# MONGODB

**DELETION OF RECORD:**

package connection;

import com.mongodb.client.\*;

import com.mongodb.client.model.Filters; import org.bson.Document;

public class Create {

public static void main(String[] args) {

// Creating a Mongo client

MongoClient mongoClient = MongoClients.*create*("mongodb://localhost:27017");

// Accessing the database

MongoDatabase database = mongoClient.getDatabase("myDb");

// Getting the collection

MongoCollection<Document> collection = database.getCollection("sampleCollection");

// Delete a document with First\_Name = "Mahesh" collection.deleteOne(Filters.*eq*("First\_Name", "Mahesh")); System.***out***.println("Document deleted.");

// Find and print all remaining documents FindIterable<Document> documents = collection.find(); System.***out***.println("Documents:");

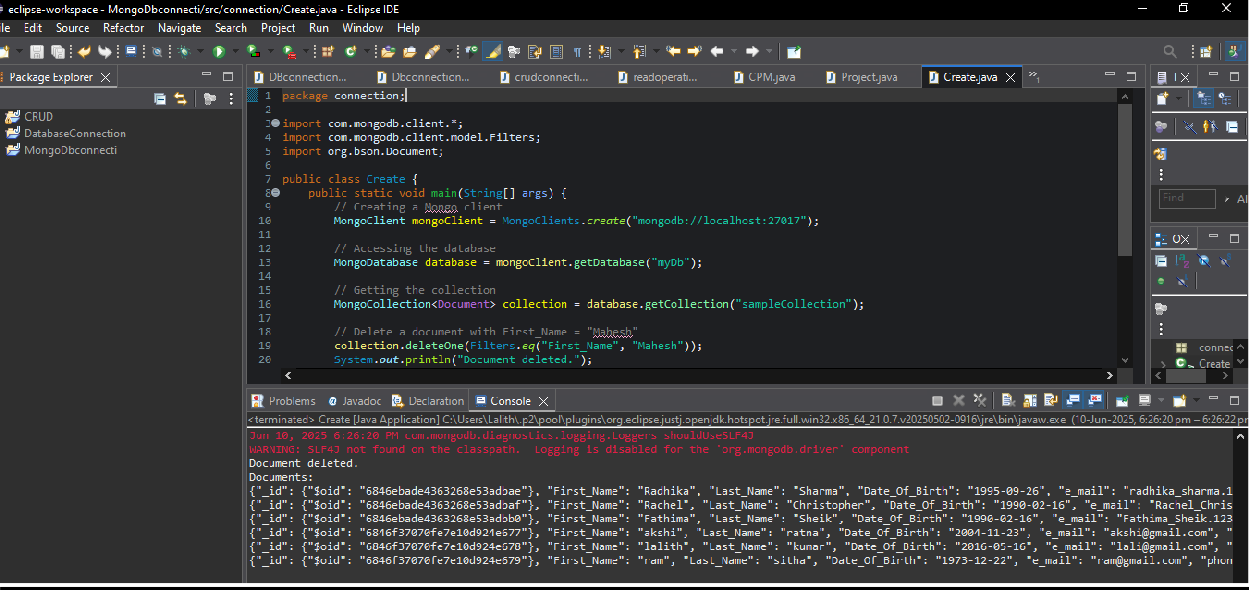
for (Document document : documents) { System.***out***.println(document.toJson());

}

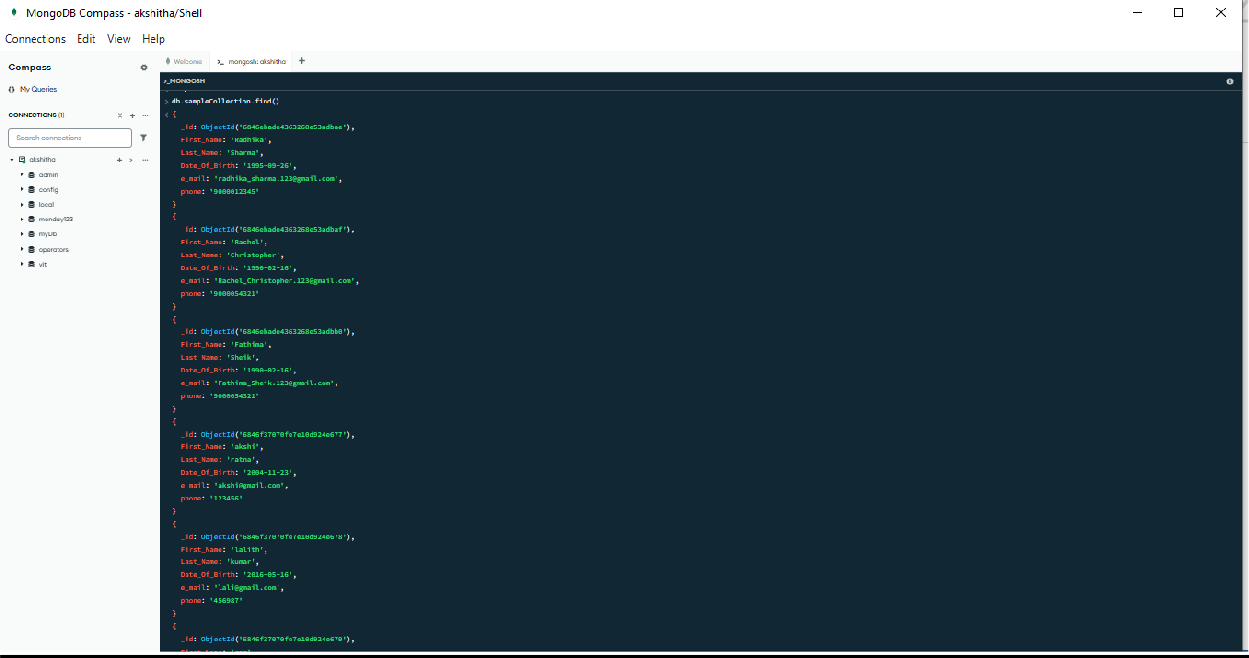
// Closing the client connection mongoClient.close();

}

}



# MONGODB:



## Deleting many recods where there is a cappuccino:

package connection;

import org.bson.Document; import com.mongodb.client.\*;

import com.mongodb.client.model.Filters;

import java.time.Instant; import java.util.Arrays;

public class Create {

public static void main(String[] args) {

// Connect to MongoDB

MongoClient mongoClient = MongoClients.*create*("mongodb://localhost:27017");

// Access the database

MongoDatabase database = mongoClient.getDatabase("myDb");

// Access the sales collection

MongoCollection<Document> salesCollection = database.getCollection("sales");

// Insert data (only if needed — comment this block out if already inserted) salesCollection.insertMany(Arrays.*asList*(

new Document("\_id", 1).append("item", "Americanos").append("price", 5).append("size", "Short").append("quantity", 22).append("date", Instant.*parse*("2022-01- 15T08:00:00Z")),

new Document("\_id", 2).append("item", "Cappuccino").append("price", 6).append("size", "Short").append("quantity", 12).append("date", Instant.*parse*("2022-01- 16T09:00:00Z")),

new Document("\_id", 3).append("item", "Lattes").append("price", 15).append("size", "Grande").append("quantity", 25).append("date", Instant.*parse*("2022-01-16T09:05:00Z")),

new Document("\_id", 4).append("item", "Mochas").append("price", 25).append("size",

"Tall").append("quantity", 11).append("date", Instant.*parse*("2022-02-17T08:00:00Z")),

new Document("\_id", 5).append("item", "Americanos").append("price", 10).append("size", "Grande").append("quantity", 12).append("date", Instant.*parse*("2022-02- 18T21:06:00Z")),

new Document("\_id", 6).append("item", "Cappuccino").append("price", 7).append("size", "Tall").append("quantity", 20).append("date", Instant.*parse*("2022-02- 20T10:07:00Z")),

new Document("\_id", 7).append("item", "Lattes").append("price", 25).append("size",

"Tall").append("quantity", 30).append("date", Instant.*parse*("2022-02-21T10:08:00Z")),

new Document("\_id", 8).append("item", "Americanos").append("price", 10).append("size", "Grande").append("quantity", 21).append("date", Instant.*parse*("2022-02- 22T14:09:00Z")),

new Document("\_id", 9).append("item", "Cappuccino").append("price", 10).append("size", "Grande").append("quantity", 17).append("date", Instant.*parse*("2022-02- 23T14:09:00Z")),

new Document("\_id", 10).append("item", "Americanos").append("price", 8).append("size", "Tall").append("quantity", 15).append("date", Instant.*parse*("2022-02- 25T14:09:00Z"))

));

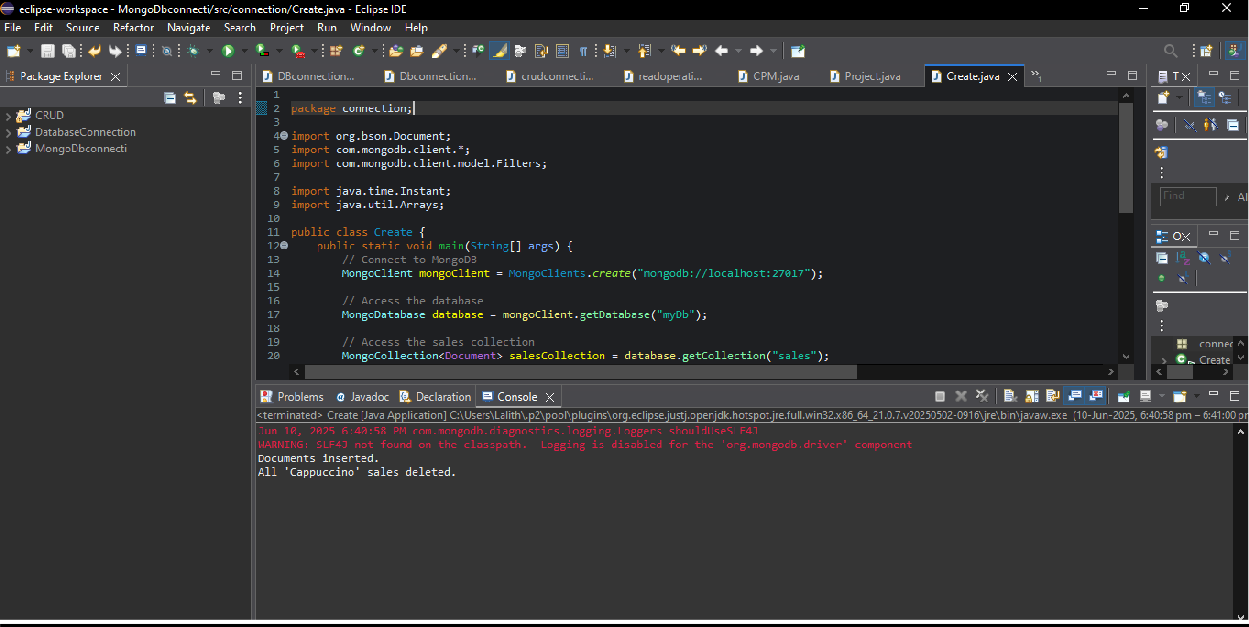
System.***out***.println("Documents inserted.");

// Delete all documents where item is "Cappuccino" salesCollection.deleteMany(Filters.*eq*("item", "Cappuccino")); System.***out***.println("All 'Cappuccino' sales deleted.");

mongoClient.close();

}

}

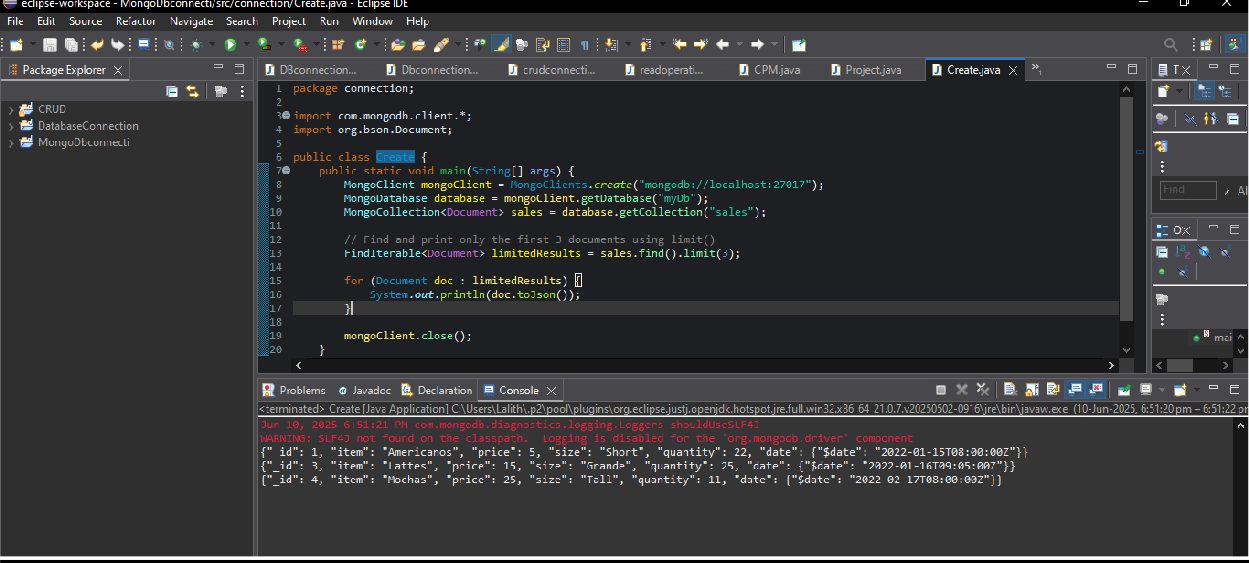


# MONGODB:



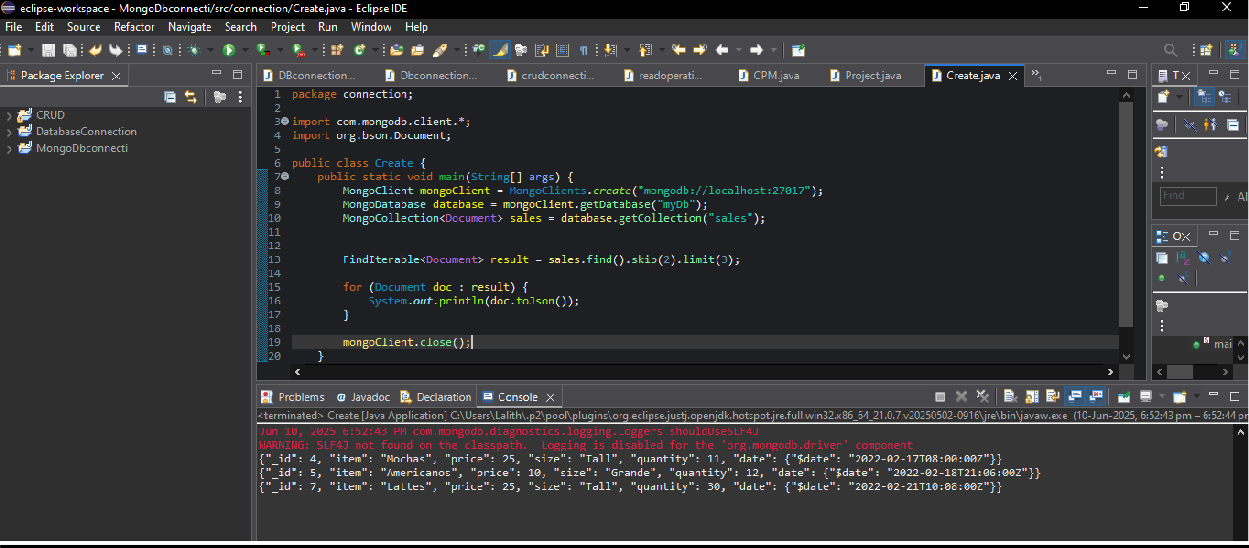
## Limit method:

FindIterable<Document> limitedResults = sales.find().limit(3);

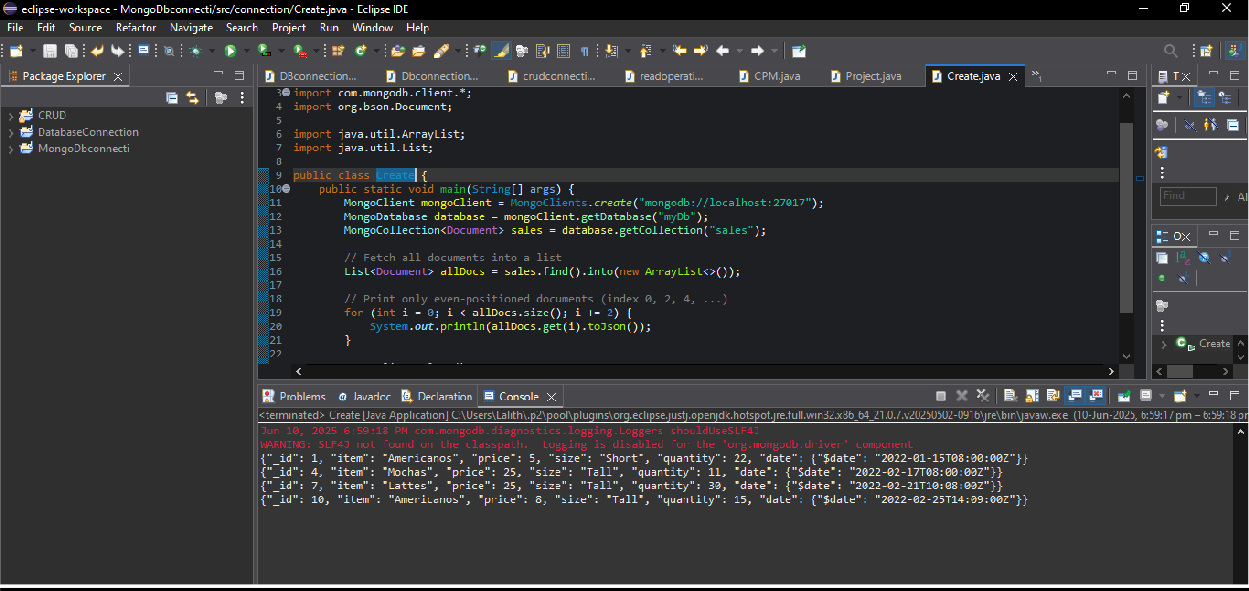


# SKIP:

FindIterable<Document> result = sales.find().skip(2).limit(3);



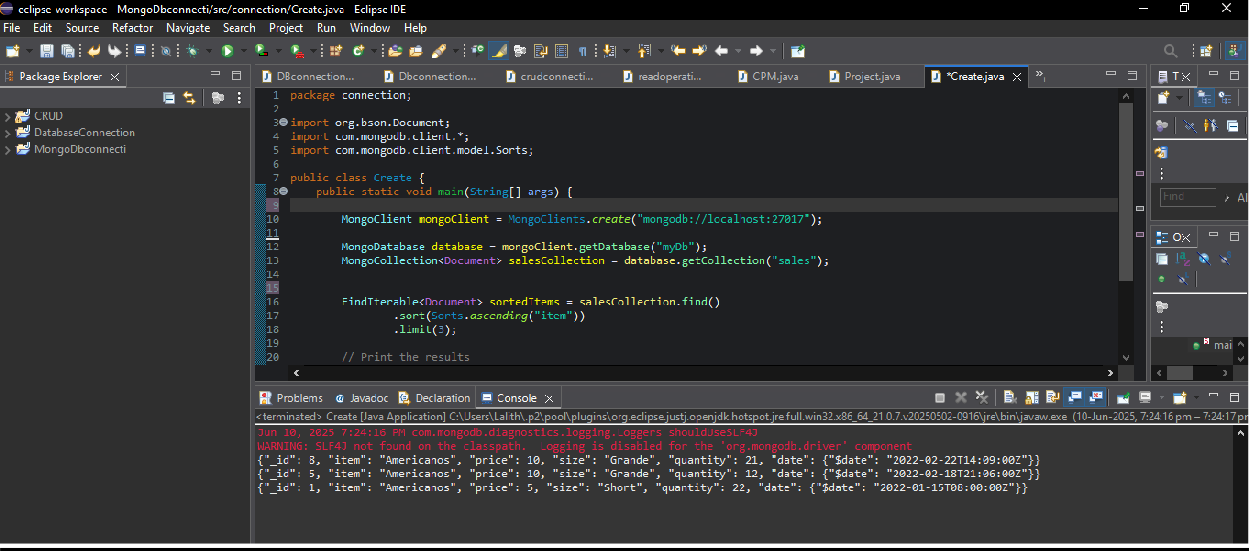
## Even position:

****

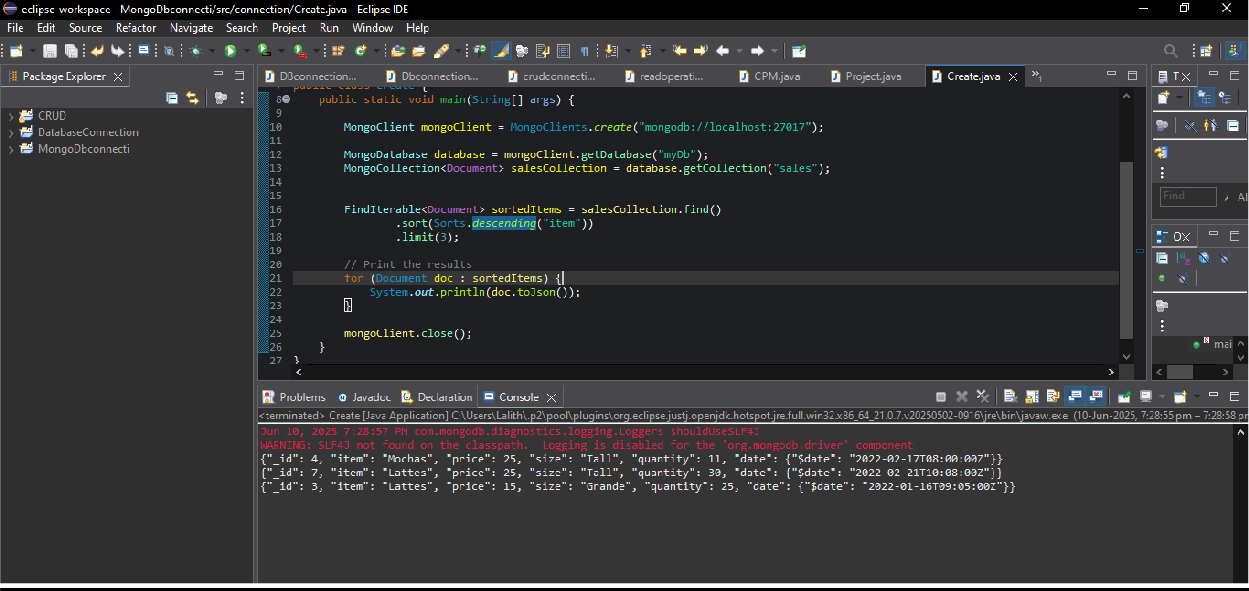
**Sort method:**

FindIterable<Document> sortedItems = salesCollection.find() .sort(Sorts.*ascending*("item"))

.limit(3);



**Descending:**

****