

CSCI 145 PA __4__ Submission

Due Date: __March 21, 2023__ Late (date and time): _____

Name(s): __Ivan Leung__ & _____

Exercise 1 -- need to submit source code and I/O

-- check if completely done __x__ ; otherwise, discuss issues below

Pseudocode below if applicable:

Source code below:

```
package pa4;
```

```
/* Java Class: CSCI 145
```

```
Modified by: Ivan Leung
```

```
Class: Mon/Wed
```

```
Date: Mar 14 2023
```

```
Description:
```

I certify that the code below is my own work.

Exception(s): N/A

```
*/
```

```
//*****
```

```
//Student.java

//

//Define a student class that stores name, score on test 1, and
//score on test 2. Methods prompt for and read in grades,
//compute the average, and return a string containing student's info.

//*****

import java.util.Scanner;

public class Student {

    // declare instance data

    private String name;

    private double test1Score;

    private double test2Score;

    private static Scanner scan = new Scanner(System.in);

    // -----

    // constructor

    // -----

    public Student(String studentName) {

        this.name = studentName;

    }

}
```

```

// -----

// inputGrades: prompt for and read in student's grades for test1 and test2.

// Use name in prompts, e.g., "Enter's Joe's score for test1".

// -----

public void inputGrades() {

    // add body of inputGrades

    System.out.print("Enter " + this.name + "'s score for test1: ");

    this.test1Score = scan.nextDouble();

    System.out.print("Enter " + this.name + "'s score for test2: ");

    this.test2Score = scan.nextDouble();

}

// -----

// getAverage: compute and return the student's test average

// -----

// add header for getAverage

public double getAverage() {

    // add body of getAverage

    return (test1Score + test2Score) / 2;

}

// -----

// getName: print the student's name

// -----

```

```

        // add header for printName

        public String getName() {

            // add body of printName

            return this.name;

        }

    }

```

package pa4;

```

/*  Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 14 2023
Description:

```

I certify that the code below is my own work.

Exception(s): N/A

```

*/

```

```

//*****
// Grades.java
//
// Use Student class to get test grades for two students
// and compute averages
//
//*****

```

```

public class Grades {
    public static void main(String[] args) {
        Student student1 = new Student("Mary");
        // create student2, "Mike"
        Student student2 = new Student("Mike");
        // input grades for Mary
        student1.inputGrades();
        // print average for Mary
    }
}

```

```

        System.out.printf("%s's average grade is %.1f%n",
student1.getName(), student1.getAverage());
        System.out.println();
        // input grades for Mike
        student2.inputGrades();
        // print average for Mike
        System.out.printf("%s's average grade is %.1f%n",
student2.getName(), student2.getAverage());
    }
}

```

Input/output below:

```

Enter Mary's score for test1: 89
Enter Mary's score for test2: 56
Mary's average grade is 72.5

```

```

Enter Mike's score for test1: 83
Enter Mike's score for test2: 87
Mike's average grade is 85.0

```

Exercise 2 -- need to submit source code and I/O

-- check if completely done __x__ ; otherwise, discuss issues below

Pseudocode below if applicable:

Source code below:

```

package pa4;

```

```

/*  Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 14 2023
Description:

```

I certify that the code below is my own work.

Exception(s): N/A

```

*/

```

```

public class Name {
    public Name(String first, String middle, String last) {
        this.first = first;
        this.middle = middle;
        this.last = last;
    }

    public String getFirst() {
        return this.first;
    }
    public String getMiddle() {
        return this.middle;
    }
    public String getLast() {
        return this.last;
    }
    public String firstMiddleLast() {
        return (this.first + " " + this.middle + " " +
this.last);
    }
    public String lastFirstMiddle() {
        return (this.last + ", " + this.first + " " +
this.middle);
    }
    public boolean equals(Name otherName) {

        return
this.first.equalsIgnoreCase(otherName.getFirst()) &&
this.middle.equalsIgnoreCase(otherName.getMiddle()) &&
this.last.equalsIgnoreCase(otherName.getLast());
    }
    public String initials() {
        return (this.first.substring(0, 1).toUpperCase() +
this.middle.substring(0, 1).toUpperCase() +
this.last.substring(0, 1).toUpperCase());
    }
    public int length() {
        return (this.first.length() + this.middle.length() +
this.last.length());
    }

    private String first;
    private String middle;
    private String last;
}

```

```
}
```

```
package pa4;
```

```
/* Java Class: CSCI 145  
Modified by: Ivan Leung  
Class: Mon/Wed  
Date: Mar 14 2023  
Description:
```

```
I certify that the code below is my own work.
```

```
Exception(s): N/A
```

```
*/
```

```
import java.util.Scanner;
```

```
public class TestNames {
```

```
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        Scanner scan = new Scanner(System.in);  
        Name user1;  
        Name user2;  
  
        System.out.print("Enter the first user's name in  
first-middle-last format with space in between: ");  
        user1 = new Name(scan.next(), scan.next(),  
scan.next());  
        System.out.print("Enter the second user's name in  
first-middle-last format with space in between: ");  
        user2 = new Name(scan.next(), scan.next(),  
scan.next());  
  
        System.out.println("The first user's name in first-  
middle-last format is: " + user1.firstMiddleLast());  
        System.out.println("The first user's name in last-  
first-middle format is: " + user1.lastFirstMiddle());  
        System.out.println("The first user's initial is: " +  
user1.initials());  
        System.out.println("The first user's full name in  
length is: " + user1.length());  
        System.out.println();  
    }  
}
```

```

        System.out.println("The second user's name in first-
middle-last format is: " + user2.firstMiddleLast());
        System.out.println("The second user's name in last-
first-middle format is: " + user2.lastFirstMiddle());
        System.out.println("The second user's initial is: " +
user2.initials());
        System.out.println("The second user's full name in
length is: " + user2.length());
        System.out.println();

        if (user1.equals(user2))
            System.out.println("The first and second users
have the same name");
        else
            System.out.println("The first and second users do
not have the same name");
    }
}

```

Input/output below:

```

Enter the first user's name in first-middle-last format with
space in between: Ivan Honyan Leung
Enter the second user's name in first-middle-last format with
space in between: Mary Jane Smith
The first user's name in first-middle-last format is: Ivan Honyan
Leung
The first user's name in last-first-middle format is: Leung, Ivan
Honyan
The first user's initial is: IHL
The first user's full name in length is: 15

The second user's name in first-middle-last format is: Mary Jane
Smith
The second user's name in last-first-middle format is: Smith,
Mary Jane
The second user's initial is: MJS
The second user's full name in length is: 13

The first and second users do not have the same name

```


Exercise 3 -- need to submit source code and I/O

-- check if completely done __x__ ; otherwise, discuss issues below

Pseudocode below if applicable:

Source code below:

```
package pa4;
```

```
//Created by T. Vo for CSCI 145 Spring 2023  
//It can be as an option for PA 4 instead of JavaFX  
//Modified by: Ivan Leung
```

```
/* Java Class: CSCI 145  
Modified by: Ivan Leung  
Class: Mon/Wed  
Date: Mar 14 2023  
Description:
```

I certify that the code below is my own work.

Exception(s): N/A

```
*/
```

```
import java.awt.Color;  
import java.awt.Dimension;  
import java.awt.Graphics;  
import java.awt.Rectangle;  
import java.util.Random;
```

```
import javax.swing.JFrame;
```

```
public class DrawingRectangleApp extends JFrame {  
    public DrawingRectangleApp() {  
        super("Drawing Rectangles");  
        // getContentPane().setBackground(Color.GREEN);  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        setPreferredSize(new Dimension(600, 400));  
        pack();  
    }  
  
    public void paint(Graphics page) {  
        super.paint(page);  
        Random rand = new Random();
```

```

// add your code to draw the 5 rectangles
int x = rand.nextInt(401);
int y = rand.nextInt(201);
int width = (rand.nextInt(501) + 100) - x;
int height = (rand.nextInt(301) + 100) - y;

page.setColor(Color.red);
page.drawRect(x, y, width, height);

x = rand.nextInt(401);
y = rand.nextInt(201);
width = (rand.nextInt(501) + 100) - x;
height = (rand.nextInt(301) + 100) - y;

page.setColor(Color.green);
page.drawRect(x, y, width, height);

x = rand.nextInt(401);
y = rand.nextInt(201);
width = (rand.nextInt(501) + 100) - x;
height = (rand.nextInt(301) + 100) - y;

page.setColor(Color.blue);
page.drawRect(x, y, width, height);

x = rand.nextInt(401);
y = rand.nextInt(201);
width = (rand.nextInt(501) + 100) - x;
height = (rand.nextInt(301) + 100) - y;

page.setColor(Color.cyan);
page.drawRect(x, y, width, height);

x = rand.nextInt(401);
y = rand.nextInt(201);
width = (rand.nextInt(501) + 100) - x;
height = (rand.nextInt(301) + 100) - y;

page.setColor(Color.magenta);
page.drawRect(x, y, width, height);

```

```

}

```

```

// -----

```

```

-----

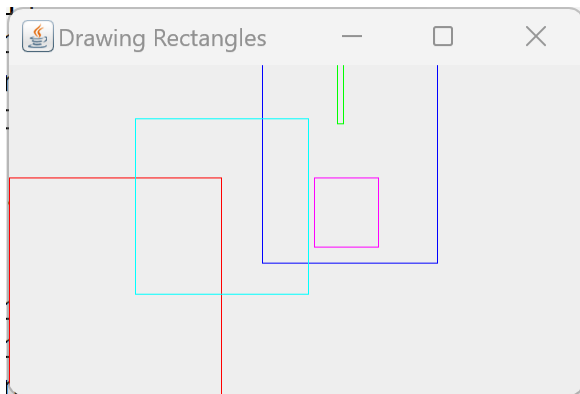
```

```

// Creates the main frame of the program.
// -----
-----
public static void main(String[] args) {
    JFrame frame = new DrawingRectangleApp();
    frame.setVisible(true);
    frame.repaint();
}
}

```

Input/output below:



Add more exercises as needed

Exercise 4 -- need to submit source code and I/O

-- check if completely done __x__ ; otherwise, discuss issues below

Pseudocode below if applicable:

Source code below:

```
package pa4;
```

```
//Class Roulette for CSCI 145 PA 4 Spring 2023
```

```
//Modified by: Ivan Leung
```

```
/* Java Class: CSCI 145
```

```
Modified by: Ivan Leung
```

```
Class: Mon/Wed
```

Date: Mar 14 2023

Description:

I certify that the code below is my own work.

Exception(s): N/A

*/

```
import java.util.*;
```

```
//Class Roulette represents a roulette betting game.
```

```
class Roulette {  
    // public name constants -- accessible to others  
    public final static int BLACK = 0; // even numbers  
    public final static int RED = 1; // odd numbers  
    public final static int GREEN = 2; // 00 OR 0  
    public final static int NUMBER = 3; // number bet  
    public final static int MIN_NUM = 1; // smallest number to  
bet  
    public final static int MAX_NUM = 36; // largest number to  
bet  
    public final static int MIN_BET = 1; // minimum amount to  
bet  
  
    // private name constants -- internal use only  
    private final static int MAX_POSITIONS = MAX_NUM + 2; //  
number of positions on wheel  
    private final static int NUMBER_PAYOFF = MAX_NUM - 1; //  
payoff for number bet  
    private final static int COLOR_PAYOFF = 2; // payoff for  
color bet  
  
    // private variables -- internal use only  
    private static int ballPosition = 0; // 00, 0, 1 .. MAX_NUM  
    private static int color = GREEN; // GREEN, RED, OR BLACK  
  
    // private variables -- testing only  
    private static int next = 0; // next value in the list  
    private static int[] randValues = { 20, 5, 0, 1, 36 }; // 5  
values  
  
    // Contains the main processing loop for the roulette game.  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);
```

```

        Player player = new Player("Jane", 100); // $100 to
start for Jane
        boolean done = false;
        int currentSpin;

        System.out.println("Author: [Your Name]\n");
        welcomeMessage();

        while (!done) {
            System.out.println("Money available for " +
player.getName() + ": " + player.getMoney());
            betOptions();

            // Add code so player can make a bet
            player.makeBet(scan);

            // spin() and display value
            currentSpin = spin();
            System.out.println("Spinning ...");
            System.out.println("Current number: " +
currentSpin);

            // Assume player lost a bet so no payment at this
point

            done = !player.playAgain(scan);

            System.out.println();
        }

        System.out.println("Game over! Thanks for playing.");
        scan.close();
    }

    //
=====
// Presents welcome message
//
=====
public static void welcomeMessage() {
    System.out.println("Welcome to a simple version of
roulette game.");
    System.out.println("You can place a bet on black, red,
or a number.");
}

```

```

        System.out.println("A color bet is paid " +
COLOR_PAYOFF + " times the bet amount.");
        System.out.println("A number bet is paid " +
NUMBER_PAYOFF + " times the bet amount.");
        System.out.println("You can bet on a number from " +
MIN_NUM + " to " + MAX_NUM + ".");
        System.out.println("Gamble responsibly. Have fun and
good luck!\n");
    }

    //
=====
====
    // Presents betting options
    //
=====
====
    public static void betOptions() {
        System.out.println("Betting Options:");
        System.out.println("    1. Bet on black (even
numbers)");
        System.out.println("    2. Bet on red (odd numbers)");
        System.out.println("    3. Bet on a number between " +
MIN_NUM + " and " + MAX_NUM);
        System.out.println();
    }

    // Spins the wheel
    public static int spin() {
        int result;

        // use nextRandom() for testing now
        result = nextRandom();

        return result;

        // comment above code and add your code to spin
    }

    // Payoff method for number bet
    public static int payoff(int betAmt, int betType, int
numberBet) {
        int pay = 0;

        return pay;
    }

```

```

        // Returns a simulated "random" value for testing
        // Assume a value between 0 and 36
        public static int nextRandom() {
            int num = randValues[next];
            next++;
            next = next % randValues.length; // back to 0 if
needed
            return num;
        }
    }
}

```

```
package pa4;
```

```

//Class Player for CSCI 145 PA 4 Spring 2023
//Modified by: Ivan Leung

```

```

/* Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 14 2023
Description:

```

I certify that the code below is my own work.

Exception(s): N/A

```
*/
```

```
import java.util.*;
```

```
//Class Player represents one roulette player.
```

```

class Player {
    private static final int RELOAD_AMOUNT = 100;
    private int bet, money, betType, number;
    private String name;

    // The Player constructor sets up name and initial available
money.
    public Player(String playerName, int initialMoney) {
        name = playerName;
        money = initialMoney;
    }
}

```

```

// Returns this player's name.
public String getName() {
    return name;
}

// Returns this player's current available money.
public int getMoney() {
    return money;
}

// Prompts the user and reads betting information.
public void makeBet(Scanner scan) {
    System.out.print("Enter a bet option, " + name + " (1,
2, or 3): ");
    betType = scan.nextInt();
    System.out.print("How much to bet: ");
    bet = scan.nextInt();
    money = money - bet;
}

// Determines if the player wants to play again.
public boolean playAgain(Scanner scan) {
    String answer;

    System.out.print("Play again, " + name + "? [y/n] ");
    answer = scan.next();
    return (answer.equals("y") || answer.equals("Y"));
}

// payment method (determines winnings)
public int payment() {
    return 0;
}
}

```

Input/output below:

Author: [Your Name]

Welcome to a simple version of roulette game.
 You can place a bet on black, red, or a number.
 A color bet is paid 2 times the bet amount.

A number bet is paid 35 times the bet amount.
You can bet on a number from 1 to 36.
Gamble responsibly. Have fun and good luck!

Money available for Jane: 100

Betting Options:

1. Bet on black (even numbers)
2. Bet on red (odd numbers)
3. Bet on a number between 1 and 36

Enter a bet option, Jane (1, 2, or 3): 1

How much to bet: 100

Spinning ...

Current number: 20

Play again, Jane? [y/n] n

Game over! Thanks for playing.

Answer for Question 1

Constructor is used to initialize instance data. The main difference between other methods is that it can only be called once only when an object is created. Also, the constructor must have the same name as the class name, and it does not have a return type as it cannot return any values.

Answer for Question 2

The import components inside a Java class include instance data, constructor, and method. My way to learn more about an existing Java class is identifies all the components int the class. Then, I would look at the comment on how each instance data, and methods work. Last, I would run the program to see what it does.

Extra Credit – provide if applicable

Pseudocode below if applicable:

Source code below:

```
package pa4;
```

```
/* Java Class: CSCI 145
```

```
Author: Ivan Leung
```

Class: Mon/Wed
Date: Mar 14 2023
Description:

I certify that the code below is my own work.

Exception(s): N/A

*/

```
public class BandBooster {
    public BandBooster(String name) {
        this.name = name;
        boxesSold = 0;
    }
    public String getName() {
        return this.name;
    }
    public void updateSales(int boxesSold) {
        this.boxesSold += boxesSold;
    }
    public String toString() {
        return this.name + ": " + this.boxesSold + " boxes";
    }

    private String name;
    private int boxesSold;
}
```

package pa4;

/* Java Class: CSCI 145
Author: Ivan Leung
Class: Mon/Wed
Date: Mar 14 2023
Description:

I certify that the code below is my own work.

Exception(s): N/A

*/

```
import java.util.Scanner;
```

```

public class TestBandBooster {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner scan = new Scanner(System.in);
        BandBooster sales1 = new BandBooster("John");
        BandBooster sales2 = new BandBooster("Mary");

        System.out.print("Enter the number of boxes sold by "
+ sales1.getName() + " the first week: ");
        sales1.updateSales(scan.nextInt());
        System.out.print("Enter the number of boxes sold by "
+ sales1.getName() + " the second week: ");
        sales1.updateSales(scan.nextInt());
        System.out.print("Enter the number of boxes sold by "
+ sales1.getName() + " the last week: ");
        sales1.updateSales(scan.nextInt());
        System.out.print("Enter the number of boxes sold by "
+ sales2.getName() + " the first week: ");
        sales2.updateSales(scan.nextInt());
        System.out.print("Enter the number of boxes sold by "
+ sales2.getName() + " the second week: ");
        sales2.updateSales(scan.nextInt());
        System.out.print("Enter the number of boxes sold by "
+ sales2.getName() + " the last week: ");
        sales2.updateSales(scan.nextInt());
        System.out.println();
        System.out.printf("%s\n%s", sales1, sales2);

    }

}

```

Input/output below:

```

Enter the number of boxes sold by John the first week: 39
Enter the number of boxes sold by John the second week: 40
Enter the number of boxes sold by John the last week: 89
Enter the number of boxes sold by Mary the first week: 10
Enter the number of boxes sold by Mary the second week: 39
Enter the number of boxes sold by Mary the last week: 50

```

John: 168 boxes

Mary: 99 boxes