

---

Módulo 8: TECNOLOGÍA Y HERRAMIENTAS BIG DATA

Nombre y apellidos: VIOLETA CHAMOSA FONDEVILA

Fecha entrega 25/05/2023

---

En este documento vamos a adjuntar los resultados con las correspondientes capturas del ejercicio final en la aplicación de MongoDB.

Una empresa multinacional decide llevar a cabo un proyecto basado en el internet de las cosas (IoT) donde estudiará distintos parámetros en 2 localizaciones diferentes donde se encuentran sus sedes con el uso de dos sensores.

Estos sensores medirán estos cuatro parámetros, dos por cada sensor, entre los días 1 y 14 de julio de 2020, para posteriormente analizar resultados a partir de los datos obtenidos.

La directiva de esta empresa multinacional ha enviado una serie de 10 cuestiones para que los analistas de datos respondan, con el objetivo de proporcionar un informe para que se puedan tomar decisiones acerca del estudio.

Estas cuestiones implican la consulta de la colección para obtener diferentes insights, así como la realización de inserciones o actualizaciones de la BBDD "sensores\_IoT".

## EJERCICIO 1

Crear una nueva colección llamada “sensores” e introducir en ella cuatro documentos, correspondientes a cada uno de los sensores que se encuentran instalados en este momento: dos en Sevilla y dos en Valladolid.

La colección deberá contener las siguientes claves:

- **Ubicacion:** tipo string, indica la ciudad en la que está instalado el sensor.
- **“medidas\_sensor”:** array de objetos JSON. Cada JSON tendrá dos subclaves: **“tipo\_medida”**, string que identifica lo que mide el sensor; **“unidad”**: string que identifica las unidades en las que mide el sensor cada tipo de medida.
- **Coordenadas:** array de numéricos. Los sensores de Valladolid tienen las coordenadas [41.638597, 4.740186] y los de Sevilla [37.409311, -5.949939].
- **“fecha\_instalación”:** timestamp. Los dos sensores de Valladolid se instalaron el día 25 de mayo 2020 a las 10:30 AM, mientras que los de Sevilla se instalaron el 28 de mayo 2020 a las 11:30 AM.
- **“location\_id”:** numérico que indica el ID de la ubicación a la que pertenece el sensor. Valor uno para los sensores de Valladolid y valor dos para los sensores de Sevilla.
- **“tipo\_sensor”:** numérico que indica el tipo de sensor. Valor uno para el tipo de sensor que mide temperatura y humedad relativa. Valor dos para el tipo de sensor que mide emisión de CO2 y consumo eléctrico.

La colección se creará automáticamente al insertar los cuatro documentos en una misma consulta. Adjuntar captura de la consulta de inserción, así como el resultado tras ejecutarla.

EL CÓDIGO DE LA INSERCIÓN ES EL SIGUIENTE:

```
db.sensores.insertMany([
  {
    "ubicacion": "Valladolid",
    "medidas_sensor": [
      {
        "tipo_medida": "Temperatura",
        "unidad": "°C"
      },
      {
        "tipo_medida": "Humedad_relativa",
        "unidad": "%"
      }
    ],
    "coordenadas": [
      41.638597, 4.740186
    ],
    "fecha_instalacion": new Date("2020-05-25T10:30:00Z"),
    "location_id": 1,
    "tipo_sensor": 1
  },
  {
    "ubicacion": "Valladolid",
    "medidas_sensor": [
      {
        "tipo_medida": "Emision_CO2",
        "unidad": "gCO2/m2"
      },
      {
        "tipo_medida": "Consumo_electrico",
        "unidad": "kWh/m2"
      }
    ],
    "coordenadas": [
      41.638597, 4.740186
    ],
    "fecha_instalacion": new Date("2020-05-25T10:30:00Z"),
    "location_id": 1,
    "tipo_sensor": 2
  },
  {
    "ubicacion": "Sevilla",
    "medidas_sensor": [
      {
        "tipo_medida": "Temperatura",
        "unidad": "°C"
      }
    ],
  },
]
```

```

{
  "tipo_medida": "Humedad_relativa",
  "unidad": "%"
},
{
  "coordenadas": [
    37.409311, -5.949939
  ],
  "fecha_instalacion": new Date("2020-05-28T11:30:00Z"),
  "location_id": 2,
  "tipo_sensor": 1
},
{
  "ubicacion": "Sevilla",
  "medidas_sensor": [
    {
      "tipo_medida": "Emision_CO2",
      "unidad": "gCO2/m2"
    },
    {
      "tipo_medida": "Consumo_electrico",
      "unidad": "kWh/m2"
    }
  ],
  "coordenadas": [
    37.409311, -5.949939
  ],
  "fecha_instalacion": new Date("2020-05-28T11:30:00Z"),
  "location_id": 2,
  "tipo_sensor": 2
})

```

## COMPROBACIÓN DE LA INSERCIÓN

```

"acknowledged" : true,
"insertedIds" : [
  ObjectId("643e7ba3b21a22f24173baff"),
  ObjectId("643e7ba3b21a22f24173bb00"),
  ObjectId("643e7ba3b21a22f24173bb01"),
  ObjectId("643e7ba3b21a22f24173bb02")
]

```

## EJERCICIO 2

Mostrar el número de datos recogidos por el sensor que mide la temperatura en Valladolid, así como el número de datos recogidos por el de Sevilla entre los días 1 y 10 julio (incluido el día 10 entero).

Haciendo cuentas, se reciben cuatro datos por hora, 96 por día, y, por tanto, en 10 días debería haber 960 datos enviados por cada sensor. Comentar si no se han recibido datos de temperatura en algún intervalo de 15 minutos para alguno de los sensores.

```
> db.datos_sensores.aggregate([
... {$match:{"sensor_id":1, "timestamp":{"$lt:"2020-07-11T00:00:00Z"}} },
... {$group:{ _id:"$location_id", count:{$sum:1}}}
... ])
{ "_id" : 1, "count" : 960 }
{ "_id" : 2, "count" : 959 }
```

Como podemos observar, para el sensor de Valladolid se han recibido todos los datos, mientras que para el de Sevilla, hay uno que falta.

### EJERCICIO 3

Se pide identificar si hay algún documento que pueda distorsionar los resultados del estudio. Por ello, se quiere identificar posibles valores erróneos de temperatura enviados por el sensor.

Se pide, por tanto, en primer lugar, identificar los documentos de la colección “datos\_sensores” que tengan una temperatura superior a 55 °C, tanto para Valladolid como para Sevilla (contar el número de casos en cada ciudad y mostrar también los documentos erróneos).

A la hora de mostrar los documentos con errores, solamente se devolverán las claves: timestamp, “location\_id” y de la clave medidas mostrar solo el objeto correspondiente a la temperatura, sin mostrar el objeto de humedad relativa.

Una vez se hayan identificado, se procede a eliminar estos documentos de la colección, es decir: para ese timestamp solo quedará el documento correspondiente a la emisión de CO2 y el consumo eléctrico.

Además de las anteriores, se deberá adjuntar también una captura que demuestre que se han eliminado correctamente dichos documentos, si es que existiera alguno.

### CONTAMOS LOS DOCUMENTOS

```
> db.datos_sensores.aggregate([
... {$match:{"sensor_id":1, "medidas.0.valor":{$gt:55}} },
... {$group:{ _id:"$location_id", count:{$sum:1}}}
... ])
{ "_id" : 2, "count" : 1 }
{ "_id" : 1, "count" : 2 }
>
```

## MOSTRAMOS LOS DOCUMENTOS CON SOLAMENTE LOS CAMPOS PEDIDOS

```
> db.datos_sensores.find({"medidas.0.valor":{$gt:55}, "sensor_id":1},{_id:0,timestamp:1,medidas:{$slice:1}}).pretty()
{
  "timestamp" : "2020-07-05T17:15:00Z",
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 67.27,
      "unidad" : "°C"
    }
  ]
}
{
  "timestamp" : "2020-07-08T03:00:00Z",
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 82.64,
      "unidad" : "°C"
    }
  ]
}
{
  "timestamp" : "2020-07-11T19:30:00Z",
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 125.48,
      "unidad" : "°C"
    }
  ]
}
```

## ELIMINAMOS LOS CAMPOS EN LOS DOCUMENTOS

```
> db.datos_sensores.update({"_id":{"$in:[ObjectId("643e78cd6453573ea0a570a4"),
ObjectId("643e78f06453573ea0a57571"), ObjectId("643e790f6453573ea0a579cc")]
}},
...
... {$pull:{medidas:{tipo_medida:"Temperatura"}}},{multi: true})
WriteResult({ "nMatched" : 3, "nUpserted" : 0, "nModified" : 3 })
```

MOSTRAMOS LOS DOCUMENTOS ACTUALIZADOS

```
> db.datos_sensores.find({"_id":{"$in:[ObjectId("643e78cd6453573ea0a570a4"),ObjectId("643e78f06453573ea0a57571"), ObjectId("643e790f6453573ea0a579cc")]} }).pretty()
{
  "_id" : ObjectId("643e78cd6453573ea0a570a4"),
  "timestamp" : "2020-07-05T17:15:00Z",
  "sensor_id" : 1,
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Humedad_relativa",
      "valor" : 8.16,
      "unidad" : "%"
    }
  ]
}
{
  "_id" : ObjectId("643e78f06453573ea0a57571"),
  "timestamp" : "2020-07-08T03:00:00Z",
  "sensor_id" : 1,
  "location_id" : 2,
  "medidas" : [
    {
      "tipo_medida" : "Humedad_relativa",
      "valor" : 45.36,
      "unidad" : "%"
    }
  ]
}
{
  "_id" : ObjectId("643e790f6453573ea0a579cc"),
  "timestamp" : "2020-07-11T19:30:00Z",
  "sensor_id" : 1,
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Humedad_relativa",
      "valor" : 51.82,
      "unidad" : "%"
    }
  ]
}
```



#### EJERCICIO 4

Buscar el valor mínimo de temperatura en Sevilla. Se considera que es un valor poco realista para Sevilla y que es necesario multiplicarlo por un factor 1,2 para hacerlo un valor algo más real.

**Nota:** una vez hallado el ID del objeto a actualizar, investigar el funcionamiento de la función `$mul` para comprobar si puede utilizarse en este caso para actualizar el valor o es necesario utilizar otro modificador.

Se deberá adjuntar una captura de la consulta de búsqueda junto con el documento al que le corresponde la temperatura mínima de Sevilla. Además, también se adjuntará la consulta de actualización y, por último, se mostrará todo el documento ya actualizado.

#### MOSTRAMOS LA CONSULTA DE BÚSQUEDA

```
> db.datos_sensores.find({"location_id":2, "sensor_id":1}).sort({"medidas.0.valor":1})
.limit(1).pretty()
{
  "_id" : ObjectId("643e79146453573ea0a57af9"),
  "timestamp" : "2020-07-11T19:30:00Z",
  "sensor_id" : 1,
  "location_id" : 2,
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 9.14,
      "unidad" : "°C"
    },
    {
      "tipo_medida" : "Humedad_relativa",
      "valor" : 33.42,
      "unidad" : "%"
    }
  ]
}
```

MOSTRAMOS LA CONSULTA DE ATUALIZACIÓN

```
> db.datos_sensores.update({_id:ObjectId("643e79146453573ea0a57af9")}, {$mul:{"medidas
.0.valor":1.2}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>
```

MOSTRAMOS QUE SE HA ACTUALIZADO

```
>
> db.datos_sensores.find({_id:ObjectId("643e79146453573ea0a57af9")}).pretty()
{
  "_id" : ObjectId("643e79146453573ea0a57af9"),
  "timestamp" : "2020-07-11T19:30:00Z",
  "sensor_id" : 1,
  "location_id" : 2,
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 10.968,
      "unidad" : "°C"
    },
    {
      "tipo_medida" : "Humedad_relativa",
      "valor" : 33.42,
      "unidad" : "%"
    }
  ]
}
```

Como podemos comprobar, si que es posible el uso del operador \$mul para la modificación de un valor de un campo en nuestro documento, en este caso, multiplicándolo por un número.

Este operador es usado para actualizar un valor de un campo en uno o más documentos.

## EJERCICIO 5

**Obtener los tres máximos valores de consumo eléctrico en un día de fin de semana en ambas ciudades y comparar sus valores (se considera fin de semana a partir de las 16:00 del viernes y hasta las 7:45 del lunes).**

**Devolver para cada ciudad un documento con el timestamp en que se producen esos máximos, el día de la semana y la hora a las que corresponden esos máximos y el valor del consumo eléctrico (subclave "tipo\_medida" "Consumo\_electrico" + subclave "valor" + subclave "unidad", es decir: no se quieren obtener los valores de ese día de emisión de CO2).**

Para realizar esta consulta hemos usado el siguiente código, variando únicamente la localización dentro del operador \$match.

```
db.datos_sensores.aggregate([
  {$project:{timestamp:{$dateFromString:{dateString:'$timestamp'}}, "_id":0, "Dia_semana":1,
    "Hora":1, "medidas":{$slice:["$medidas",-1]}, "location_id":1 }},
  {$addFields:{Dia_semana:{$dayOfWeek: "$timestamp"}, Hora:{$dateToString: {date:
    "$timestamp", format: "%H:%M"}}}},
  {$match:{"Dia_semana":{$in:[1,2,6,7]}, "medidas.tipo_medida":"Consumo_electrico",
    "location_id":1}},
  {$match: {$or:[$and: [{"Dia_semana":6}, {"Hora":{$gt:"16:00"}}], {"Dia_semana":{$in:[7,1]}},
    {$and:[{"Dia_semana":2}, {"Hora":{$lt:"07:45"}}]}]},
  {$sort:{"medidas.valor":-1}},
  {$limit:3}
]).pretty()
```

## DOCUMENTOS PARA LA CIUDAD DE VALLADOLID

```
{
  "location_id" : 1,
  "timestamp" : ISODate("2020-07-11T01:45:00Z"),
  "medidas" : [
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.00235,
      "unidad" : "kWh/m2"
    }
  ],
  "Dia_semana" : 7,
  "Hora" : "01:45"
}
{
  "location_id" : 1,
  "timestamp" : ISODate("2020-07-11T22:15:00Z"),
  "medidas" : [
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.00225,
      "unidad" : "kWh/m2"
    }
  ],
  "Dia_semana" : 7,
  "Hora" : "22:15"
}
{
  "location_id" : 1,
  "timestamp" : ISODate("2020-07-11T20:00:00Z"),
  "medidas" : [
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.00214,
      "unidad" : "kWh/m2"
    }
  ],
  "Dia_semana" : 7,
  "Hora" : "20:00"
}
```

## DOCUMENTOS PARA LA CIUDAD DE SEVILLA

```
{
  "location_id" : 2,
  "timestamp" : ISODate("2020-07-05T15:00:00Z"),
  "medidas" : [
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.01711,
      "unidad" : "kWh/m2"
    }
  ],
  "Dia_semana" : 1,
  "Hora" : "15:00"
}
{
  "location_id" : 2,
  "timestamp" : ISODate("2020-07-05T08:00:00Z"),
  "medidas" : [
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.01706,
      "unidad" : "kWh/m2"
    }
  ],
  "Dia_semana" : 1,
  "Hora" : "08:00"
}
{
  "location_id" : 2,
  "timestamp" : ISODate("2020-07-05T13:00:00Z"),
  "medidas" : [
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.01698,
      "unidad" : "kWh/m2"
    }
  ],
  "Dia_semana" : 1,
  "Hora" : "13:00"
}
```

## EJERCICIO 6

Se considera como perjudicial para la salud cualquier día en que el acumulado de las emisiones de CO2 durante el día completo supere los 420 g CO2/m2. De los 14 días que comprende nuestro estudio, se pretende recuperar cuántos días se superó ese límite para Valladolid y, por otro lado, cuántos días se superó para Sevilla.

Se deben adjuntar tanto la captura del número de veces que se supera el límite para cada ciudad (clave "días\_sobre\_limite\_permitido"), así como otra captura que muestre todos los documentos de Valladolid, por un lado, agrupados por la clave "dia\_mes" y el acumulado total de emisión CO2 bajo la clave "suma\_Emision\_CO2", ordenados de mayor a menor emisión.

Presentar las mismas dos capturas para Sevilla y comentar los resultados. ¿Qué tienen los cuatro días de ambas ciudades en los que la emisión de CO2 es más pequeña?

Las consultas serán idénticas para Valladolid y Sevilla, distinguiéndose únicamente por el filtrado sobre la clave "location\_id".

En este caso, el código usado para ambas consultas en las 2 ciudades es, como en el ejercicio anterior, el mismo, a excepción del campo de "location\_id".

En primer lugar, haremos la consulta del número de días en los que se supera el límite y, seguidamente mostraremos los documentos en los cuales se superan estos valores.

Consulta del número de días que se supera el límite:

```
db.datos_sensores.aggregate([
  {$project:{timestamp:{$dateFromString:{dateString:'$timestamp'}},' _id':0,
  "medidas":{$slice:["$medidas",1]},"location_id":1}},
  {$addFields:{dia_mes:{$dayOfMonth:"$timestamp"}}},
  {$unwind:"$medidas"},
  {$match:{"location_id":1,"medidas.tipo_medida":"Emision_CO2"}},
  {$group:{_id:{"dia_mes":"$dia_mes"},suma_Emision_CO2:{$sum:"$medidas.valor"}},{$match:{suma_Emision_CO2:{$gt:420}}},
  {$count: "dias_sobre_limite_CO2_permitido"]})
```

Demostración de los documentos:

```
db.datos_sensores.aggregate([
  {$project:{timestamp:{$dateFromString:{dateString:'$timestamp'}},' _id':0,
  "medidas":{$slice:["$medidas",1]},"location_id":1}},
  {$addFields:{dia_mes:{$dayOfMonth:"$timestamp"}}},
  {$unwind:"$medidas"},
  {$match:{"location_id":1,"medidas.tipo_medida":"Emission_CO2"}},
  {$group:{_id:{"dia_mes":"$dia_mes"},suma_Emission_CO2:{$sum:"$medidas.valor"}},{ $match:{suma_Emission_CO2:{$gt:420}}},
  {$sort:{"suma_Emission_CO2":-1}}]).pretty()
```

PARA EL CASO DE VALLADOLID

```
> db.datos_sensores.aggregate([
... {$project:{timestamp:{$dateFromString:{dateString:'$timestamp'}},' _id':0, "medidas":{$slice:["$medidas",1]},"location_id":1}},
... {$addFields:{dia_mes:{$dayOfMonth:"$timestamp"}}},
... {$unwind:"$medidas"},
... {$match:{"location_id":1,"medidas.tipo_medida":"Emission_CO2"}},
... {$group:{_id:{"dia_mes":"$dia_mes"},suma_Emission_CO2:{$sum:"$medidas.valor"}},{ $match:{suma_Emission_CO2:{$gt:420}}},
... {$count:"dias_sobre_limite_CO2_permitido"}})
{ "dias_sobre_limite_CO2_permitido" : 4 }
```

```
> db.datos_sensores.aggregate([
... {$project:{timestamp:{$dateFromString:{dateString:'$timestamp'}},' _id':0, "medidas":{$slice:["$medidas",1]},"location_id":1}},
... {$addFields:{dia_mes:{$dayOfMonth:"$timestamp"}}},
... {$unwind:"$medidas"},
... {$match:{"location_id":1,"medidas.tipo_medida":"Emission_CO2"}},
... {$group:{_id:{"dia_mes":"$dia_mes"},suma_Emission_CO2:{$sum:"$medidas.valor"}},{ $match:{suma_Emission_CO2:{$gt:420}}},
... {$sort:{"suma_Emission_CO2":-1}}]).pretty()
{ "_id" : { "dia_mes" : 9 }, "suma_Emission_CO2" : 425.432 }
{ "_id" : { "dia_mes" : 13 }, "suma_Emission_CO2" : 422.799 }
{ "_id" : { "dia_mes" : 7 }, "suma_Emission_CO2" : 421.447 }
{ "_id" : { "dia_mes" : 2 }, "suma_Emission_CO2" : 420.337 }
```

## PARA EL CASO DE SEVILLA

```
> db.datos_sensores.aggregate([
... {$project:{timestamp:{$dateFromString:{dateString:'$timestamp'}}, "_id":0, "medidas":{$slice:["$medidas",1]}, "location_id":1}},
... {$addFields:{dia_mes:{$dayOfMonth:"$timestamp"}}},
... {$unwind:"$medidas"},
... {$match:{location_id:2,"medidas.tipo_medida":"Emision_CO2"}},
... {$group:{_id:{"dia_mes":"$dia_mes"}, suma_Emision_CO2:{$sum:"$medidas.valor"}}}, {$match:{suma_Emision_CO2:{$gt:420}}},
... {$count: "dias_sobre_limite_CO2_permitido"}})
{ "dias_sobre_limite_CO2_permitido" : 8 }
```

```
> db.datos_sensores.aggregate([
... {$project:{timestamp:{$dateFromString:{dateString:'$timestamp'}}, "_id":0, "medidas":{$slice:["$medidas",1]}, "location_id":1}},
... {$addFields:{dia_mes:{$dayOfMonth:"$timestamp"}}},
... {$unwind:"$medidas"},
... {$match:{location_id:2,"medidas.tipo_medida":"Emision_CO2"}},
... {$group:{_id:{"dia_mes":"$dia_mes"}, suma_Emision_CO2:{$sum:"$medidas.valor"}}}, {$match:{suma_Emision_CO2:{$gt:420}}},
... {$sort:{suma_Emision_CO2:-1}}]).pretty()
{ "_id" : { "dia_mes" : 1 }, "suma_Emision_CO2" : 509.796 }
{ "_id" : { "dia_mes" : 7 }, "suma_Emision_CO2" : 507.367 }
{ "_id" : { "dia_mes" : 6 }, "suma_Emision_CO2" : 503.715 }
{ "_id" : { "dia_mes" : 2 }, "suma_Emision_CO2" : 501.444 }
{ "_id" : { "dia_mes" : 9 }, "suma_Emision_CO2" : 501.261 }
{ "_id" : { "dia_mes" : 13 }, "suma_Emision_CO2" : 500.754 }
{ "_id" : { "dia_mes" : 8 }, "suma_Emision_CO2" : 499.94100000000003 }
{ "_id" : { "dia_mes" : 14 }, "suma_Emision_CO2" : 483.623 }
```



## EJERCICIO 7

Obtener la media de emisiones de CO2 de cada ciudad por separado en la hora punta de personas en la oficina, que se considera las 10 AM. Por tanto, se obtendrán por un lado los registros con hora 10 en el timestamp, que midan CO2 y que sean de Valladolid y obtener la media de emisiones de CO2 durante los 14 días en esa hora como la clave "Avg\_Emission\_CO2" (recordar que cada hora se reciben cuatro valores desde el sensor para cada ciudad). Mostrar los resultados ordenados por la clave "Avg\_Emission\_CO2", mostrando como primer documento la media más baja. Redondear esta clave a dos decimales. Realizar la misma operativa con Sevilla.

Vamos a hacer la consulta de los documentos correspondientes para la ciudad de Valladolid, primero, y después, para los de Sevilla.

Una vez obtenidos los documentos realizaremos la consulta para saber la media de las emisiones de CO2 en los 14 días agrupándola por la localización.

MOSTRAMOS ALGUNOS DOCUMENTOS DEL VALOR DE LAS EMISIONES A LAS 10 HORAS PARA LA CIUDAD DE VALLADOLID

```
> db_datos_sensores.aggregate([
...   {'$project':{'timestamp':{$dateFromString:(dateString:'$timestamp')}, '_id':0, "medidas":{$slice:['$medidas',1]}, "location_id":1}},
...   {'$addFields':{'Hora':{$hour:'$timestamp'}}},
...   {'$unwind':{'$medidas'}},
...   {'$match':{'location_id':1, 'Hora':{$in:[10]}, "medidas.tipo_medida":"'Enliston_CO2'" }},
...   {'$sort':{'medidas.valor":-1}}])
{ "location_id" : 1, "timestamp" : ISODate("2020-07-12T18:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.667, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-11T18:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.678, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-11T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.698, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-04T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.722, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-04T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.739, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-05T18:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.771, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-11T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.786, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-05T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.823, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-12T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.823, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-12T18:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.827, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-12T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.835, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-05T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.836, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-11T18:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.837, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-04T18:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.872, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-05T18:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.883, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-04T18:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 1.903, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-07T18:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 7.132, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-02T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 7.445, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-01T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 7.849, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-14T18:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.021, "unidad" : "gCO2/m2", "Hora" : 10 },
type "lt" for more
> type "lt" for more
{ "location_id" : 1, "timestamp" : ISODate("2020-07-07T18:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.173, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-06T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.344, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-06T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.383, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-07T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.475, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-02T18:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.517, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-08T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.523, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-09T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.555, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-03T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.621, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-10T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.629, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-13T18:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.674, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-02T18:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.704, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-03T18:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.796, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-10T18:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.829, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-01T18:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.897, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-01T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.915, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-09T18:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.925, "unidad" : "gCO2/m2", "Hora" : 10 },
{ "location_id" : 1, "timestamp" : ISODate("2020-07-14T18:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.989, "unidad" : "gCO2/m2", "Hora" : 10 }
```

## MOSTRAMOS LOS DOCUMENTOS PARA SEVILLA

```
> db.datos_sensores.aggregate([
...   {$project:{$timestamp:{$dateFromString:{dateString:'$timestamp'}}}, "_id":0, "medidas":{$slice:[$medidas,1]}, "location_id":1}},
...   {$addFields:{Hora:{$hour:"$timestamp"}}},
...   {$unwind:"$medidas"},
...   {$match:{location_id:2, Hora:{"$in":["10"]}, "medidas.tipo_medida":"Emision_CO2"}},
...   {$sort:{medidas.valor:1}}])
{ "location_id" : 2, "timestamp" : ISODate("2020-07-11T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.121, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-12T10:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.175, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-12T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.178, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-05T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.182, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-04T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.199, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-11T10:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.203, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-05T10:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.213, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-05T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.216, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-11T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.225, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-11T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.244, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-12T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.249, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-04T10:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.25, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-04T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.251, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-05T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.253, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-04T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.277, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-12T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.279, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-10T10:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 2.279, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-10T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 7.577, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-10T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 7.761, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-03T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 7.862, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-03T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.02, "unidad" : "gCO2/m2", "Hora" : 10 } }
Type "it" for more
> it
{ "location_id" : 2, "timestamp" : ISODate("2020-07-03T10:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.057, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-10T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.231, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-10T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.412, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-03T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.54, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-08T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.692, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-07T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 8.956, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-06T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.633, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-14T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.646, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-06T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.664, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-02T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.69, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-08T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.768, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-07T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.788, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-09T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.863, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-14T10:45:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.948, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-13T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.961, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-08T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 9.993, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-14T10:30:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 10.041, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-13T10:15:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 10.056, "unidad" : "gCO2/m2", "Hora" : 10 } }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-07T10:00:00Z"), "medidas" : { "tipo_medida" : "Emision_CO2", "valor" : 10.071, "unidad" : "gCO2/m2", "Hora" : 10 } }
```

## MOSTRAMOS LAS MEDIAS DE LAS EMISIONES PARA AMBAS CUIDADES

```
> db.datos_sensores.aggregate([
...   {$project:{$timestamp:{$dateFromString:{dateString:'$timestamp'}}}, "_id":0, "medidas":{$slice:[$medidas,1]}, "location_id":1}},
...   {$addFields:{Hora:{$hour:"$timestamp"}}},
...   {$unwind:"$medidas"},
...   {$match:{Hora:{"$in":["10"]}, "medidas.tipo_medida":"Emision_CO2"}},
...   {$group:{_id:"$location_id", Avg_Emi_CO2:{$avg:"$medidas.valor"}}},
...   {$addFields:{Avg_Emission_CO2:{$round:["$Avg_Emi_CO2",2]}}},
...   {$sort:{Avg_Emission_CO2:1}}])
{ "_id" : 1, "Avg_Emi_CO2" : 6.996178571428572, "Avg_Emission_CO2" : 7 }
{ "_id" : 2, "Avg_Emi_CO2" : 7.599410714285715, "Avg_Emission_CO2" : 7.6 }
```



## EJERCICIO 8

Se ha descubierto que el sensor de temperatura de Valladolid mide 1,5 °C de más. Por ello, se pide actualizar todos los valores correspondientes a este sensor decrementando el valor de la temperatura en 1,5 °C.

Antes de realizar la actualización, se ordenarán mostrando primero el de mayor temperatura, y muestra los dos primeros documentos del sensor de Valladolid con la mayor temperatura. Solo mostrar las siguientes claves: timestamp, "location\_id". Del array de medidas solo mostrar el primer ítem, por ejemplo: "medidas" : [ { "tipo\_medida" : "Temperatura", "valor" : 15.75, "unidad" : "°C" } ] }. No incluir el ObjectId)

Realizar el mismo proceso una vez que has actualizado los valores y realiza capturas de los dos documentos anteriores para comprobar que se han actualizado.

MOSTRAMOS LOS DOS DOCUMENTOS CON MAYOR TEMPERATURA PARA VALLADILID

```
> db.datos_sensores.find({location_id:1, sensor_id:1,"medidas.tipo_medida":"Temperatura"},
... {_id:0,timestamp:1,location_id:1,medidas:{$slice:1}}).sort({"medidas.0.valor":-1}).limit(2).pretty()
{
  "timestamp" : "2020-07-14T17:30:00Z",
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 40.88,
      "unidad" : "°C"
    }
  ]
}
{
  "timestamp" : "2020-07-03T16:30:00Z",
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 40.87,
      "unidad" : "°C"
    }
  ]
}
```

## ACTUALIZAMOS LOS DOCUMENTOS

```
> db.datos_sensores.update({location_id:1, sensor_id:1,"medidas.tipo_medida":"Temperatura"},
... {$inc:{"medidas.0.valor":-1.5}},{multi: true})
WriteResult({ "nMatched" : 1342, "nUpserted" : 0, "nModified" : 1342 })
>
```

Como vemos se han modificado 1342, que vienen siendo los documentos que corresponden a la ciudad de Valladolid para el sensor 1.

## MOSTRAMOS DOS DOCUMENTOS ACTUALIZADOS

```
> db.datos_sensores.find({location_id:1, sensor_id:1,"medidas.tipo_medida":"Temperatura"},
... {_id:0,timestamp:1,location_id:1,medidas:{$slice:1}}).sort({"medidas.0.valor":-1}).limit(2).pretty()
{
  "timestamp" : "2020-07-14T17:30:00Z",
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 39.38,
      "unidad" : "°C"
    }
  ]
}
{
  "timestamp" : "2020-07-03T16:30:00Z",
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 39.37,
      "unidad" : "°C"
    }
  ]
}
```

Vemos que los documentos han sido actualizados correctamente, modificándose las temperaturas iniciales de 40.88 y 40.87 a los nuevos valores de 39.38 y 39.37, respectivamente.

## EJERCICIO 9

Se quieren analizar los porcentajes de humedad relativa en horario laboral (8-18:00) de ambas ciudades. Recuperar por un lado los cinco documentos con los valores mínimos de humedad relativa en Sevilla y los cinco documentos con humedad relativa mínima en Valladolid por separado (ambos en horario laboral).

Sólo se quieren recuperar de estos documentos el timestamp y la subclave `medidas.tipo_medida = Humedad_relativa`, junto con el valor y la unidad asociados.

Analizar si los mínimos se producen siempre en la misma franja horaria (mismo intervalo de 1-2 horas, o es variable). Comentar si varían de una ciudad a otra.

Consideramos horario laboral los días de la semana entre lunes y viernes

### MOSTRAMOS LOS DOCUMENTOS PARA VALLADOLID

```
> db.datos_sensores.aggregate([
... {$project:{timestamp:{$dateFromString:{dateString:'timestamp'}}, "_id":0, "medidas":{$slice:[$medidas,-1]}, "location_id":1}},
... {$addFields:{Dia_semana:{$dayOfWeek: "timestamp"}, Hora:{$hour: "timestamp"}}},
... {$match:{"Dia_semana":{$in:[2,3,4,5,6]}, "Hora":{$lt:18}, "medidas.tipo_medida":"Humedad_relativa", "location_id":1}},
... {$match: {"Hora":{$gte:8}}},
... {$sort:{"medidas.valor":1}},
... {$limit:5}
... ])
{ "location_id" : 1, "timestamp" : ISODate("2020-07-03T16:45:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 5.74,
"unidad" : "%" } ], "Dia_semana" : 6, "Hora" : 16 }
{ "location_id" : 1, "timestamp" : ISODate("2020-07-06T17:00:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 6.81,
"unidad" : "%" } ], "Dia_semana" : 2, "Hora" : 17 }
{ "location_id" : 1, "timestamp" : ISODate("2020-07-10T17:45:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 7.6,
"unidad" : "%" } ], "Dia_semana" : 6, "Hora" : 17 }
{ "location_id" : 1, "timestamp" : ISODate("2020-07-14T17:30:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 8.14,
"unidad" : "%" } ], "Dia_semana" : 3, "Hora" : 17 }
{ "location_id" : 1, "timestamp" : ISODate("2020-07-07T14:45:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 8.53,
"unidad" : "%" } ], "Dia_semana" : 3, "Hora" : 14 }
>
```

## MOSTRAMOS LOS DOCUMENTOS PARA SEVILLA

```
> db.datos_sensores.aggregate([
... {$project:{timestamp:{$dateFromString:{dateString:'$timestamp'}}, "_id":0, "medidas":{$slice:["$medidas",-1]}, "location_id":1}},
... {$addFields:{Dia_semana:{$dayOfWeek: "$timestamp"}, Hora:{$hour: "$timestamp"}}},
... {$match:{Dia_semana:{$in:[2,3,4,5,6]}, "Hora":{$lt:18}, "medidas.tipo_medida":"Humedad_relativa", "location_id":2}},
... {$match: {"Hora":{$gte:8}}},
... {$sort:{"medidas.valor":1}},
... {$limit:5}
... ])
{ "location_id" : 2, "timestamp" : ISODate("2020-07-01T16:15:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 0.61,
"unidad" : "%" } ], "Dia_semana" : 4, "Hora" : 16 }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-02T15:15:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 0.94,
"unidad" : "%" } ], "Dia_semana" : 5, "Hora" : 15 }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-14T14:45:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 1.25,
"unidad" : "%" } ], "Dia_semana" : 3, "Hora" : 14 }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-10T13:45:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 2.1,
"unidad" : "%" } ], "Dia_semana" : 6, "Hora" : 13 }
{ "location_id" : 2, "timestamp" : ISODate("2020-07-07T13:30:00Z"), "medidas" : [ { "tipo_medida" : "Humedad_relativa", "valor" : 2.25,
"unidad" : "%" } ], "Dia_semana" : 3, "Hora" : 13 }
>
```

Observando los datos obtenidos en las consultas, vemos que, para la ciudad de Valladolid, los mínimos se encuentran en la franja horaria de entre 2 y 5 horas de la tarde, dándose un mayor registro a las 5 de la tarde. En el caso de la ciudad de Sevilla, la franja horaria es desde la 1 hasta las 4 de la tarde.

Podemos decir que los mínimos se producen en unas franjas horarias similares en las dos ciudades.

## EJERCICIO 10

Se quieren actualizar todos los documentos para introducir en el array de medidas dos nuevos elementos que van a ser constantes y que van a tener el valor del precio del kWh y de la superficie total de la sede.

Solamente se añadirán a los documentos que tengan como medida el consumo eléctrico (los que tienen temperatura y humedad no).

Los ítems a introducir son para Valladolid:

```
{"precio_kWh":0.102,"unidad":"€/kWh"},"superficie":450,"unidad":"m2"}
```

Los ítems a introducir son para Sevilla:

```
{"precio_kWh":0.107,"unidad":"€/kWh"},"superficie":550,"unidad":"m2"}
```

Para comprobar que se ha actualizado correctamente la colección, muestra cuatro documentos de Sevilla y cuatro de Valladolid, correspondientes a los timestamps: 2020-07-01T08:00:00Z, 2020-07-01T23:15:00Z. Cada intervalo se envían dos documentos desde cada ciudad: (Temperatura y Humedad\_relativa) + (Emision\_CO2 y Consumo\_electrico).

## ACTUALIZACIÓN DE LOS DOCUMENTOS PARA AMBAS CUIDADES

```
>
> db.datos_sensores.updateMany({"sensor_id":2, "location_id":1},
... { $push: { medidas:{$each: [{"precio_kWh":0.102,"unidad":"€/kWh"},"superficie":450,"unidad":"m2"}]} }
... )
{ "acknowledged" : true, "matchedCount" : 1344, "modifiedCount" : 1344 }
>
>
> db.datos_sensores.updateMany({"sensor_id":2, "location_id":2},
... { $push: { medidas:{$each: [{"precio_kWh":0.107,"unidad":"€/kWh"},"superficie":550,"unidad":"m2"}]} }
... )
{ "acknowledged" : true, "matchedCount" : 1344, "modifiedCount" : 1344 }
>
```

## MOSTRAMOS LOS DOCUMENTOS DE VALLADOLID

```
> db.datos_sensores.find({"timestamp":{"$in":["2020-07-01T08:00:00Z", "2020-07-01T23:15:00Z"]}, "location_id":1, "sensor_id": 2}).pretty()
{
  "_id" : ObjectId("643e78a06453573ea0a56a6c"),
  "timestamp" : "2020-07-01T08:00:00Z",
  "sensor_id" : 2,
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Emision_CO2",
      "valor" : 6.7,
      "unidad" : "gCO2/m2"
    },
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.01788,
      "unidad" : "kWh/m2"
    },
    {
      "precio_kWh" : 0.102,
      "unidad" : "€/kWh"
    },
    {
      "superficie" : 450,
      "unidad" : "m2"
    }
  ]
}
{
  "_id" : ObjectId("643e78a56453573ea0a56b05"),
  "timestamp" : "2020-07-01T23:15:00Z",
  "sensor_id" : 2,
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Emision_CO2",
      "valor" : 1.629,
      "unidad" : "gCO2/m2"
    },
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.00164,
      "unidad" : "kWh/m2"
    },
    {
      "precio_kWh" : 0.102,
      "unidad" : "€/kWh"
    },
    {
      "superficie" : 450,
      "unidad" : "m2"
    }
  ]
}
```



## MOSTRAMOS LOS DOCUMENTOS DE SEVILLA

```
> db.datos_sensores.find({"timestamp":{"$in":["2020-07-01T08:00:00Z", "2020-07-01T23:15:00Z"]}, "location_id":2, "sensor_id": 2}).pretty()
{
  "_id" : ObjectId("643e78aa6453573ea0a56b4a"),
  "timestamp" : "2020-07-01T08:00:00Z",
  "sensor_id" : 2,
  "location_id" : 2,
  "medidas" : [
    {
      "tipo_medida" : "Emision_CO2",
      "valor" : 10.262,
      "unidad" : "gCO2/m2"
    },
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.01467,
      "unidad" : "kWh/m2"
    },
    {
      "precio_kWh" : 0.107,
      "unidad" : "€/kWh"
    },
    {
      "superficie" : 550,
      "unidad" : "m2"
    }
  ]
}
{
  "_id" : ObjectId("643e78af6453573ea0a56c3e"),
  "timestamp" : "2020-07-01T23:15:00Z",
  "sensor_id" : 2,
  "location_id" : 2,
  "medidas" : [
    {
      "tipo_medida" : "Emision_CO2",
      "valor" : 1.992,
      "unidad" : "gCO2/m2"
    },
    {
      "tipo_medida" : "Consumo_electrico",
      "valor" : 0.00259,
      "unidad" : "kWh/m2"
    },
    {
      "precio_kWh" : 0.107,
      "unidad" : "€/kWh"
    },
    {
      "superficie" : 550,
      "unidad" : "m2"
    }
  ]
}
```