

Create your account

Have an account? [Log in now](#)



— Or with email and password —

Email Address

We recommend using your work email



First Name



Last Name



Password



- ✓ Must be at least 8 characters
- ✓ Does not contain your email address

Company Name



I accept the [Privacy Policy](#) and
the [Terms of Service](#)

Pantalla de regstre amb el formulari de dades.

Crear una Base de Dades NoSQL amb MongoDB

ORGANIZATION
Bilal's Org - 2025-12-18 ➤ PROJECT
Project 0 ▾

We are deploying your changes (current action: creating a plan)

Clusters

Find a database deployment...

Cluster0

ORGANIZATION
Bilal's Org - 2025-12-18 ➤ PROJECT
Project 0 ▾

Configurations

Name Quick setup

Provider AWS Google Cloud Azure

Region

Tag (optional)
Create your first tag to categorize and label your resources; more tags can be added later. [Learn more.](#)

:

Included features

Auto-scale
Atlas enables auto-scaling for cluster storage and cluster tier. [Learn more.](#)

Storage Scaling [i](#) Cluster Tier Scaling [i](#)

Cloud Backup
Snapshots are taken automatically and stored according to your backup and retention policy. [Learn more.](#)

Continuous Cloud Backup [i](#)

Un cop creat, cliquem **collection**.

Clica Create Database



Create Database

Database Name
tasques

Collection Name
entrades

Time-Series
Time-series collections efficiently store sequences of measurements over a period of time. [Learn More](#)

➤ Additional preferences (e.g. Custom collation, Clustered collections)

[Cancel](#) [Create Database](#)

A detailed screenshot of the 'Create Database' dialog box. It contains fields for 'Database Name' (tasques) and 'Collection Name' (entrades). There's a checkbox for 'Time-Series' which is unchecked. Below it is a descriptive text about time-series collections and a link to 'Learn More'. A section titled 'Additional preferences' is shown with a right-pointing arrow. At the bottom are 'Cancel' and 'Create Database' buttons, with a red arrow pointing to the 'Create Database' button.

ORGANIZATION Bilal's Org - 2025-12-18 ▾

PROJECT Project 0 ▾

Data Explorer

{} My Queries

▪ Data Modeling

CLUSTERS (1) X

Search clusters T

- ▼ Cluster0
 - ▶ admin
 - ▶ local
 - ▶ sample_mflix
 - ▼ tasques
 - ▶ entrades
 - ▶ tasques.entrades

Clica la colección entradas.

Clica Insert Document.

Copia i enganxa el següent codi JSON:

Insert Document

To collection tasques.entrades

VIEW

{}

≡



```
1 ▾ [  
2 ▾ {  
3   "nomAlumne": "Laura",  
4   "cognom1": "Martínez",  
5   "cognom2": "Soler",  
6   "dataEntradaTasca": "2025-12-10T09:30:00Z",  
7   "completa": true,  
8   "observacions": "Tasca entregada abans de la data límit"  
9 },  
10 ▾ {  
11   "nomAlumne": "Pol",  
12   "cognom1": "García",  
13   "cognom2": "López",  
14   "dataEntradaTasca": "2025-12-12T14:15:00Z",  
15   "completa": false,  
16   "observacions": "Falten alguns exercicis"  
17 },  
18 ▾ {  
19   "nomAlumne": "Anna",  
20   "cognom1": "Vidal",  
21   "cognom2": "Riera",  
22   "dataEntradaTasca": "2025-12-14T18:45:00Z",  
23   "completa": true,  
24   "observacions": "Molt bona presentació i contingut"  
25 }  
26 ]  
27 |
```

Cancel

Insert



The screenshot shows the MongoDB Compass interface with the following details:

- Top Navigation:** Welcome, entrades, tasques.entrades, +
- Breadcrumbs:** Cluster0 > tasques > tasques.entrades
- Buttons:** View monitoring, Visualize Your Data
- Tabs:** Documents (3), Aggregations, Schema, Indexes (1), Validation
- Toolbar Buttons:** Explain, Reset, Find (highlighted in green), Options ▾
- Search and Pagination:** 25 ▾ 1 - 3 of 3, back, forward, refresh, dropdown menu, search bar

Document 1:

```
_id: ObjectId('695eedf878347211682c4d27')
nomAlumne : "Laura"
cognom1 : "Martínez"
cognom2 : "Soler"
dataEntradaTasca : "2025-12-10T09:30:00Z"
completa : true
observacions : "Tasca entregada abans de la data límit"
```

Document 2:

```
_id: ObjectId('695eedf878347211682c4d28')
nomAlumne : "Pol"
cognom1 : "García"
cognom2 : "López"
dataEntradaTasca : "2025-12-12T14:15:00Z"
completa : false
observacions : "Falten alguns exercicis"
```

Document 3:

```
_id: ObjectId('695eedf878347211682c4d29')
nomAlumne : "Anna"
cognom1 : "Vidal"
cognom2 : "Riera"
dataEntradaTasca : "2025-12-14T18:45:00Z"
completa : true
observacions : "Molt bona presentació i contingut"
```

Realitzar cerques a la col·lecció

The screenshot shows the MongoDB Compass interface. In the top navigation bar, there are tabs for 'Connections' (selected), 'Edit', 'View', 'Collection', and 'Help'. Below the tabs, the connection name 'cluster0ye8yvir.mongodb.net' is shown. The main area displays a collection named 'tasques.entrades'. A red arrow points from the text 'Open MongoDB shell' in the top right corner of the interface towards the bottom right corner of the screenshot.

In the query bar at the top, the command 'db.tasques.entrades.find({ completa: true })' is entered. Below the query bar, there are buttons for 'Generate query', 'Explain', 'Reset', 'Find', and 'Options'. Underneath these buttons, there are buttons for 'ADD DATA', 'EXPORT DATA', 'UPDATE', and 'DELETE'. The results section shows three documents:

```
_id: ObjectId('695eedf878347211682c4d27')
nomAlumne: "Laura"
cognom1: "Martínez"
cognom2: "Soler"
dataEntradaTasca: "2025-12-10T09:30:00Z"
completa: true
observacions: "Tasca entregada abans de la data límit"

_id: ObjectId('695eedf878347211682c4d28')
nomAlumne: "Poi"
cognom1: "Garcia"
cognom2: "López"
dataEntradaTasca: "2025-12-12T14:15:00Z"
completa: false
observacions: "Falten alguns exercicis"

_id: ObjectId('695eedf878347211682c4d29')
nomAlumne: "Anna"
cognom1: "Vidal"
cognom2: "Riera"
dataEntradaTasca: "2025-12-14T18:45:00Z"
completa: true
```

Llistar totes les tasques

```
> db.tasques.entrades.find({ completa: true })
< [
    {
        _id: ObjectId('695eedf878347211682c4d27'),
        nomAlumne: 'Laura',
        cognom1: 'Martínez',
        cognom2: 'Soler',
        dataEntradaTasca: '2025-12-10T09:30:00Z',
        completa: true,
        observacions: 'Tasca entregada abans de la data límit'
    },
    {
        _id: ObjectId('695eedf878347211682c4d28'),
        nomAlumne: 'Poi',
        cognom1: 'Garcia',
        cognom2: 'López',
        dataEntradaTasca: '2025-12-12T14:15:00Z',
        completa: false,
        observacions: 'Falten alguns exercicis'
    },
    {
        _id: ObjectId('695eedf878347211682c4d29'),
        nomAlumne: 'Anna',
        cognom1: 'Vidal',
        cognom2: 'Riera',
        dataEntradaTasca: '2025-12-14T18:45:00Z',
        completa: true
    }
]
```

Listar les tasques completades

```
db.tasques.entrades.find({ completa: true })  
{  
  _id: ObjectId('695eedf878347211682c4d27'),  
  nomAlumne: 'Laura',  
  cognom1: 'Martínez',  
  cognom2: 'Soler',  
  dataEntradaTasca: '2025-12-10T09:30:00Z',  
  completa: true,  
  observacions: 'Tasca entregada abans de la data límit'  
}  
{  
  _id: ObjectId('695eedf878347211682c4d29'),  
  nomAlumne: 'Anna',  
  cognom1: 'Vidal',  
  cognom2: 'Riera',  
  dataEntradaTasca: '2025-12-14T18:45:00Z',  
  completa: true,  
  observacions: 'Molt bona presentació i contingut'  
}  
db.tasques.entrades.find()  
{  
  _id: ObjectId('695eedf878347211682c4d27'),  
  nomAlumne: 'Laura',  
  cognom1: 'Martínez',
```

// Llistar totes les tasques.

```
db.tasques.entrades.find()
```

// Llistar les tasques completades.

```
db.tasques.entrades.find({ completa: true })
```

// Llistar les tasques d'un alumne específic.

```
db.tasques.entrades.find({ nomAlumne: "Laura" })
```

// Cerca les tasques de determinat alumne i completades.

```
db.tasques.entrades.find({ nomAlumne: "Anna", completa: true })
```

// Llista les tasques completades i amb observacions.

```
db.tasques.entrades.find({ completa: true, observacions: { $exists: true, $ne: "" } })
```

// Llista les tasques completades després d'una certa data.

```
db.tasques.entrades.find({
```

```
completa: true,  
dataEntradaTasca: { $gt: ISODate("2025-12-11T00:00:00Z") }  
})
```

```
// Llista les tasques amb observacions que continguin una paraula específica.  
db.tasques.entrades.find({ observacions: /exercicis/ })
```

```
// Llista les tasques no completades ordenades alfabèticament.  
db.tasques.entrades.find({ completa: false }).sort({ nomAlumne: 1 })
```

```
// Llista les tasques entre dues dates específiques.  
db.tasques.entrades.find(  
  dataEntradaTasca: {  
    $gte: ISODate("2025-12-10T00:00:00Z"),  
    $lte: ISODate("2025-12-14T23:59:59Z")  
  }  
)
```

```
// Llista les tasques amb observacions que comencin amb una determinada paraula  
// ordenades per data més nova a més antiga.  
db.tasques.entrades.find({ observacions: /^Tasca/ }).sort({ dataEntradaTasca: -1 })
```

Activitat 2

Proposta d'aplicació NO-SQL

Nom de l'aplicació:

EventBooker

Descripció

Aquesta aplicació és per gestionar esdeveniments i reserves.

Permet crear esdeveniments, buscar-los per data, categoria i preu, i fer operacions CRUD.

Justificació NoSQL

MongoDB és bo perquè:

- Les dades són flexibles
- No tots els esdeveniments tenen els mateixos camps
- Fem moltes consultes per filtres
- És escalable

Tipus de dades

- Text
- Números
- Dates
- Categories

Disseny de la Base de Dades

Col·lecció principal: events

Aquesta col·lecció té tots els esdeveniments de l'aplicació.

Camps:

- title (string): nom de l'esdeveniment
- category (string): tipus d'esdeveniment
- price (double): preu
- eventDate (date): data de l'esdeveniment (és obligatori)
- location (string): lloc
- available (boolean): si està disponible

Justificació

Hem fet una sola col·lecció principal perquè:

- Les dades són molt relacionades
- Evitem unions innecessàries
- Fa facilitat les consultes

Esquema JSON Schema

```
json:  
{  
  "bsonType": "object",  
  "required": ["title", "category", "eventDate"],  
  "properties": {  
    "title": { "bsonType": "string" },  
    "category": { "bsonType": "string" },  
    "price": { "bsonType": "double" },  
    "eventDate": { "bsonType": "date" },  
    "location": { "bsonType": "string" },  
    "available": { "bsonType": "bool" }  
  }  
}
```

The screenshot shows the MongoDB Compass application interface. At the top, it displays the organization and project details: 'ORGANIZATION Bilal's Org - 2025-12-18' and 'PROJECT Project 0'. The main area is the 'Data Explorer' tab, which includes sections for 'My Queries', 'Data Modeling', and 'CLUSTERS'. The 'CLUSTERS' section shows a tree view with 'Cluster0' expanded, containing 'admin', 'local', and 'practica3' collections. 'practica3' is further expanded to show 'events', 'sample_mflix', and 'tasques' sub-collections. The 'events' collection is currently selected. On the right side, the document list for 'events' is displayed under the 'Documents' tab, showing 25 results. One document is highlighted, showing its details:

```
_id: ObjectId('695fd02e4390f743c1969d9a')  
title : "Concert Rock"  
category : "music"  
price : 25  
eventDate : "2025-01-10T18:00:00Z"  
location : "Barcelona"  
available : true
```

Below this, another document is partially visible:

```
_id: ObjectId('695fd02e4390f743c1969d9b')
```

mongo compas

The screenshot shows the Compass MongoDB interface. On the left, there's a sidebar with 'My Queries', 'Data Modeling', and a 'CONNECTIONS' section listing 'bilal', 'cluster0.ye8yv1r.mongodb.net' (selected), and 'local'. The 'local' section contains 'practica3' which has an 'events' collection. A red box highlights the 'events' collection. The main panel shows the 'events' collection with 25 documents. One document is expanded to show its fields: '_id: ObjectId('695fd02e4390f743c1969d9a'), title: 'Concert Rock', category: 'music', price: 25, eventDate: '2025-01-10T18:00:00Z', location: 'Barcelona', available: true'. Another document is partially visible below it.

Consultes:

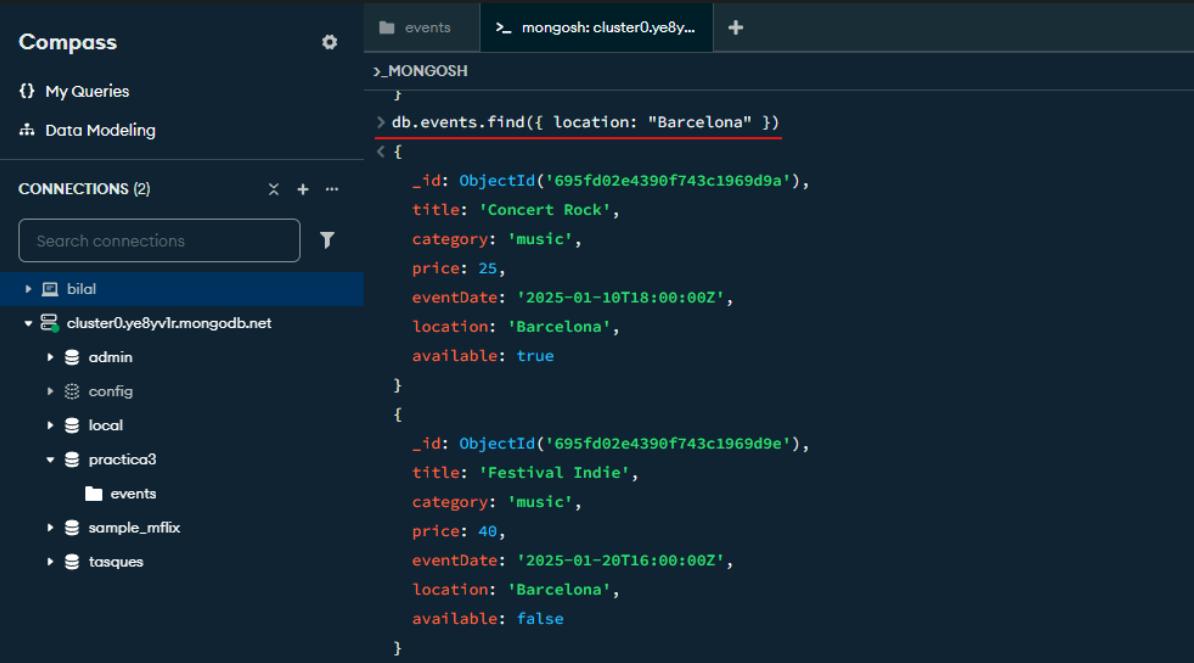
Filtrar per preu més gran que 20:

```
db.events.find({ price: { $gt: 20 } })
```

The screenshot shows the Compass MongoDB interface with the query results. The results pane displays two documents that match the filter: 'price: { \$gt: 20 }'. The first document is the same as the one shown in the previous screenshot: '_id: ObjectId('695fd02e4390f743c1969d9a'), title: 'Concert Rock', category: 'music', price: 25, eventDate: '2025-01-10T18:00:00Z', location: 'Barcelona', available: true'. The second document is: '_id: ObjectId('695fd02e4390f743c1969d9c'), title: 'Teatre Clàssic', category: 'theatre', price: 30, eventDate: '2025-01-15T20:00:00Z', location: 'Tarragona', available: true'.

Filtrar per ciutat

```
db.events.find({ location: "Barcelona" })
```



```
_id: ObjectId('695fd02e4390f743c1969d9a'),
title: 'Concert Rock',
category: 'music',
price: 25,
eventDate: '2025-01-10T18:00:00Z',
location: 'Barcelona',
available: true

{
  _id: ObjectId('695fd02e4390f743c1969d9e'),
  title: 'Festival Indie',
  category: 'music',
  price: 40,
  eventDate: '2025-01-20T16:00:00Z',
  location: 'Barcelona',
  available: false
}
```

Connexió Java-MongoDB i CRUD:

link github:

https://github.com/BILALNAMIR/practica3-nosql-mongodb/blob/main/mongoEvents/src/main/java/dam/uf3/mongo/App_practica3.java