SHEET 1 SOLUTIONS

SHEHAB MAHMOUD SALAH | 2100320

GENERAL NOTE: to **copy** code **from** this PDF document, **copy each block of code** separately to **not** lose the code's formatting, alternatively you can find all the **source code** to **PROGRAMMING EXERCISES** on my **GitHub**: https://bit.ly/CSE231Sheets , **happy compiling!**

1. (**AreaAndPerim.java** on GitHub)

```
public class AreaAndPerim{
    // a) rectangle method
    public void rectangle(float width, float height){
        float perimeter = 2*(width + height);
        float area = width * height;
        System.out.println("The rectangle's perimeter = 2 x ("+ width
+" + "+ height +") = " + perimeter);
        System.out.println("The rectangle's area = "+ width + " x " +
height + " = " + area);
    }

    // b) square method
    public void square(float side){
        float perimeter = 4 * side;
        float area = side * side;
        System.out.println("The square's perimeter = 4 x " + side + "
= " + perimeter);
        System.out.println("The square's area = " + side + " x " +
side + " = " + area);
    }

    // c) circle method
    public void circle(float radius){
        float perimeter = (float) (2 * Math.PI * radius); // cast to
"float" because Math.PI is originally a "double" value
        float area = (float) (Math.PI * radius * radius);
        System.out.println("The circle's circumference = 2 x pi x " +
radius + " = " + perimeter);
        System.out.println("The circle's area = " + "pi x " + radius +
" x " + radius + " = " + area);

    }

    // main method:
    public static void main(String[] args){
        AreaAndPerim testCases = new AreaAndPerim();
        testCases.rectangle(4.0f,5.5f);
        testCases.square(10.0f);
        testCases.circle(3.2f);
    }
}
```

2. For the code:

```
public class ContinueTest {
      public static void main(String[] args) {
      for(int i=0;i<10;i++){
            if (i%2 == 0)
                  continue;
            System.out.println(i+ " " + i * i);
            }
      }
}
```

The output would be:

```
1 1
3 9
5 25
7 49
9 81
```

(it prints out the odd numbers from 1 up to 10, and their square values).

3. If `break;` was used instead of `continue;` this would result in the program exiting the for loop after the first even number occurrence, meaning it would not give an output since `( 0 % 2 == 0 )`.

4. (**Triangle.java** on GitHub)

```
public class Triangle {
    public static void main(String args[]){
        for (int i = 0; i < 6; i++){
            for (int j = 0; j < i; j++){
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```

5. Using a For-each loop: (**ArraySum.java** on GitHub)

```
public class ArraySum {
    // array summation method
    public static int sumArray(int[] array){
        int sum = 0;
        for (int element : array){
            sum += element;
        }
        return sum;
    }
    public static void main(String args[]){
        int[] testCase = {1,2,3,4,5};
        System.out.println(sumArray(testCase));
    }
}
```

Alternatively using a For loop:

```
public class ArraySum {
    // array summation method
    public int sumArray(int[] array){
        int sum = 0;
        for (int i = 0; i < array.length; i++){
            sum += array[i];
        }
        return sum;
    }
    public static void main(String args[]){
        ArraySum testCases = new ArraySum();
        int[] testCase = {1,2,3,4,5};
        System.out.println(testCases.sumArray(testCase));
    }
}
```

This concludes Sheet (1) Solutions, this document + source code to all programming exercises available on https://bit.ly/CSE231Sheets **.**