

Programación

ESTRUCTURAS DE CONTROL

Manuel Molino Milla Luis Molina Garzón

16 de noviembre de 2017

Índice

1. API de Java	1
1.1. Ejercicio 1	1
2. Estructuras de control	2
2.1. Ejercicio 1	2
2.2. Ejercicio 2 -OPCIONAL-	2
3. API de Java	3
3.1. Ejercicio 1	3
3.2. Ejercicio 2	3
3.3. Ejercicio 3	4
3.4. Ejercicio 4	4
3.5. Ejercicio 5	4
3.6. Ejercicio 6	4

1. API de Java

1.1. Ejercicio 1

La API String de Java, contiene entre otros, los métodos abajo indicados en la tabla. Rellena la información correspondiente:

nombre método	valor retorno	parámetros	Breve descripción	Ejemplo
contains				
endsWith				
equalsIgnoreCase				
isEmpty				
startsWith				
charAt				
length				
replace				
toUpperCase				
toLowerCase				
substring				

2. Estructuras de control

2.1. Ejercicio 1

En una clase denominada *Numeros* implementa métodos que realicen:

- Imprimir los números del 1 al 10.
- Imprimir la siguiente serie: 20 25 30 35 ... 70 75 80
- Imprimir la siguiente serie: 100 98 96 94 ... 56 54 52 50
- Calcular la suma de los números enteros del 1 a N, siendo N un número entero y es el parámetro que se pase al método. El método debe devolver un *int*
- Calcular la la suma de los cuadrados de los N primeros números naturales. El método debe devolver un *int*
- Imprimir independientemente la suma de los pares e impares comprendidos entre 1 y N

Realiza el ejercicio usando una vez el bucle for, otra vez el bucle while y también debes usar la estructura do-while

2.2. Ejercicio 2 -OPCIONAL-

En español hay dos maneras para formar el plural de los sustantivos y adjetivos: -s y -es. Las reglas en las que se basan son:

- **Sustantivos y adjetivos terminados en vocal átona o en -e tónica.** Forman el plural con -s: *casas, estudiantes, taxis, planos, tribus, comités.*
- **Sustantivos y adjetivos terminados en -a o en -o tónicas.** Forman el plural con -s: *papás, sofás, bajás, burós, rococós, dominós.*

- **Sustantivos y adjetivos terminados en -i o en -u tónicas.** Admiten generalmente dos formas de plural, una con -es y otra con -s, aunque en la lengua culta suele preferirse la primera: *bisturíes o bisturís, carmesíes o carmesís, tisúes o tisús, tabúes o tabús*.
- **Sustantivos y adjetivos terminados en -y precedida de vocal.** Forman tradicionalmente su plural con -es: *rey, pl. reyes; ley, pl. leyes; buey, pl. bueyes; ay, pl. ayes; convoy, pl. convoyes; bocoy, pl. bocoyes*
- **Sustantivos y adjetivos terminados en -s o en -x.** Permanecen invariables: *crisis, pl. crisis; tórax, pl. tórax; fórceps, pl. fórceps*.
- **Sustantivos y adjetivos terminados en -l, -r, -n, -d, -z, -j.** Forman el plural con -es: *dócil, pl. dóciles; color, pl. colores; pan, pl. panes; césped, pl. céspedes; cáliz, pl. cálices; reloj, pl. relojes*.

Crea una clase llamada *Plural* que tenga un método que devuelva el plural de dicha palabra.

Comprueba el funcionamiento de dicho método en otra clase que aporte el método *main*

Ayuda: consulta el API de la clase *String*

3. API de Java

3.1. Ejercicio 1

Busca información sobre la clase *Scanner*, usada para leer datos desde el teclado o desde un fichero y explica que realizan los siguientes metodos:

- *Scanner(System.in)*
- *hasNext()*
- *hasNextInt()*
- *hasNextDouble()*
- *next()*
- *nextInt()*
- *nextDouble()*

3.2. Ejercicio 2

Realiza un programa en Java, que lea desde le entrada estandar tu nombre y que presente por pantalla el siguiente mensaje: *Hola nombreLeido*

3.3. Ejercicio 3

Realiza un programa en Java, que lea desde la entrada un número entero de tres cifras y posteriormente muestre sus cifras por separado. Haz todo en un metodo *main* de una clase denominadas Cifras.

3.4. Ejercicio 4

Programa que se le pase como argumentos tres números enteros H, M, S que contienen hora, minutos y segundos respectivamente. Comprueba si la hora que indican es una hora válida. Ejemplo de ejecución:

```
java Tiempo 22 10 15
Hora válida
java Tiempo 25 10 15
Hora no válida
```

3.5. Ejercicio 5

Programa que lea doce números decimales por teclado, crea una clase denominada DoceNumeros. Posteriormente muestra por pantalla todos los valores tabulados(con mas de un espacio blanco entre ellos) en cuatro filas y con dos decimales. (Debes usar el comando *printf*)

3.6. Ejercicio 6

Los sistemas de ecuaciones lineales, en el caso de dos ecuaciones con dos incógnitas se pueden representar de forma genérica de la siguiente forma:

$$\left. \begin{array}{l} a \cdot x + b \cdot y = e \\ c \cdot x + d \cdot y = f \end{array} \right\}$$

Un sistema de ecuaciones tiene solución si:

$$a \cdot d - b \cdot c \neq 0$$

La solución de un sistema de dos ecuaciones con dos incógnitas, en el caso que lo tenga, viene dado por la siguiente regla, derivada de la regla de Cramer:

$$x = \frac{e \cdot d - b \cdot f}{a \cdot d - b \cdot c}$$
$$y = \frac{a \cdot f - e \cdot c}{a \cdot d - b \cdot c}$$

Se quiere realizar una clase denomina *Ecuacion* que contenga dos métodos:

- Un método boolean que devuelva cierto o falso si el sistema es resoluble o no.

- Dos métodos que devuelvan el valor de x e y . Usa *double* como tipos, al menos en los valores de x e y
- Los argumentos en los métodos serán los coeficientes que acompañan a las incógnitas, además del término independiente (a, b, c, \dots)

Crea una clase TestEcuación, con el método *main* y que resuelva las siguientes ecuaciones:

$$\left. \begin{array}{l} x+y=1 \\ 2\cdot x+2\cdot y=2 \end{array} \right\}$$

$$\left. \begin{array}{l} 2\cdot x+y=7 \\ -x+2\cdot y=-1 \end{array} \right\}$$

Debe indicar por pantalla, si el sistema es resoluble o no. Y en el caso que sea resoluble debe mostrar la resolución de dicho sistema de ecuaciones.