A screen shot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

CODE:

**package** connection;

**import** org.bson.Document;

**import** com.mongodb.client.FindIterable;

**import** com.mongodb.client.MongoClient;

**import** com.mongodb.client.MongoClients;

**import** com.mongodb.client.MongoCollection;

**import** com.mongodb.client.MongoDatabase;

**import** com.mongodb.client.MongoIterable;

**public** **class** collection\_two {

**public** **static** **void** main(String[] args) {

// Creating a Mongo client

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

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

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

Document priceRangeQuery = **new** Document("price", **new** Document("$gte", 700).append("$lte", 900));

FindIterable<Document> products = collection.find(priceRangeQuery);

**for** (Document product : products) {

System.***out***.println(product.toJson());

}

}

}

A screenshot of a computer screen

AI-generated content may be incorrect.

CODE:

**package** connection;

**import** java.util.Arrays;

**import** org.bson.Document;

**import** com.mongodb.client.AggregateIterable;

**import** com.mongodb.client.FindIterable;

**import** com.mongodb.client.MongoClient;

**import** com.mongodb.client.MongoClients;

**import** com.mongodb.client.MongoCollection;

**import** com.mongodb.client.MongoDatabase;

**import** com.mongodb.client.MongoIterable;

**public** **class** collection\_two {

**public** **static** **void** main(String[] args) {

// Creating a Mongo client

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

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

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

AggregateIterable<Document> result = collection.aggregate(Arrays.*asList*(

**new** Document("$match", **new** Document("price", **new** Document("$ne", **null**))), // filter documents where price is not null

**new** Document("$group", **new** Document("\_id", **null**)

.append("totalPrice", **new** Document("$sum", "$price"))

.append("count", **new** Document("$sum", 1)))

));

**for** (Document doc : result) {

System.***out***.println("Total Price: " + doc.getInteger("totalPrice"));

System.***out***.println("Count: " + doc.getInteger("count"));

}

}

}

A screenshot of a computer program

AI-generated content may be incorrect.

CODE:

**package** connection;

**import** java.util.Arrays;

**import** org.bson.Document;

**import** com.mongodb.client.AggregateIterable;

**import** com.mongodb.client.FindIterable;

**import** com.mongodb.client.MongoClient;

**import** com.mongodb.client.MongoClients;

**import** com.mongodb.client.MongoCollection;

**import** com.mongodb.client.MongoDatabase;

**import** com.mongodb.client.MongoIterable;

**public** **class** collection\_two {

**public** **static** **void** main(String[] args) {

// Creating a Mongo client

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

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

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

AggregateIterable<Document> result = collection.aggregate(Arrays.*asList*(

**new** Document("$sort", **new** Document("item", 1).append("price", -1)),

**new** Document("$group", **new** Document("\_id", "$item")

.append("maxPrice", **new** Document("$first", "$price"))

.append("document", **new** Document("$first", "$$ROOT"))),

**new** Document("$replaceRoot", **new** Document("newRoot", "$document"))

));

**for** (Document doc : result) {

System.***out***.println(doc.toJson());

}

}

}