

Mostrar las personas de 18 años ordenando sus amigos de modo descendente por edad.

La consulta en MongoDB Compass es:

```
find(eq( fieldName: "Edad", value: 18));
```

Recordar que tenemos en la colección Amigos dentro de la base de datos ejemplo.

```
{
  "_id": ObjectId("5a6f05fbb56b58b447e66801"),
  "Nombre": "Luis",
  "Apellidos": "Martinez",
  "Edad": 18,
  "Aficiones": ["fútbol", "senderismo", "tenis"],
  "Amigos": [
    {"Nombre": "Monica", "Edad": 20},
    {"Nombre": "Andrés", "Edad": 24}
  ]
},
{
  "_id": ObjectId("5a6f05fbb56b58b447e66802"),
  "Nombre": "Marisa",
  "Apellidos": "Garcia",
  "Edad": 18,
  "Aficiones": ["senderismo", "tenis", "pintura"],
  "Amigos": [
    {"Nombre": "Monica", "Edad": 20}
  ]
},
{
  "_id": ObjectId("5a6f05fbb56b58b447e66803"),
  "Nombre": "Joquin",
  "Apellidos": "Rodriguez",
  "Edad": 21,
  "Aficiones": ["fútbol", "natación"],
  "Amigos": [
    {"Nombre": "Andrés", "Edad": 24}
  ]
},
{
  "_id": ObjectId("5a6f05fbb56b58b447e66804"),
  "Nombre": "Luis",
  "Apellidos": "Martinez",
  "Edad": 20,
  "Aficiones": ["natación", "culturismo"],
  "Amigos": [
    {"Nombre": "Juan", "Edad": 40},
    {"Nombre": "Antonio", "Edad": 52}
  ]
}
];
```

El resultado sería

Se encontraron resultados para la consulta.

Nombre: Luis Martinez

Edad: 18

Los Amigos de Luis (ordenados por edad descendente):

- Andrés (24 años)
- Monica (20 años)

Nombre: Marisa Garcia

Edad: 18

Los Amigos de Marisa (ordenados por edad descendente):

- Monica (20 años)
-

Usar las librerías

```
import com.mongodb.client.*;  
import org.bson.Document;  
import java.util.Arrays;  
import static com.mongodb.client.model.Filters.eq;  
import static com.mongodb.client.model.Sorts.descending;
```

CODIGO

```
FindIterable<Document> resultados = collection.find(eq("Edad", 18));
```

```

public class Operaciones { 1 usage
    public static void consulta() { 1 usage
        // Conectar a MongoDB
        String uri = "mongodb://localhost:27017";
        MongoClient mongoClient = MongoClient.create(uri);
        // Seleccionar la base de datos y la colección
        MongoDB database = mongoClient.getDatabase( s: "ejemplos");
        MongoCollection<Document> collection = database.getCollection( s: "amigos");
        // Mostrar las colecciones para verificar que existe 'amigos'
        System.out.println("Colecciones disponibles:");
        for (String name : database.listCollectionNames()) {
            System.out.println(name);
        }
        // Consulta para obtener personas con 18 años
        FindIterable<Document> resultados = collection.find(eq( fieldName: "Edad", value: 18));
        // Verificar si hay resultados
        if (resultados.iterator().hasNext()) {
            System.out.println("Se encontraron resultados para la consulta.");
        } else {
            System.out.println("No se encontraron resultados para la consulta.");
        }
        // Procesar resultados
        try {
            for (Document doc : resultados) {
                // System.out.println("Documento encontrado: " + doc.toJson()); // Mostrar documento completo

                String nombre = doc.getString( key: "Nombre");
                String apellidos = doc.getString( key: "Apellidos");
                int edad = doc.getInteger( key: "Edad");

                System.out.println("Nombre: " + nombre + " " + apellidos);
                System.out.println("Edad: " + edad);
                System.out.println("Los Amigos de " + nombre + " (ordenados por edad descendente):");

                // Obtener y ordenar los amigos por edad descendente
                Document[] amigos = doc.getList( key: "Amigos", Document.class).toArray(new Document[0]);
                Arrays.sort(amigos, (a, b) -> b.getInteger( key: "Edad") - a.getInteger( key: "Edad"));

                for (Document amigo : amigos) {
                    System.out.println(" - " + amigo.getString( key: "Nombre") + " (" + amigo.getInteger( key: "Edad")
                }
                System.out.println("-----");
            }
        } catch (Exception e) {
            System.out.println("Error procesando los documentos: " + e.getMessage());
        }
    }
}

```


