Ws46ExamReview

bad packages

1   ACALO CRUZ CRISTIAN JOSE
DB: mongodb+srv://duck:duck@cluster0.14yiBd.mongodb.net/test
Inspector: Yaranga Leonardo
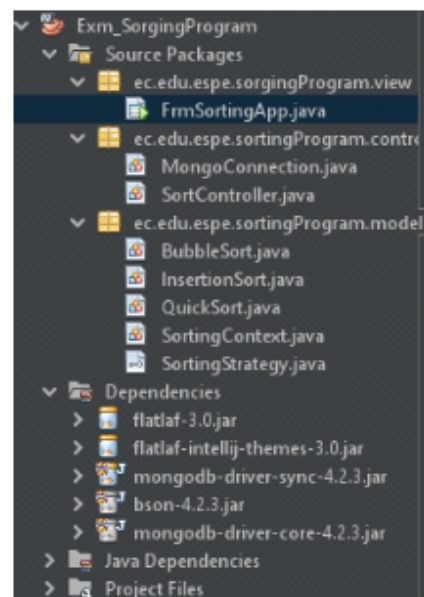
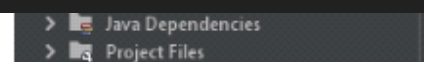RUBRIC

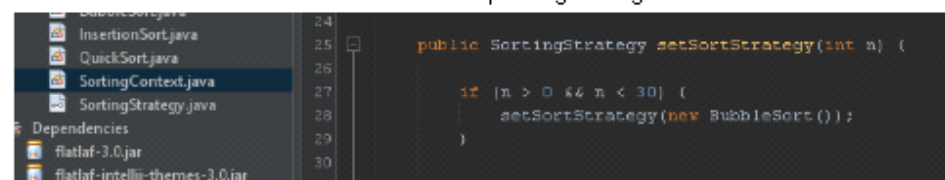|		
Strategy pattern:.	6/10	ExamApp, SortingContext, SortingStrategy, Bubble, range 1-3, 4-7, 8 … -**3**
		SortingStrategy -> abstract class or interface
		sort abstract method
		at least two methods per algorithm -1
		QuickSort, Insertion Sort-**1**
GUI/table:............	0/10	
MongoDB Collection:...	10/10	MongoDb Collection 5 (structure), 5 (update with execution)
working algorithms:.....	9/10	message showing ordered array (No message-**1**), strategy -3)
Clean code/principles:.	7/10	MVC(-**3**)

Total: 32
Evidence

Bad packages. Algorithms in model

Bad packages. Algorithms in model

Bad packages. Algorithms in model

```
24
25      public SortingStrategy setSortStrategy(int n) {
26
27          if (n > 0 && n < 30) {
28              setSortStrategy(new BubbleSort());
29          }
30
31          if (n >= 30 && n < 100) {
32              setSortStrategy(new InsertionSort());
33          }
34
35          if (n >= 100) {
36              setSortStrategy(new QuickSort());
37          }
38
39          return getSortStrategy();
```

Java Dependencies
Project Files
InsertionSort.java
QuickSort.java
SortingContext.java
SortingStrategy.java
Dependencies
flatlaf-3.0.jar
flatlaf-intellij-themes-3.0.jar
mongodb-driver-sync-4.2.3.jar
bson-4.2.3.jar
mongodb-driver-core-4.2.3.jar
Java Dependencies
Project Files
arm
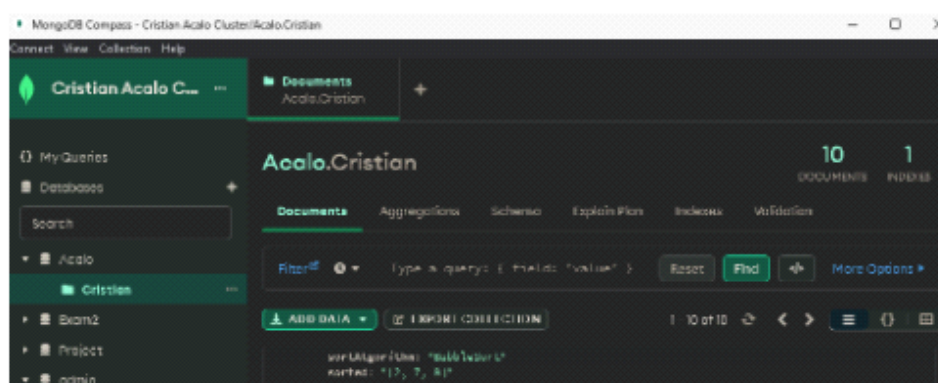ptSnippetPrototype
nvestorProgram
ingletonGUI

```java
package ec.edu.espe.sortingProgram.model;

/**
 *
 * @author Cristian Acalo, Scriptal, DCCO-ESPE
 */
public interface SortingStrategy {
    public  int[] sort(int data[]);
}
```
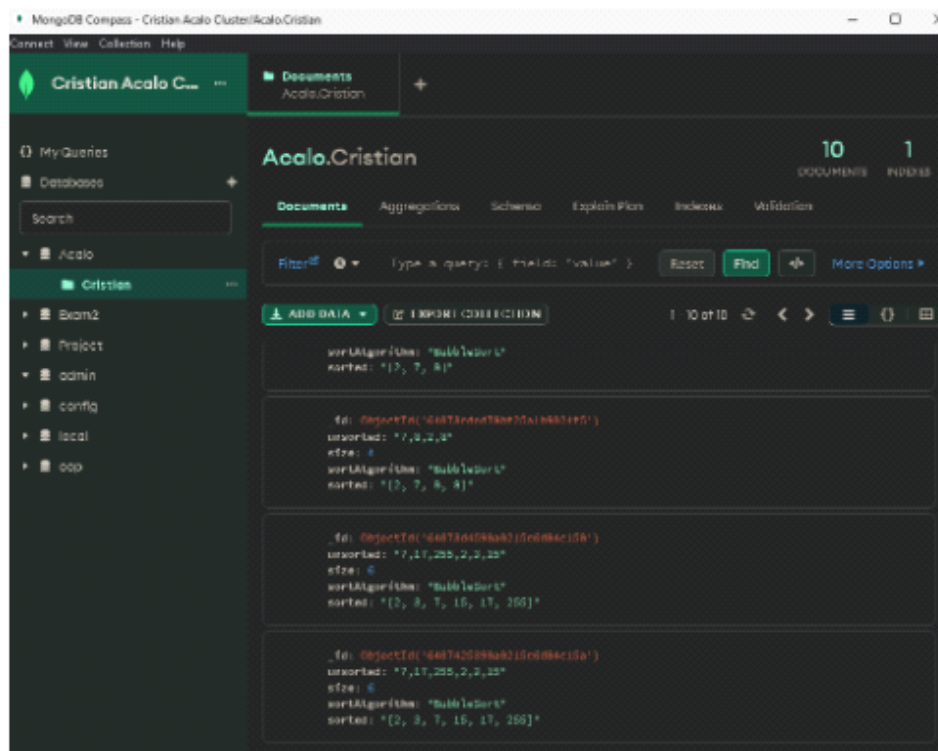
```java
 * @author Cristian Acalo, Scriptal, DCCO-ESPE
 */
public class BubbleSort implements SortingStrategy {
    @Override
    public int[] sort(int data[]) {

        System.out.println(x: "BubbleSort");

        int n = data.length;
        for (int i = 0; i < n - 1; i++) {
            for (int j = 0; j < n - 1 - i; j++) {
                if (data[j] > data[j + 1]) {
                    int temp = data[j];
                    data[j] = data[j + 1];
                    data[j + 1] = temp;
                }
            }
        }

        return data;
```

Acalo.Cristian

Documents   Aggregations   Schema   Explain Plan   Indexes   Validation

```
sortAlgorithm: "BubbleSort"
sorted: "[2, 7, 8]"

_id: ObjectId('66d73cded78af25a1a803dd5')
unsorted: "7,8,2,8"
size: 4
sortAlgorithm: "BubbleSort"
sorted: "[2, 7, 8, 8]"

_id: ObjectId('66d73dd596a02d5c6d6ac1db')
unsorted: "7,17,255,2,3,15"
size: 6
sortAlgorithm: "BubbleSort"
sorted: "[2, 3, 7, 15, 17, 255]"

_id: ObjectId('66d742589ba02d5c6d6ac1da')
unsorted: "7,17,255,2,3,15"
size: 6
sortAlgorithm: "BubbleSort"
sorted: "[2, 3, 7, 15, 17, 255]"
```

SortApp

Sorting Program

Insert the list of numbers to sort separated by a acomma (Ex: 6,5,4,7,8):

> 7,17,255,2,3,15

Your sorted list:

> [2, 3, 7, 15, 17, 255]

Sort   Clean

No sortAlgorithm messague

```java
SortingContext context = new SortingContext();
int[] sortedNumbers = context.sort(data: insertedData);

txtSortedNumbers.setText(s: Arrays.toString(a: sortedNumbers));

String sortStrategy = context.getSortStrategy().getClass().getSimpleName();

ArrayList<Integer> sortArr = new ArrayList<>();

for (int i = 0; i < sortedNumbers.length; i++) {
    sortArr.add(sortedNumbers[i]);
}
```

```java
        sortArr.add(sortedNumbers[i]);
    }

    Document sortDocument = new Document().append(key: "_id", new ObjectId())
            .append(key: "unsorted", value: txtEnterNumbers.getText())
            .append(key: "size", value: insertedData.length)
            .append(key: "sortAlgorithm", value: sortStrategy)
            .append(key: "sorted", value: sortArr.toString());

    addSortCase(sortCase: sortDocument);
```

Maybe this should be a method in a controller class.

```
addSortCase[ sortCase: sortDocument];
```

Maybe this should be a method in a controller class.

41   YARANGA SUQUILLO LEONARDO JAVIER
DB: mongodb+srv://lyaranga:tortilla@espe2210-oopsw7996.77wv341.mongodb.net/test
Inspector: Verdugo Sebastian

RUBRIC
Strategy pattern:.   10/10   ExamApp, SorignContext, SortingStrategy, Bubble,
QuickSort, Insertio Sort
GUI/table:............   0/10 There's no table with the data
MongoDB Collection:...   10/10 Connect successfully
working algorithms:.....   10/10
Clean code/principles:.   10/10

Total Grade: 40/50