

Advanced Programming Language

Lab Assignment 2

Pankaj Kumar (Lcs2020065)

Sol1.

```
class solution1 {
    boolean isPrime(int n) {
        int count = 0;
        for (int i = 2; i * i <= n; i++) {
            if (n % i == 0)
                count++;
        }
        if (count == 0)
            return true;
        else
            return false;
    }

    void calcTwin(int n) {
        for (int i = 2; i <= n; i++) {
            int check = 0;
            int count = 0;
            if (isPrime(i)) {
                for (int j = i + 1; j <= n; j++) {
                    if (!isPrime(j))
                        count++;
                    if (isPrime(j)) {
                        check = j;
                        break;
                    }
                }
                if (count > 0 && check != 0) {
                    System.out.print("(" + i + ", " + check + " ), ");
                }
            }
        }
    }
}

Run | Debug
public static void main(String[] args) {
    int lmt = 100;
    solution1 ob = new solution1();
    ob.calcTwin(lmt);
}
```

Output:

```
( 3,5 ), ( 5,7 ), ( 7,11 ), ( 11,13 ), ( 13,17 ), ( 17,19 ), ( 19,23 ), ( 23,29 ), ( 29,31 )  
, ( 31,37 ), ( 37,41 ), ( 41,43 ), ( 43,47 ), ( 47,53 ), ( 53,59 ), ( 59,61 ), ( 61,67 ), ( 67,71 ), ( 71,73 ), ( 73,79 ), ( 79,83 ), ( 83,89 ), ( 89,97 ),
```

Sol.2.

(i)Using getter:

```
class GetterOnlyClass {  
    private int value;  
  
    public GetterOnlyClass(int value) {  
        this.value = value;  
    }  
  
    public int getValue() {  
        return value;  
    }  
}  
  
public class q2getter {  
    Run | Debug  
    public static void main(String[] args) {  
        var obj = new GetterOnlyClass(12);  
        System.out.println("Getting object value: " + obj.getValue());  
    }  
}
```

(ii)using setter:

```
class SetterOnlyClass {
    int value;

    public SetterOnlyClass(int value) {
        this.value = value;
    }

    public void setValue(int value) {
        this.value = value;
    }
}

public class q2setter {
    Run | Debug
    public static void main(String[] args) {
        var obj = new SetterOnlyClass(12);
        System.out.println("Setting value to 42.");
        obj.setValue(42);
        System.out.println("Value set to 42.");
    }
}
```

Sol.3.

```

class mobileINfo {
    private double cost;
    private String camera;
    public String username;

    // getter method for cost
    public double getCost() {
        return cost;
    }

    // setter method for cost
    public void setCost(double cost) {
        this.cost = cost;
    }

    // getter method for camera
    public String getCamera() {
        return camera;
    }

    // setter method for camera
    public void setCamera(String camera) {
        this.camera = camera;
    }
}

class Mobile {
    Run | Debug
    public static void main(String[] args) {
        mobileINfo user1 = new mobileINfo();
        mobileINfo user2 = new mobileINfo();
        mobileINfo user3 = new mobileINfo(); // user1
        user1.username = "Pankaj";
    }
}

```

```

user1.setCost(25000.00);
user1.setCamera("64 MP");
// user2
user2.username = "Rajesh";
user2.setCost(12000.00);
user2.setCamera("24 MP");
// user3
user3.username = "Ramnish";
user3.setCost(64000.00);
user3.setCamera("48 MP");
// user1 print details
System.out.println("User 1: ");
System.out.println("Username: " + user1.username);
System.out.println("Cost: " + user1.getCost());
System.out.println("Camera: " + user1.getCamera());
System.out.println();
// user2 print details
System.out.println("User 2: ");
System.out.println("Username: " + user2.username);
System.out.println("Cost: " + user2.getCost());
System.out.println("Camera: " + user2.getCamera());
System.out.println();
// user3 print details
System.out.println("User 3: ");
System.out.println("Username: " + user3.username);
System.out.println("Cost: " + user3.getCost());
System.out.println("Camera: " + user3.getCamera());
System.out.println();
}
}

```

Output:

```
User 1:
Username: Pankaj
Cost: 25000.0
Camera: 64 MP

User 2:
Username: Rajesh
Cost: 12000.0
Camera: 24 MP

User 3:
Username: Ramnish
Cost: 64000.0
Camera: 48 MP
```

Sol4.

The concept used here is Encapsulation. From the given problem, the setter and getter methods are used to set and return the values of the object, which can bring us to a conclusion that the fields are declared in the private section and to access them, we are using the getter method to return the values associated with the fields.

Main.java

```
import java.util.Arrays;

public class q4_main {
    public static void main(String[] args) {
        var vehicle = new q4_unmannedvehicle(12, 42);
        vehicle.setLocation(12.123, 42.4321);
        System.out.println("Vehicle position in kilometers: " + Arrays.toString(
            vehicle.getLocationKilometers()));
        System.out.println("Vehicle position in meters: " + Arrays.toString(ve
            hicle.getLocationMeters()));
    }
}
```

UnmannedVehicle.java

```

public class q4_unmannedvehicle {
    double locX, locY; // location in km from some origin

    public q4_unmannedvehicle(double locX, double locY) {
        this.locX = locX;
        this.locY = locY;
    }

    public int[] getLocationKilometers() {
        return new int[]{(int) Math rint(locX), (int) Math rint(locY)};
    }
    public int[] getLocationMeters() {
        return new int[]{(int) Math rint(locX*1000), (int) Math rint(locY*1000
    )});
    }

    void setLocation(double x, double y) {
        locX = x;
        locY = y;
    }
}

```

```

Vehicle position in kilometers: [12, 42]
Vehicle position in meters: [12123, 42432]

```

Sol5.

```
import java.util.Scanner;

public class q5 {

    public static void main(String[] args) {
        var sc = new Scanner(System.in);
        System.out.print("Enter type of shape: ");
        String shape = sc.nextLine();
        System.out.println(shape);

        double area = 0;
        switch (shape.toLowerCase()) {
            case "circle":
                System.out.print("Enter radius: ");
                double radius = sc.nextDouble();
                System.out.println(radius);
                area = Math.PI * radius * radius;
                break;
            case "pentagon":
                System.out.print("Enter side length: ");
                double length = sc.nextDouble();
                System.out.println(length);
                area = 1.72048 * length * length;
                break;
            case "triangle":
                System.out.print("Enter base: ");
                double base = sc.nextDouble();
                System.out.println(base);
                System.out.print("Enter height: ");
                double height = sc.nextDouble();
                System.out.println(height);
                area = 0.5 * base * height;
                break;
            case "square":
                System.out.print("Enter side length: ");
                length = sc.nextDouble();
```

```
                System.out.println(length);
                area = length * length;
                break;
            case "rectangle":
                System.out.print("Enter length: ");
                length = sc.nextDouble();
                System.out.println(length);
                System.out.print("Enter width: ");
                double width = sc.nextDouble();
                System.out.println(width);
                area = length * width;
            }
        System.out.println("Area is: " + area);
    }
}
```


Output:

```
Enter type of shape: circle
circle
Enter radius: 3
3.0
Area is: 28.274333882308138
```

```
Enter type of shape: triangle
triangle
Enter base: 5
5.0
Enter height: 4
4.0
Area is: 10.0
```

```
Enter type of shape: square
square
Enter side length: 8
8.0
Area is: 64.0
```

```
Enter type of shape: rectangle
rectangle
Enter length: 4
4.0
Enter width: 6
6.0
Area is: 24.0
```

```
Enter type of shape: pentagon
pentagon
Enter side length: 5
5.0
Area is: 43.012
```

Sol6.

```
public class q6 {

    public static void main(String[] args) {
        int x = 1;
        int increase = 4;

        for (int i = 0; i < 50; i++) {
            System.out.println(x);
            x += increase;
            increase += 3;
        }
    }
}
```

Output:

1
5
12
22
35
51
70
92
117
145
176
210
247
287
330
376
425
477
532
590
651
715
782
852
925
1001
1080

1162
1247
1335
1426
1520
1617
1717
1820
1926
2035
2147
2262
2380
2501
2625
2752
2882
3015
3151
3290
3432
3577
3725

Sol7.

Main.java

```
(i) public class q7_main1 {

    static int multiply(int x, int y) {
        return x * y;
    }
    static int multiply(int x, int y, int z) {
        return x * y * z;
    }
    static int multiply(double x, double y) {
        return (int) (x * y);
    }
    public static void main(String[] args) {

        System.out.println("Multiplying 2 and 3: " + multiply(2, 3));
        System.out.println("Multiplying 2, 3, and 4: " + multiply(2, 3, 4));
        System.out.println("Multiplying 2 and 3.5: " + multiply(2, 3.5));
    }
}

-----

(ii) public class q7_main2 {

    public static void main(String[] args) {
        var obj1 = new q7_multiply(2, 3);
        var obj2 = new q7_multiply(2, 3, 4);
        var obj3 = new q7_multiply(2, 3.5);
    }
}
```

Multiply.java

```
public class q7_multiply {
    public q7_multiply(int x, int y) {
        System.out.println(x + " times " + y + " is: " + x * y);
    }
    public q7_multiply(int x, int y, int z) {
        System.out.println(x + " times " + y + " times " + z + " is: " + x * y
* z);
    }
    public q7_multiply(double x, double y) {
        System.out.println(x + " times " + y + " is: " + x * y);
    }
}
```

Output:

```
Multiplying 2 and 3: 6  
Multiplying 2, 3, and 4: 24  
Multiplying 2 and 3.5: 7
```

```
2 times 3 is: 6  
2 times 3 times 4 is: 24  
2.0 times 3.5 is: 7.0
```

Sol8.

```
public class q8 {  
    // Write a Java program to print the Fibonacci series of the number up to  
    100  
    public static void main(String[] args) {  
        int a = 1, b = 1;  
        while (a < 100) {  
            System.out.println(a);  
            b += a;  
            a = b-a;  
        }  
    }  
}
```

Output:

```
1  
1  
2  
3  
5  
8  
13  
21  
34  
55  
89
```