Average:

**package** Connection;

**import** com.mongodb.client.\*;

**import** org.bson.Document;

**public** **class** average {

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

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

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

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

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

**double** totalPrice = 0;

**int** count = 0;

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

Number priceValue = doc.get("price", Number.**class**);

**if** (priceValue != **null**) {

totalPrice += priceValue.doubleValue();

count++;

}

}

**if** (count > 0) {

**double** average = totalPrice / count;

System.***out***.println("Average Price: " + average);

} **else** {

System.***out***.println("No products found.");

}

mongoClient.close();

}

2>

**package** Connection;

**import** com.mongodb.client.\*;

**import** org.bson.Document;

**import** **static** com.mongodb.client.model.Aggregates.*group*;

**import** **static** com.mongodb.client.model.Accumulators.*avg*;

**import** java.util.Arrays;

**public** **class** avg2 {

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

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*(

*group*(**null**, *avg*("avgPrice", "$price"))

));

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

**double** avgDouble = doc.getDouble("avgPrice");

**int** avgInt = (**int**) Math.*round*(avgDouble);

System.***out***.println("Average Price: " + avgInt);

}

mongoClient.close();

}

}

MAXIMUM:

**package** Connection;

**import** com.mongodb.client.\*;

**import** org.bson.Document;

**import** **static** com.mongodb.client.model.Aggregates.*group*;

**import** **static** com.mongodb.client.model.Accumulators.*max*;

**import** java.util.Arrays;

**public** **class** maximumprice {

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

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*(

*group*("$item", *max*("maxPrice", "$price"))

));

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

String item = doc.getString("\_id");

**int** maxPrice = doc.getInteger("maxPrice");

System.***out***.println("Item: " + item + ", Max Price: " + maxPrice);

}

mongoClient.close();

}

}

FILTERATION:

**package** Connection;

**import** com.mongodb.client.\*;

**import** org.bson.Document;

**import** **static** com.mongodb.client.model.Filters.\*;

**public** **class** filter1 {

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

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 doc : products) {

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

}

mongoClient.close();

}

}