

Métodos

```
public class DAOAerolinea {

    MongoClient mongoClient;
    MongoDB database;
    MongoCollection<Document> collection;

    public DAOAerolinea() {
        mongoClient = new MongoClient();
        database = mongoClient.getDatabase("airport");
        collection = database.getCollection("airlines");
    }

    public ArrayList<Aerolinea> obtenerAerolineas() {
        ArrayList<Aerolinea> aerolineas = new ArrayList<>();
        MongoCursor<Document> cursor = collection.find().iterator();
        try {
            while (cursor.hasNext()) {
                Document d = cursor.next();
                boolean lowcost = false;
                if (d.containsKey("lowcost")) {
                    lowcost = d.getBoolean("lowcost");
                }
                Aerolinea a = new Aerolinea(d.getObjectId("_id"), d.getString("name"), d.getString("country"), d.getString("currency"), lowcost);
                aerolineas.add(a);
            }
        } finally {
            cursor.close();
        }
        return aerolineas;
    }

    public void AgregarAerolinea(Aerolinea a) {
        Document aerolinea = new Document("name", a.getNombre())
            .append("country", a.getPais())
            .append("currency", a.getMoneda())
            .append("lowcost", a.isEconomica());
        collection.insertOne(aerolinea);
    }
}
```

```
public boolean eliminarAerolinea(String id) {
    ObjectId objectId;
    try {
        objectId = new ObjectId(id);
    } catch (IllegalArgumentException e) {
        System.out.println(x:"El ID proporcionado no es válido.");
        return false;
    }

    Document query = new Document("_id", objectId);
    Document result = collection.find(query).first();

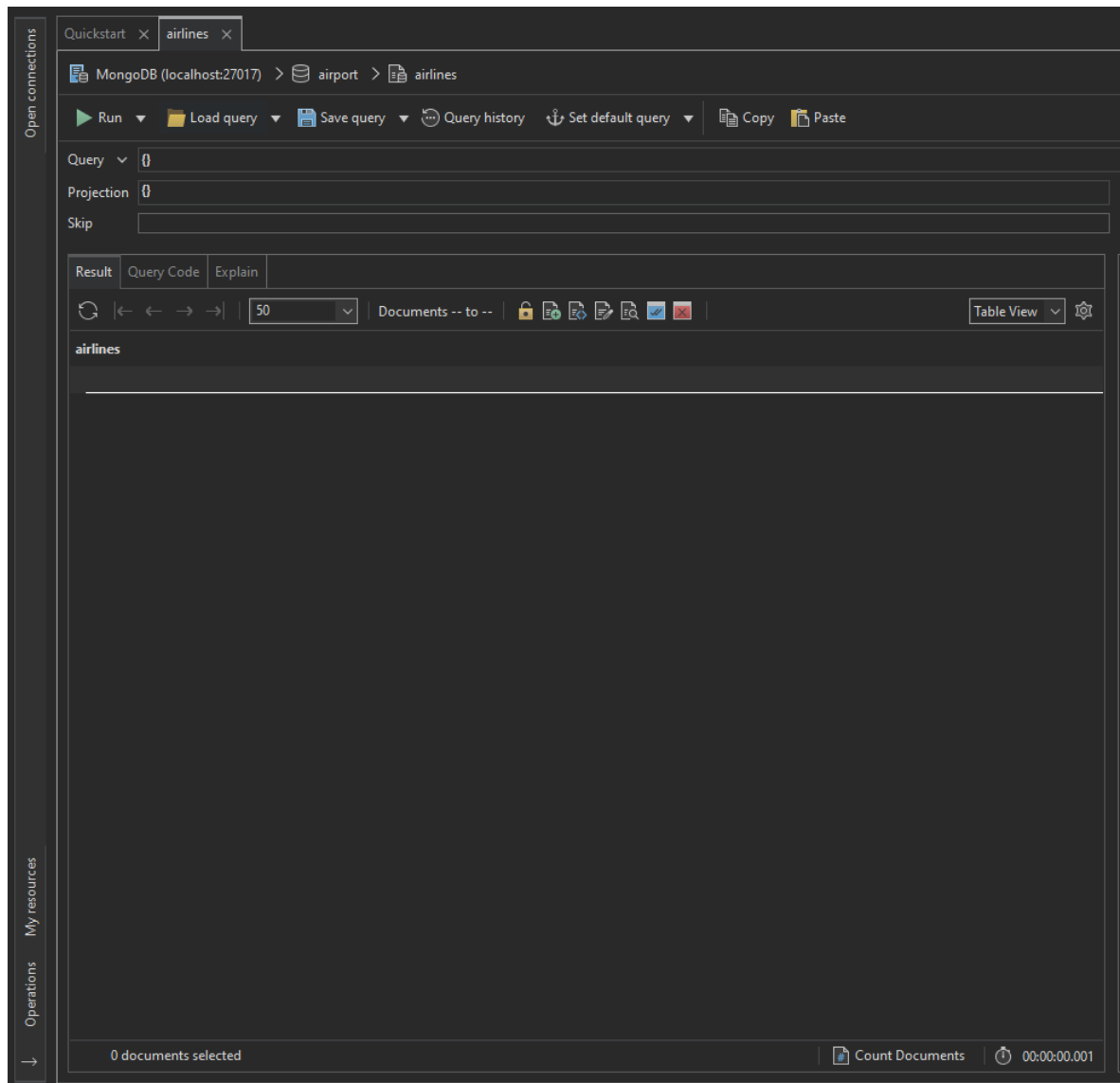
    if (result == null) {
        System.out.println(x:"No se encontró ninguna aerolínea con el ID proporcionado.");
        return false;
    }

    collection.deleteOne(query);
    return true;
}

public String obtenerPrimerIdAerolinea() {
    Document primero = collection.find().first();
    if (primero != null) {
        ObjectId objectId = primero.getObjectId("_id");
        return objectId.toHexString();
    } else {
        System.out.println(x:"No se encontraron aerolíneas en la colección.");
        return null;
    }
}
```

FRANCISCO DE JESÚS LÓPEZ RUIZ

Aerolíneas



Antes de utilizar las pruebas y métodos.

FRANCISCO DE JESÚS LÓPEZ RUIZ

Aerolíneas

```
*
* @author ID145
*/
public class AirportMongo {

    Run | Debug
    public static void main(String[] args) {
        DAOAerolinea a = new DAOAerolinea();

        // Método agregar.
        Aerolinea aerolinea = new Aerolinea(nombre:"Volaris", pais:"mexico", moneda:"MXN", economica:true);
        Aerolinea aerolinea1 = new Aerolinea(nombre:"Interjet", pais:"Mexico", moneda:"MXN", economica:true);
        Aerolinea aerolinea2 = new Aerolinea(nombre:"American Airlines", pais:"Estados Unidos", moneda:"USD", economica:false);
        Aerolinea aerolinea3 = new Aerolinea(nombre:"Emirates", pais:"Emiratos Árabes Unidos", moneda:"AED", economica:false);
        Aerolinea aerolinea4 = new Aerolinea(nombre:"British Airways", pais:"Reino Unido", moneda:"GBP", economica:true);
        a.AgregarAerolinea(aerolinea);
        a.AgregarAerolinea(aerolinea1);
        a.AgregarAerolinea(aerolinea2);
        a.AgregarAerolinea(aerolinea3);
        a.AgregarAerolinea(aerolinea4);

        // Método obtener todas.
        ArrayList<Aerolinea> lista = a.obtenerAerolineas();
        for (Aerolinea as : lista) {
            System.out.println(a.toString());
        }

        // Método eliminar.
        String primerId = a.obtenerPrimerIdAerolinea();
        if (primerId != null) {
            System.out.println("ID del primer documento encontrado: " + primerId);

            boolean eliminado = a.eliminarAerolinea(primerId);
            if (eliminado) {
                System.out.println(x:"Aerolínea eliminada correctamente.");
            } else {
                System.out.println(x:"La aerolínea no pudo ser eliminada.");
            }
        }
    }
}
```

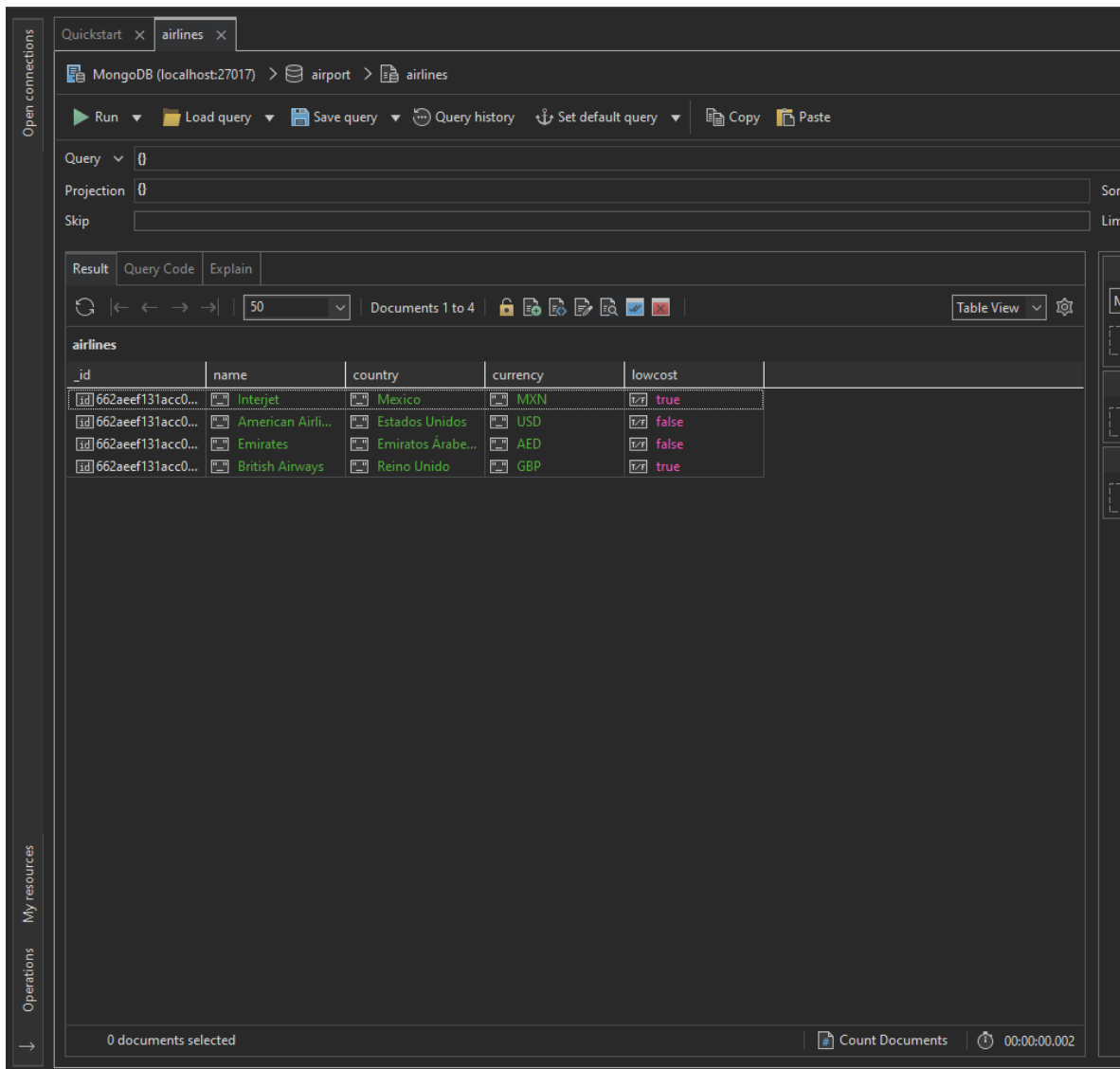
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\ID145\OneDrive\Documents\Github\AerolineasMongo_FranciscoLopez> & 'C:\Program Files\Java\jdk-21\bin\java.exe' -gc:\Users\ID145\AppData\Local\Temp\cp_9kq7n8f1462wes86fshg5f11.argfile' -bda:itson.airportmongo
..AirportMongo'
Apr 25, 2024 5:01:53 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Cluster created with settings (hosts=[127.0.0.1:27017], mode=SINGLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30000 ms', maxWaitQueueSize=500)
Apr 25, 2024 5:01:53 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Cluster description not yet available. Waiting for 30000 ms before timing out
Apr 25, 2024 5:01:53 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:1}] to 127.0.0.1:27017
Apr 25, 2024 5:01:53 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Monitor thread successfully connected to server with description ServerDescription{address=127.0.0.1:27017, type=STANDALONE, state=CONNECTED, ok=true, version=ServerVersion{versionList=[7, 0, 8]}, minWireVersion=0, maxWireVersion=21, maxDocumentSize=16777216, logicalSessionTimeoutMinutes=30, roundTripTimeNanos=1895400}
Apr 25, 2024 5:01:53 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:2}] to 127.0.0.1:27017
datos.DAOAerolinea@29626d54
datos.DAOAerolinea@29626d54
datos.DAOAerolinea@29626d54
datos.DAOAerolinea@29626d54
datos.DAOAerolinea@29626d54
ID del primer documento encontrado: 662aeef131acc064f2f05837
Aerolínea eliminada correctamente.
PS C:\Users\ID145\OneDrive\Documents\Github\AerolineasMongo_FranciscoLopez>

Couldn't connect to Discord via RPC: Error: Could not connect
Reconnect to Discord
Ln 1, Col 1 Spaces: 4 UTF-8 LF ( ) Java
```

FRANCISCO DE JESÚS LÓPEZ RUIZ

Aerolíneas

Después.



The screenshot shows the MongoDB Compass interface. The top bar indicates the connection to 'MongoDB (localhost:27017)' and the database 'airport' with the collection 'airlines'. The query editor shows a query of '0'. The results section displays a table with the following data:

_id	name	country	currency	lowcost
662aeef131acc0...	Interjet	Mexico	MXN	true
662aeef131acc0...	American Airli...	Estados Unidos	USD	false
662aeef131acc0...	Emirates	Emiratos Árabe...	AED	false
662aeef131acc0...	British Airways	Reino Unido	GBP	true

The bottom status bar shows '0 documents selected' and a 'Count Documents' button.

[FranciscoLrf/aerolineasMongo FranciscoLopez: Ejercicio Aerolineas \(github.com\)](https://github.com/FranciscoLopez/aerolineasMongo)