JAVA ASSIGNMENT ASSIGNMENT-2 DATE:25-03-23

PROBLEM STATEMENT : classes, array of objects, instance members, constructors in java.

PROBLEM PART : Write a menu-driven Java Program to study the concepts of classes, array of objects, instance members, constructors in java. Assignment description: Create a Student class describing attributes of a student like prn, name, DoB, marks etc. Create an array of objects of Student class and perform operations like:

Add students, Display, Search (by prn, by name, by position), Update/Edit and Delete.
-The program should contain different java files.
-Each operation should be a separate function.
-Must use Array or ArrayList to create an array of objects.
- Program should contain at top of Main file in comments : Name, PRN, Batch
- Program should follow all the coding guidelines.
- Program should contain comments for a particular block of logic.
- It is recommended to upload a Program on your GitHub account.
- Your Student_Data_Array_of_Objects repository on GitHub should contain a README file describing all functions or methods or definitions.

CODE:

```
/*
Problem Statement : Write a menu-driven Java Program to study the concepts of classes, array
                    of objects, instance members, constructors in java. Assignment description:
                    Create a Student class describing attributes of a student like prn, name, DoB,
                    marks etc. Create an array of objects of Student class and perform operations like:

                    Add students, Display, Search (by prn, by name, by position), Update/Edit and
Delete.
                    -The program should contain different java files.
                    -Each operation should be a separate function.
                    -Must use Array or ArrayList to create an array of objects.
                    - Program should contain at top of Main file in comments : Name, PRN, Batch
                    - Program should follow all the coding guidelines.
                    - Program should contain comments for a particular block of logic.
                    - It is recommended to upload a Program on your GitHub account.
                    - Your Student_Data_Array_of_Objects repository on GitHub should contain a
README file describing all functions or methods or definitions.
```

Name - Nisarg Patel
PRN - 21070126060
Batch - AIML A3

LAB ASSIGNMENT - 3
*/

```
package Assignment_3;
import java.util.*;
public class ListArray
{
   public static void main(String[] args) // main function
   {
      functions student_obj = new functions(); // create an instance of the functions class

      // menu for add, display, search, update, delete
      while(true) // loop indefinitely until user decides to exit
      {
         // display the menu options for the user
         System.out.println("Student Database");
         System.out.println("----------------------------------------");
         System.out.println("Menu");
         System.out.println("1. Add student details");
         System.out.println("2. Display all");
         System.out.println("3. Search student");
         System.out.println("4. Update Details");
         System.out.println("5. Delete record");
         System.out.println("6. Exit");
         System.out.println();
         System.out.print("Select the operation to modify database: ");

         Scanner sc = new Scanner(System.in); // create a scanner to read user input
         int ch = sc.nextInt(); // read user's choice of operation

         switch(ch){ // perform the operation based on user's choice
            case 0:
               System.out.println("Exiting..."); // exit the program
               break;
            case 1:
               student_obj.addStudent(); // add a new student record
               break;
            case 2:
               student_obj.display(); // display all student records
               break;
```

```
            case 3:
                student_obj.search(); // search for a specific student record
                break;
            case 4:
                student_obj.update(); // update an existing student record
                break;
            case 5:
                student_obj.delete(); // delete an existing student record
                break;
            case 6:
                System.exit(0);
            default:
                System.out.println("Invalid ch"); // display an error message for an invalid choice
        }
        if (ch == 0){ // if user chooses to exit, break out of the while loop
            break;
        }
    }
  }
}


class Student {                                                    // student class
   private int prn_num;
   private String st_name;
   private String st_dob;
   private int st_marks;

   public Student(int prn_num, String st_name, String st_dob, int st_marks)                    //
constructor
   {
      this.prn_num = prn_num;
      this.st_name = st_name;
      this.st_dob = st_dob;
      this.st_marks = st_marks;
   }

   public int getprn_num() {
      return prn_num;
   }

   public void setprn_num(int prn_num) {
      this.prn_num = prn_num;
   }
```

```
    public String getst_name() {
        return st_name;
    }

    public void setst_name(String st_name) {
        this.st_name = st_name;
    }

    public String getDob() {
        return st_dob;
    }

    public void setDob(String st_dob) {
        this.st_dob = st_dob;
    }

    public int getst_marks() {
        return st_marks;
    }

    public void setst_marks(int st_marks) {
        this.st_marks = st_marks;
    }
}

class functions
{
    ArrayList<Student> studentList = new ArrayList<>();


    public void printStudent(int i)
    {
        System.out.print("st_name: " + studentList.get(i).getst_name()+" | ");
        System.out.print("prn_num: " + studentList.get(i).getprn_num()+" | ");
        System.out.print("DOB: "+ studentList.get(i).getDob()+" | ");
        System.out.print("st_marks: " + studentList.get(i).getst_marks()+"  | \n\n");
    }

    public void addStudent()                                    // add student function
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of students to be added: ");
        int n = sc.nextInt();
```

```
        for (int i = 0; i < n; i++) {
            System.out.println("Enter the details of the student in the following format: prn_num,
st_name, Date of Birth " +
                    "(dd/mm/yyyy), st_marks");

            String details = sc.next();

            String[] details_array = details.split(",");
            int prn_num = Integer.parseInt(details_array[0]);

            String st_name = details_array[1];

            String dobString = details_array[2];

            int st_marks = Integer.parseInt(details_array[3]);

            Student student1 = new Student(prn_num, st_name, dobString, st_marks);

            ArrayList<Integer> prn_numList = new ArrayList<>();
            for (int j = 0; j < studentList.size(); j++) {
                prn_numList.add(studentList.get(j).getprn_num());
            }

            if (prn_numList.contains(prn_num)) {
                System.out.println("Student with prn_num " + prn_num + " already exists");
            } else {
                studentList.add(student1);
            }

        }
    }

    public void display()
    {
        for (int i = 0; i < studentList.size(); i++)
        {
            printStudent(i);
        }
    }

    public void search(){

        System.out.println("Select the search criteria: ");
```

```
        System.out.println("1. prn_num");
        System.out.println("2. st_name");
        System.out.println("3. Position");

        Scanner sc = new Scanner(System.in);
        int ch = sc.nextInt();

        switch (ch)                                                    // search function
        {
            case 1 ->                                                  // search by prn_num
            {

                System.out.println("Enter the prn_num to be searched: ");
                int prn_num = sc.nextInt();
                for (int i = 0; i < studentList.size(); i++) {
                    if (studentList.get(i).getprn_num() == prn_num) {
                        printStudent(i);
                    }
                }
            }
            case 2 ->
            {                                                          // search by st_name
                System.out.println("Enter the st_name to be searched: ");
                String st_name = sc.next();
                for (int i = 0; i < studentList.size(); i++) {
                    if (Objects.equals(studentList.get(i).getst_name(), st_name)) {
                        printStudent(i);
                    }
                }
            }
            case 3 -> { //position
                System.out.println("Enter the Position to be searched: ");                  // search by
position
                int position = sc.nextInt();
                for (int i = 0; i < studentList.size(); i++)
                {
                    if (i == position)
                    {
                        printStudent(i);
                    }
                }
            }
            default -> System.out.println("Invalid ch");
        }
```

```
    }

    public void update()                                        // update function
    {
        System.out.println("Enter the prn_num of the student to be updated: ");
        Scanner sc = new Scanner(System.in);
        int prn_num = sc.nextInt();

        for (int i = 0; i < studentList.size(); i++) {
            if (studentList.get(i).getprn_num() == prn_num) {
                System.out.println("Enter the details of the student in the following format: prn_num,
st_name, " +
                        "Date of Birth (dd/mm/yyyy), st_marks");
                String details = sc.next();

                String[] details_array = details.split(",");
                int prn_num_new = Integer.parseInt(details_array[0]);

                String st_name_new = details_array[1];

                String dob_string_new = details_array[2];

                int st_marks_new = Integer.parseInt(details_array[3]);

                Student new_student = new Student(prn_num_new, st_name_new, dob_string_new,
st_marks_new);
                studentList.set(i, new_student);
                printStudent(i);
            }
        }
    }

    public void delete()
    {
        System.out.println("Enter the prn_num of the student to be deleted: ");
        Scanner sc = new Scanner(System.in);
        int prn_num = sc.nextInt();

        for (int i = 0; i < studentList.size(); i++)
        {
            if (studentList.get(i).getprn_num() == prn_num)
            {
```

```
            System.out.println("Student st_named:"+ studentList.get(i).getst_name() + " deleted
successfully");
            studentList.remove(i);
        }
    }
  }

}
```

OUTPUT :

```
Menu
1. Add student details
2. Display all
3. Search student
4. Update Details
5. Delete record
6. Exit

Select the operation to modify database: 1
Enter the number of students to be added:
2
Enter the details of the student in the following format: prn_num, st_name, Date of Birth (dd/mm/yyyy), st_marks
60,Nisarg,10/07/2003,98
Enter the details of the student in the following format: prn_num, st_name, Date of Birth (dd/mm/yyyy), st_marks
39,Jainil,21/05/2003,99
Student Database
----------------------------------------
```

```
Menu
1. Add student details
2. Display all
3. Search student
4. Update Details
5. Delete record
6. Exit

Select the operation to modify database: 2
st_name: Nisarg | prn_num: 60 | DOB: 10/07/2003 | st_marks: 98  |

st_name: Jainil | prn_num: 39 | DOB: 21/05/2003 | st_marks: 99  |

Student Database
----------------------------------------
```

```
Menu
1. Add student details
2. Display all
3. Search student
4. Update Details
5. Delete record
6. Exit

Select the operation to modify database: 3
Select the search criteria:
1. prn_num
2. st_name
3. Position
1
Enter the prn_num to be searched:
60
st_name: Nisarg | prn_num: 60 | DOB: 10/07/2003 | st_marks: 98  |
```

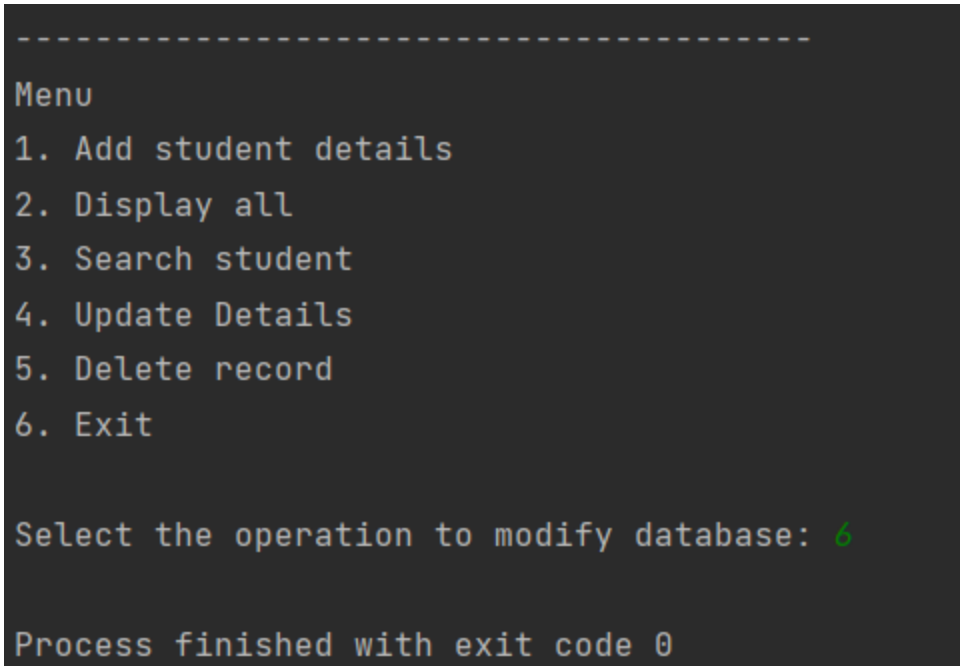
```
Menu
1. Add student details
2. Display all
3. Search student
4. Update Details
5. Delete record
6. Exit

Select the operation to modify database: 4
Enter the prn_num of the student to be updated:
39
Enter the details of the student in the following format: prn_num, st_name, Date of Birth (dd/mm/yyyy), st_marks
39,Jainl,21/05/2003,98
st_name: Jainl | prn_num: 39 | DOB: 21/05/2003 | st_marks: 98  |

Student Database
----------------------------------------
```

```
Menu
1. Add student details
2. Display all
3. Search student
4. Update Details
5. Delete record
6. Exit

Select the operation to modify database: 5
Enter the prn_num of the student to be deleted:
39
Student st_named:Jainl deleted successfully
Student Database
----------------------------------------
```



GitHub Repository: https://github.com/SnakeEyes1308/Java-Assignment-