidad.aggregate(aggPipeline);
207
```

```
var diasParaRestar = 3;
```

```
var aggPipeline = [{ $addFields: { "fechaCreacion": { $subtract: [{ $toDate: "$fechaCreacion" }, {
    $multiply: [diasParaRestar * 24 * 60 * 60 * 1000, -1] } } } }];
```

```
db.Felicidad.aggregate(aggPipeline);
```

CONCLUSION:

Bueno, después de sumergirme en los análisis de este dataset de los países más felices, puedo decir que ha sido fascinante descubrir patrones y peculiaridades.

Al observar las estadísticas globales, pude obtener una visión general de cómo se distribuyen los niveles de felicidad, el PIB per cápita y la esperanza de vida. Descubrir los países con el mayor apoyo social o la mayor libertad de elección proporcionó insights sobre qué factores podrían estar contribuyendo más a la felicidad.

Además, jugar con proyecciones y filtrados me permitió enfocarme en detalles específicos, como la distribución de la generosidad. Cada análisis fue como abrir una nueva ventana a la complejidad de lo que significa ser feliz en diferentes partes del mundo. Sin duda, este proyecto me ha enseñado más allá de los números y ha añadido un toque de realismo a lo que aprendemos en clase.