



CONSULTAS EN MONGODB

Bases de datos

DESCRIPCIÓN BREVE

Migración SQL a MongoDB

Andrés Morales González

IES Gonzalo Nazareno

Contenido

Enunciado:	2
Inserción de datos	6
Eliminación de datos	9
Actualizar	11
Consultas simples	16
ARRAYS:	18
Consultas embebidas	20
Agrupación:	24

Enunciado:

1. Debes seleccionar al menos cuatro tablas de tu proyecto que incluyan una relación N:M y una 1:N con el mayor número de tipos de datos soportados en MongoDB. Antes de continuar la profesora deberá dar el visto bueno a las mismas.
2. Debes pasar de SQL a MongoDB.
3. Inserta varios documentos utilizando los dos métodos de inserción de MongoDB
4. Elimina varios documentos utilizando los dos métodos de eliminación de MongoDB
5. Actualiza varios documentos utilizando los tres métodos de actualización de MongoDB
6. Consultas:
 5. Al menos incluye 5 consultas de datos simples
 6. Al menos 3 consultas con arrays
 7. Al menos 3 consultas con documentos embebidos
 8. Al menos 1 consulta de agrupación

NOTA: Debes utilizar en las consultas proyecciones, operadores, ordenación y los modificadores sort y limit. Se valorará la complejidad de las mismas

Para esta parte lo que he hecho ha sido una migración desde SQL a MongoDB

Donde si nos metemos en la consola de `mongosh → show dbs;`

Nos aparecerá lo que es nuestro proyecto que teníamos en SQL en MongoDB

```
test> show dbs;
HIPOPOTAMO          680.00 KiB
Proyecto_mongo      72.00 KiB
Proyecto_universidad 8.00 KiB
admin               40.00 KiB
colegio             40.00 KiB
config              60.00 KiB
local               72.00 KiB
```

Usamos el comando:

`Use hipopotamo;`

```
test> use HIPOPOTAMO;
switched to db HIPOPOTAMO
```

Y mostramos las tablas que contienen:

`Show collections;`

```
HIPOPOTAMO> show collections;
ALUMNO
APP_PROCESOS_INDRUSTRIALES
AREA_CONOCIMIENTO
ASIGNATURA
CATEGORIA
CONTRATO
CONVOCATORIA
DEPARTAMENTO
DIRECCION
EMPRESA
GRUPO
NOTAS
ORGANISMO_PUBLICO
PROFESOR
PROYECTO_INVESTIGACION
RESPONSABILIDAD
TITULACION
```

Y vemos que tenemos totalmente nuestra base de datos que teníamos de SQL ahora mismo en MongoDB.

Ahora procederemos a hacer la inserción de varios documentos usando los dos métodos de inserción de datos que hemos aprendido gracias al curso de MongoDB.

Meteremos algunos alumnos , profesores y alguna asignatura, primeramente veremos los datos que tenemos metidos en las tablas que usare:

```
db.ALUMNO.find()
```

```
HIPOPOTAMO> db.ALUMNO.find()
[
  {
    _id: Long('1'),
    ID_ALUMNO: Long('1'),
    NOMBRE: 'Juan',
    APELLIDO1: 'Pérez',
    APELLIDO2: 'García',
    DNI: '12345678A',
    ID_GRUPO: Long('1'),
    CODIGO_POSTAL: Long('28001'),
    ID_DIRECCION: Long('1')
  },
  {
    _id: Long('3'),
    ID_ALUMNO: Long('3'),
    NOMBRE: 'Carlos',
    APELLIDO1: 'García',
    APELLIDO2: 'Fernández',
    DNI: '87654321C',
    ID_GRUPO: Long('2'),
    CODIGO_POSTAL: Long('28002'),
    ID_DIRECCION: Long('2')
  },
  {
    _id: Long('4'),
    ID_ALUMNO: Long('4'),
    NOMBRE: 'Sofía',
    APELLIDO1: 'Hernández',
    APELLIDO2: 'Martínez',
    DNI: '7654321D',
    ID_GRUPO: Long('2'),
    CODIGO_POSTAL: Long('28002'),
    ID_DIRECCION: Long('2')
  },
  {
    _id: Long('5'),
    ID_ALUMNO: Long('5'),
    NOMBRE: 'Javier',
    APELLIDO1: 'López',
    APELLIDO2: 'González',
    DNI: '6543210E',
    ID_GRUPO: Long('3'),
    CODIGO_POSTAL: Long('28003'),
    ID_DIRECCION: Long('3')
  },
  {
    _id: Long('6'),
    ID_ALUMNO: Long('6'),
    NOMBRE: 'Laura',
    APELLIDO1: 'Fernández',
    APELLIDO2: 'Pérez',
    DNI: '5432109F',
    ID_GRUPO: Long('3'),
    CODIGO_POSTAL: Long('28003'),
    ID_DIRECCION: Long('3')
  }
]
```

```
db.PROFESOR.find()
```

```
HIPOPOTAMO> db.PROFESOR.find()
[
  {
    _id: Long('1'),
    ID_PROFESOR: Long('1'),
    NOMBREPROFESOR: 'María',
    APELLIDO1: 'González',
    APELLIDO2: 'López',
    ID_DEPARTAMENTO: Long('1'),
    ID_CATEGORIA: Long('1'),
    CODIGO_POSTAL: Long('28001'),
    ID_DIRECCION: Long('1')
  },
  {
    _id: Long('2'),
    ID_PROFESOR: Long('2'),
    NOMBREPROFESOR: 'José',
    APELLIDO1: 'Martínez',
    APELLIDO2: 'García',
    ID_DEPARTAMENTO: Long('2'),
    ID_CATEGORIA: Long('2'),
    CODIGO_POSTAL: Long('8002'),
    ID_DIRECCION: Long('2')
  },
  {
    _id: Long('3'),
    ID_PROFESOR: Long('3'),
    NOMBREPROFESOR: 'Ana',
    APELLIDO1: 'Sánchez',
    APELLIDO2: 'Fernández',
    ID_DEPARTAMENTO: Long('3'),
    ID_CATEGORIA: Long('3'),
    CODIGO_POSTAL: Long('41001'),
    ID_DIRECCION: Long('3')
  },
  {
    _id: Long('4'),
    ID_PROFESOR: Long('4'),
    NOMBREPROFESOR: 'David',
    APELLIDO1: 'Rodríguez',
    APELLIDO2: 'Pérez',
    ID_DEPARTAMENTO: Long('4'),
    ID_CATEGORIA: Long('4'),
    CODIGO_POSTAL: Long('46001'),
    ID_DIRECCION: Long('4')
  },
  {
    _id: Long('5'),
    ID_PROFESOR: Long('5'),
    NOMBREPROFESOR: 'Laura',
    APELLIDO1: 'López',
    APELLIDO2: 'Martínez',
    ID_DEPARTAMENTO: Long('5'),
    ID_CATEGORIA: Long('5'),
    CODIGO_POSTAL: Long('28003'),
    ID_DIRECCION: Long('5')
  }
]
```

Inserción de datos

Ahora procederemos a meter algunos alumnos nuevos:

Con insertOne:

```
db.ALUMNO.insertOne({
  _id: Long(10),
  ID_ALUMNO: Long(10),
  NOMBRE: "Andrea",
  APELLIDO1: "Santos",
  APELLIDO2: "Esteban",
  DNI: "12345678C",
  ID_GRUPO: Long(2),
  CODIGO_POSTAL: Long(28002),
  ID_DIRECCION: Long(2)
})
```

```
HIPOPOTAMO> db.ALUMNO.insertOne({
...   _id: Long(10),
...   ID_ALUMNO: Long(10),
...   NOMBRE: "Andrea",
...   APELLIDO1: "Santos",
...   APELLIDO2: "Esteban",
...   DNI: "12345678C",
...   ID_GRUPO: Long(2),
...   CODIGO_POSTAL: Long(28002),
...   ID_DIRECCION: Long(2)
... })
{ acknowledged: true, insertedId: Long('10') }
```

Con insertMany:

```
db.ALUMNO.insertMany([
  {
    _id: Long(11),
    ID_ALUMNO: Long(11),
    NOMBRE: "Concha",
    APELLIDO1: "De la Rosa",
    APELLIDO2: "Cabezas",
    DNI: "23456789D",
    ID_GRUPO: Long(2),
    CODIGO_POSTAL: Long(41013),
    ID_DIRECCION: Long(2)
  },
  {
    _id: Long(12),
    ID_ALUMNO: Long(12),
    NOMBRE: "Claudia",
    APELLIDO1: "Morales",
    APELLIDO2: "González",
    DNI: "34567890E",
    ID_GRUPO: Long(2),
    CODIGO_POSTAL: Long(41704),
    ID_DIRECCION: Long(2)
  }
])
```

```
HIPOPOTAMO> db.ALUMNO.insertMany([
...   {
...     _id: Long(11),
...     ID_ALUMNO: Long(11),
...     NOMBRE: "Concha",
...     APELLIDO1: "De la Rosa",
...     APELLIDO2: "Cabezas",
...     DNI: "23456789D",
...     ID_GRUPO: Long(2),
...     CODIGO_POSTAL: Long(41013),
...     ID_DIRECCION: Long(2)
...   },
...   {
...     _id: Long(12),
...     ID_ALUMNO: Long(12),
...     NOMBRE: "Claudia",
...     APELLIDO1: "Morales",
...     APELLIDO2: "González",
...     DNI: "34567890E",
...     ID_GRUPO: Long(2),
...     CODIGO_POSTAL: Long(41704),
...     ID_DIRECCION: Long(2)
...   }
... ])
{
  acknowledged: true,
  insertedIds: { '0': Long('11'), '1': Long('12') }
}
```


Comprobación de que esta todo, lo vemos con el comando anterior:

```
db.ALUMNO.find()
```

```
HIPOPOTAMO> db.ALUMNO.find()
[
  {
    _id: Long('1'),
    ID_ALUMNO: Long('1'),
    NOMBRE: 'Juan',
    APELLIDO1: 'Pérez',
    APELLIDO2: 'García',
    DNI: '12345678A',
    ID_GRUPO: Long('1'),
    CODIGO_POSTAL: Long('28001'),
    ID_DIRECCION: Long('1')
  },
  {
    _id: Long('3'),
    ID_ALUMNO: Long('3'),
    NOMBRE: 'Carlos',
    APELLIDO1: 'García',
    APELLIDO2: 'Fernández',
    DNI: '87654321C',
    ID_GRUPO: Long('2'),
    CODIGO_POSTAL: Long('28002'),
    ID_DIRECCION: Long('2')
  },
  {
    _id: Long('4'),
    ID_ALUMNO: Long('4'),
    NOMBRE: 'Sofía',
    APELLIDO1: 'Hernández',
    APELLIDO2: 'Martínez',
    DNI: '7654321D',
    ID_GRUPO: Long('2'),
    CODIGO_POSTAL: Long('28002'),
    ID_DIRECCION: Long('2')
  },
  {
    _id: Long('5'),
    ID_ALUMNO: Long('5'),
    NOMBRE: 'Javier',
    APELLIDO1: 'López',
    APELLIDO2: 'González',
    DNI: '6543210E',
    ID_GRUPO: Long('3'),
    CODIGO_POSTAL: Long('28003'),
    ID_DIRECCION: Long('3')
  },
  {
    _id: Long('6'),
    ID_ALUMNO: Long('6'),
    NOMBRE: 'Laura',
    APELLIDO1: 'Fernández',
    APELLIDO2: 'Pérez',
    DNI: '5432109F',
    ID_GRUPO: Long('3'),
    CODIGO_POSTAL: Long('28003'),
    ID_DIRECCION: Long('3')
  },
  {
    _id: Long('10'),
    ID_ALUMNO: Long('10'),
    NOMBRE: 'Andrea',
    APELLIDO1: 'Santos',
    APELLIDO2: 'Esteban',
    DNI: '12345678C',
    ID_GRUPO: Long('2'),
    CODIGO_POSTAL: Long('28002'),
    ID_DIRECCION: Long('2')
  },
  {
    _id: Long('11'),
    ID_ALUMNO: Long('11'),
    NOMBRE: 'Concha',
    APELLIDO1: 'De la Rosa',
    APELLIDO2: 'Cabezas',
    DNI: '23456789D',
    ID_GRUPO: Long('2'),
    CODIGO_POSTAL: Long('41013'),
    ID_DIRECCION: Long('2')
  },
  {
    _id: Long('12'),
    ID_ALUMNO: Long('12'),
    NOMBRE: 'Claudia',
    APELLIDO1: 'Morales',
    APELLIDO2: 'González',
    DNI: '34567890E',
    ID_GRUPO: Long('2'),
    CODIGO_POSTAL: Long('41704'),
    ID_DIRECCION: Long('2')
  }
]
```

Como vemos esta todo perfectamente metido en la tabla de ALUMNO.

Eliminación de datos

Ahora procederemos a eliminar los alumnos que acabamos de meter, con los métodos aprendidos por el curso de MongoDB.

Usando `deleteOne`:

```
db.ALUMNO.deleteOne({ NOMBRE: "Andrea", APELLIDO1: "Santos",  
APELLIDO2: "Esteban" })
```

```
HIPOPOTAMO> db.ALUMNO.deleteOne({ NOMBRE: "Andrea", APELLIDO1: "Santos", APELLIDO2: "Esteban" })  
{ acknowledged: true, deletedCount: 1 }  
HIPOPOTAMO> |
```

Usando `deleteMany`:

```
db.ALUMNO.deleteMany({  
  $or: [  
    { NOMBRE: "Concha", APELLIDO1: "De la Rosa", APELLIDO2:  
      "Cabezas" },  
    { NOMBRE: "Claudia", APELLIDO1: "Morales", APELLIDO2:  
      "González" }  
  ]  
})
```

```
HIPOPOTAMO> db.ALUMNO.deleteMany({  
...   $or: [  
...     { NOMBRE: "Concha", APELLIDO1: "De la Rosa", APELLIDO2: "Cabezas" },  
...     { NOMBRE: "Claudia", APELLIDO1: "Morales", APELLIDO2: "González" }  
...   ]  
... })  
{ acknowledged: true, deletedCount: 2 }  
HIPOPOTAMO> |
```

Comprobación:

db.ALUMNO.find()

```
HIPOPOTAMO> db.ALUMNO.find()
[
  {
    _id: Long('1'),
    ID_ALUMNO: Long('1'),
    NOMBRE: 'Juan',
    APELLIDO1: 'Pérez',
    APELLIDO2: 'García',
    DNI: '12345678A',
    ID_GRUPO: Long('1'),
    CODIGO_POSTAL: Long('28001'),
    ID_DIRECCION: Long('1')
  },
  {
    _id: Long('3'),
    ID_ALUMNO: Long('3'),
    NOMBRE: 'Carlos',
    APELLIDO1: 'García',
    APELLIDO2: 'Fernández',
    DNI: '87654321C',
    ID_GRUPO: Long('2'),
    CODIGO_POSTAL: Long('28002'),
    ID_DIRECCION: Long('2')
  },
  {
    _id: Long('4'),
    ID_ALUMNO: Long('4'),
    NOMBRE: 'Sofía',
    APELLIDO1: 'Hernández',
    APELLIDO2: 'Martínez',
    DNI: '7654321D',
    ID_GRUPO: Long('2'),
    CODIGO_POSTAL: Long('28002'),
    ID_DIRECCION: Long('2')
  },
  {
    _id: Long('5'),
    ID_ALUMNO: Long('5'),
    NOMBRE: 'Javier',
    APELLIDO1: 'López',
    APELLIDO2: 'González',
    DNI: '6543210E',
    ID_GRUPO: Long('3'),
    CODIGO_POSTAL: Long('28003'),
    ID_DIRECCION: Long('3')
  },
  {
    _id: Long('6'),
    ID_ALUMNO: Long('6'),
    NOMBRE: 'Laura',
    APELLIDO1: 'Fernández',
    APELLIDO2: 'Pérez',
    DNI: '5432109F',
    ID_GRUPO: Long('3'),
    CODIGO_POSTAL: Long('28003'),
    ID_DIRECCION: Long('3')
  }
]
HIPOPOTAMO> |
```

Actualizar

Ahora procederemos a actualizar los salarios de los profesores, con los métodos aprendidos por el curso de MongoDB.

Primero vemos los datos que tenemos:

```
db.PROYECTO_INVESTIGACION.find()
```

```

HIPOPOTAMO> db.PROYECTO_INVESTIGACION.find()
[
  {
    _id: Long('1'),
    ID_PROYECTO: Long('1'),
    NOMBRE_PROYECTO: 'Desarrollo de nuevos materiales',
    FECHA_INICIO: ISODate('2023-10-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-09-30T00:00:00.000Z'),
    ID_PROFESOR: Long('1'),
    ID_ORGANISMO_PUBLICO: Long('1'),
    SUBVENCION: Long('10000')
  },
  {
    _id: Long('3'),
    ID_PROYECTO: Long('3'),
    NOMBRE_PROYECTO: 'Investigación en Inteligencia Artifi
    FECHA_INICIO: ISODate('2024-05-10T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-08-20T00:00:00.000Z'),
    ID_PROFESOR: Long('2'),
    ID_ORGANISMO_PUBLICO: Long('2'),
    SUBVENCION: Long('20000')
  },
  {
    _id: Long('4'),
    ID_PROYECTO: Long('4'),
    NOMBRE_PROYECTO: 'Estudio de impacto ambiental',
    FECHA_INICIO: ISODate('2024-07-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-12-31T00:00:00.000Z'),
    ID_PROFESOR: Long('3'),
    ID_ORGANISMO_PUBLICO: Long('3'),
    SUBVENCION: Long('18000')
  },
  {
    _id: Long('5'),
    ID_PROYECTO: Long('5'),
    NOMBRE_PROYECTO: 'Desarrollo de vacunas',
    FECHA_INICIO: ISODate('2024-09-15T00:00:00.000Z'),
    FECHA_FIN: ISODate('2027-01-10T00:00:00.000Z'),
    ID_PROFESOR: Long('4'),
    ID_ORGANISMO_PUBLICO: Long('4'),
    SUBVENCION: Long('25000')
  },
  {
    _id: Long('6'),
    ID_PROYECTO: Long('6'),
    NOMBRE_PROYECTO: 'Análisis de datos',
    FECHA_INICIO: ISODate('2024-11-03T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-10-15T00:00:00.000Z'),
    ID_PROFESOR: Long('5'),

```

Con `updateOne`

```

db.proyecto_investigacion.updateOne(
  { _id: Long('1') },
  { $set: { SUBVENCION: Long('15000') } }
)

```

Y comprobamos con el comando anterior si se ha producido el cambio:

```
HIPOPOTAMO> db.PROYECTO_INVESTIGACION.updateOne(
...   { _id: Long('1') },
...   { $set: { SUBVENCION: Long('15000') } }
... )
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
HIPOPOTAMO> db.PROYECTO_INVESTIGACION.find()
[
  {
    _id: Long('1'),
    ID_PROYECTO: Long('1'),
    NOMBRE_PROYECTO: 'Desarrollo de nuevos materiales',
    FECHA_INICIO: ISODate('2023-10-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-09-30T00:00:00.000Z'),
    ID_PROFESOR: Long('1'),
    ID_ORGANISMO_PUBLICO: Long('1'),
    SUBVENCION: Long('15000')
  },
  ...
]
```

Con `updateMany`:

```
db.PROYECTO_INVESTIGACION.updateMany(
  { _id: { $in: [Long('3'), Long('4')] } },
  { $mul: { SUBVENCION: 1.75 } }
)
```

Lo que he hecho ha sido incrementar en un 75% los proyectos con id 3 y 4.

```

HIPOPOTAMO> db.PROYECTO_INVESTIGACION.find()
[
  {
    _id: Long('1'),
    ID_PROYECTO: Long('1'),
    NOMBRE_PROYECTO: 'Desarrollo de nuevos materiales',
    FECHA_INICIO: ISODate('2023-10-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-09-30T00:00:00.000Z'),
    ID_PROFESOR: Long('1'),
    ID_ORGANISMO_PUBLICO: Long('1'),
    SUBVENCION: Long('15000')
  },
  {
    _id: Long('3'),
    ID_PROYECTO: Long('3'),
    NOMBRE_PROYECTO: 'Investigación en Inteligencia Artificial',
    FECHA_INICIO: ISODate('2024-05-10T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-08-20T00:00:00.000Z'),
    ID_PROFESOR: Long('2'),
    ID_ORGANISMO_PUBLICO: Long('2'),
    SUBVENCION: 35000
  },
  {
    _id: Long('4'),
    ID_PROYECTO: Long('4'),
    NOMBRE_PROYECTO: 'Estudio de impacto ambiental',
    FECHA_INICIO: ISODate('2024-07-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-12-31T00:00:00.000Z'),
    ID_PROFESOR: Long('3'),
    ID_ORGANISMO_PUBLICO: Long('3'),
    SUBVENCION: 31500
  },
  {
    _id: Long('5'),
    ID_PROYECTO: Long('5'),
    NOMBRE_PROYECTO: 'Desarrollo de vacunas',
    FECHA_INICIO: ISODate('2024-09-15T00:00:00.000Z'),
    FECHA_FIN: ISODate('2027-01-10T00:00:00.000Z'),
    ID_PROFESOR: Long('4'),
    ID_ORGANISMO_PUBLICO: Long('4'),
    SUBVENCION: Long('25000')
  }
]

```

Con `replaceOne`:

```
db.PROYECTO_INVESTIGACION.replaceOne(
  { _id: Long('6') },
  {
    _id: Long('6'),
    ID_PROYECTO: Long('6'),
    NOMBRE_PROYECTO: 'Análisis de datos',
    FECHA_INICIO: ISODate('2024-11-03T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-10-15T00:00:00.000Z'),
    ID_PROFESOR: Long('5'),
    ID_ORGANISMO_PUBLICO: Long('4'),
    SUBVENCION: Long('33000')
  }
)
```

```
{
  _id: Long('6'),
  ID_PROYECTO: Long('6'),
  NOMBRE_PROYECTO: 'Análisis de datos',
  FECHA_INICIO: ISODate('2024-11-03T00:00:00.000Z'),
  FECHA_FIN: ISODate('2026-10-15T00:00:00.000Z'),
  ID_PROFESOR: Long('5'),
  ID_ORGANISMO_PUBLICO: Long('4'),
  SUBVENCION: Long('33000')
}
]
HIPOPOTAMO> |
```


Consultas simples:

Mostrar los alumnos de forma alfabética, por nombres:

```
db.ALUMNO.find({}, { NOMBRE: 1, APELLIDO1: 1, APELLIDO2: 1,
_id: 0 }).sort({ NOMBRE: 1 }).collation({ locale: "en" })
```

```
HIPOPOTAMO> db.ALUMNO.find({}, { NOMBRE: 1, APELLIDO1: 1, APELLIDO2: 1, _id: 0 }).sort({ NOMBRE: 1 }).collation({ locale: "en" })
[
  { NOMBRE: 'Carlos', APELLIDO1: 'García', APELLIDO2: 'Fernández' },
  { NOMBRE: 'Javier', APELLIDO1: 'López', APELLIDO2: 'González' },
  { NOMBRE: 'Juan', APELLIDO1: 'Pérez', APELLIDO2: 'García' },
  { NOMBRE: 'Laura', APELLIDO1: 'Fernández', APELLIDO2: 'Pérez' },
  { NOMBRE: 'Sofía', APELLIDO1: 'Hernández', APELLIDO2: 'Martínez' }
]
```

Dame las notas de mayor a menor, y dime las fechas:

```
db.NOTAS.find({}, { CALIFICACION: 1, FECHA_CALIFICACION: 1,
_id: 0 }).sort({ CALIFICACION: -1 })
```

```
HIPOPOTAMO> db.NOTAS.find({}, { CALIFICACION: 1, FECHA_CALIFICACION: 1, _id: 0 }).sort({ CALIFICACION: -1 })
[
  {
    CALIFICACION: Long('9'),
    FECHA_CALIFICACION: ISODate('2024-02-10T00:00:00.000Z')
  },
  {
    CALIFICACION: Long('8'),
    FECHA_CALIFICACION: ISODate('2024-01-15T00:00:00.000Z')
  },
  {
    CALIFICACION: Long('8'),
    FECHA_CALIFICACION: ISODate('2024-02-12T00:00:00.000Z')
  },
  {
    CALIFICACION: Long('7'),
    FECHA_CALIFICACION: ISODate('2024-02-15T00:00:00.000Z')
  },
  {
    CALIFICACION: Long('6'),
    FECHA_CALIFICACION: ISODate('2024-02-18T00:00:00.000Z')
  }
]
```

Ordenar por orden alfabético los profesores por apellidos, y slagan su codgo postal:

```
db.PROFESOR.find({}, { NOMBREPROFESOR: 1, APELLIDO1: 1,
APELLIDO2: 1, CODIGO_POSTAL: 1, _id: 0 }).sort({ APELLIDO1:
1 });
```

```
HIPOPOTAMO> db.PROFESOR.find({}, { NOMBREPROFESOR: 1, APELLIDO1: 1, APELLIDO2: 1, CODIGO_POSTAL: 1, _id: 0 }).sort({ APELLIDO1: 1 });
[
  {
    NOMBREPROFESOR: 'María',
    APELLIDO1: 'González',
    APELLIDO2: 'López',
    CODIGO_POSTAL: Long('28001')
  },
  {
    NOMBREPROFESOR: 'Laura',
    APELLIDO1: 'López',
    APELLIDO2: 'Martínez',
    CODIGO_POSTAL: Long('28003')
  },
  {
    NOMBREPROFESOR: 'José',
    APELLIDO1: 'Martínez',
    APELLIDO2: 'García',
    CODIGO_POSTAL: Long('8002')
  },
  {
    NOMBREPROFESOR: 'David',
    APELLIDO1: 'Rodríguez',
    APELLIDO2: 'Pérez',
    CODIGO_POSTAL: Long('46001')
  },
  {
    NOMBREPROFESOR: 'Ana',
    APELLIDO1: 'Sánchez',
    APELLIDO2: 'Fernández',
    CODIGO_POSTAL: Long('41001')
  }
]
HIPOPOTAMO> |
```

Usando proyecciones serias:

```
db.PROFESOR.aggregate([ { $addFields: { nombreCompleto: { $concat: [ { $toUpper: "$NOMBREPROFESOR" }, " ", { $toUpper: "$APELLIDO1" }, " ", { $toUpper: "$APELLIDO2" }] } } }, { $project: { _id: 0, nombreCompleto: 1 } }, { $sort: { nombreCompleto: 1 } } ] );
```

```
HIPOPOTAMO> db.PROFESOR.aggregate([ { $addFields: { nombreCompleto: { $concat: [ { $toUpper: "$NOMBREPROFESOR" }, " ", { $toUpper: "$APELLIDO1" }, " ", { $toUpper: "$APELLIDO2" }] } } }, { $project: { _id: 0, nombreCompleto: 1 } }, { $sort: { nombreCompleto: 1 } } ] );
[
  { nombreCompleto: 'ANA SÁNCHEZ FERNÁNDEZ' },
  { nombreCompleto: 'DAVID RODRÍGUEZ PÉREZ' },
  { nombreCompleto: 'JOSÉ MARTÍNEZ GARCÍA' },
  { nombreCompleto: 'LAURA LÓPEZ MARTÍNEZ' },
  { nombreCompleto: 'MARÍA GONZÁLEZ LÓPEZ' }
]
```

Ordenar los proyectos de investigación de mayor a menor según su subvención:

```
db.PROYECTO_INVESTIGACION.find().sort({ SUBVENCION: -1 })
```

```
HIPOPOTAMO> db.PROYECTO_INVESTIGACION.find().sort({ SUBVENCION: -1 })
[
  {
    _id: Long('5'),
    ID_PROYECTO: Long('5'),
    NOMBRE_PROYECTO: null,
    FECHA_INICIO: null,
    FECHA_FIN: null,
    ID_PROFESOR: null,
    ID_ORGANISMO_PUBLICO: null,
    SUBVENCION: 43750
  },
  {
    _id: Long('3'),
    ID_PROYECTO: Long('3'),
    NOMBRE_PROYECTO: 'Investigación en Inteligencia Artificial',
    FECHA_INICIO: ISODate('2024-05-10T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-08-20T00:00:00.000Z'),
    ID_PROFESOR: Long('2'),
    ID_ORGANISMO_PUBLICO: Long('2'),
    SUBVENCION: 35000
  },
  {
    _id: Long('6'),
    ID_PROYECTO: Long('6'),
    NOMBRE_PROYECTO: 'Análisis de datos',
    FECHA_INICIO: ISODate('2024-11-03T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-10-15T00:00:00.000Z'),
    ID_PROFESOR: Long('5'),
    ID_ORGANISMO_PUBLICO: Long('4'),
    SUBVENCION: Long('33000')
  },
  {
    _id: Long('4'),
    ID_PROYECTO: Long('4'),
    NOMBRE_PROYECTO: 'Estudio de impacto ambiental',
    FECHA_INICIO: ISODate('2024-07-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-12-31T00:00:00.000Z'),
    ID_PROFESOR: Long('3'),
    ID_ORGANISMO_PUBLICO: Long('3'),
    SUBVENCION: 31500
  },
  {
    _id: Long('1'),
    ID_PROYECTO: Long('1'),
    NOMBRE_PROYECTO: 'Desarrollo de nuevos materiales',
    FECHA_INICIO: ISODate('2023-10-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-09-30T00:00:00.000Z'),
    ID_PROFESOR: Long('1'),
    ID_ORGANISMO_PUBLICO: Long('1'),
    SUBVENCION: Long('15000')
  }
]
HIPOPOTAMO>
```

ARRAYS:

Profesores ordenados alfabéticamente por apellido y limitados a 2 resultados:

```
db.PROFESOR.aggregate([ { $project: { _id: 0, "NOMBRE": { $toUpper: "$NOMBREPROFESOR" }, "APELLIDO1": { $toUpper: "$APELLIDO1" }, "APELLIDO2": { $toUpper: "$APELLIDO2" } } }, { $sort: { "APELLIDO1": 1 } }, { $limit: 2 } ] )
```

```
HIPOPOTAMO> db.PROFESOR.aggregate([ { $project: { _id: 0, "NOMBRE": { $toUpper: "$NOMBREPROFESOR" }, "APELLIDO1": { $toUpper: "$APELLIDO1" }, "APELLIDO2": { $toUpper: "$APELLIDO2" } } }, { $sort: { "APELLIDO1": 1 } }, { $limit: 2 } ] )
[
  { NOMBRE: 'MARÍA', APELLIDO1: 'GONZÁLEZ', APELLIDO2: 'LÓPEZ' },
  { NOMBRE: 'LAURA', APELLIDO1: 'LÓPEZ', APELLIDO2: 'MARTÍNEZ' }
]
```

Alumnos con DNI que comienzan con '76' y ordenados por nombre:

```
db.ALUMNO.aggregate([ { $match: { DNI: /^76/ } }, { $project: { _id: 0, NOMBRE: { $toUpper: "$NOMBRE" }, APELLIDO1: { $toUpper: "$APELLIDO1" }, APELLIDO2: { $toUpper: "$APELLIDO2" }, DNI: 1 } }, { $addFields: { NOMBRE_COMPLETO: { $concat: ["$NOMBRE", " ", "$APELLIDO1", " ", "$APELLIDO2"] } } }, { $sort: { NOMBRE_COMPLETO: 1 } }, { $limit: 2 } ] )
```

```
HIPOPOTAMO> db.ALUMNO.aggregate([ { $match: { DNI: /^76/ } }, { $project: { _id: 0, NOMBRE: { $toUpper: "$NOMBRE" }, APELLIDO1: { $toUpper: "$APELLIDO1" }, APELLIDO2: { $toUpper: "$APELLIDO2" }, DNI: 1 } }, { $addFields: { NOMBRE_COMPLETO: { $concat: ["$NOMBRE", " ", "$APELLIDO1", " ", "$APELLIDO2"] } } }, { $sort: { NOMBRE_COMPLETO: 1 } }, { $limit: 2 } ] )
[
  {
    DNI: '7654321D',
    NOMBRE: 'SOFÍA',
    APELLIDO1: 'HERNÁNDEZ',
    APELLIDO2: 'MARTÍNEZ',
    NOMBRE_COMPLETO: 'SOFÍA HERNÁNDEZ MARTÍNEZ'
  }
]
```

Recupera los detalles de los proyectos de investigación que tienen una subvención mayor a 20000, incluyendo información como el título del proyecto, las fechas de inicio y finalización, la identificación del profesor a cargo, la identificación del organismo público asociado y el valor de la subvención. Estos detalles deben ser presentados en orden descendente según el monto de la subvención. En caso de que algún dato como el título del proyecto, las fechas, el ID del profesor o del organismo público no esté disponible, indica "No especificado" en su lugar.

```
db.PROYECTO_INVESTIGACION.aggregate([
  {
    $match: { "SUBVENCION": { $gt: 20000 } }
  },
  {
    $project: {
      _id: 0,
      "ID_PROYECTO": 1,
      "NOMBRE_PROYECTO": { $ifNull: ["$NOMBRE_PROYECTO", "No especificado"] },
      "FECHA_INICIO": { $ifNull: ["$FECHA_INICIO", new Date("1970-01-01")] },
      "FECHA_FIN": { $ifNull: ["$FECHA_FIN", new Date("1970-01-01")] },
      "ID_PROFESOR": { $ifNull: ["$ID_PROFESOR", "No especificado"] },
      "ID_ORGANISMO_PUBLICO": { $ifNull: ["$ID_ORGANISMO_PUBLICO", "No especificado"] },
      "SUBVENCION": 1
    }
  },
  {
    $sort: { "SUBVENCION": -1 }
  }
])
```

```

    }
  })
  {
    $match: { "SUBVENCION": { $gt: 20000 } }
  },
  {
    $project: {
      _id: 0,
      "ID_PROYECTO": 1,
      "NOMBRE_PROYECTO": { $ifNull: ["$NOMBRE_PROYECTO", "No especificado"] },
      "FECHA_INICIO": { $ifNull: ["$FECHA_INICIO", new Date("1970-01-01")] },
      "FECHA_FIN": { $ifNull: ["$FECHA_FIN", new Date("1970-01-01")] },
      "ID_PROFESOR": { $ifNull: ["$ID_PROFESOR", "No especificado"] },
      "ID_ORGANISMO_PUBLICO": { $ifNull: ["$ID_ORGANISMO_PUBLICO", "No especificado"] },
      "SUBVENCION": 1
    }
  },
  {
    $sort: { "SUBVENCION": -1 }
  }
})
[
  {
    ID_PROYECTO: Long('5'),
    SUBVENCION: 43750,
    NOMBRE_PROYECTO: 'No especificado',
    FECHA_INICIO: ISODate('1970-01-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('1970-01-01T00:00:00.000Z'),
    ID_PROFESOR: 'No especificado',
    ID_ORGANISMO_PUBLICO: 'No especificado'
  },
  {
    ID_PROYECTO: Long('3'),
    SUBVENCION: 35000,
    NOMBRE_PROYECTO: 'Investigación en Inteligencia Artificial',
    FECHA_INICIO: ISODate('2024-05-10T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-08-20T00:00:00.000Z'),
    ID_PROFESOR: Long('2'),
    ID_ORGANISMO_PUBLICO: Long('2')
  },
  {
    ID_PROYECTO: Long('6'),
    SUBVENCION: Long('33000'),
    NOMBRE_PROYECTO: 'Análisis de datos',
    FECHA_INICIO: ISODate('2024-11-03T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-10-15T00:00:00.000Z'),
    ID_PROFESOR: Long('5'),
    ID_ORGANISMO_PUBLICO: Long('4')
  },
  {
    ID_PROYECTO: Long('4'),
    SUBVENCION: 31500,
    NOMBRE_PROYECTO: 'Estudio de impacto ambiental',
    FECHA_INICIO: ISODate('2024-07-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-12-31T00:00:00.000Z'),
    ID_PROFESOR: Long('3'),
    ID_ORGANISMO_PUBLICO: Long('3')
  }
]

```

Consultas embebidas;

Consulta para obtener los detalles de un proyecto de investigación con información sobre el profesor a cargo y el departamento, ordenados por nombre del proyecto:

```
db.PROYECTO_INVESTIGACION.aggregate([
  {
    $lookup: {
      from: "PROFESOR",
      localField: "ID_PROFESOR",
      foreignField: "ID_PROFESOR",
      as: "PROFESOR"
    }
  },
  {
    $unwind: "$PROFESOR"
  },
  {
    $lookup: {
      from: "DEPARTAMENTO",
      localField: "PROFESOR.ID_DEPARTAMENTO",
      foreignField: "ID_DEPARTAMENTO",
      as: "DEPARTAMENTO"
    }
  },
  {
    $unwind: "$DEPARTAMENTO"
  },
  {
    $project: {
      _id: 0,
      ID_PROYECTO: 1,
      NOMBRE_PROYECTO: 1,
      FECHA_INICIO: 1,
      FECHA_FIN: 1,
      PROFESOR: {
        NOMBRE: { $toUpper: "$PROFESOR.NOMBREPROFESOR" },
        APELLIDO1: { $toUpper: "$PROFESOR.APELLIDO1" },
        APELLIDO2: { $toUpper: "$PROFESOR.APELLIDO2" }
      },
      DEPARTAMENTO: { $toUpper: "$DEPARTAMENTO.NOMBRE" }
    }
  },
  {
    $sort: { "NOMBRE_PROYECTO": 1 }
  },
  {
    $limit: 5
  }
])
```

```
[
  {
    ID_PROYECTO: Long('6'),
    NOMBRE_PROYECTO: 'Análisis de datos',
    FECHA_INICIO: ISODate('2024-11-03T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-10-15T00:00:00.000Z'),
    PROFESOR: { NOMBRE: 'LAURA', APELLIDO1: 'LÓPEZ', APELLIDO2: 'MARTÍNEZ' },
    DEPARTAMENTO: 'DEPARTAMENTO DE ECONOMÍA'
  },
  {
    ID_PROYECTO: Long('1'),
    NOMBRE_PROYECTO: 'Desarrollo de nuevos materiales',
    FECHA_INICIO: ISODate('2023-10-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-09-30T00:00:00.000Z'),
    PROFESOR: { NOMBRE: 'MARÍA', APELLIDO1: 'GONZÁLEZ', APELLIDO2: 'LÓPEZ' },
    DEPARTAMENTO: 'INGENIERÍA INDUSTRIAL'
  },
  {
    ID_PROYECTO: Long('4'),
    NOMBRE_PROYECTO: 'Estudio de impacto ambiental',
    FECHA_INICIO: ISODate('2024-07-01T00:00:00.000Z'),
    FECHA_FIN: ISODate('2025-12-31T00:00:00.000Z'),
    PROFESOR: { NOMBRE: 'ANA', APELLIDO1: 'SÁNCHEZ', APELLIDO2: 'FERNÁNDEZ' },
    DEPARTAMENTO: 'DEPARTAMENTO DE INGENIERÍA INDUSTRIAL'
  },
  {
    ID_PROYECTO: Long('3'),
    NOMBRE_PROYECTO: 'Investigación en Inteligencia Artificial',
    FECHA_INICIO: ISODate('2024-05-10T00:00:00.000Z'),
    FECHA_FIN: ISODate('2026-08-20T00:00:00.000Z'),
    PROFESOR: { NOMBRE: 'JOSÉ', APELLIDO1: 'MARTÍNEZ', APELLIDO2: 'GARCÍA' },
    DEPARTAMENTO: 'FÍSICA NUCLEAR'
  }
]
```

Consulta para obtener información sobre un alumno junto con los detalles de su grupo y dirección, ordenados por apellido del alumno:

```
db.ALUMNO.aggregate([
  {
    $lookup: {
      from: "GRUPO",
      localField: "ID_GRUPO",
      foreignField: "ID_GRUPO",
      as: "GRUPO"
    }
  },
  {
    $unwind: "$GRUPO"
  },
  {
    $lookup: {
      from: "DIRECCION",
      localField: "ID_DIRECCION",
      foreignField: "ID_DIRECCION",
      as: "DIRECCION"
    }
  },
  {
    $unwind: "$DIRECCION"
  },
  {
    $project: {
      _id: 0,
      ID_ALUMNO: 1,
      NOMBRE: { $toUpper: "$NOMBRE" },
      APELLIDO1: { $toUpper: "$APELLIDO1" },
      APELLIDO2: { $toUpper: "$APELLIDO2" },
      DNI: { $toUpper: "$DNI" },
      GRUPO: { $toUpper: "$GRUPO.NOMBRE_GRUPO" },
      DIRECCION: {
        CALLE: { $toUpper: "$DIRECCION.CALLE" },
        CIUDAD: { $toUpper: "$DIRECCION.CIUDAD" },
        "CÓDIGO POSTAL": "$DIRECCION.CÓDIGO_POSTAL"
      }
    }
  }
])
```

```

},
{
  $sort: { "APELLIDO1": 1 }
},
{
  $limit: 5
}
])

```

```

{
  ID_ALUMNO: Long('6'),
  DIRECCION: { CALLE: 'CALLE SIERPES', CIUDAD: 'SEVILLA' },
  NOMBRE: 'LAURA',
  APELLIDO1: 'FERNÁNDEZ',
  APELLIDO2: 'PÉREZ',
  DNI: '5432109F',
  GRUPO: ''
},
{
  ID_ALUMNO: Long('3'),
  DIRECCION: { CALLE: 'AVENIDA DIAGONAL', CIUDAD: 'BARCELONA' },
  NOMBRE: 'CARLOS',
  APELLIDO1: 'GARCÍA',
  APELLIDO2: 'FERNÁNDEZ',
  DNI: '87654321C',
  GRUPO: ''
},
{
  ID_ALUMNO: Long('4'),
  DIRECCION: { CALLE: 'AVENIDA DIAGONAL', CIUDAD: 'BARCELONA' },
  NOMBRE: 'SOFÍA',
  APELLIDO1: 'HERNÁNDEZ',
  APELLIDO2: 'MARTÍNEZ',
  DNI: '7654321D',
  GRUPO: ''
},
{
  ID_ALUMNO: Long('5'),
  DIRECCION: { CALLE: 'CALLE SIERPES', CIUDAD: 'SEVILLA' },
  NOMBRE: 'JAVIER',
  APELLIDO1: 'LÓPEZ',
  APELLIDO2: 'GONZÁLEZ',
  DNI: '6543210E',
  GRUPO: ''
},
{
  ID_ALUMNO: Long('1'),
  DIRECCION: { CALLE: 'CALLE MAYOR', CIUDAD: 'MADRID' },
  NOMBRE: 'JUAN',
  APELLIDO1: 'PÉREZ',
  APELLIDO2: 'GARCÍA',
  DNI: '12345678A',
  GRUPO: ''
}
]

```

Obtén los detalles de los profesores que están asignados al departamento de Ingeniería Industrial y que tienen al menos un proyecto de investigación asociado. Mostrar el ID del profesor, su nombre y apellidos en mayúsculas.

```

db.PROFESOR.aggregate([ { $match: { ID_DEPARTAMENTO: 1 } }, { $lookup: {
from: "PROYECTO_INVESTIGACION", localField: "ID_PROFESOR", foreignField:
"ID_PROFESOR", as: "proyectos" } }, { $match: { proyectos: { $exists:
true, $ne: [] } } }, { $project: { _id: 0, ID_PROFESOR: 1,
NOMBREPROFESOR: { $toUpper: "$NOMBREPROFESOR" }, APELLIDO1: { $toUpper:
"$APELLIDO1" }, APELLIDO2: { $toUpper: "$APELLIDO2" } } } ] )

```

```

HIPOPOTAMO> db.PROFESOR.aggregate([ { $match: { ID_DEPARTAMENTO: 1 } }, { $lookup: { from: "PROYECTO_INVESTIGACION", localField: "ID_PROFESOR", foreignField: "ID_PROFESOR", as: "proyectos" } }, { $match: { proyectos: { $exists: true, $ne: [] } } }, { $project: { _id: 0, ID_PROFESOR: 1, NOMBREPROFESOR: { $toUpper: "$NOMBREPROFESOR" }, APELLIDO1: { $toUpper: "$APELLIDO1" }, APELLIDO2: { $toUpper: "$APELLIDO2" } } } ] )
[
  {
    ID_PROFESOR: Long('1'),
    NOMBREPROFESOR: 'MARIA',
    APELLIDO1: 'GONZÁLEZ',
    APELLIDO2: 'LÓPEZ'
  }
]
HIPOPOTAMO> |

```


Agrupación:

Se desea obtener una lista de profesores junto con el número de proyectos de investigación en los que han participado, ordenados de mayor a menor número de proyectos. Además, se requiere que los nombres de los profesores se muestren en mayúsculas para una mejor legibilidad

```
db.PROFESOR.aggregate([
  {
    $lookup: {
      from: "PROYECTO_INVESTIGACION",
      localField: "ID_PROFESOR",
      foreignField: "ID_PROFESOR",
      as: "PROYECTOS_INVESTIGACION"
    }
  },
  {
    $group: {
      _id: {
        ID_PROFESOR: "$ID_PROFESOR",
        NOMBREPROFESOR: "$NOMBREPROFESOR",
        APELLIDO1: "$APELLIDO1",
        APELLIDO2: "$APELLIDO2"
      },
      NUM_PROYECTOS_INVESTIGACION: { $sum: 1 }
    }
  },
  {
    $sort: { NUM_PROYECTOS_INVESTIGACION: -1 }
  },
  {
    $project: {
      _id: 0,
      ID_PROFESOR: { $toUpper: "$_id.ID_PROFESOR" },
      NOMBREPROFESOR: { $toUpper: "$_id.NOMBREPROFESOR" },
      APELLIDO1: { $toUpper: "$_id.APELLIDO1" },
      APELLIDO2: { $toUpper: "$_id.APELLIDO2" },
      NUM_PROYECTOS_INVESTIGACION: 1
    }
  }
])
```

```

[
  {
    NUM_PROYECTOS_INVESTIGACION: 1,
    ID_PROFESOR: '2',
    NOMBREPROFESOR: 'JOSÉ',
    APELLIDO1: 'MARTÍNEZ',
    APELLIDO2: 'GARCÍA'
  },
  {
    NUM_PROYECTOS_INVESTIGACION: 1,
    ID_PROFESOR: '1',
    NOMBREPROFESOR: 'MARÍA',
    APELLIDO1: 'GONZÁLEZ',
    APELLIDO2: 'LÓPEZ'
  },
  {
    NUM_PROYECTOS_INVESTIGACION: 1,
    ID_PROFESOR: '3',
    NOMBREPROFESOR: 'ANA',
    APELLIDO1: 'SÁNCHEZ',
    APELLIDO2: 'FERNÁNDEZ'
  },
  {
    NUM_PROYECTOS_INVESTIGACION: 1,
    ID_PROFESOR: '5',
    NOMBREPROFESOR: 'LAURA',
    APELLIDO1: 'LÓPEZ',
    APELLIDO2: 'MARTÍNEZ'
  }
]
HIPOPOTAMO>

```