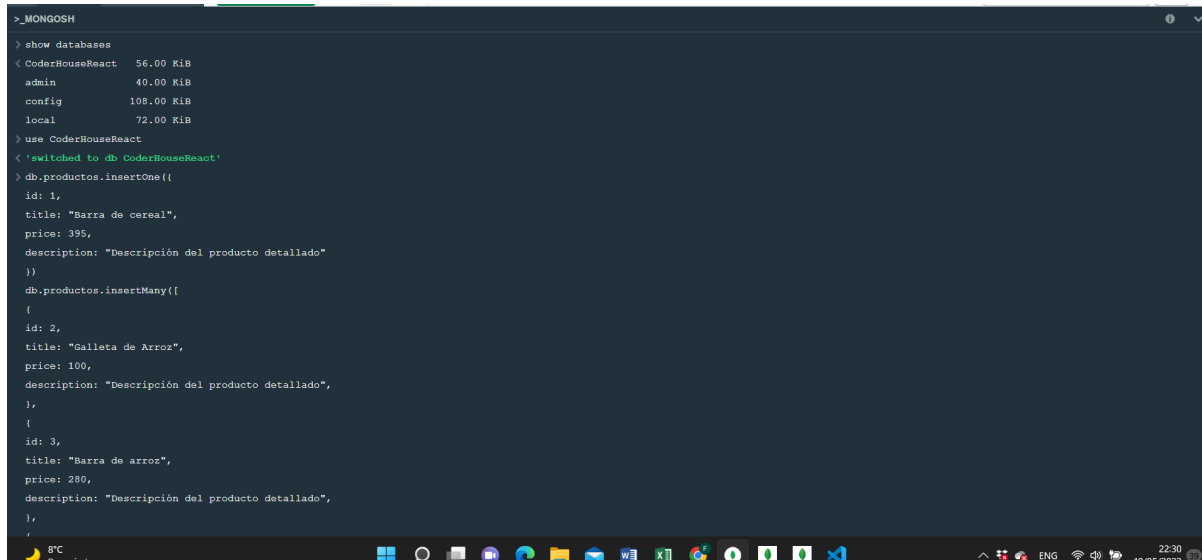


Paso a paso

- 1) Instale mongodb y configurarlo.
- 2) Abrí mongod.exe (servidor) y mongo.exe (cliente)
- 3) Copie url de mongo.exe para conectarlo en mongodb Compass
- 4) Cree base de datos "CoderHouseReact".
- 5) Show databases;
 - a) Entre a "CoderHouseReact".
- 6) Create Collection:
 - a) productos
- 7) Agregue 10 documentos:
 - i)

```
db.productos.insertOne({ id: 1, title: "Barra de cereal", price: 395, description: "Descripción del producto detallado" })
db.productos.insertMany([ { id: 2, title: "Galleta de Arroz", price: 100, description: "Descripción del producto detallado", }, { id: 3, title: "Barra de arroz", price: 280, description: "Descripción del producto detallado", }, { id: 4, title: "Alfajor de arroz", price: 450, description: "Descripción del producto detallado", }, { id: 5, title: "Snack de maíz", price: 100, description: "Descripción del producto detallado", }, { id: 6, title: "Mix de frutos secos", price: 300, description: "Descripción del producto detallado", }, { id: 7, title: "Maní Japones", price: 90, description: "Descripción del producto detallado", }, { id: 8, title: "Chips Mix vegetales", price: 340, description: "Descripción del producto detallado", }, { id: 9, title: "Wraps sin gluten", price: 2425, description: "Descripción del producto detallado", }, { id: 10, title: "Galletitas Santiveri", price: 110, description: "Descripción del producto detallado" }, ])
```



```
> MONGOSH
> show databases
{
  CoderHouseReact: 56.00 KiB,
  admin: 40.00 KiB,
  config: 108.00 KiB,
  local: 72.00 KiB
}
> use CoderHouseReact
switched to db CoderHouseReact
> db.productos.insertOne({
  id: 1,
  title: "Barra de cereal",
  price: 395,
  description: "Descripción del producto detallado"
})
{
  acknowledged: true,
  insertedId: 1
}
> db.productos.insertMany([
  {
    id: 2,
    title: "Galleta de Arroz",
    price: 100,
    description: "Descripción del producto detallado",
  },
  {
    id: 3,
    title: "Barra de arroz",
    price: 280,
    description: "Descripción del producto detallado",
  },
  {
    id: 4,
    title: "Alfajor de arroz",
    price: 450,
    description: "Descripción del producto detallado",
  },
  {
    id: 5,
    title: "Snack de maíz",
    price: 100,
    description: "Descripción del producto detallado",
  },
  {
    id: 6,
    title: "Mix de frutos secos",
    price: 300,
    description: "Descripción del producto detallado",
  },
  {
    id: 7,
    title: "Maní Japones",
    price: 90,
    description: "Descripción del producto detallado",
  },
  {
    id: 8,
    title: "Chips Mix vegetales",
    price: 340,
    description: "Descripción del producto detallado",
  },
  {
    id: 9,
    title: "Wraps sin gluten",
    price: 2425,
    description: "Descripción del producto detallado",
  },
  {
    id: 10,
    title: "Galletitas Santiveri",
    price: 110,
    description: "Descripción del producto detallado",
  }
])
{
  acknowledged: true,
  insertedIds: [
    2, 3, 4, 5, 6, 7, 8, 9, 10
  ]
}
```

- 8)
- 9) Ver cantidad de documentos en coleccion "productos"
 - a) `db.productos.find()`
- 10) Creo coleccion "mensajes" y agrego 10 docs
 - a)

```
db.mensajes.insertMany([ { text: "texto1", date: new Date(), }, { text: "texto2", date: new Date(), }, { text: "texto3", date: new Date(), }, ])
db.mensajes.insertMany([ { text: "texto4", date: new Date(), }, { text:
```

```
"texto5", date: new Date(), }, { text: "texto6", date: new Date(), }, { text:
"texto7", date: new Date(), }, { text: "texto8", date: new Date(), }, { text:
"texto9", date: new Date(), }, { text: "texto10", date: new Date(), }, ])
```

11) Ver cantidad de documentos en coleccion "mensajes"

a) db.mensajes.find()

12) Agregar 1 doc mas a "productos"

a) db.productos.insertOne()

Cosultas especificas:

13) Pido producto con precio igual a 395

a) db.productos.find({price: 395})

14) Prod con \$ <500

a) db.productos.find({price: {\$lt: 500}})

15) >\$500

a) db.productos.find({price: {\$gt: 500}})

16) Agregar stock 1 a todos los productos

a) db.productoss.updateMany({price: {\$lt: 3000}}, { \$set: {stock: 1 } })

17) Ahora stock a 0

a) db.productoss.updateMany({price: {\$gt: 4000}}, { \$set: {stock: 0 } })

18) Eliminar prods con \$ <200

a) db.productos.deleteMany({price: {\$lt: 200}})

19) Crear usuario "pepe" clave: "asd456" que solo pueda leer la base de datos

a) db.createUser({user: "pepe", pwd: "asd456", roles: [{role: "read", db: "CoderHouseReact"}]})

20) Me conecto con el usuario y pwd de "pepe" y al tratar de crear una collection no me lo permite ya que tiene acceso de "solo-lectura".