WIX1002 Fundamentals of Programming
Lab 8: Class (Answer)

1. Define a class name Number. This class is used to handle an integer array. The class Number will display all the item of the array, the even number, the prime number, the maximum number, the minimum number, the average, the square number. Create a Tester class that tests the program.

```
Number a = new Number() // generate 10 random integers within 0 to 100.
Number b = new Number(5) // generate 5 random integers within 0 to 100.
Number c = new Number(4, 50) // generate 4 random integers within 0 to 50.
```

```
public class Number {
   private int[] num;
   private int size;
   private final int MAX = 100;

   public  Number() {
      size = 10;
      num = new int[size];
      Random g = new Random();
      for(int i=0; i<size; i++)
         num[i] = g.nextInt(MAX);
   }

   public  Number(int n) {
      size = n;
      num = new int[size];
      Random g = new Random();
      for(int i=0; i<size; i++)
         num[i] = g.nextInt(MAX);
   }

   public Number(int n, int r) {
      size = n;
      num = new int[size];
      Random g = new Random();
      for(int i=0; i<size; i++)
         num[i] = g.nextInt(r);
   }

   public void output() {
      System.out.println("The array items are: ");
      for(int i=0; i<size; i++)
         System.out.print(num[i] + " ");
      System.out.println();
```

```
    System.out.print("The even numbers are: ");
    for(int i=0; i<size; i++)
        if (isEven(num[i]))
            System.out.print(num[i] + " ");
    System.out.println();
    System.out.print("The prime numbers are: ");
    for(int i=0; i<size; i++)
        if (isPrime(num[i]))
            System.out.print(num[i] + " ");
    System.out.println();
    System.out.println("The maximum numbers is: " + getMax(num));
    System.out.println("The minimum numbers is: " + getMin(num));
    System.out.println("The average is: " + getAverage(num));
    System.out.print("The square numbers are: ");
    for(int i=0; i<size; i++)
        if (isSquare(num[i]))
            System.out.print(num[i] + " ");
    System.out.println();
}

public int getMax(int[] a) {
    int max = a[0];
    for(int i=1; i<size; i++)
        if (a[i]>max)
            max = a[i];
    return max;
}

public int getMin(int[] a) {
    int min = a[0];
    for(int i=1; i<size; i++)
        if (a[i]<min)
            min = a[i];
    return min;
}

public double getAverage(int[] a) {
    double sum = 0.0;
    for(int i=0; i<size; i++)
        sum+=a[i];
    return sum / size;
}

public boolean isPrime(int a) {
    if (a==2)
        return true;
```

```
        else if(a==0 || a==1)
            return false;
        else if(a%2==0)
            return false;
        else {
            for(int n=3; n<=a; n+=2)
                if(n==a)
                    return true;
                else if (a%n==0)
                    return false;
        }
        return false;
    }

    public boolean isEven(int a) {
        if (a%2==0)
            return true;
        return false;
    }

    public boolean isSquare(int a) {
        if (a==0)
            return false;
        else {
            for(int i=1; i<=a; i = (i+1)*(i+1))
                if (i==a)
                    return true;
        }
        return false;
    }
}
```

2. Define a class name BankAccount. The class allows the customer to open the saving account with their name, IC or passport number and the deposit amount. The class consists of deposit method and withdraws method. Besides, the class can display the current balance. Create a Tester class that tests the program.

```
public class BankAccount {
   private String fullname, ic;
   private double amount;

   public BankAccount(String f, String i, double a)
   {
      fullname = f;
      ic = i;
      amount = a;
```

```
    }

    public void deposit(double a) {
        amount+=a;
        System.out.println("Deposit " + a + " successfully");
    }

    public void withdraw(double a) {
        if (a>amount)
            System.out.println("Your withdraw more than your balance");
        else {
            amount-=a;
            System.out.println("Withdraw " + a + " successfully");
        }
    }
    public void output() {
         System.out.println("The current balance is " + amount);
    }
}
```

3. Define a class name WeightCalculator. The class has a input method that accepts the user's age and height. Besides, the class consists a method that calculating the recommend weight with this formula

   recommend weight = (height – 100 + age / 10) * 0.9

   The class will display the user's age, height and the recommend weight. Create a Tester class that tests the program.

```
public class WeightCalculator {
    private int age;
    private double height, weight;

    public void input() {
        Scanner k = new Scanner(System.in);
        String temp;
        System.out.print("Enter age: ");
        age = k.nextInt();
        temp = k.nextLine();
        System.out.print("Enter height: ");
        height = k.nextDouble();
        temp = k.nextLine();
    }

    public void getWeight() {
        weight = (height-100+age/10)* 0.9;
    }
```

```
    public void output() {
        System.out.println("Age :" + age);
        System.out.println("Height :" + height);
        System.out.println("Recommended Weight :" + weight);
    }
}
```

4. Define a class Fraction. The class has an input method that accepts the numerator and the denominator from the user. Use the mutator method to set the numerator and denominator and the accessor method to get the value of numerator and denominator. This class also has an method to display the fraction reduced to lowest terms. (find the greatest common divisor for the numerator and denominator. Create a Tester class that tests the program.

```
public class Fraction {
    private int numerator, denominator;

    public void setNumerator(int n) {
        numerator = n;
    }
    public int getNumerator() {
        return numerator;
    }
    public void setDenominator(int d) {
        denominator = d;
    }
    public int getDenominator() {
        return denominator;
    }

    public void input() {
        Scanner k = new Scanner(System.in);
        String temp;
        System.out.print("Enter numerator: ");
        setNumerator(k.nextInt());
        temp = k.nextLine();
        System.out.print("Enter denominator: ");
        setDenominator(k.nextInt());
        temp = k.nextLine();
    }

    public void output() {
        int a = gcd();
        System.out.println("The fraction is : "+ getNumerator()/a + " / " +
getDenominator()/a);
```

```
}

public int gcd() {
   int m, n, temp;
   m = getNumerator();
   n = getDenominator();
   if (m<n) {
     temp = m;
     m = n;
     n = temp;
   }
   while(n>0) {
      temp = m%n;
      m = n;
      n = temp;
   }
   return m;
}
```

5. Define a class Game. The class has a constructor that accept player name. Besides, the class contains a method move that roll the dice. Create a Tester class that tests the program with two players and the player that reach 100 or more win the game.

```
public class Game {
   private String fullname;
   private int dice, amount;
   private Random g;

   public Game(String n) {
      fullname = n;
      amount = 0;
      g = new Random();
   }

   public int getAmount() {
      return amount;
   }

   public String getFullName() {
      return fullname;
   }
   public void move() {
      dice = g.nextInt(Integer.MAX_VALUE)%6 + 1;
      amount+=dice;
      System.out.println(fullname + " rolls" + dice + " Total: " + amount);
   }
```

```
    }

        public static void main(String[] args) {
            Game a = new Game("John");
            Game b = new Game("Alvin");

            while(true) {
                a.move();
                if (a.getAmount()>=100) {
                    System.out.println(a.getFullName() + " WIN!");
                    break;
                }
                b.move();
                if (b.getAmount()>=100) {
                    System.out.println(b.getFullName() + " WIN!");
                    break;
                }
            }

        }
```

6. You operate several burger stalls distributed throughout town. Each burger stall has an ID and the number of burger sold for the day. The class consists of a constructor to initialize the instance variables and a sold method that increment the number of burger sold by parameter value. The class also contains a variable and method that tracks the total number of burgers sold in all stalls. Display the total burger sold by each stall and the total number of burgers sold in all stalls.

```
public class Burger {
    private String ID;
    private int burgerSold;
    private static int totalBurgerSold = 0;

    public Burger(String id) {
        burgerSold = 0;
        this.ID = id;
    }

    public void sold(int burger) {
        burgerSold = burger;
        totalBurgerSold+=burger;
    }

    public void display() {
        System.out.println("Burger Stall - " + ID);
        System.out.println("Total burger sold today is " + burgerSold);
```

```
    }

    public void displayAll() {
            System.out.println("Total number of burger sold from all stalls " +
totalBurgerSold);
    }
}
```

7. Define a class money. The class is used to calculate the number of note and coins for a given amount of money. The available notes are RM100, RM50, RM10, RM5 and RM1. The available coins are 50cent, 20cent, 10cent and 5 cent. The class needs to round up the amount according to the table below. The class also contains addition and subtraction method to add or subtract two objects.

Before Rounding	After Rounding
10.11, 10.12	10.10
10.13, 10.14, 10.16, 10.17	10.15
10.18, 10.19	10.20

```
public class Money {
    private double amount;
    private int note100, note50, note10, note5, note1;
    private int cent50, cent20, cent10, cent5;

    public Money(double a) {
        amount = a;
        note100 = 0;
        note50 = 0;
        note10 = 0;
        note5 = 0;
        note1 = 0;
        cent50 = 0;
        cent20 = 0;
        cent10 = 0;
        cent5 = 0;
    }

    public double getAmount() { return amount; }

    public void addition(Money m) {
        this.amount+=m.getAmount();
    }
```

```
    public void subtraction(Money m) {
        this.amount-=m.getAmount();
    }

    public void display() {
        compute();
        System.out.printf("Total Amount: %.2f\n", amount);
          System.out.printf("RM100: %d RM50: %d RM10: %d RM5: %d RM1: %d\n",
note100, note50, note10, note5, note1);
           System.out.printf("50cent: %d 20cent: %d 10cent: %d 5cent: %d\n", cent50,
cent20, cent10, cent5);
    }

    public void compute() {
        double temp;
        int total;
        temp = amount*100;
        total = (int) temp;

        if (total%10 == 9 || total%10 == 4)
            total+=1;
        else if (total%10 == 8 || total == 3)
            total+=2;
        else if (total%10 == 7 || total%10 == 2)
            total-=2;
        else if (total%10 == 6 || total%10 == 1)
            total-=1;

        note100 = total / 10000;
        total%=10000;
        note50 = total / 5000;
        total%=5000;
        note10 = total / 1000;
        total%=1000;
        note5 = total / 500;
        total%=500;
        note1 = total / 100;
        total%=100;
        cent50 = total / 50;
        total%=50;
        cent20 = total / 20;
        total%=20;
        cent10 = total / 10;
        total%=10;
        cent5 = total / 5;
    }
```

}