

Práctica 3

Bases de datos 2

Índice

1. Parámetros de conexión a la BD.	2
1.1. Servidor	2
1.2. Usuario y contraseña	2
2. Instrucciones lanzadas y resultados obtenidos .	3
• Ejercicio 1	3
• Ejercicio 2	4
• Ejercicio 3	6
• Ejercicio 4	7
• Ejercicio 5	9
• Ejercicio 6	12
• Ejercicio 7	14

1. Parámetros de conexión a la BD.

1.1. Servidor

- Compass

```
mongodb+srv://815352:815352@cluster0.mwumhb6.mongodb.net/
```

- Shell

```
mongosh "mongodb+srv://cluster0.mwumhb6.mongodb.net" --apiVersion 1  
--username 815352 --password 815352
```

Para acceder a la base de datos sakila, una vez dentro de Mongo ejecutando el comando anterior, introducir:

```
use sakila
```

1.2. Usuario y contraseña

- Usuario:
815352
- Contraseña:
815352

2. Instrucciones lanzadas y resultados obtenidos .

- Ejercicio 1

Instrucción Mongo Shell que obtenga el rating más habitual en las películas en las que participa SANDRA PECK.

- Instrucción lanzada

```
db.getCollection('films').aggregate([
  {
    $match: {
      "Actors.First name": "SANDRA",
      "Actors.Last name": "PECK"
    }
  },
  {
    $group: {
      _id: "$Rating",
      count: { $sum: 1 }
    }
  },
  {
    $sort: { count: -1 }
  },
  {
    $limit: 1
  },
  {
    $project: {
      _id: 0,
      rating: "$_id"
    }
  }
])
```

- Resultados obtenidos

```
[ { rating: 'G' } ]
```

Figura 1: Resultado obtenido del ejercicio 1.

- Ejercicio 2

Instrucción Mongo Shell que obtenga un listado con título, categoría y duración de las películas que ha alquilado WENDY HARRISON.

- Instrucción lanzada

```
db.getCollection('customers').aggregate([
  {
    $match: {
      "First Name": "WENDY",
      "Last Name": "HARRISON"
    },
  },
  {
    $lookup: {
      from: "films",
      localField: "Rentals.filmId",
      foreignField: "_id",
      as: "films"
    }
  },
  {
    $unwind: "$films"
  },
  {
    $replaceRoot: {
      newRoot: "$films"
    }
  },
  {
    $project: {
      _id: 0,
      título: "$Title",
      categoría: "$Category",
      duración: "$Length"
    }
  }
])
```

- Resultados obtenidos

```
{
  'título': 'HYDE DOCTOR',
  'categoría': 'Classics',
  'duración': '100'
},
{
  'título': 'JET NEIGHBORS',
  'categoría': 'Foreign',
  'duración': '59'
},
{
  'título': 'LEAGUE HELLFIGHTERS',
  'categoría': 'Classics',
  'duración': '110'
},
{
  'título': 'MIDNIGHT WESTWARD',
  'categoría': 'Action',
  'duración': '86'
},
{
  'título': 'MODERN DORADO',
  'categoría': 'Documentary',
  'duración': '74'
},
{
  'título': 'MOULIN WAKE', 'categoría': 'Travel', 'duración': '79' },
{
  'título': 'OSCAR GOLD',
  'categoría': 'Animation',
  'duración': '115'
},
{
  'título': 'PACIFIC AMISTAD',
  'categoría': 'Documentary',
  'duración': '144'
},
{
  'título': 'PERSONAL LADYBUGS',
  'categoría': 'Music',
  'duración': '118'
}
```

Figura 2: Resultado obtenido del ejercicio 2.

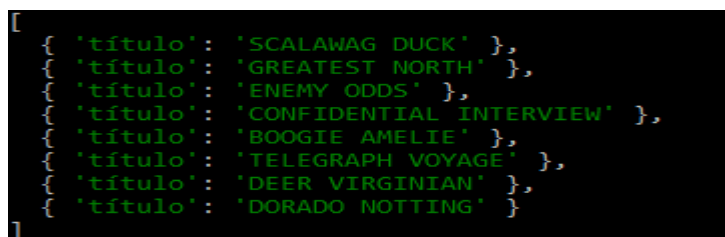
● Ejercicio 3

Instrucción Mongo Shell que obtenga un listado con el título de las ocho películas de género musical con más alquileres.

- Instrucción lanzada

```
db.getCollection('films').aggregate([
  {
    $match: {
      "Category": "Music"
    }
  },
  {
    $lookup: {
      from: "customers",
      localField: "_id",
      foreignField: "Rentals.filmId",
      as: "customer"
    }
  },
  {
    $unwind: "$customer"
  },
  {
    $group: {
      _id: { id: "$_id", title: "$Title" },
      count: { $sum: 1 }
    }
  },
  {
    $sort: { count: -1 }
  },
  {
    $limit: 8
  },
  {
    $project: {
      _id: 0,
      título: "$_id.title"
    }
  }
])
```

- Resultados obtenidos



```
[
  { 'título': 'SCALAWAG DUCK' },
  { 'título': 'GREATEST NORTH' },
  { 'título': 'ENEMY ODDS' },
  { 'título': 'CONFIDENTIAL INTERVIEW' },
  { 'título': 'BOOGIE AMELIE' },
  { 'título': 'TELEGRAPH VOYAGE' },
  { 'título': 'DEER VIRGINIAN' },
  { 'título': 'DORADO NOTTING' }
]
```

Figura 3: Resultado obtenido del ejercicio 3.

● Ejercicio 4

Instrucción Mongo Shell que crea una vista con las películas de género musical y las personas que los han alquilado.

- Instrucción lanzada

Si ya hay una vista creada con el mismo nombre, hay que borrarla antes de ejecutar la instrucción para crear la vista. Para borrarla hay que ejecutar esta instrucción:

```
db.genero_musical.drop()
```

Para crear la vista:

```
db.createView("genero_musical",
"customers",
[
  {
    $lookup: {
      from: "films",
      localField: "Rentals.filmId",
      foreignField: "_id",
      as: "Film"
    }
  },
  {
    $unwind: "$Film"
  },
  {
    $match: {
      "Film.Category": "Music"
    }
  },
  {
    $group: {
      _id: "$Film",
      customers: {
        $addToSet: {
          "_id": "$_id",
          "First Name": "$First Name",
          "Last Name": "$Last Name"
        }
      }
    }
  }
])
```

Para acceder a la vista:

```
db.genero_musical.find({})
```


- Resultados obtenidos

```
{
  _id: {
    _id: 477,
    Actors: [
      {
        'First name': 'CHRISTIAN',
        'Last name': 'GABLE',
        actorId: 10
      },
      { 'First name': 'JULIA', 'Last name': 'MCQUEEN', actorId: 27 },
      { 'First name': 'VAL', 'Last name': 'BOLGER', actorId: 37 },
      {
        'First name': 'CHRISTIAN',
        'Last name': 'AKROYD',
        actorId: 58
      },
      { 'First name': 'JESSICA', 'Last name': 'BAILEY', actorId: 67 }
    ],
    Category: 'Music',
    Description: 'A Stunning Reflection of a Boat And a Pastry Chef who must Succumb a A Shark in A Jet Boat',
    Length: '118',
    Rating: 'PG',
    'Rental Duration': '5',
    'Replacement Cost': '15.99',
    'Special Features': 'Trailers,Behind the Scenes',
    Title: 'JAWBREAKER BROOKLYN'
  },
  customers: [
    { _id: 147, 'First Name': 'JOANNE', 'Last Name': 'ROBERTSON' },
    { _id: 407, 'First Name': 'DALE', 'Last Name': 'RATCLIFF' },
    { _id: 455, 'First Name': 'JON', 'Last Name': 'WILES' },
    { _id: 570, 'First Name': 'IVAN', 'Last Name': 'CROMWELL' },
    { _id: 308, 'First Name': 'THOMAS', 'Last Name': 'GRIGSBY' },
    { _id: 252, 'First Name': 'MATTIE', 'Last Name': 'HOFFMAN' },
    { _id: 344, 'First Name': 'HENRY', 'Last Name': 'BILLINGSLEY' },
    { _id: 38, 'First Name': 'MARTHA', 'Last Name': 'GONZALEZ' },
    { _id: 563, 'First Name': 'KEN', 'Last Name': 'PREWITT' },
    { _id: 161, 'First Name': 'GERALDINE', 'Last Name': 'PERKINS' }
  ]
},
```

```
{
  _id: {
    _id: 454,
    Actors: [
      { 'First name': 'VIVIEN', 'Last name': 'BERGEN', actorId: 14 },
      {
        'First name': 'GINA',
        'Last name': 'DEGENERES',
        actorId: 107
      },
      {
        'First name': 'WARREN',
        'Last name': 'JACKMAN',
        actorId: 119
      }
    ],
    Category: 'Music',
    Description: 'A Epic Character Study of a Frisbee And a Moose who must Outgun a Technical Writer in A Shark Tank',
    Length: '180',
    Rating: 'PG-13',
    'Rental Duration': '6',
    'Replacement Cost': '20.99',
    'Special Features': 'Trailers,Deleted Scenes',
    Title: 'IMPACT ALADDIN'
  },
  customers: [
    { _id: 231, 'First Name': 'GEORGIA', 'Last Name': 'JACOBS' },
    { _id: 444, 'First Name': 'MARCUS', 'Last Name': 'HIDALGO' },
    { _id: 509, 'First Name': 'RAUL', 'Last Name': 'FORTIER' },
    { _id: 595, 'First Name': 'TERRENCE', 'Last Name': 'GUNDERSON' },
    { _id: 374, 'First Name': 'JEREMY', 'Last Name': 'HURTADO' },
    { _id: 121, 'First Name': 'JOSEPHINE', 'Last Name': 'GOMEZ' },
    { _id: 61, 'First Name': 'KATHERINE', 'Last Name': 'RIVERA' },
    { _id: 565, 'First Name': 'JAIME', 'Last Name': 'NETTLES' },
    { _id: 214, 'First Name': 'KRISTIN', 'Last Name': 'JOHNSTON' }
  ]
}
]
Type "it" for more
```

Figura 4: Resultado obtenido del ejercicio 4.

• Ejercicio 5

Importar a vuestra BD como dos colecciones los datos de:

Colección oferta_unizar y colección notas_corte_unizar.

[Oferta y ocupación de plazas. Curso 2021-2022. Universidad de Zaragoza - Repositorio Institucional de Documentos \(unizar.es\)](#)

[Notas de corte definitivas del cupo general a estudios de grado. Curso 2021-2022. Universidad de Zaragoza - Repositorio Institucional de Documentos \(unizar.es\)](#)

Como en el enunciado no especifica si realizar la importación a través de instrucciones Mongo Shell o a través de interfaz gráfica, he optado por realizar la importación con la interfaz gráfica Compass.

Para ello, primero he descargado los ficheros CSV correspondientes a cada enlace proporcionados en el enunciado. Luego, desde la interfaz gráfica, accedes a la base de datos que deseas, en este caso la de sakila y le das a *create collection*.

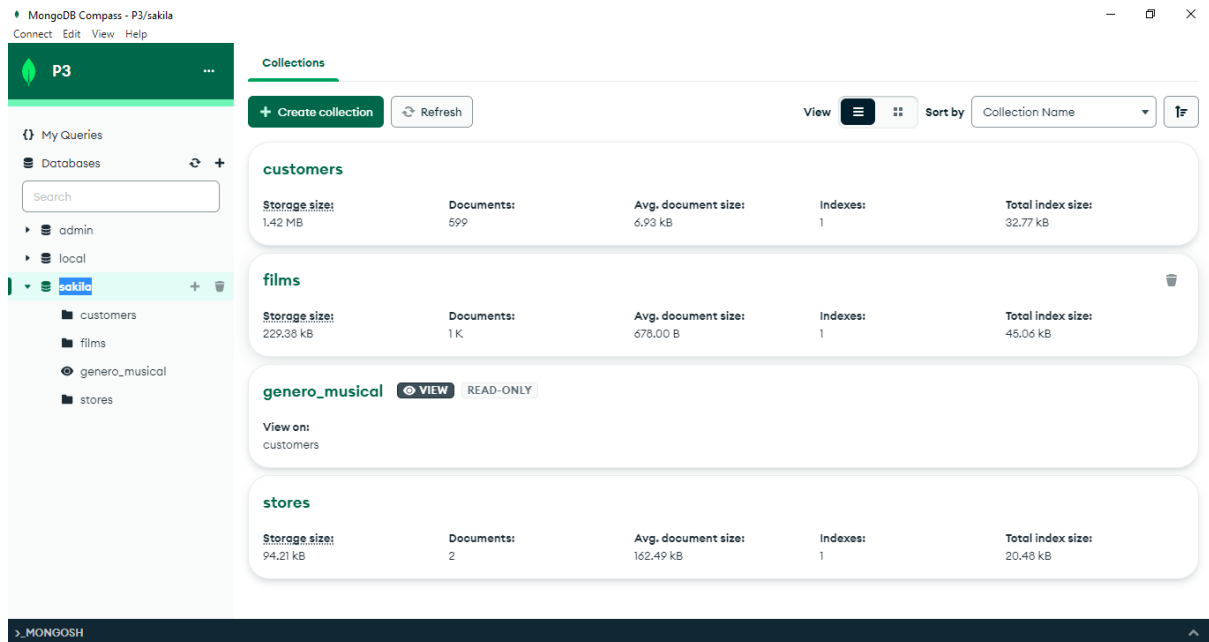


Figura 5: Captura de pantalla seleccionando la base de datos sakila y el botón create collection.

A continuación, insertas el nombre de la colección y clicas sobre *create collection*.

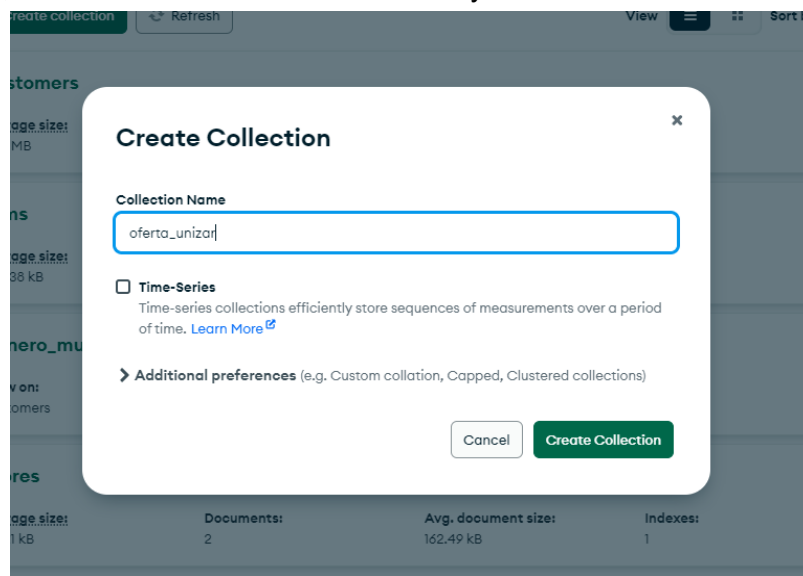


Figura 6: Captura de pantalla para crear la colección.

Dentro de la colección, hay un botón *add data*, se le clicla y adjuntamos el archivo previamente descargado.

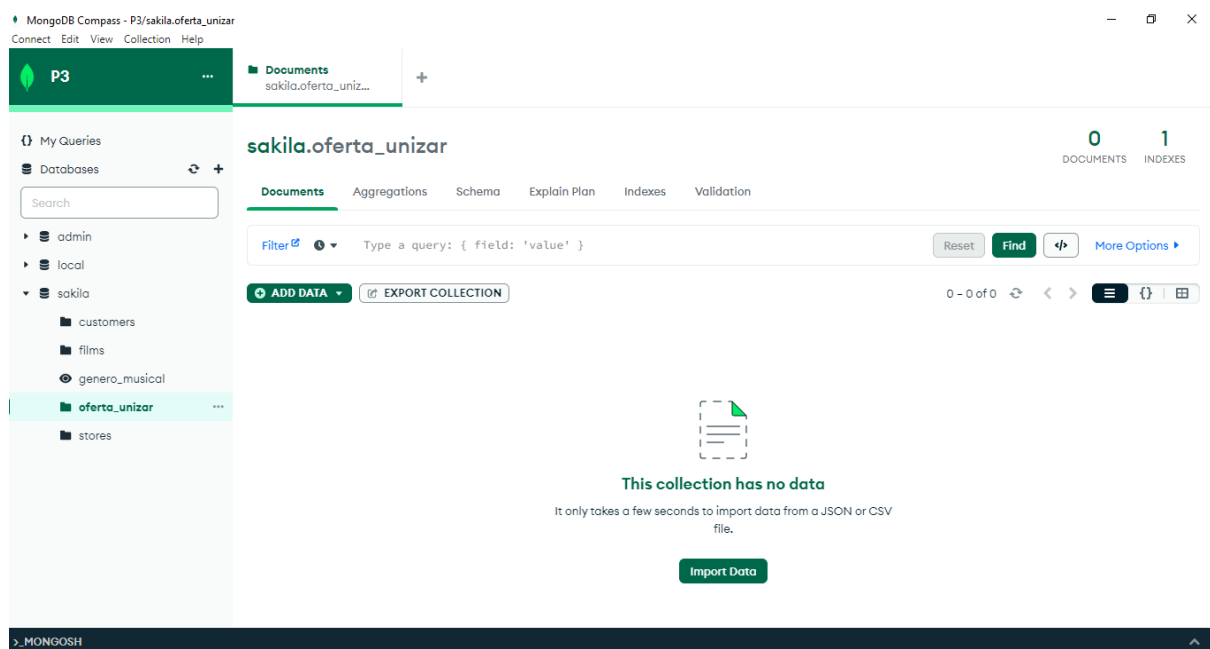


Figura 7: Captura de pantalla de la colección creada sin datos.

Una vez importado el documento ya se dispone de la colección completa con los datos.

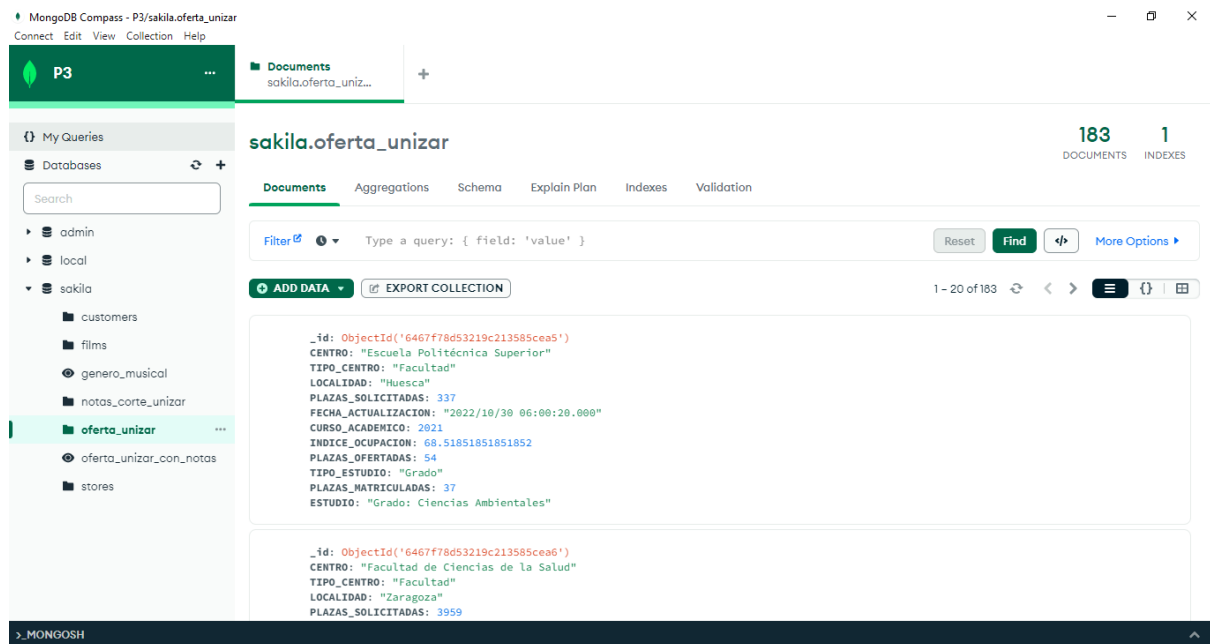


Figura 8: Captura de pantalla de la colección creada con datos.

He realizado los mismos pasos para importar la colección notas_corte_unizar.

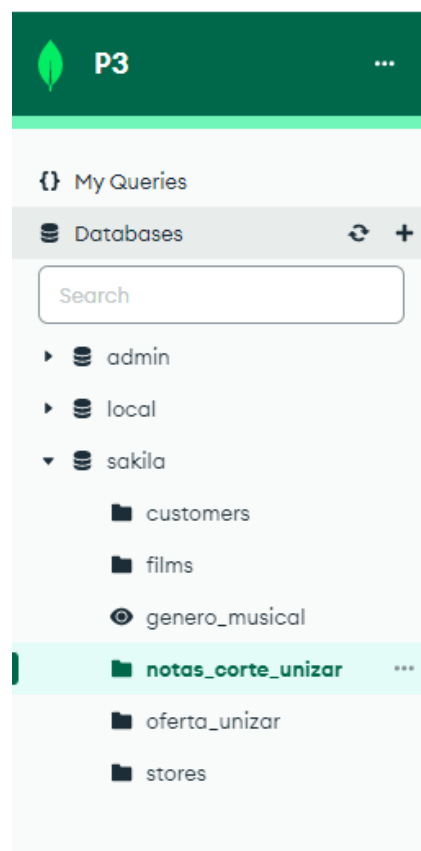


Figura 9: Captura de pantalla de ambas colecciones creadas.

● Ejercicio 6

Añadir a la colección `notas_corte_unizar` los campos numéricos `nota_corte_julio` y `nota_corte_septiembre` que será el resultado de convertir a número los datos de `NOTA_CORTE_DEFINITIVA_JULIO` y `NOTA_CORTE_DEFINITIVA_SEPTIEMBRE`, que por defecto se importan como texto.

- Instrucción lanzada

Para modificar el valor:

```
db.notas_corte_unizar.updateMany(
  {},
  [
    {
      $set: {
        NOTA_CORTE_JULIO: { $toDouble:
"$NOTA_CORTE_DEFINITIVA_JULIO" },
        NOTA_CORTE_SEPTIEMBRE: { $toDouble:
"$NOTA_CORTE_DEFINITIVA_SEPTIEMBRE" }
      }
    }
  ]
)
```

Para ver el tipo de dato:

```
db.notas_corte_unizar.aggregate([
  {
    $project: {
      cursoAcademico: { $type: "$CURSO_ACADEMICO" },
      estudio: { $type: "$ESTUDIO" },
      localidad: { $type: "$LOCALIDAD" },
      centro: { $type: "$CENTRO" },
      prelaConvoNotaDef: { $type: "$PRELA_CONVO_NOTA_DEF" },
      notaCorteJulio: { $type: "$NOTA_CORTE_JULIO" },
      notaCorteSeptiembre: { $type: "$NOTA_CORTE_SEPTIEMBRE" },
      notaCorteDefJulio: { $type: "$NOTA_CORTE_DEFINITIVA_JULIO" },
      notaCorteDefSeptiembre: { $type: "$NOTA_CORTE_DEFINITIVA_SEPTIEMBRE" },
      fechaActualizacion: { $type: "$FECHA_ACTUALIZACION" }
    }
  }
])
```

- Resultados obtenidos

```
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 72,
  modifiedCount: 0,
  upsertedCount: 0
}

{
  {
    _id: ObjectId("6467f6e653219c213585ce69"),
    cursoAcademico: 'int',
    estudio: 'string',
    localidad: 'string',
    centro: 'string',
    prelaConvoNotaDef: 'null',
    notaCorteJulio: 'double',
    notaCorteSeptiembre: 'null',
    notaCorteDefJulio: 'double',
    notaCorteDefSeptiembre: 'null',
    fechaActualizacion: 'string'
  },
  {
    _id: ObjectId("6467f6e653219c213585ce6a"),
    cursoAcademico: 'int',
    estudio: 'string',
    localidad: 'string',
    centro: 'string',
    prelaConvoNotaDef: 'null',
    notaCorteJulio: 'double',
    notaCorteSeptiembre: 'null',
    notaCorteDefJulio: 'double',
    notaCorteDefSeptiembre: 'null',
    fechaActualizacion: 'string'
  },
  {
    _id: ObjectId("6467f6e653219c213585ce6b"),
    cursoAcademico: 'int',
    estudio: 'string',
    localidad: 'string',
    centro: 'string',
    prelaConvoNotaDef: 'null',
    notaCorteJulio: 'double',
    notaCorteSeptiembre: 'double',
    notaCorteDefJulio: 'double',
    notaCorteDefSeptiembre: 'double',
    fechaActualizacion: 'string'
  },
}
```

Figura 10: Resultado obtenido del ejercicio 6.

● Ejercicio 7

Crear una vista o una nueva colección, llamada oferta_unizar_con_notas que combine los datos de ambas colecciones.

- Instrucción lanzada

Si ya hay una vista creada con el mismo nombre, hay que borrarla antes de ejecutar la instrucción para crear la vista. Para borrarla hay que ejecutar esta instrucción:

```
db.oferta_unizar_con_notas.drop()
```

Para crear la vista:

```
db.createView("oferta_unizar_con_notas",
"oferta_unizar",
[
{
  $lookup: {
    from: "notas_corte_unizar",
    let: {
      estudio: "$ESTUDIO",
      centro: "$CENTRO",
      localidad: "$LOCALIDAD"
    },
    pipeline: [
      {
        $match: {
          $expr: {
            $and: [
              { $eq: ["$ESTUDIO", "$$estudio"] },
              { $eq: ["$CENTRO", "$$centro"] },
              { $eq: ["$LOCALIDAD", "$$localidad"] }
            ]
          }
        }
      }
    ]
  },
  as: "notas_corte"
}
],
{
  $unwind: "$notas_corte"
},
{
  $project: {
    _id: 0,
    CURSO_ACADEMICO: 1,
    ESTUDIO: 1,
    LOCALIDAD: 1,
    CENTRO: 1,
    PRELA_CONVO_NOTA_DEF: "$notas_corte.PRELA_CONVO_NOTA_DEF",
    NOTA_CORTE_DEFINITIVA_JULIO: "$notas_corte.NOTA_CORTE_DEFINITIVA_JULIO",
    NOTA_CORTE_DEFINITIVA_SEPTIEMBRE: "$notas_corte.NOTA_CORTE_DEFINITIVA_SEPTIEMBRE",
    TIPO_CENTRO: 1,
  }
}
```

```

TIPO_ESTUDIO: 1,
PLAZAS_OFERTADAS: 1,
PLAZAS_MATRICULADAS: 1,
PLAZAS_SOLICITADAS: 1,
INDICE_OCUPACION: 1,
NOTA_CORTE_JULIO: "$notas_corte.NOTA_CORTE_DEFINITIVA_JULIO",
NOTA_CORTE_SEPTIEMBRE: "$notas_corte.NOTA_CORTE_DEFINITIVA_SEPTIEMBRE",
FECHA_ACTUALIZACION: {
  $dateFromString: {
    dateString: "$notas_corte.FECHA_ACTUALIZACION",
    format: "%Y/%m/%d %H:%M:%S.%L"
  }
}
}
}
}
}
1)

```

Para acceder a la vista:

```
db.oferta_unizar_con_notas.find({})
```

- Resultados obtenidos

```

{
  CENTRO: 'Facultad de Ciencias Sociales y Humanas de Teruel',
  TIPO_CENTRO: 'Facultad',
  LOCALIDAD: 'Teruel',
  PLAZAS_SOLICITADAS: 988,
  CURSO_ACADEMICO: 2021,
  INDICE_OCUPACION: 95.83333333333333,
  PLAZAS_OFERTADAS: 120,
  TIPO_ESTUDIO: 'Grado',
  PLAZAS_MATRICULADAS: 115,
  ESTUDIO: 'Grado: Magisterio en Educación Primaria',
  PRELA_CONVO_NOTA_DEF: null,
  NOTA_CORTE_DEFINITIVA_JULIO: 5,
  NOTA_CORTE_DEFINITIVA_SEPTIEMBRE: 5,
  NOTA_CORTE_JULIO: 5,
  NOTA_CORTE_SEPTIEMBRE: 5,
  FECHA_ACTUALIZACION: ISODate("2022-10-30T06:00:20.000Z")
},
{
  CENTRO: 'Facultad de Educación',
  TIPO_CENTRO: 'Facultad',
  LOCALIDAD: 'Zaragoza',
  PLAZAS_SOLICITADAS: 1329,
  CURSO_ACADEMICO: 2021,
  INDICE_OCUPACION: 100,
  PLAZAS_OFERTADAS: 120,
  TIPO_ESTUDIO: 'Grado',
  PLAZAS_MATRICULADAS: 120,
  ESTUDIO: 'Grado: Magisterio en Educación Infantil',
  PRELA_CONVO_NOTA_DEF: null,
  NOTA_CORTE_DEFINITIVA_JULIO: 9.803,
  NOTA_CORTE_DEFINITIVA_SEPTIEMBRE: null,
  NOTA_CORTE_JULIO: 9.803,
  NOTA_CORTE_SEPTIEMBRE: null,
  FECHA_ACTUALIZACION: ISODate("2022-10-30T06:00:20.000Z")
}

```



```

{
  CENTRO: 'Facultad de Educación',
  TIPO_CENTRO: 'Facultad',
  LOCALIDAD: 'Zaragoza',
  PLAZAS_SOLICITADAS: 1773,
  CURSO_ACADEMICO: 2021,
  INDICE_OCUPACION: 99.16666666666667,
  PLAZAS_OFERTADAS: 240,
  TIPO_ESTUDIO: 'Grado',
  PLAZAS_MATRICULADAS: 238,
  ESTUDIO: 'Grado: Magisterio en Educación Primaria',
  PRELA_CONVO_NOTA_DEF: null,
  NOTA_CORTE_DEFINITIVA_JULIO: 9.897,
  NOTA_CORTE_DEFINITIVA_SEPTIEMBRE: null,
  NOTA_CORTE_JULIO: 9.897,
  NOTA_CORTE_SEPTIEMBRE: null,
  FECHA_ACTUALIZACION: ISODate("2022-10-30T06:00:20.000Z")
},
{
  CENTRO: 'Facultad de Ciencias Humanas y de la Educación',
  TIPO_CENTRO: 'Facultad',
  LOCALIDAD: 'Huesca',
  PLAZAS_SOLICITADAS: 895,
  CURSO_ACADEMICO: 2021,
  INDICE_OCUPACION: 100,
  PLAZAS_OFERTADAS: 120,
  TIPO_ESTUDIO: 'Grado',
  PLAZAS_MATRICULADAS: 120,
  ESTUDIO: 'Grado: Magisterio en Educación Primaria',
  PRELA_CONVO_NOTA_DEF: null,
  NOTA_CORTE_DEFINITIVA_JULIO: 7.53,
  NOTA_CORTE_DEFINITIVA_SEPTIEMBRE: null,
  NOTA_CORTE_JULIO: 7.53,
  NOTA_CORTE_SEPTIEMBRE: null,
  FECHA_ACTUALIZACION: ISODate("2022-10-30T06:00:20.000Z")
},
}

```

Figura 11: Resultado obtenido del ejercicio 7.