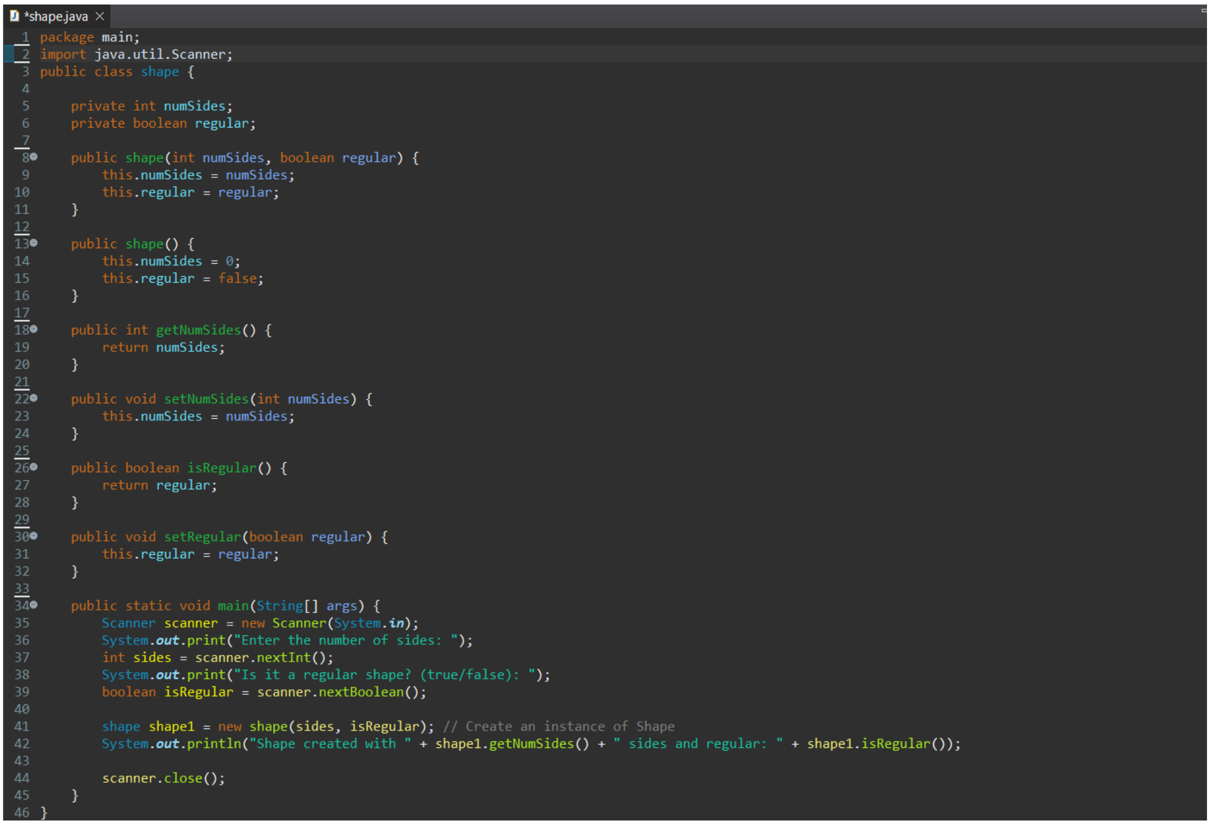
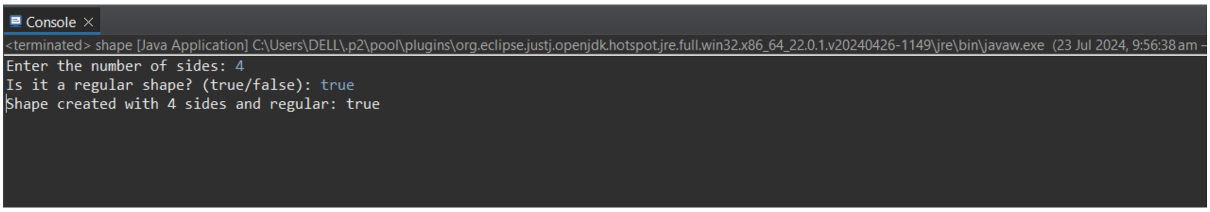
7.1

1. Create a simple class Shape that will represent a 2-dimensional shape with line segments for edges. It should have the following instance variables: numSides (int), regular (boolean). Create at least two constructors and getter and setter methods.



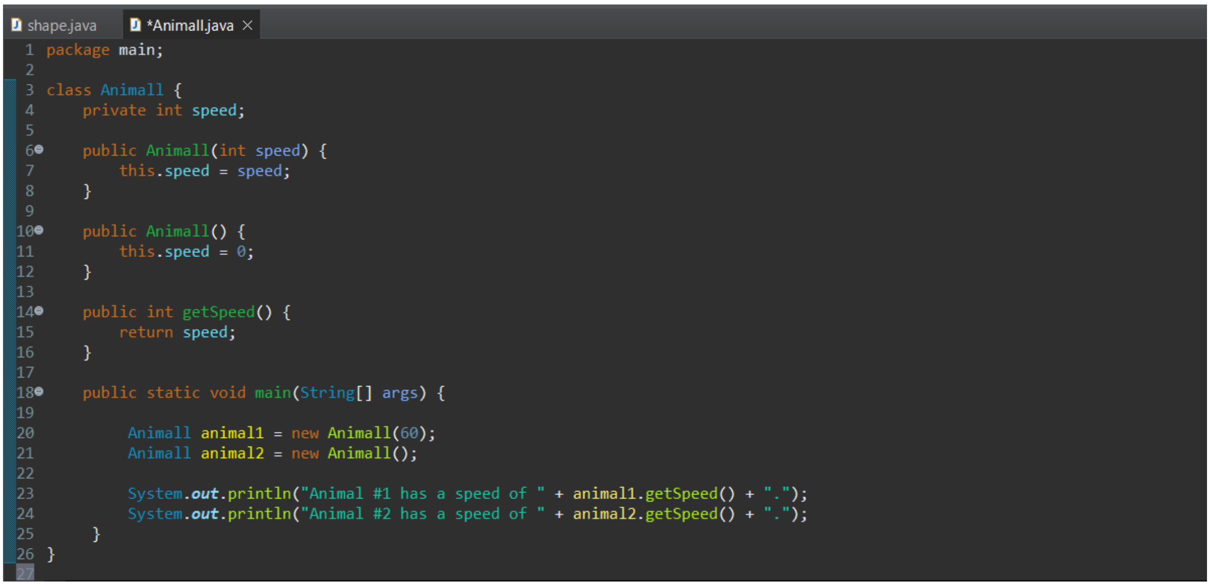
```java
package main;
import java.util.Scanner;
public class shape {

    private int numSides;
    private boolean regular;

    public shape(int numSides, boolean regular) {
        this.numSides = numSides;
        this.regular = regular;
    }

    public shape() {
        this.numSides = 0;
        this.regular = false;
    }

    public int getNumSides() {
        return numSides;
    }

    public void setNumSides(int numSides) {
        this.numSides = numSides;
    }

    public boolean isRegular() {
        return regular;
    }

    public void setRegular(boolean regular) {
        this.regular = regular;
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of sides: ");
        int sides = scanner.nextInt();
        System.out.print("Is it a regular shape? (true/false): ");
        boolean isRegular = scanner.nextBoolean();

        shape shape1 = new shape(sides, isRegular); // Create an instance of Shape
        System.out.println("Shape created with " + shape1.getNumSides() + " sides and regular: " + shape1.isRegular());

        scanner.close();
    }
}
```

Output:



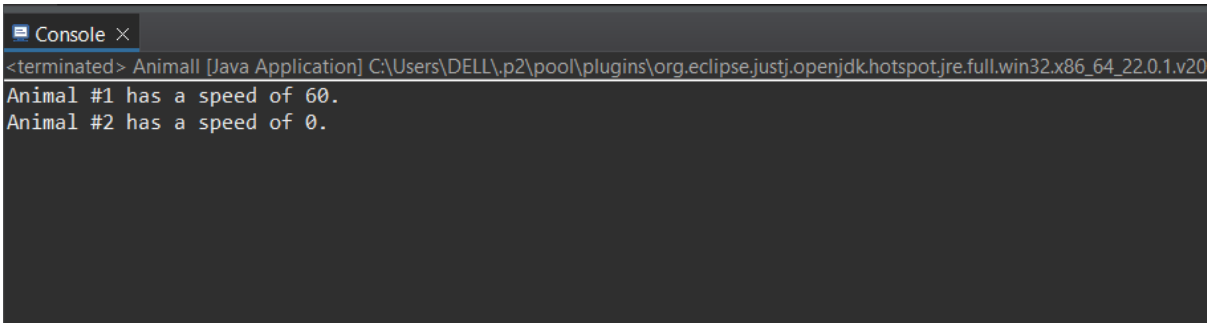
```
<terminated> shape [Java Application] C:\Users\DELL\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.1.v20240426-1149\jre\bin\javaw.exe  (23 Jul 2024, 9:56:38 am -
Enter the number of sides: 4
Is it a regular shape? (true/false): true
Shape created with 4 sides and regular: true
```

3. Write code to create two instances of the Animal class template listed in problem #2. Be sure to use each of the two constructors provided. Then add Java code that will print the following: a. Animal #1 has a speed of ___. b. Animal #2 has a speed of ___. Be sure that the blanks are automatically filled in with the actual speeds. Use the methods provided to access the speeds.



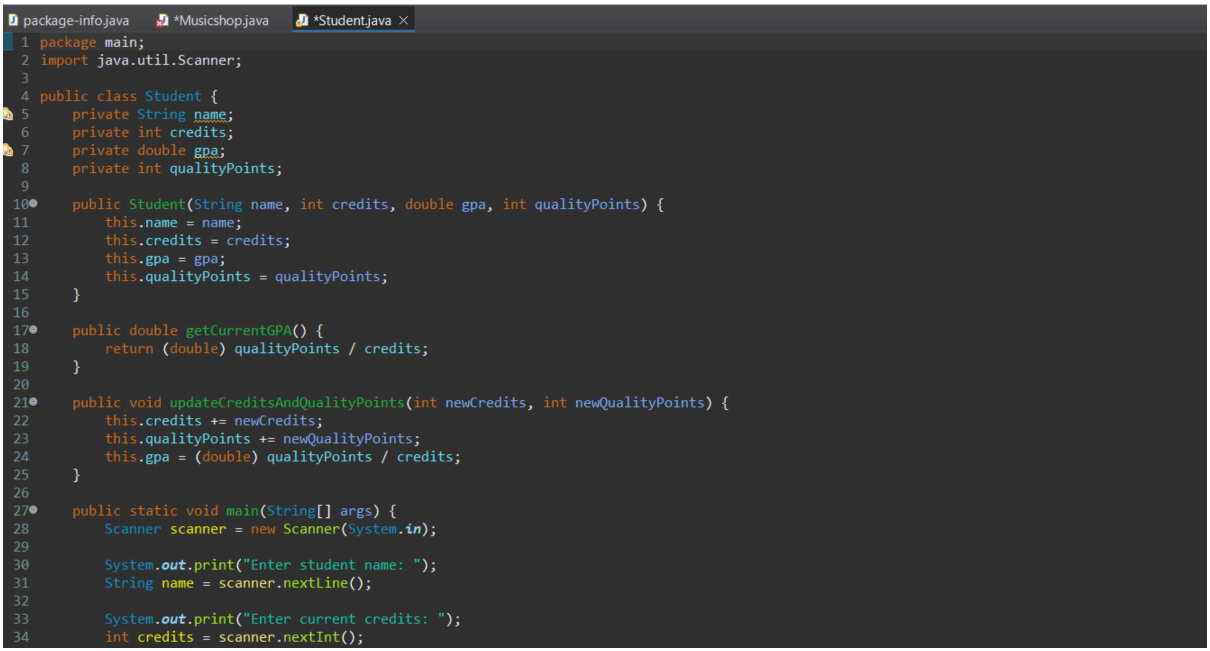
```java
package main;

class Animall {
    private int speed;

    public Animall(int speed) {
        this.speed = speed;
    }

    public Animall() {
        this.speed = 0;
    }

    public int getSpeed() {
        return speed;
    }

    public static void main(String[] args) {

         Animall animal1 = new Animall(60);
         Animall animal2 = new Animall();

         System.out.println("Animal #1 has a speed of " + animal1.getSpeed() + ".");
         System.out.println("Animal #2 has a speed of " + animal2.getSpeed() + ".");
     }
}
```

Output:

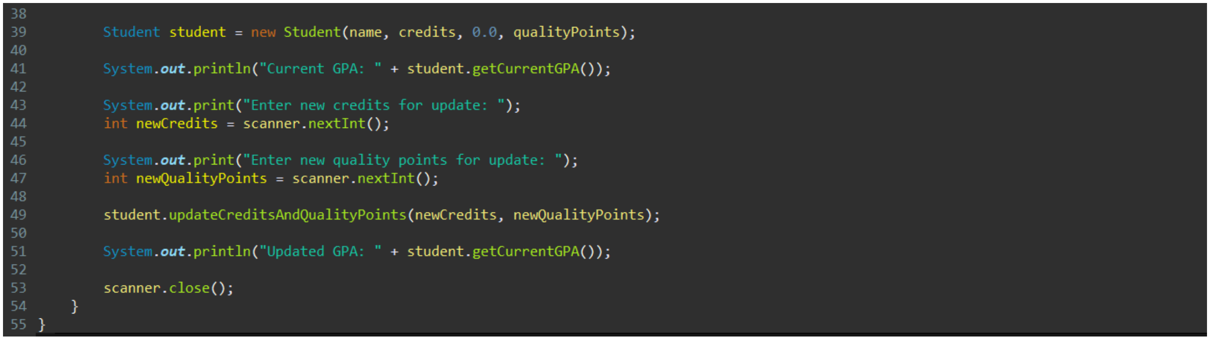


```
<terminated> Animall [Java Application] C:\Users\DELL\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.1.v20
Animal #1 has a speed of 60.
Animal #2 has a speed of 0.
```

4. Write a class Student. It should have the following instance variables for the name, credits, grade point average (GPA), and quality Points. Create a constructor method. Create two other methods as follows: a. A method that will return the current grade point average which will be the quality points divided by the credits. b. A method that will take in the credits for a class or semester along with the quality points. It should update the credits, the quality points, and the GPA.

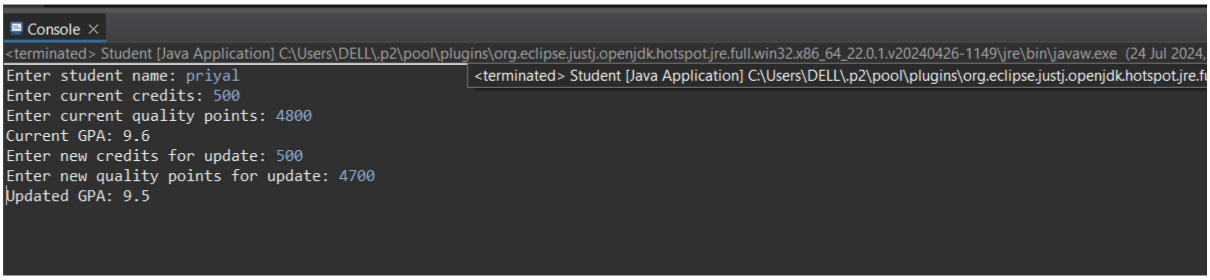


```java
package main;
import java.util.Scanner;

public class Student {
    private String name;
    private int credits;
    private double gpa;
    private int qualityPoints;

    public Student(String name, int credits, double gpa, int qualityPoints) {
        this.name = name;
        this.credits = credits;
        this.gpa = gpa;
        this.qualityPoints = qualityPoints;
    }

    public double getCurrentGPA() {
        return (double) qualityPoints / credits;
    }

    public void updateCreditsAndQualityPoints(int newCredits, int newQualityPoints) {
        this.credits += newCredits;
        this.qualityPoints += newQualityPoints;
        this.gpa = (double) qualityPoints / credits;
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter student name: ");
        String name = scanner.nextLine();

        System.out.print("Enter current credits: ");
        int credits = scanner.nextInt();
```



```java

        Student student = new Student(name, credits, 0.0, qualityPoints);

        System.out.println("Current GPA: " + student.getCurrentGPA());

        System.out.print("Enter new credits for update: ");
        int newCredits = scanner.nextInt();

        System.out.print("Enter new quality points for update: ");
        int newQualityPoints = scanner.nextInt();

        student.updateCreditsAndQualityPoints(newCredits, newQualityPoints);

        System.out.println("Updated GPA: " + student.getCurrentGPA());

        scanner.close();
    }
}
```

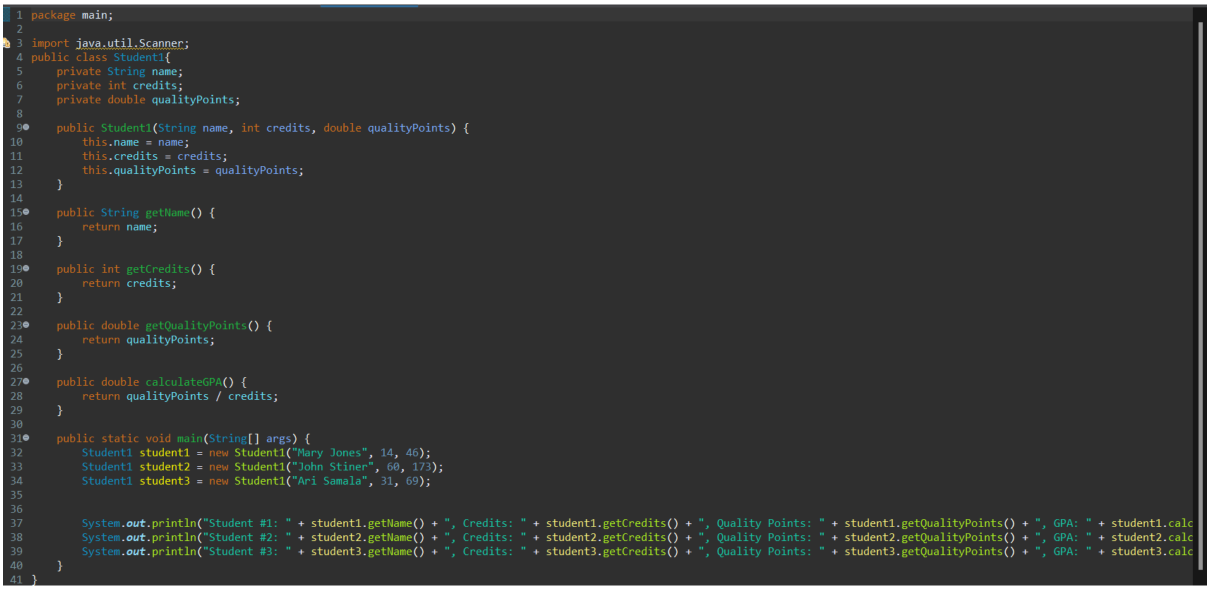
Output:



```
<terminated> Student [Java Application] C:\Users\DELL\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.1.v20240426-1149\jre\bin\javaw.exe  (24 Jul 2024,
Enter student name: priyal
Enter current credits: 500
Enter current quality points: 4800
Current GPA: 9.6
Enter new credits for update: 500
Enter new quality points for update: 4700
Updated GPA: 9.5
```

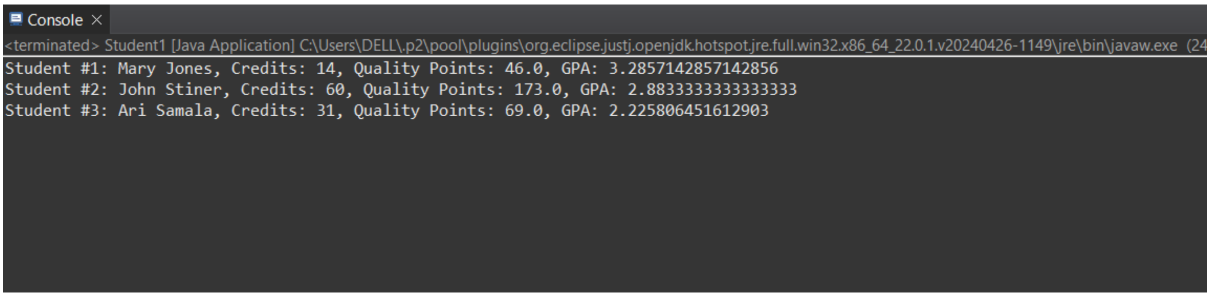
<terminated> Student [Java Application] C:\Users\DELL\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.f

5. Using the class you created in #4, create three instances of the Student Class from the table below:



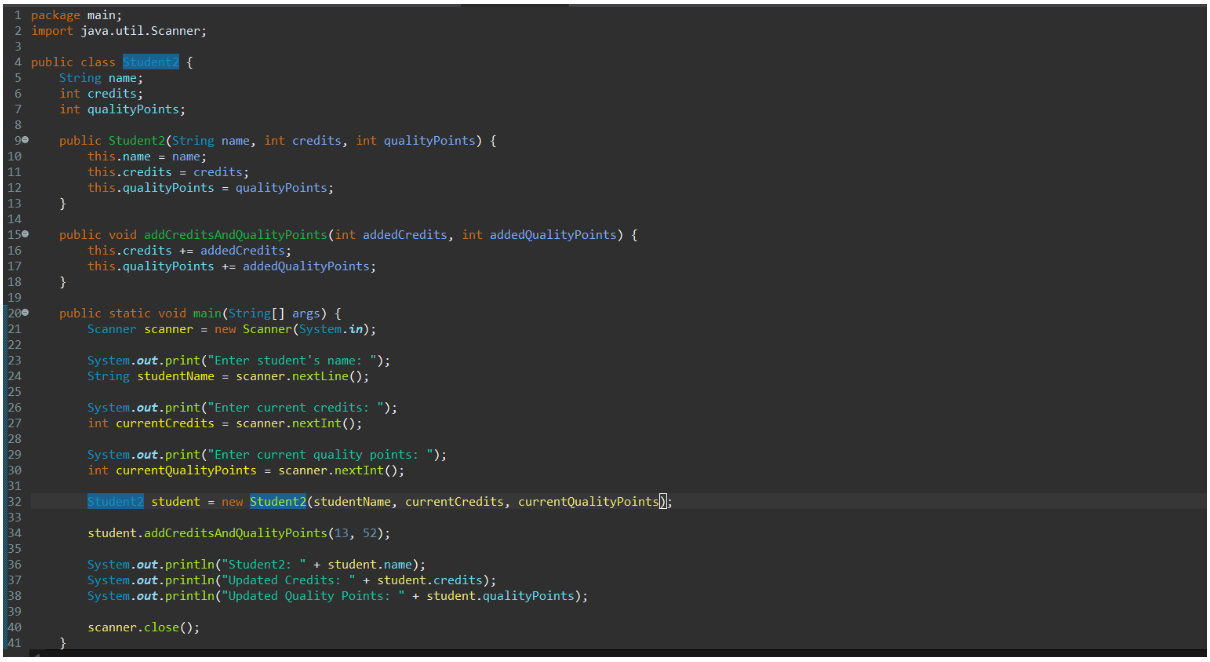
```java
package main;

import java.util.Scanner;
public class Student1{
    private String name;
    private int credits;
    private double qualityPoints;

    public Student1(String name, int credits, double qualityPoints) {
        this.name = name;
        this.credits = credits;
        this.qualityPoints = qualityPoints;
    }

    public String getName() {
        return name;
    }

    public int getCredits() {
        return credits;
    }

    public double getQualityPoints() {
        return qualityPoints;
    }

    public double calculateGPA() {
        return qualityPoints / credits;
    }

    public static void main(String[] args) {
        Student1 student1 = new Student1("Mary Jones", 14, 46);
        Student1 student2 = new Student1("John Stiner", 60, 173);
        Student1 student3 = new Student1("Ari Samala", 31, 69);


        System.out.println("Student #1: " + student1.getName() + ", Credits: " + student1.getCredits() + ", Quality Points: " + student1.getQualityPoints() + ", GPA: " + student1.calc
        System.out.println("Student #2: " + student2.getName() + ", Credits: " + student2.getCredits() + ", Quality Points: " + student2.getQualityPoints() + ", GPA: " + student2.calc
        System.out.println("Student #3: " + student3.getName() + ", Credits: " + student3.getCredits() + ", Quality Points: " + student3.getQualityPoints() + ", GPA: " + student3.calc
    }
}
```

Output:



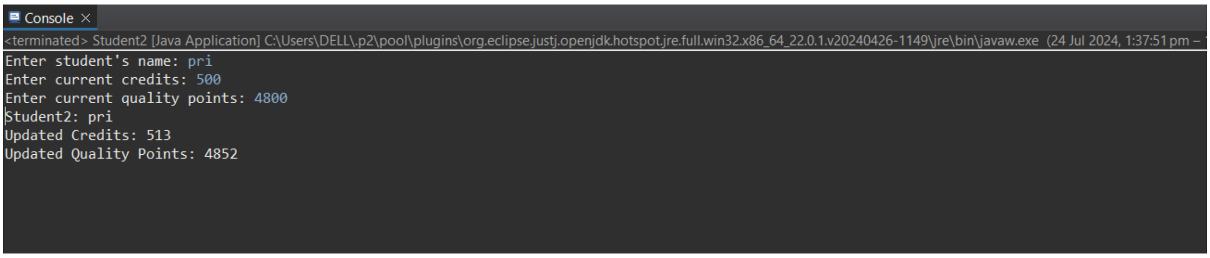
```
<terminated> Student1 [Java Application] C:\Users\DELL\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.1.v20240426-1149\jre\bin\javaw.exe  (24
Student #1: Mary Jones, Credits: 14, Quality Points: 46.0, GPA: 3.2857142857142856
Student #2: John Stiner, Credits: 60, Quality Points: 173.0, GPA: 2.8833333333333333
Student #3: Ari Samala, Credits: 31, Quality Points: 69.0, GPA: 2.225806451612903
```

6. Using the instance variables created in #5, add 13 credits and 52 quality points to the student "Ari Samala".



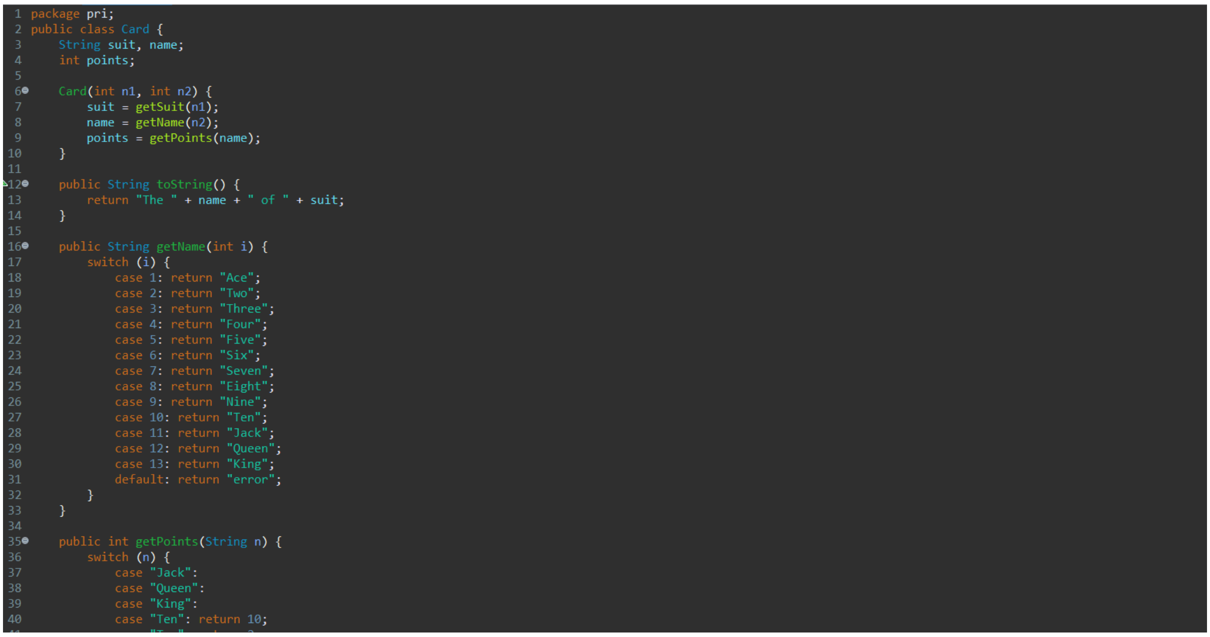
```java
package main;
import java.util.Scanner;

public class Student2 {
    String name;
    int credits;
    int qualityPoints;

    public Student2(String name, int credits, int qualityPoints) {
        this.name = name;
        this.credits = credits;
        this.qualityPoints = qualityPoints;
    }

    public void addCreditsAndQualityPoints(int addedCredits, int addedQualityPoints) {
        this.credits += addedCredits;
        this.qualityPoints += addedQualityPoints;
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter student's name: ");
        String studentName = scanner.nextLine();

        System.out.print("Enter current credits: ");
        int currentCredits = scanner.nextInt();

        System.out.print("Enter current quality points: ");
        int currentQualityPoints = scanner.nextInt();

        Student2 student = new Student2(studentName, currentCredits, currentQualityPoints);

        student.addCreditsAndQualityPoints(13, 52);

        System.out.println("Student2: " + student.name);
        System.out.println("Updated Credits: " + student.credits);
        System.out.println("Updated Quality Points: " + student.qualityPoints);

        scanner.close();
    }
```

Output:

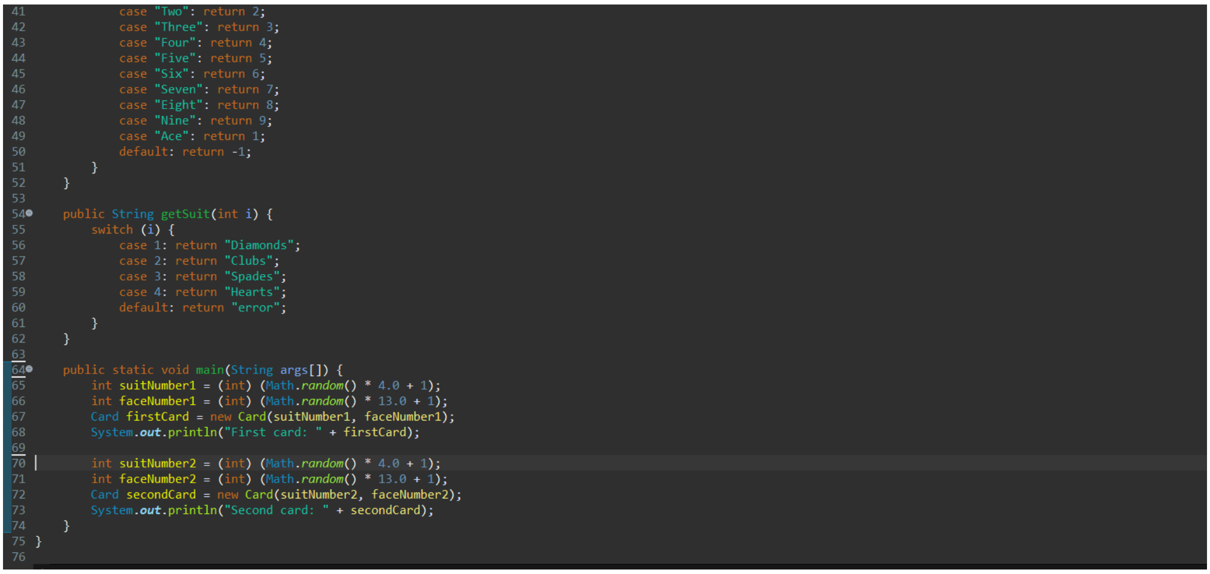


```
<terminated> Student2 [Java Application] C:\Users\DELL\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.1.v20240426-1149\jre\bin\javaw.exe  (24 Jul 2024, 1:37:51 pm –
Enter student's name: pri
Enter current credits: 500
Enter current quality points: 4800
Student2: pri
Updated Credits: 513
Updated Quality Points: 4852
```

7. Using the Card class from the slides and test the program to make sure it works. Add a second random Card

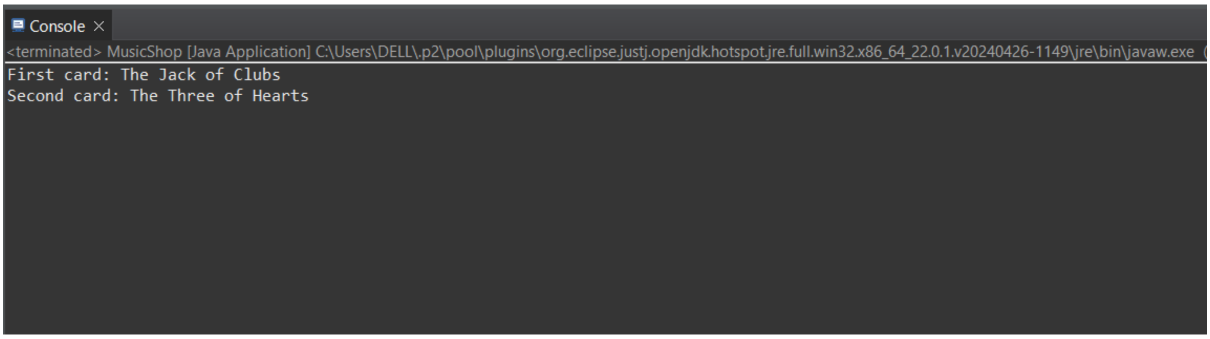


```java
package pri;
public class Card {
    String suit, name;
    int points;

    Card(int n1, int n2) {
        suit = getSuit(n1);
        name = getName(n2);
        points = getPoints(name);
    }

    public String toString() {
        return "The " + name + " of " + suit;
    }

    public String getName(int i) {
        switch (i) {
            case 1: return "Ace";
            case 2: return "Two";
            case 3: return "Three";
            case 4: return "Four";
            case 5: return "Five";
            case 6: return "Six";
            case 7: return "Seven";
            case 8: return "Eight";
            case 9: return "Nine";
            case 10: return "Ten";
            case 11: return "Jack";
            case 12: return "Queen";
            case 13: return "King";
            default: return "error";
        }
    }

    public int getPoints(String n) {
        switch (n) {
            case "Jack":
            case "Queen":
            case "King":
            case "Ten": return 10;
```



```java
            case "Two": return 2;
            case "Three": return 3;
            case "Four": return 4;
            case "Five": return 5;
            case "Six": return 6;
            case "Seven": return 7;
            case "Eight": return 8;
            case "Nine": return 9;
            case "Ace": return 1;
            default: return -1;
        }
    }

    public String getSuit(int i) {
        switch (i) {
            case 1: return "Diamonds";
            case 2: return "Clubs";
            case 3: return "Spades";
            case 4: return "Hearts";
            default: return "error";
        }
    }

    public static void main(String args[]) {
        int suitNumber1 = (int) (Math.random() * 4.0 + 1);
        int faceNumber1 = (int) (Math.random() * 13.0 + 1);
        Card firstCard = new Card(suitNumber1, faceNumber1);
        System.out.println("First card: " + firstCard);

        int suitNumber2 = (int) (Math.random() * 4.0 + 1);
        int faceNumber2 = (int) (Math.random() * 13.0 + 1);
        Card secondCard = new Card(suitNumber2, faceNumber2);
        System.out.println("Second card: " + secondCard);
    }
}
```

Output:



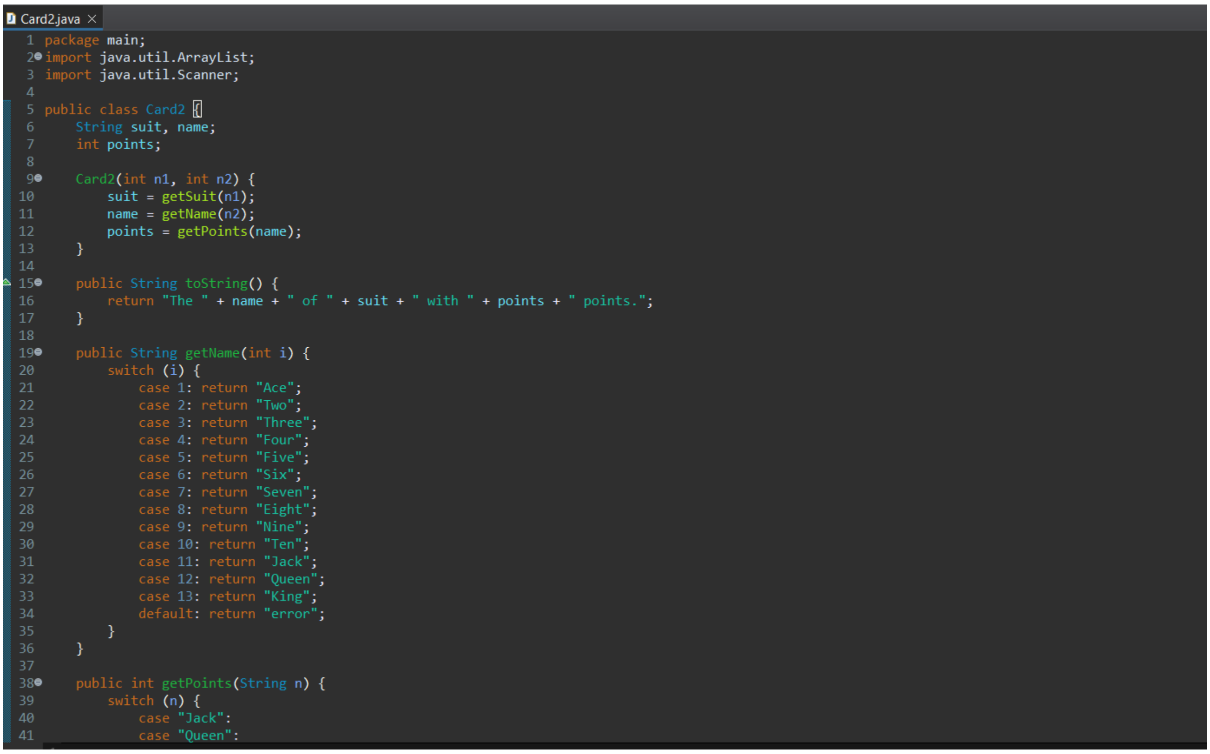
```
<terminated> MusicShop [Java Application] C:\Users\DELL\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.1.v20240426-1149\jre\bin\javaw.exe
First card: The Jack of Clubs
Second card: The Three of Hearts
```

8. Add code to the Main class in exercise #7 to the following:

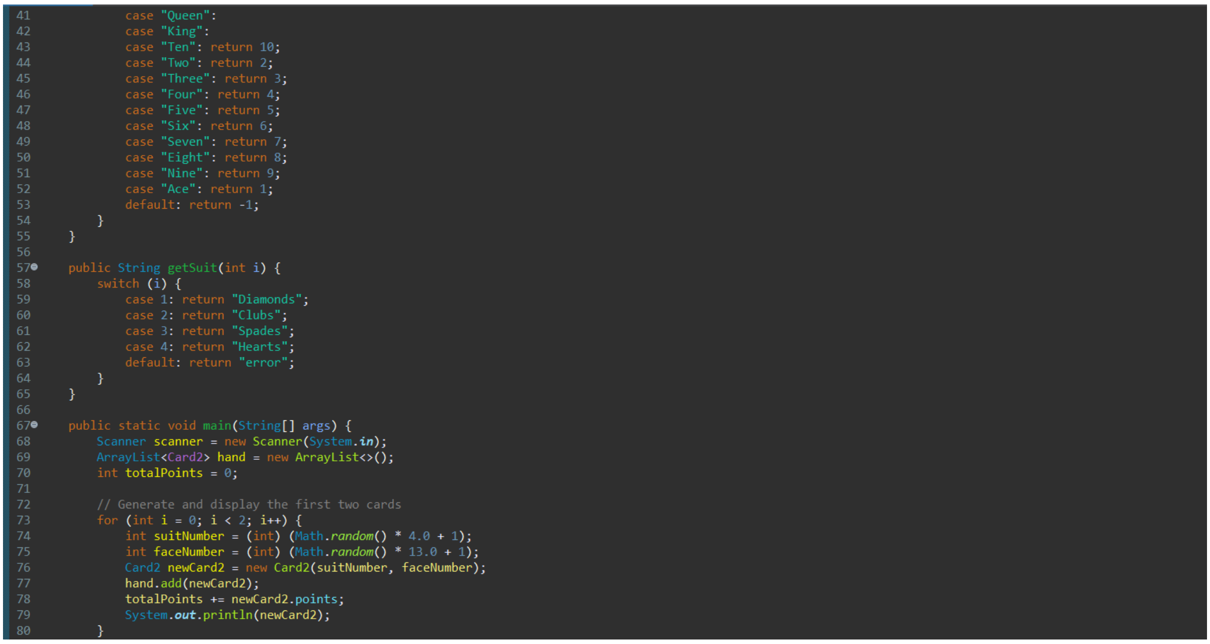
a. Display the total point value for the two random cards.

b. Ask the user if they would like another card. If they say yes display the new card and the points for all 3 cards in their "Hand".

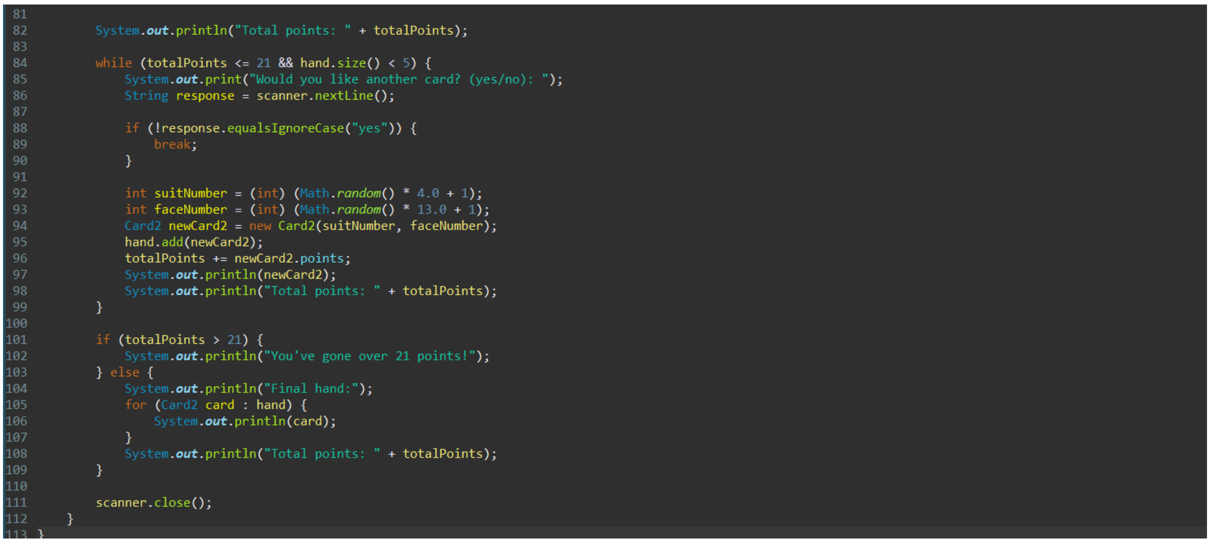
c. Loop to allow the user to continue to add cards to the hand until the number of points goes over 21 or the user decides not to add any more cards or the total number of cards is 5.



```java
package main;
import java.util.ArrayList;
import java.util.Scanner;

public class Card2 {
    String suit, name;
    int points;

    Card2(int n1, int n2) {
        suit = getSuit(n1);
        name = getName(n2);
        points = getPoints(name);
    }

    public String toString() {
        return "The " + name + " of " + suit + " with " + points + " points.";
    }

    public String getName(int i) {
        switch (i) {
            case 1: return "Ace";
            case 2: return "Two";
            case 3: return "Three";
            case 4: return "Four";
            case 5: return "Five";
            case 6: return "Six";
            case 7: return "Seven";
            case 8: return "Eight";
            case 9: return "Nine";
            case 10: return "Ten";
            case 11: return "Jack";
            case 12: return "Queen";
            case 13: return "King";
            default: return "error";
        }
    }

    public int getPoints(String n) {
        switch (n) {
            case "Jack":
            case "Queen":
```

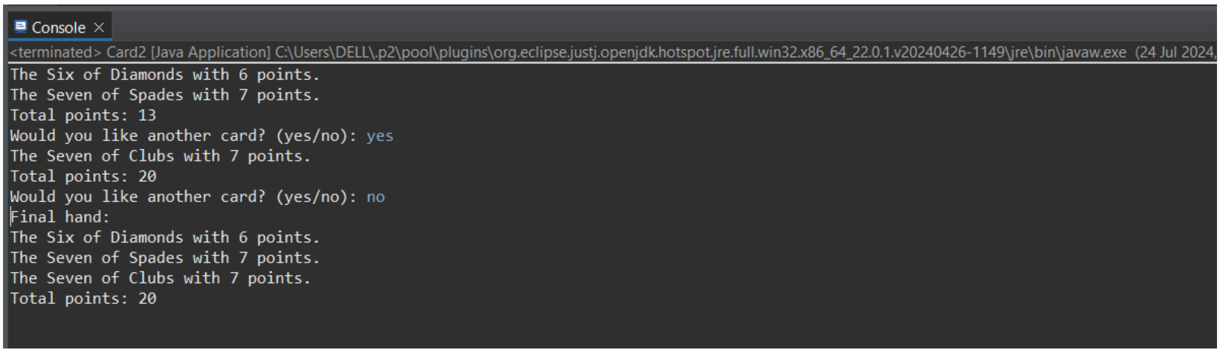


```java
            case "Queen":
            case "King":
            case "Ten": return 10;
            case "Two": return 2;
            case "Three": return 3;
            case "Four": return 4;
            case "Five": return 5;
            case "Six": return 6;
            case "Seven": return 7;
            case "Eight": return 8;
            case "Nine": return 9;
            case "Ace": return 1;
            default: return -1;
        }
    }

    public String getSuit(int i) {
        switch (i) {
            case 1: return "Diamonds";
            case 2: return "Clubs";
            case 3: return "Spades";
            case 4: return "Hearts";
            default: return "error";
        }
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        ArrayList<Card2> hand = new ArrayList<>();
        int totalPoints = 0;

        // Generate and display the first two cards
        for (int i = 0; i < 2; i++) {
            int suitNumber = (int) (Math.random() * 4.0 + 1);
            int faceNumber = (int) (Math.random() * 13.0 + 1);
            Card2 newCard2 = new Card2(suitNumber, faceNumber);
            hand.add(newCard2);
            totalPoints += newCard2.points;
            System.out.println(newCard2);
        }
```



```java

        System.out.println("Total points: " + totalPoints);

        while (totalPoints <= 21 && hand.size() < 5) {
            System.out.print("Would you like another card? (yes/no): ");
            String response = scanner.nextLine();

            if (!response.equalsIgnoreCase("yes")) {
                break;
            }

            int suitNumber = (int) (Math.random() * 4.0 + 1);
            int faceNumber = (int) (Math.random() * 13.0 + 1);
            Card2 newCard2 = new Card2(suitNumber, faceNumber);
            hand.add(newCard2);
            totalPoints += newCard2.points;
            System.out.println(newCard2);
            System.out.println("Total points: " + totalPoints);
        }

        if (totalPoints > 21) {
            System.out.println("You've gone over 21 points!");
        } else {
            System.out.println("Final hand:");
            for (Card2 card : hand) {
                System.out.println(card);
            }
            System.out.println("Total points: " + totalPoints);
        }

        scanner.close();
    }
}
```

Output:



```
<terminated> Card2 [Java Application] C:\Users\DELL\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.1.v20240426-1149\jre\bin\javaw.exe  (24 Jul 2024
The Six of Diamonds with 6 points.
The Seven of Spades with 7 points.
Total points: 13
Would you like another card? (yes/no): yes
The Seven of Clubs with 7 points.
Total points: 20
Would you like another card? (yes/no): no
Final hand:
The Six of Diamonds with 6 points.
The Seven of Spades with 7 points.
The Seven of Clubs with 7 points.
Total points: 20
```