

Unidad 4

Ejercicios 1 al 3

1. **Crear una BD en MongoDB con tu nombre y dentro de ella crear una colección llamada “juegos”. De los juegos se puede almacenar la siguiente información: título, género, precio y fecha de lanzamiento.**

```
use Francisco
```

- a) **Insertar un juego rellenando todos los campos.**

```
db.createCollection("juegos")
```

```
db.juegos.insertOne({"titulo":"Stardew  
Valley","genero":"Simulación","precio":13.99,"lanzamiento":new  
ISODate("2016-02-26")})
```

- b) **Insertar un juego sin indicar la fecha de lanzamiento.**

```
db.juegos.insertOne({"titulo":"Minecraft","genero":"Sandbox","precio":2  
9.99})
```

- c) **Realizar una búsqueda de todos los juegos del mismo género y ordenarlos por título (ascendente).**

```
db.juegos.find({genero:"Simulación"}).sort({titulo:1})
```

2. **Crear la colección “empleados” dentro de la BD “mibd” y añadir los siguientes documentos:**

- *emp_no: 1, nombre: “Juan”, dep: 10, salario: 1000, fecha_alta: “10/10/1999”.*
- *emp_no: 2, nombre: “Alicia”, dep: 10, salario: 1400, fecha_alta: “07/08/200”, oficio: “Profesora”.*
- *emp_no: 3, nombre: “María Jesús”, dep: 20, salario: 1500, fecha_alta: “05/01/2005”, oficio: “Analista”, comision: 100.*
- *emp_no: 4, nombre: “Alberto”, dep: 20, salario: 1100, fecha_alta: “15/11/2001”.*
- *emp_no: 5, nombre: “Fernando”, dep: 30, salario: 1400, fecha_alta: “20/11/1999”, comision: 200, oficio: “Analista”.*

Proponer las sentencias a ejecutar para realizar las siguientes operaciones:

- a) **Crear la BD y la colección.**

```
use mibd
```

```
db.createCollection("empleados")
```

- b) **Insertar los documentos dados.**

```
db.empleados.insertMany([  
  { "emp_no": 1, "nombre": "Juan", "dep": 10, "salario": 1000, "fecha_alta":  
ISODate("1999-10-10T00:00:00Z") },  
  { "emp_no": 2, "nombre": "Alicia", "dep": 10, "salario": 1400, "fecha_alta":  
ISODate("2000-08-07T00:00:00Z"), "oficio": "Profesora" },
```

```
{ "emp_no": 3, "nombre": "María Jesús", "dep": 20, "salario": 1500,
"fecha_alta": ISODate("2005-01-05T00:00:00Z"), "oficio": "Analista", "comision":
100 },
{ "emp_no": 4, "nombre": "Alberto", "dep": 20, "salario": 1100, "fecha_alta":
ISODate("2001-11-15T00:00:00Z") },
{ "emp_no": 5, "nombre": "Fernando", "dep": 30, "salario": 1400, "fecha_alta":
ISODate("1999-11-20T00:00:00Z"), "comision": 200, "oficio": "Analista" }
])
```

c) Mostrar los empleados del departamento 10.

```
db.empleados.find({dep:10})
```

d) Mostrar los empleados del departamento 10 y 20.

```
db.empleados.find({dep:{$in: [10,20]}})
```

e) Mostrar los empleados con salario > 1300 y oficio “Profesora”.

```
db.empleados.find({salario:{$gt:1300},oficio:"Profesora"})
```

f) Subir el salario a todos los analistas 100€.

```
db.empleados.updateMany({oficio:"Analista"},{$inc:{salario:100}})
```

g) Decrementar la comisión en 20€ (-20€), sólo a los que tengan comisión.

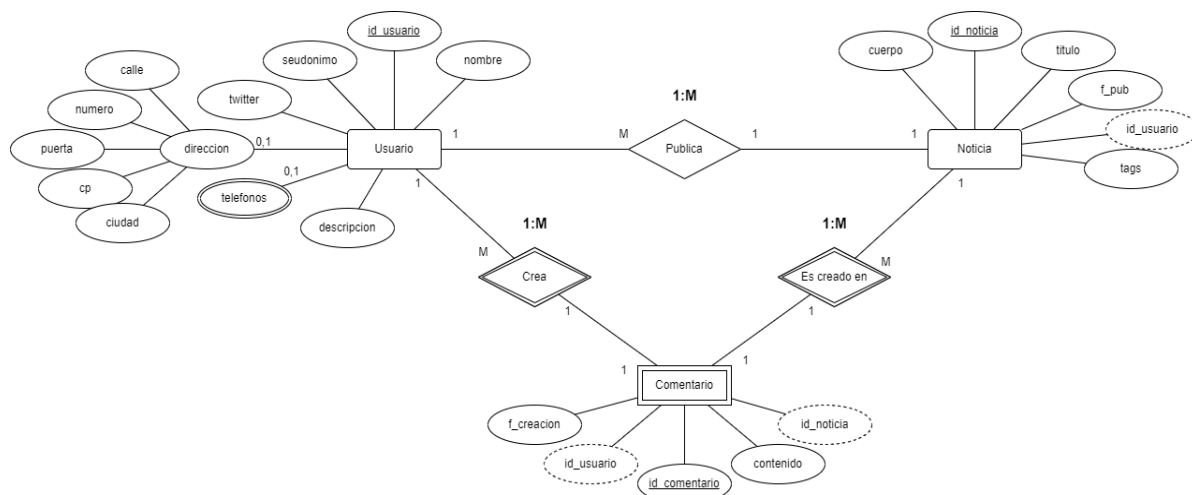
```
db.empleados.updateMany({comision:{$exists : true}},{$inc:
{comision:-20}})
```

3. Se precisa diseñar un blog de noticias donde los usuarios registrados puedan publicar sus comentarios:

- *Cada usuario tiene un identificador único, un nombre, un seudónimo, una cuenta de Twitter y una descripción. Además, de forma opcional, los usuarios pueden proporcionar como datos su dirección postal (calle, número, puerta, código postal y ciudad) o sus teléfonos de contacto (pueden tener varios).*
- *Las noticias tienen un identificador único, un título, un cuerpo y una fecha de publicación. Son publicadas por un usuario y pueden contener o no, una lista de tags.*
- *Las noticias reciben comentarios, quedando registrado el usuario que lo escribió, el comentario escrito y el momento en el que lo hizo.*

Se pide:

a) Realizar el diagrama entidad/relación.



```
use blog
db.createCollection("usuario")
db.createCollection("comentario")
db.createCollection("noticia")
```

```

D
db.usuario.insertOne({
  _id: ObjectId(), nombre: "Juan", seudonimo: "JPerez", twitter: "@juanperez",
  descripcion: "Periodista",
  direccion: { calle: "Gran Vía", numero: 10, puerta: "B", cp: "28013", ciudad:
  "Madrid" }, telefonos: ["666123456", "915678900"]
})

db.noticia.insertOne({
  _id: ObjectId(), titulo: "Nuevo descubrimiento en Marte", cuerpo: "Han
  encontrado agua subterránea...",
  f_pub: ISODate("2025-02-25T10:30:00Z"), id_usuario: db.usuario.findOne({ nombre:
  "Juan" })._id, tags: ["ciencia", "espacio", "Marte"]
})

db.comentarios.insertOne({
  _id: ObjectId(), id_noticia: db.noticia.findOne({ titulo: "Nuevo descubrimiento
  en Marte" })._id,
  id_usuario: db.usuario.findOne({ nombre: "Juan" })._id,
  contenido: "Gran descubrimiento", f_creacion: ISODate("2025-02-25T11:00:00Z")
})

```

d) Mostrar los datos de un usuario dado su nombre de usuario y cuenta de Twitter.

```
db.usuario.findOne({seudonimo:"JPerez",twitter:"@juanperez"})
```

e) Mostrar el número de usuarios de cada código postal.

```
db.usuario.aggregate([
  { $group: { _id: 28013, totalUsuarios: { $sum: 1 } } }
])
```

f) Mostrar las 10 últimas noticias publicadas (más recientes).

```
db.noticia.find().sort({ f_pub: -1 }).limit(10)
```

g) Mostrar el número de comentarios por noticia o por usuario (dos sentencias distintas).

```
db.comentario.aggregate([
  { $group: { _id: db.noticia.findOne({ titulo: "Nuevo descubrimiento en
Marte" })._id, totalComentarios: { $sum: 1 } } }
])
```

```
db.comentarios.aggregate([
  { $group: { _id: db.usuario.findOne({ nombre: "Juan" })._id,
totalComentarios: { $sum: 1 } } }
])
```

