

Beginner Java(JD521)

Group Activity 6

0000

MARCH 2

COMPANY NAME

Authored by: Your Name



Logo
Name

Table of Contents

Introduction	3
Program Content	4
Source Code:.....	4
class Apartment:.....	4
class TestApartments:	6
Bibliography & References:.....	10
Appendix	11

Introduction

Program Content

Source Code:

class Apartment:

```
package group.act4.project;

import javax.swing.JOptionPane;

/**
 *
 * @author Albert Michael Ludick(5119)
 */
```

```
/**
 *
 * @author Albert Michael Ludick(5119)
 */
public class Apartment {
    private String apartmentNumber;
    private static int rentAmount;
    private static int numberOfBedrooms;
    private static int numberOfBaths;
    public static int[][] newArray;
```

```
    public Apartment(String apartmentNumber,int rentAmount,int numberOfBedrooms,
        int numberOfBaths)//custom constructor
    {
        this.apartmentNumber=apartmentNumber;
        Apartment.rentAmount=rentAmount;
        Apartment.numberOfBedrooms=numberOfBedrooms;
        Apartment.numberOfBaths=numberOfBaths;
    }
```

```
    public Apartment() {}//default constructor
    /**
     *
     * @return apartmentNumber
     */
```

```
/**
 *
 * @return apartmentNumber
 */
public String getApartmentNumber() { //Read-Only Method(Property)
    return apartmentNumber;
}

/**
 *
 * @return rentAmount
 */
public static int getRentAmount() { //Read-Only Method(Property)
    return rentAmount;
}

/**
```

```
/**
 *
 * @return numberOfBedrooms
 */
    public static int getNumberOfBedrooms() { //Read-Only Method(Property)
        return numberOfBedrooms;
    }

/**
 *
 * @return numberOfBaths
 */
    public static int getNumberOfBaths() { //Read-Only Method(Property)
        return numberOfBaths;
    }
}
```

```

/**
 * This method Takes data from this class apartments and instatiated
 * Fields and adds the fields to an array
 * @param j is an object of a class
 */
public static void ToString(Object j){
    try{
        newArray = new int[5][3];
        for(int i = 0; i < 5;i++){
            newArray[i][0]=getRentAmount();
            newArray[i][1]=getNumberOfBedrooms();
            newArray[i][2]=getNumberOfBaths();
        }
    }
    catch (ArrayIndexOutOfBoundsException ex){
        JOptionPane.showMessageDialog(null,ex,"Error Message"
        ,JOptionPane.ERROR_MESSAGE);
        throw ex;
    }
}

```

class TestApartments:

```

package group.act4.project;
import java.awt.HeadlessException;
import javax.swing.JOptionPane;
/**
 *
 * @author Albert Michael Ludick(5119)
 */

```

```

public class TestApartments {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        //making 5 Object of class apartment
        Apartment Apartment1 = new Apartment("0",4000,2,1);
        Apartment Apartment2 = new Apartment("1",7500,3,2);
        Apartment Apartment3 = new Apartment("2",6000,2,2);
        Apartment Apartment4 = new Apartment("3",9000,3,3);
        Apartment Apartment5 = new Apartment("4",10000,4,2);
        //Adding the Apartment Data to a 2D Array
        Apartment.ToString(Apartment1);
        Apartment.ToString(Apartment2);
        Apartment.ToString(Apartment3);
        Apartment.ToString(Apartment4);
        Apartment.ToString(Apartment5);
    }
}

```

```

try{
    // 3 user statements that ask user for the room of his dreams
    int rentAmount =Integer.parseInt
    (JOptionPane.showInputDialog("plz enter the minimum amount of "
        + "rent you "
        + "wish to pay"));

    int numberOfBedrooms =Integer.parseInt
    (JOptionPane.showInputDialog("plz enter the minimum number of "
        + "bedrooms "
        + "required"));

    int numberOfBaths =Integer.parseInt
    (JOptionPane.showInputDialog("plz enter the minimum number of "
        + "baths "
        + "required"));

    // a method call to compare the user data with the available rooms
    //and to say which is available
    LookForUserData(numberOfBedrooms,numberOfBaths,rentAmount);
}
catch(HeadlessException | NumberFormatException ex){
    JOptionPane.showMessageDialog(null,ex,"Error Message"
        ,JOptionPane.ERROR_MESSAGE);
    throw ex;
}
}

```

```

/**
 * this method is to check user data and compare with the available rooms
 * and to notify user of the rooms he can choose from.
 * @param numberOfBedrooms
 * @param numberOfBaths
 * @param rentAmount
 */
public static void LookForUserData(int numberOfBedrooms,int numberOfBaths,
    int rentAmount)
{
    try{
        for(int i = 0; i < 5;i++){
            if(rentAmount <= Apartment.newArray[i][0]
                && numberOfBedrooms <= Apartment.newArray[i][1]
                && numberOfBaths <= Apartment.newArray[i][2])
            {
                JOptionPane.showMessageDialog(null,
                    "the availble room is :"+" "+i+" "
                    +"rent amount :"+" "+Apartment.newArray[i][0]+" "
                    +"numofbedrooms :"+" "+Apartment.newArray[i][1]+" "
                    +"numofbaths :"+" "+ Apartment.newArray[i][2] );
            }
            else{
                JOptionPane.showMessageDialog(null,"There is no Rooms "

```

```

public static void LookForUserData(int numberOfBedrooms,int numberOfBaths,
    int rentAmount)
{
    try{
        for(int i = 0; i < 5;i++){
            if(rentAmount <= Apartment.newArray[i][0]
                && numberOfBedrooms <= Apartment.newArray[i][1]
                && numberOfBaths <= Apartment.newArray[i][2])
            {
                JOptionPane.showMessageDialog(null,
                    "the availble room is :"+" "+i+" "
                    +"rent amount :"+" "+Apartment.newArray[i][0]+" "
                    +"numofbedrooms :"+" "+Apartment.newArray[i][1]+" "
                    +"numofbaths :"+" "+ Apartment.newArray[i][2] );
            }
            else{
                JOptionPane.showMessageDialog(null,"There is no Rooms "
                    + "Matching your "
                    + "criteria that you are looking for"
                    + "Plz try again");
                break;
            }
        }
    }
}

```

```
    }  
    catch (ArrayIndexOutOfBoundsException ex) {  
        JOptionPane.showMessageDialog(null, ex, "Error Message"  
            , JOptionPane.ERROR_MESSAGE);  
        throw ex;  
    }  
}  
}
```

Bibliography & References:

Websites:

GeeksforGeek : <https://www.geeksforgeeks.org/>

And Javapoint : <https://www.javatpoint.com/>

Wiki How to do anything : <https://www.wikihow.com/Calculate-Weighted-Average>

Books:

Java_ A Beginner's Guide, Eighth Edition

Appendix

No Extra files or media.

Except for the project itself