Saturday Assignment

22BLC1194

- 1st question

```
import java.util.ArrayList;
import org.bson.Document;
import com.mongodb.BasicDBObject;
import com.mongodb.client.*;
import com.mongodb.client.model.Filters;

public class Connection_practice {
    public static void main(String[] args) {
        MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017");
        MongoDatabase database = mongoClient.getDatabase("Saturday");
        MongoCollection<Document> collection = database.getCollection("student");
        collection.drop();
        Document document1 = new Document("First_Name", "Sai")
            .append("Last_Name", "kushal")
            .append("Mark", 90)
            .append("age", 18);
        Document document2 = new Document("First_Name", "suneeth")
            .append("Last_Name", "reddy")
            .append("Mark", 90)
            .append("age", 22);
        Document document3 = new Document("First_Name", "krishna")
            .append("Last_Name", "kishore")
            .append("Mark", 91)
            .append("age", 20);
        Document document4 = new Document("First_Name", "Prem")
```

```
            .append("Last_Name", "sai")
            .append("Mark", 89)
            .append("age", 19);
        ArrayList<Document> documents = new ArrayList<Document>();
        documents.add(document1);
        documents.add(document2);
        documents.add(document3);
        documents.add(document4);
        collection.insertMany(documents);
        Document highestMark = collection.find().sort(new BasicDBObject("Mark", -1)).first();
        if (highestMark != null) {
            int Mark = highestMark.getInteger("Mark");
            FindIterable<Document> highestMarkStudents = collection.find(Filters.eq("Mark", Mark));
            System.out.println("Students with the highest mark:");
            for (Document doc : highestMarkStudents) {
                System.out.println(doc);
            }
        } else {
            System.out.println("No students found");
        }
    }
}
```

- 2nd question

```
import java.util.ArrayList;
import org.bson.Document;
import com.mongodb.BasicDBObject;
import com.mongodb.client.*;

public class Assignment_2_2 {
    public static void main(String[] args) {
```

```
MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017");
MongoDatabase database = mongoClient.getDatabase("Saturday");

MongoCollection<Document> collection = database.getCollection("employee");
collection.drop(); // Drop the collection if it exists to start fresh

Document document1 = new Document("First_Name", "Sai")
   .append("Last_Name", "Kushal")
   .append("salary", 18000)
   .append("age", 30);
Document document2 = new Document("First_Name", "Krishna")
   .append("Last_Name", "Kishore")
   .append("salary", 21000)
   .append("age", 35);
Document document3 = new Document("First_Name", "Feroz")
   .append("Last_Name", "shaik")
   .append("salary", 17000)
   .append("age", 42);
Document document4 = new Document("First_Name", "karthik")
   .append("Last_Name", "ram")
   .append("salary", 22000)
   .append("age", 37);
Document document5 = new Document("First_Name", "suneeth")
      .append("Last_Name", "reddy")
      .append("salary", 38000)
      .append("age", 44);
Document document6 = new Document("First_Name", "lalith")
      .append("Last_Name", "sri")
      .append("salary", 29000)
      .append("age", 32);
ArrayList<Document> documents = new ArrayList<>();
```

```
        documents.add(document1);
        documents.add(document2);
        documents.add(document3);
        documents.add(document4);
        documents.add(document5);
        documents.add(document6);
        collection.insertMany(documents);
        FindIterable<Document> sortedsalary = collection.find().sort(new BasicDBObject("salary", 1));
        Document lowestSalary = null;
        for (Document doc : sortedsalary) {
            int age = doc.getInteger("age");
            if (age >= 30 && age <= 40) {
                lowestSalary = doc;
                break;
            }
        }
        if (lowestSalary != null) {
            System.out.println("Employee with the lowest salary in the age range 30 to 40:");
            System.out.println(lowestSalary.toJson());
        } else {
            System.out.println("No employees found");
        }
    }
}
```