

Programación 3

TEMA 2: Conceptos Básicos - Introducción al lenguaje Java

Práctica nº 2 - A

NOTA: Para realizar esta práctica cree un proyecto llamado **Practica2**

1. Estructuras de Control (for, while) y recursión

- 1.1. Cree una clase llamada **IteradorAB**. Al momento de crear la clase, **verifique** que la casilla de creación automática del método **main** se encuentre chequeada.

NOTA: Para probar las siguientes implementaciones use el "método main" en la misma clase. Además, **NO** necesita leer valores desde el teclado, simplemente asignarles un valor antes de la invocación.

- 1.2. Implemente el **método de instancia** llamado **iteracionConFor** que recibe 2 argumentos de tipo entero (llamados a y b, considere $a < b$) e imprime todos los números comprendidos entre a y b inclusive, usando el **for** como estructura de control.
Es decir, dado $a=2$ y $b=7$ imprime 2 3 4 5 6 7.
- 1.3. Implemente el **método de instancia** llamado **iteracionConWhile** que recibe 2 argumentos de tipo entero (llamados a y b, considere $a < b$) e imprime todos los números comprendidos entre a y b inclusive, usando el **while** como estructura de control.
- 1.4. Implemente el **método de instancia** llamado **recursivo** que recibe 2 argumentos de tipo entero (llamados a y b, considere $a < b$) e imprime todos los números comprendidos entre a y b inclusive, usando **recursión**.
- 1.5. Modifique los métodos para que sean **métodos de clase**.
- 1.5.1. ¿Qué cambios tiene que realizar en la definición del método?
- 1.5.2. ¿Qué cambios tiene que realizar en la invocación a los métodos?
- 1.6. Analice qué modificaciones serían necesarias para estos 3 métodos funcionen si $a > b$ e implemente para el caso recursivo.

2. Arreglos

- 2.1. Cree una clase llamada **Multiplos**.
- 2.2. Escriba un **método de clase** llamado **getMultiplos** que dado un número entero **n** como parámetro, devuelve un arreglo de **n** posiciones cuyo contenido son los primeros n primeros múltiplos mayores a 1.

Ejemplo:

```
getMultiplos(5) devuelve [5; 10; 15; 20; 25];  
getMultiplos(4) devuelve [4; 8; 12; 16];
```

- 2.3. Pruebe que su método funciona correctamente con distintos valores de **n** (indique manualmente los valores de n, **NO** los lea de teclado)

3. Creación de instancias (operador new)

- 3.1. Cree una clase llamada **Tablet**.
- 3.2. Agregue las siguientes **variables de instancia privadas** a la clase Tablet:
- marca (String)
 - sistemaOperativo (String)
 - modelo (String)
 - costo (double)
 - pulgadas (float)
- 3.3. Genere los métodos "getter" y "setter" que permitan asignar valores a las variables de instancia y devolverlos. **Use el menú contextual que provee eclipse para generarlos automáticamente.**
- 3.4. Escriba un método de instancia llamado **devolverDatos()** en la clase Tablet que **devuelve** información de la instancia sobre la cual es invocado. Para el caso de una instancia de Tablet se espera que devuelva todos los valores de sus variables de instancia "concatenados".
- 3.5. Cree una clase llamada TestTablet que contenga el método **main**. Dentro del método **main** cree un **arreglo de Tablet** de 3 posiciones.
- 3.6. Cree 3 instancias de Tablet con distintos valores y agreguelos a cada una de las posiciones del arreglo.
- 3.7. Recorra el arreglo usando una estructura de control iterativa **for** o **foreach** e imprima la información de las Tablets, enviando a cada tablet el mensaje **devolverDatos()**.

4. **Lectura por teclado:** Agregue al programa del ejercicio 2 la posibilidad de probar con distintos valores de n ingresándolos por teclado, mediante el uso de las clases Scanner y System. La clase Scanner permite leer de forma sencilla valores de entrada.

Ayuda: Como ejemplo de uso, el siguiente código cuenta la cantidad de números leídos hasta el primer 42 ingresado:

```
public static int contar42 ( ) {  
    Scanner s = new Scanner(System.in) ;  
    int cantidad = 0 ;  
    while (s.nextInt() != 42) { cantidad++; }  
    return cantidad;  
}
```