KK14203(SEKSYEN 1-HC00) – OBJECT ORIENTED PROGRAMMING

OOP GROUP PROJECT: KINDERGARTEN REGISTRATION

Lecturer: DR. MOHD SHAMRIE SAININ

Group Members:

No.	Name	Matric No.
1	DANIEL DEAN PHILLIP	BI19110192
2	MCLAREN W.YUSOF	BI19110151
3	MOHD IKMAL BIN INDAN	BI19160360
4	JACINTHA LUCIA RUMPUNGAN	BI19110109
5	JACQUELIN RINYA ANAK ENJANG	BI19110144

Project's Code:

```
1    //This Program store information of a child age between 4 - 7 years old
2    //It also will store the child's parent information
3    //This program can at least store up to 20 record
4    import java.util.Scanner;
5    import java.text.ParseException;
6    import java.time.LocalDate;
7    import java.time.Period;
8
9    public class KindergatenRegistrationSystem {
10       public static void main(String args[]) throws ParseException{
11          int registerNo = 1, counter =  1, age = 0, userChoice = 0, userRegisterNo = 0;
12          String userInput;
13          char ch;
14          Scanner scan = new Scanner(System.in);
15          Child Record1[] = new Child[22];//can hold at least 20 record
16          Father Record2[] = new Father[22];
17          Mother Record3[] = new Mother[22];
18          System.out.println("Do you want to register? (yes / no)");
19          userInput = scan.next();
20          ch = userInput.charAt(0);
21          while(ch == 'y' || ch == 'Y')
22          {
23             if(counter == 21)
24             {
25                System.out.println("Sorry all Class is full");
26                break;
27             }
28             System.out.println("Your registration number is: " + registerNo );//important for checking record of registration
29             Record1[counter] = new Child();//can hold at least 20 record
30             Record2[counter] = new Father();
31             Record3[counter] = new Mother();
32             Record1[counter].InputChildData();
33             age = Record1[counter].child_age;
34             if(age < 4 || age > 7)
35             {
36                System.out.println("Sorry we only accepting child age between 4 - 7 years old");
```

```
37                Record2[counter] = null;
38                Record3[counter] = null;
39                Record1[counter] = null;
40                break;
41            }
42            Record2[counter].InputFatherData();
43            Record3[counter].InputMotherData();
44            System.out.println("Registered Successfull!");
45            System.out.println("What do you want to do?
[1.Register, 2.Delete Registration, 3.Exit]");
46            userChoice = scan.nextInt();
47            if(userChoice == 1)
48            {
49                System.out.println("Do you want to register
someone else? (yes / no)");
50                userInput = scan.next();
51                ch = userInput.charAt(0);
52            }
53            else if(userChoice == 2)
54            {
55                System.out.print("Enter your register no: ");
56                userRegisterNo = scan.nextInt();
57                Record2[userRegisterNo] = null;
58                Record3[userRegisterNo] = null;
59                Record1[userRegisterNo] = null;
60                ch = 'n';
61            }
62            else
63            {
64                ch = 'n';
65            }
66            registerNo++;
67            counter++;
68        }
69
70        if((age <= 7 && age >= 4) || registerNo > 1)
71        {
72            String respond; int inputNumber = 0; boolean flag =
true, loop = true; char ch2, exit;
73            System.out.println("Show registered record? (yes /
no)");
74            respond = scan.next();
75            ch2 = respond.charAt(0);
76            if(ch2 == 'y' || ch2 == 'Y')
77            {
```

```
78                loop = true;
79                while(loop)
80                {
81                    while(flag)
82                    {
83                        System.out.println("Enter your registration
number(between 1 - 20): ");
84                        inputNumber = scan.nextInt();
85                        if(inputNumber != 0 && inputNumber <= 20)
86                            flag = false;
87                    }
88                    if(Record2[inputNumber] != null &&
Record3[inputNumber] != null && Record1[inputNumber] != null)
89                    {
90
                      System.out.println("==========================
=========");
91                        Record2[inputNumber].ShowFatherData();
92
                      System.out.println("==========================
=========");
93                        Record3[inputNumber].ShowMotherData();
94
                      System.out.println("==========================
=========");
95                        Record1[inputNumber].ShowChildData();
96
                      System.out.println("==========================
=========");
97                    }
98                    else
99                    {
100                       System.out.println("No Registration");
101                   }
102
103                   System.out.println("Show another registered
record? (yes / no)");
104                   respond = scan.next();
105                   exit = respond.charAt(0);
106                   flag = true;
107                   if(exit == 'n' || exit == 'N')
108                       loop = false;
109               }
110           }
111       }
112       else
```

```
113     {
114         System.out.print("");
115     }
116 }
117 }
118
119 class Find_Age{//Determine the child age
120     public static int setAge(int year, int month, int
date)throws ParseException{
121         LocalDate l1 = LocalDate.of(year, month, date);
122         LocalDate now1 = LocalDate.now();
123         Period different1 = Period.between(l1, now1);
124         return different1.getYears();
125     }
126 }
127
128 class Child {// Stored children Information
129     private  String childName;
130     private  String childID;
131     private  String childbirth;
132     private  String childgender;
133     int child_age;
134
135     public void InputChildData()throws ParseException{
136
137         int age;
138         int year;
139         int month;
140         int date;
141         Scanner scan = new Scanner(System.in);
142         System.out.println("Child name:");
143         childName = scan.nextLine();
144         System.out.println("Child ID:");
145         childID = scan.nextLine();
146         System.out.println("Gender");
147         childgender = scan.nextLine();
148         System.out.print("Date of Birth(year):");
149         year = scan.nextInt();
150         System.out.print("Date of Birth(month):");
151         month = scan.nextInt();
152         System.out.print("Date of Birth(Day):");
153         date = scan.nextInt();
154         child_age = Find_Age.setAge(year,month,date);
155         String str_y = String.valueOf(year); String str_m =
String.valueOf(month); String str_d = String.valueOf(date);
```

```
156         childbirth = (str_y + "/" + str_m + "/" + str_d);
157     }
158 
159 
160     public  void ShowChildData(){
161     if(child_age >= 4 && child_age <= 5)
162         {
163         System.out.println("Children age between 4 - 5 years
old");
164         System.out.println("Teacher: Maclaren Yusof");
165 
166         System.out.println("Child name: " + childName);
167         System.out.println("Child ID: " + childID);
168         System.out.println("Child Age: " + child_age);
169         System.out.println("Gender: " + childgender);
170         System.out.println("Date of Birth(yyyy/MM/dd): " +
childbirth);
171          }
172     else if (child_age >= 6 && child_age <= 7)
173         {
174         System.out.println("Children age between 6 - 7 years
old");
175         System.out.println("Teacher: Ikmal");
176         System.out.println("Child name: " + childName);
177         System.out.println("Child ID: " + childID);
178         System.out.println("Gender: " + childgender);
179         System.out.println("Date of Birth(yyyy/MM/dd): " +
childbirth);
180          }
181     }
182 }
183 
184 class Mother{//Stored parent information
185     private  String M_name;
186     private  String M_contNo;
187     private  String M_address;
188 
189     public void InputMotherData(){
190 
191     Scanner scan = new Scanner(System.in);
192     System.out.println("Mother's name:");
193     M_name= scan.nextLine();
194     System.out.println("Contact Number:");
195     M_contNo = scan.nextLine();
196     System.out.println("Address:");
```

```
197     M_address = scan.nextLine();
198     }
199
200     public void ShowMotherData(){
201     System.out.println("Mother Name: " + M_name);
202     System.out.println("Contact Number: " + M_contNo);
203     System.out.println("Address: " + M_address);
204     }
205
206 }
207
208 class Father {//Stored parent information
209     private String F_name;
210     private String F_contNo;
211     private String F_address;
212
213     public void InputFatherData(){
214
215     Scanner scan = new Scanner(System.in);
216     System.out.println("Father's name:");
217     F_name = scan.nextLine();
218     System.out.println("Contact Number:");
219     F_contNo = scan.nextLine();
220     System.out.println("Address:");
221     F_address = scan.nextLine();
222     }
223
224     public  void ShowFatherData(){
225     System.out.println("Father Name: " + F_name);
226     System.out.println("Contact Number: " + F_contNo);
227     System.out.println("Address: " + F_address);
228     }
229 }
```