

## Descripción corta del challenge

Realizar una serie de consultas a una colección de MongoDB, que almacena documentos ingestados desde sensores.

## La base de datos

Adjunto captura de pantalla:

```
{
  "timestamp": "2020-07-01T00:00:00Z",
  "sensor_id": 1,
  "location_id": 1,
  "medidas": [
    { "tipo_medida": "Temperatura", "valor": 16.61, "unidad": "°C" },
    { "tipo_medida": "Humedad_relativa", "valor": 83.74, "unidad": "%" }
  ]
}

{
  "timestamp": "2020-07-01T00:00:00Z",
  "sensor_id": 2,
  "location_id": 1,
  "medidas": [
    { "tipo_medida": "Emision_CO2", "valor": 1.572, "unidad": "gCO2/m2" },
    { "tipo_medida": "Consumo_electrico", "valor": 0.00188, "unidad": "kWh/m2" }
  ]
}
```

Cada documento es una medición con:

- timestamp: momento de la medición
- sensor\_id: tipo de sensor de la medición (valores: 1 / 2)
- location\_id: ciudad donde está ubicado el sensor (1: Valladolid, 2: Sevilla)
- medidas: dependiendo del tipo de sensor, array de documentos indicando los tipos de medida recogidos, valores y unidades.

5375 documentos en la BD

## Consulta 1

### Enunciado

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.

## Respuesta

```
db.sensores.insert([
  {
    "Ubicacion": "Valladolid",
    "medidas_sensor": [
      { "tipo_medida": "Temperatura",
        { "unidad": "°C" }
      ],
    "Coordenadas": [41.638597, 4.740186],
    "fecha_instalacion": new ISODate("2020-05-25T10:30:00Z"),
    "location_id": 1,
    "tipo_sensor": 1
  },
  {
    "Ubicacion": "Valladolid",
    "medidas_sensor": [
      { "tipo_medida": "Humedad_relativa",
        { "unidad": "%" }
      ],
    "Coordenadas": [41.638597, 4.740186],
    "fecha_instalacion": new ISODate("2020-05-25T10:30:00Z"),
    "location_id": 1,
    "tipo_sensor": 2
  },
  {
```

```
"Ubicacion": "Sevilla",
"medidas_sensor": [
  {"tipo_medida": "Temperatura"},
  {"unidad": "°C"}
],
"Coordenadas": [37.409311, -5.949939],
"fecha_instalacion": new ISODate("2020-05-28T11:30:00Z"),
"location_id": 2,
"tipo_sensor": 1
},
{
  "Ubicacion": "Sevilla",
  "medidas_sensor": [
    {"tipo_medida": "Humedad_relativa"},
    {"unidad": "%"}
  ],
  "Coordenadas": [37.409311, -5.949939],
  "fecha_instalacion": new ISODate("2020-05-28T11:30:00Z"),
  "location_id": 2,
  "tipo_sensor": 2
}]
)
```

```

... {
...   "Ubicacion": "Sevilla",
...   "medidas_sensor": [
...     {"tipo_medida": "Humedad_relativa"},
...     {"unidad": "%"}
...   ],
...   "Coordenadas": [37.409311, -5.949939],
...   "fecha_instalacion": new ISODate("2020-05-28T11:30:00Z"),
...   "location_id": 2,
...   "tipo_sensor": 2
... }
... )
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 4,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})

```

## Consulta 2

### Enunciado

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.

### Respuesta

```

db.datos_sensores.find({
  "timestamp":{$lte:"2020-07-10T23:45:00Z"},
  "location_id":1,
  "sensor_id":1
})
).count()

```

```

db.datos_sensores.find({
  "timestamp":{$lte:"2020-07-10T23:45:00Z"},
  "location_id":2,
  "sensor_id":1
})

```

```
).count()
```

/\* Para Sevilla devuelve 959 en vez de los esperados 960, quizá ese dato faltante sea el que fue a trash en el flujo de streamsets \*/

```
> db.datos_sensores.find({
... "timestamp":{$lte:"2020-07-10T23:45:00Z"},
... "location_id":1,
... "sensor_id":1
... })
... }.count()
960
> db.datos_sensores.find({
... "timestamp":{$lte:"2020-07-10T23:45:00Z"},
... "location_id":2,
... "sensor_id":1
... })
... }.count()
959
> █
```

## Consulta 3

### Enunciado

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.

### Respuesta

```
db.datos_sensores.aggregate([
  {
    $match:{
      "medidas":{
        $elemMatch:{
```

```

        "tipo_medida":"Temperatura",
        "valor":{$gt:55}
    }
}
}
},
{
    $group:{
        _id:"$location_id",
        "veces_superados_55_grados":{$sum:1}
    }
}
])

```

```

{ "_id" : 1, "veces_superados_55_grados" : 2 }
{ "_id" : 2, "veces_superados_55_grados" : 1 }
>

```

```

db.datos_sensores.find({
    "medidas":{
        $elemMatch:{
            "tipo_medida":"Temperatura",
            "valor":{$gt:55}
        }
    }
},
{
    "_id":0,
    "timestamp":1,
    "location_id": 1,
    "medidas": {$elemMatch: {"tipo_medida":"Temperatura"}}
})

```

```

... })
{ "timestamp" : "2020-07-05T17:15:00Z", "location_id" : 1, "medidas" : [ { "tipo_medida" : "Temperatura", "valor" : 67.27, "unidad" : "°C" } ] }
{ "timestamp" : "2020-07-08T03:00:00Z", "location_id" : 2, "medidas" : [ { "tipo_medida" : "Temperatura", "valor" : 82.64, "unidad" : "°C" } ] }
{ "timestamp" : "2020-07-11T19:30:00Z", "location_id" : 1, "medidas" : [ { "tipo_medida" : "Temperatura", "valor" : 125.48, "unidad" : "°C" } ] }
>

```

```

db.datos_sensores.update({

```

```

    "medidas":{
      $elemMatch:{
        "tipo_medida":"Temperatura",
        "valor":{$gt:55}
      }
    }
  },
  {
    $unset:{
      "medidas.$":1}
  },
  {
    multi:true
  }
)

```

```

... )
WriteResult({ "nMatched" : 3, "nUpserted" : 0, "nModified" : 3 })
>

```

## Consulta 4

### Enunciado

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.

### Respuesta

```

db.datos_sensores.find({
  "medidas":
    {$elemMatch:{tipo_medida:"Temperatura"}}
}).sort({
  "medidas.0.valor": 1

```

```
}).limit(1).pretty()
```

```
{
  "_id" : ObjectId("61aa57848a037a6912386eb1"),
  "timestamp" : "2020-07-12T07:45:00Z",
  "sensor_id" : 1,
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 1.31,
      "unidad" : "°C"
    },
    {
      "tipo_medida" : "Humedad_relativa",
      "valor" : 86.68,
      "unidad" : "%"
    }
  ]
}
```

```
db.datos_sensores.update({
  "_id":ObjectId("61aa57848a037a6912386eb1")
},
{
  $mul: {"medidas.0.valor": 1.2}
})
```

```
db.datos_sensores.find({"_id":ObjectId("61aa57848a037a6912386eb1")}).pretty()
```

```
{
  "_id" : ObjectId("61aa57848a037a6912386eb1"),
  "timestamp" : "2020-07-12T07:45:00Z",
  "sensor_id" : 1,
  "location_id" : 1,
  "medidas" : [
    {
      "tipo_medida" : "Temperatura",
      "valor" : 1.572,
      "unidad" : "°C"
    },
    {
      "tipo_medida" : "Humedad_relativa",
      "valor" : 86.68,
      "unidad" : "%"
    }
  ]
}
```



## Consulta 5

### Enunciado

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).

### Respuesta

```
db.datos_sensores.aggregate([ {
    $match: {
        $and: [
            {"location_id":1},
            {"medidas.tipo_medida":"Consumo_electrico"}
        ]
    }
},
{
    $addFields: {
        "DateFormat": {$dateFromString:{"dateString":"$timestamp"}}
    }
},
{
    $addFields: {
        "DMHFormat": {$dateToString:{"format": "%u-%H-%M", date: "$DateFormat"}}
    }
},
{
    $match: {
        $or: [
            { "DMHFormat": {$lte:"1-07-45"}},
            { "DMHFormat": {$gte:"5-16-00"}}
        ]
    }
}
```

```
},
{
    $sort: {"medidas.1.valor": -1}
},
{
    $project: {
        "_id":0,
        "ciudad": "Valladolid",
        "timestamp":1,
        "Dia de la semana y hora":{$dateToString:{format: "%u-%H", date:
"$DateFormat"}},
        "medidas":{$arrayElemAt:["$medidas",1]}
    }
},
{
    $limit: 3
}
]).pretty()
```

```
{
  "timestamp" : "2020-07-11T01:45:00Z",
  "ciudad" : "Valladolid",
  "Dia de la semana y hora" : "6-01",
  "medidas" : {
    "tipo_medida" : "Consumo_electrico",
    "valor" : 0.00235,
    "unidad" : "kWh/m2"
  }
}
{
  "timestamp" : "2020-07-11T22:15:00Z",
  "ciudad" : "Valladolid",
  "Dia de la semana y hora" : "6-22",
  "medidas" : {
    "tipo_medida" : "Consumo_electrico",
    "valor" : 0.00225,
    "unidad" : "kWh/m2"
  }
}
{
  "timestamp" : "2020-07-11T20:00:00Z",
  "ciudad" : "Valladolid",
  "Dia de la semana y hora" : "6-20",
  "medidas" : {
    "tipo_medida" : "Consumo_electrico",
    "valor" : 0.00214,
    "unidad" : "kWh/m2"
  }
}
```

```
db.datos_sensores.aggregate([
  $match: {
    $and: [
      {"location_id":2},
      {"medidas.tipo_medida":"Consumo_electrico"}
    ]
  },
  {
    $addFields: {"DateFormat": {$dateFromString:{"dateString":"$timestamp"}}}
  },
  {
    $addFields: {"DMHFormat": {$dateToString:{"format": "%u-%H-%M", date:
"$DateFormat"}}}
  },
  {
    $match: {
```

```

        $or: [
            { "DMHFormat": {$lte:"1-07-45"}},
            { "DMHFormat": {$gte:"5-16-00"}}
        ]
    }
},
{
    $sort: {"medidas.1.valor": -1}
},
{
    $project: {
        "_id":0,
        "ciudad": "Sevilla",
        "timestamp":1,
        "Dia de la semana y hora":{$dateToString:{format: "%u-%H", date:
"$DateFormat"}},
        "medidas":{$arrayElemAt:["$medidas",1]}
    }
},
{
    $limit: 3
} ]).pretty()

```

```
{
  "timestamp" : "2020-07-05T15:00:00Z",
  "ciudad" : "Sevilla",
  "Dia de la semana y hora" : "7-15",
  "medidas" : {
    "tipo_medida" : "Consumo_electrico",
    "valor" : 0.01711,
    "unidad" : "kWh/m2"
  }
}
{
  "timestamp" : "2020-07-05T08:00:00Z",
  "ciudad" : "Sevilla",
  "Dia de la semana y hora" : "7-08",
  "medidas" : {
    "tipo_medida" : "Consumo_electrico",
    "valor" : 0.01706,
    "unidad" : "kWh/m2"
  }
}
{
  "timestamp" : "2020-07-05T13:00:00Z",
  "ciudad" : "Sevilla",
  "Dia de la semana y hora" : "7-13",
  "medidas" : {
    "tipo_medida" : "Consumo_electrico",
    "valor" : 0.01698,
    "unidad" : "kWh/m2"
  }
}
```

/\* Valores mucho más altos en la oficina de Sevilla que en la de Valladolid. Para cada ciudad, los 3 consumos máximos se producen durante el mismo día. Se podría investigar más a fondo para entender los motivos (picos de trabajo, temperatura ambiente muy alta, etc.) \*/

## Consulta 6

### Enunciado

Se considera como perjudicial para la salud cualquier día en que el acumulado de las emisiones de CO<sub>2</sub> durante el día completo supere los 420 g CO<sub>2</sub>/m<sup>2</sup>. 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 CO<sub>2</sub> 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 CO<sub>2</sub> es más pequeña?

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

### Respuesta

```
db.datos_sensores.aggregate([  
    $unwind: "$medidas"  
},  
{  
    $match: {  
        "location_id":1,  
        "sensor_id":2,  
        "medidas.tipo_medida":"Emision_CO2"  
    }  
},  
{  
    $addFields: {  
        "DateFormat": {$dateFromString:{"dateString":"$timestamp"}}  
    }  
},  
{  
    $addFields: {"Dia_Mes": {$dayOfMonth:"$DateFormat"}}  
},  
{  
    $group: {  
        _id:"$Dia_Mes",  
        "suma_Emision_CO2":{$sum:"$medidas.valor"}  
    }  
},  
{  
    $match: {"suma_Emision_CO2":{$gte:420}}  
},  
{  
    $count: "dias_sobre_limite_permitido"  
}]  
{ "dias_sobre_limite_permitido" : 4 }  
db.datos_sensores.aggregate([
```

```

    $unwind: "$medidas"
  },
  {
    $match: {
      "location_id":2,
      "sensor_id":2,
      "medidas.tipo_medida":"Emision_CO2"}
    },
  {
    $addFields: {"DateFormat": {$dateFromString:{"dateString":"$timestamp"}}}
  },
  {
    $addFields: {"Dia_Mes": {$dayOfMonth:"$DateFormat"}}
  },
  {
    $group: {
      _id:"$Dia_Mes",
      "suma_Emision_CO2":{$sum:"$medidas.valor"}
    }
  },
  {
    $match: {"suma_Emision_CO2":{$gte:420}}
  },
  {
    $count: "dias_sobre_limite_permitido"
  }
])
... ])
{ "dias_sobre_limite_permitido" : 8 }

```

```

db.datos_sensores.aggregate([
  $unwind: "$medidas"
},
{
  $match: {

```

```

        "location_id":2,
        "sensor_id":2,
        "medidas.tipo_medida":"Emission_CO2"}
    },
    {
        $addFields: {"DateFormat": {$dateFromString:{"dateString":"$timestamp"}}}
    },
    {
        $addFields: {"Dia_Mes": {$dayOfMonth:"$DateFormat"}}
    },
    {
        $group: {
            _id:"$Dia_Mes",
            "suma_Emission_CO2":{$sum:"$medidas.valor"}
        }
    },
    {
        $match: {"suma_Emission_CO2":{$gte:420}}
    },
    {
        $count: "dias_sobre_limite_permitido"
    }
])

```

```

db.datos_sensores.aggregate([
    $unwind: "$medidas"
],
{
    $match: {
        "location_id":1,
        "sensor_id":2,
        "medidas.tipo_medida":"Emission_CO2"
    }
}

```



```

    },
    {
        $addFields: {"DateFormat": {$dateFromString:{"dateString":"$timestamp"}}}
    },
    {
        $addFields: {"Dia_Mes": {$dayOfMonth:"$DateFormat"}}
    },
    {
        $group: {
            _id:"$Dia_Mes",
            "suma_Emission_CO2":{$sum:"$medidas.valor"}
        }
    },
    {
        $sort: {"suma_Emission_CO2":-1}
    }
])

```

```

db.datos_sensores.aggregate([
    $unwind: "$medidas"
],
[
    $match: {
        "location_id":2,
        "sensor_id":2,
        "medidas.tipo_medida":"Emission_CO2"
    },
    {
        $addFields: {"DateFormat": {$dateFromString:{"dateString":"$timestamp"}}}
    },
    {
        $addFields: {"Dia_Mes": {$dayOfMonth:"$DateFormat"}}
    },
    {
        $group: {

```

```

        _id:"$Dia_Mes",
        "suma_Emission_CO2":{$sum:"$medidas.valor"}
    }
},
{
    $sort: {"suma_Emission_CO2":-1}
}
])

```

```

... ])
{ "_id" : 9, "suma_Emission_CO2" : 425.432 }
{ "_id" : 13, "suma_Emission_CO2" : 422.799 }
{ "_id" : 7, "suma_Emission_CO2" : 421.447 }
{ "_id" : 2, "suma_Emission_CO2" : 420.337 }
{ "_id" : 6, "suma_Emission_CO2" : 418.368 }
{ "_id" : 8, "suma_Emission_CO2" : 417.688 }
{ "_id" : 14, "suma_Emission_CO2" : 416.577 }
{ "_id" : 1, "suma_Emission_CO2" : 411.776 }
{ "_id" : 10, "suma_Emission_CO2" : 374.376 }
{ "_id" : 3, "suma_Emission_CO2" : 369.989 }
{ "_id" : 11, "suma_Emission_CO2" : 164.694 }
{ "_id" : 5, "suma_Emission_CO2" : 164.522 }
{ "_id" : 4, "suma_Emission_CO2" : 164.474 }
{ "_id" : 12, "suma_Emission_CO2" : 163.997 }

```

```

{ "_id" : 1, "suma_Emission_CO2" : 509.796 }
{ "_id" : 7, "suma_Emission_CO2" : 507.367 }
{ "_id" : 6, "suma_Emission_CO2" : 503.715 }
{ "_id" : 2, "suma_Emission_CO2" : 501.444 }
{ "_id" : 9, "suma_Emission_CO2" : 501.261 }
{ "_id" : 13, "suma_Emission_CO2" : 500.754 }
{ "_id" : 8, "suma_Emission_CO2" : 499.941000000000003 }
{ "_id" : 14, "suma_Emission_CO2" : 483.623 }
{ "_id" : 3, "suma_Emission_CO2" : 364.4 }
{ "_id" : 10, "suma_Emission_CO2" : 362.891 }
{ "_id" : 12, "suma_Emission_CO2" : 203.911 }
{ "_id" : 4, "suma_Emission_CO2" : 203.811 }
{ "_id" : 11, "suma_Emission_CO2" : 203.318 }
{ "_id" : 5, "suma_Emission_CO2" : 203.205 }

```

/\* Los 4 valores más bajos, para ambas ciudades, corresponden a los días que caen entre sábado y domingo (4, 5, 11 y 12). Incluso se nota el efecto del fin de semana en que, para las dos ciudades, los registros más bajos de consumo sin contar sábado y domingo, corresponden a los días que caen en viernes (días 3 y 10) \*/

## Consulta 7

### Enunciado

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.

### Respuesta

```
db.datos_sensores.aggregate([  
    $unwind: "$medidas"  
},  
{  
    $match: {  
        "location_id":1,  
        "sensor_id":2,  
        "medidas.tipo_medida":"Emision_CO2"  
    }  
},  
{  
    $addFields: {"DateFormat": {$dateFromString:{"dateString":"$timestamp"}}}  
},  
{  
    $addFields: {"Dia_Mes": {$dayOfMonth:"$DateFormat"},  
"Hora":{$hour:"$DateFormat"}}  
},  
{  
    $match: {"Hora":10}  
},  
{  
    $group: {  
        _id:"$Dia_Mes",  
        "media_sensores":{$avg:"$medidas.valor"}  
    }  
},  
{  
    $project: {  
        "_id":1,  
        "avg_Emision_CO2":{$round:[$multiply:["$media_sensores",4]],2]}  
    }  
},  
{
```

```

    $sort: {"avg_Emission_CO2":1}
}
])

db.datos_sensores.aggregate([
    $unwind: "$medidas"
},
{
    $match: {
        "location_id":2,
        "sensor_id":2,
        "medidas.tipo_medida":"Emission_CO2"
    }
},
{
    $addFields: {"DateFormat": {$dateFromString:{"dateString":"$timestamp"}}}
},
{
    $addFields: {
        "Dia_Mes": {$dayOfMonth:"$DateFormat"},
        "Hora":{$hour:"$DateFormat"}
    }
},
{
    $match: {"Hora":10}
},
{
    $group: {
        _id:"$Dia_Mes",
        "media_sensores":{$avg:"$medidas.valor"}
    }
},
{
    $project: {

```

```

        "_id":1,
        "avg_Emission_CO2":{$round:[$multiply:["$media_sensores",4]],2]}
    }
},
{
    $sort: {"avg_Emission_CO2":1}
}
])

```

```

{ "_id" : 11, "avg_Emission_CO2" : 7 }
{ "_id" : 12, "avg_Emission_CO2" : 7.15 }
{ "_id" : 4, "avg_Emission_CO2" : 7.24 }
{ "_id" : 5, "avg_Emission_CO2" : 7.31 }
{ "_id" : 7, "avg_Emission_CO2" : 33.84 }
{ "_id" : 1, "avg_Emission_CO2" : 35.24 }
{ "_id" : 6, "avg_Emission_CO2" : 35.7 }
{ "_id" : 2, "avg_Emission_CO2" : 36.28 }
{ "_id" : 10, "avg_Emission_CO2" : 36.49 }
{ "_id" : 14, "avg_Emission_CO2" : 36.62 }
{ "_id" : 3, "avg_Emission_CO2" : 36.7 }
{ "_id" : 9, "avg_Emission_CO2" : 36.97 }
{ "_id" : 8, "avg_Emission_CO2" : 37.22 }
{ "_id" : 13, "avg_Emission_CO2" : 38.04 }

```

```

{ "_id" : 11, "avg_Emission_CO2" : 8.79 }
{ "_id" : 5, "avg_Emission_CO2" : 8.86 }
{ "_id" : 12, "avg_Emission_CO2" : 8.88 }
{ "_id" : 4, "avg_Emission_CO2" : 8.98 }
{ "_id" : 10, "avg_Emission_CO2" : 31.98 }
{ "_id" : 3, "avg_Emission_CO2" : 32.48 }
{ "_id" : 8, "avg_Emission_CO2" : 38.66 }
{ "_id" : 7, "avg_Emission_CO2" : 38.94 }
{ "_id" : 14, "avg_Emission_CO2" : 40.17 }
{ "_id" : 6, "avg_Emission_CO2" : 40.67 }
{ "_id" : 2, "avg_Emission_CO2" : 40.94 }
{ "_id" : 13, "avg_Emission_CO2" : 41.68 }
{ "_id" : 9, "avg_Emission_CO2" : 42.05 }
{ "_id" : 1, "avg_Emission_CO2" : 42.48 }

```

## Consulta 8

### Enunciado

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.

#### Respuesta

```
db.datos_sensores.find({
  location_id:1,
  "medidas.tipo_medida":"Temperatura"
},
{
  "_id":0,
  "timestamp":1,
  "location_id":1,
  "medidas.$":1
}).sort({"medidas.0.valor": -1}).limit(2).pretty()
```

```
db.datos_sensores.update({
  location_id:1,
  "medidas.tipo_medida":"Temperatura"
},
{
  $inc:{"medidas.$.valor":-1.5}
},
{
  multi:true
})
```

```

{
  "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"
    }
  ]
}

```

```
WriteResult({"nMatched" : 1342, "nUpserted" : 0, "nModified" : 1342 })
```

```

{
  "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"
    }
  ]
}

```

## Consulta 9

### Enunciado

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.

### Respuesta

```
db.datos_sensores.aggregate([{\n    $addField: {"Hora": {$hour:{$dateFromString:{"dateString":"$timestamp"}}}}\n},\n{\n    $match: {\n        "location_id":1,\n        "sensor_id":1,\n        "Hora":{$gte:8, $lte:18}\n    }\n},\n{\n    $sort: {"medidas.1.valor":1}\n},\n{\n    $limit: 5\n},\n{\n    $project: {\n        "_id":0,\n        "timestamp":1,\n        "medidas":{$arrayElemAt:["$medidas",1]}\n    }\n}\n]).pretty()
```

```
db.datos_sensores.aggregate([{\n    $addField: {"Hora": {$hour:{$dateFromString:{"dateString":"$timestamp"}}}}\n
```



```
},
{
  $match: {
    "location_id":2,
    "sensor_id":1,
    "Hora":{"$gte:8, $lte:18}
  }
},
{
  $sort: {"medidas.1.valor":1}
},
{
  $limit: 5
},
{
  $project: {
    "_id":0,
    "timestamp":1,
    "medidas":{"$arrayElemAt:["$medidas",1]}
  }
}
]).pretty()
```

```
{
  "timestamp" : "2020-07-03T16:45:00Z",
  "medidas" : {
    "tipo_medida" : "Humedad_relativa",
    "valor" : 5.74,
    "unidad" : "%"
  }
}
{
  "timestamp" : "2020-07-06T17:00:00Z",
  "medidas" : {
    "tipo_medida" : "Humedad_relativa",
    "valor" : 6.81,
    "unidad" : "%"
  }
}
{
  "timestamp" : "2020-07-10T17:45:00Z",
  "medidas" : {
    "tipo_medida" : "Humedad_relativa",
    "valor" : 7.6,
    "unidad" : "%"
  }
}
{
  "timestamp" : "2020-07-14T17:30:00Z",
  "medidas" : {
    "tipo_medida" : "Humedad_relativa",
    "valor" : 8.14,
    "unidad" : "%"
  }
}
{
  "timestamp" : "2020-07-05T17:15:00Z",
  "medidas" : {
    "tipo_medida" : "Humedad_relativa",
    "valor" : 8.16,
    "unidad" : "%"
  }
}
>
```

```

{
    "timestamp" : "2020-07-05T14:00:00Z",
    "medidas" : {
        "tipo_medida" : "Humedad_relativa",
        "valor" : 0.3,
        "unidad" : "%"
    }
}
{
    "timestamp" : "2020-07-01T16:15:00Z",
    "medidas" : {
        "tipo_medida" : "Humedad_relativa",
        "valor" : 0.61,
        "unidad" : "%"
    }
}
{
    "timestamp" : "2020-07-02T15:15:00Z",
    "medidas" : {
        "tipo_medida" : "Humedad_relativa",
        "valor" : 0.94,
        "unidad" : "%"
    }
}
{
    "timestamp" : "2020-07-14T14:45:00Z",
    "medidas" : {
        "tipo_medida" : "Humedad_relativa",
        "valor" : 1.25,
        "unidad" : "%"
    }
}
{
    "timestamp" : "2020-07-10T13:45:00Z",
    "medidas" : {
        "tipo_medida" : "Humedad_relativa",
        "valor" : 2.1,
        "unidad" : "%"
    }
}
}

```

/\* Parece que la franja horaria, para ambas ciudades, es un factor más decisivo sobre los valores mínimos que el hecho de considerar un día concreto. De hecho, vemos que para Valladolid los valores mínimos se concentran en la franja entre las 16:45 y las 17:45, siendo cada una de las 5 medidas pertenecientes a días distintos. En Sevilla ocurre algo parecido, sólo que la franja con los valores mínimos se mueve a las 13:45-16:15.

Otra diferencia entre las dos ciudades, es que los valores de humedad relativa de Sevilla son bastante menores. \*/

## Consulta 10

### Enunciado

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)**.

### Respuesta

```
db.datos_sensores.update({
  "location_id":1,
  "sensor_id":2
},
{
  $push:{
    "medidas": {
      $each: [
        {"precio_kWh":0.102,"unidad":"€/kWh"},
        {"superficie":450,"unidad":"m2"}
      ]
    }
  }
},
{
  multi:true
})
```

```

db.datos_sensores.update({
  "location_id":2,
  "sensor_id":2
},
{
  $push:{
    "medidas": {
      $each: [
        {"precio_kWh":0.107,"unidad":"€/kWh"},
        {"superficie":550,"unidad":"m2"}
      ]
    }
  }
},
{
  multi:true
}
)

```

```

... )
WriteResult({ "nMatched" : 1344, "nUpserted" : 0, "nModified" : 1344 })
>

```

```

... ,
WriteResult({ "nMatched" : 1344, "nUpserted" : 0, "nModified" : 1344 })

```

```

db.datos_sensores.find({
  "timestamp": {
    $in:[
      "2020-07-01T08:00:00Z",
      "2020-07-01T23:15:00Z"
    ]
  }
}
)

```

```
{ "_id" : ObjectId("61aa57158a037a6912385e37"), "timestamp" : "2020-07-01T08:00:00Z", "sensor_id" : 1, "location_id" : 1,
"medidas" : [ { "tipo_medida" : "Temperatura", "valor" : 18.7, "unidad" : "°C" }, { "tipo_medida" : "Humedad_relativa",
"valor" : 77.36, "unidad" : "%" } ] }
{ "_id" : ObjectId("61aa57158a037a6912385e39"), "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_electri
co", "valor" : 0.01788, "unidad" : "kwh/m2" }, { "precio_kwh" : 0.102, "unidad" : "€/kwh" }, { "superficie" : 450, "unida
d" : "m2" } ] }
{ "_id" : ObjectId("61aa571a8a037a6912385e9a"), "timestamp" : "2020-07-01T08:00:00Z", "sensor_id" : 1, "location_id" : 2,
"medidas" : [ { "tipo_medida" : "Temperatura", "valor" : 22.6, "unidad" : "°C" }, { "tipo_medida" : "Humedad_relativa",
"valor" : 40.73, "unidad" : "%" } ] }
{ "_id" : ObjectId("61aa571a8a037a6912385e9c"), "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_elec
trico", "valor" : 0.01467, "unidad" : "kwh/m2" }, { "precio_kwh" : 0.107, "unidad" : "€/kwh" }, { "superficie" : 550, "un
idad" : "m2" } ] }
{ "_id" : ObjectId("61aa571a8a037a6912385f2b"), "timestamp" : "2020-07-01T23:15:00Z", "sensor_id" : 1, "location_id" : 1,
"medidas" : [ { "tipo_medida" : "Temperatura", "valor" : 10.06, "unidad" : "°C" }, { "tipo_medida" : "Humedad_relativa",
"valor" : 85.96, "unidad" : "%" } ] }
{ "_id" : ObjectId("61aa571a8a037a6912385f2d"), "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_elect
rico", "valor" : 0.00164, "unidad" : "kwh/m2" }, { "precio_kwh" : 0.102, "unidad" : "€/kwh" }, { "superficie" : 450, "un
idad" : "m2" } ] }
{ "_id" : ObjectId("61aa571f8a037a6912385f8e"), "timestamp" : "2020-07-01T23:15:00Z", "sensor_id" : 1, "location_id" : 2,
"medidas" : [ { "tipo_medida" : "Temperatura", "valor" : 25.58, "unidad" : "°C" }, { "tipo_medida" : "Humedad_relativa",
"valor" : 30.86, "unidad" : "%" } ] }
{ "_id" : ObjectId("61aa571f8a037a6912385f90"), "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_elect
rico", "valor" : 0.00259, "unidad" : "kwh/m2" }, { "precio_kwh" : 0.107, "unidad" : "€/kwh" }, { "superficie" : 550, "un
idad" : "m2" } ] }
```