

## Práctica #7

### Introducción a Java

#### 1. Java

Java es un lenguaje de programación de propósito general creado por James Gosling en 1996, cuando trabajaba para Sun Microsystems. Hoy en día, los derechos de Java son de la compañía Oracle, quienes lo adquirieron en 2010.



La tecnología Java se utiliza para desarrollar aplicaciones para una amplia gama de entornos, desde dispositivos de consumo hasta sistemas empresariales heterogéneos. El paradigma de programación del lenguaje Java se basa en el concepto de programación orientada a objetos (POO). Su sintaxis deriva en gran medida de C y C++, pero tiene menos utilidades de bajo nivel que cualquiera de ellos. Las aplicaciones creadas con este lenguaje son portables y pueden correr en cualquier tipo de computador. Ellas son compiladas a bytecode, que puede ejecutarse en cualquier máquina virtual Java (JVM) sin importar la arquitectura de la computadora subyacente. Al crear código con Java logramos aplicaciones de alto rendimiento, con computación multi-hilo.

#### 2. Swing

Es una biblioteca de la API de Java para el desarrollo de Interfaces gráficas de usuario. La biblioteca proporciona un conjunto de herramientas para la construcción de interfaces gráficas que tienen una apariencia y se comportan de forma semejante en todas las plataformas en las que se ejecuten.

La estructura básica de la biblioteca gira en torno a componentes y contenedores. Los contenedores contienen componentes y son componentes a su vez, de forma que los eventos pueden tratarse tanto en contenedores como en componentes.

La API está constituida por clases, interfaces y derivaciones. Swing se apoya sobre AWT (otra biblioteca de Java) y añade JComponents. Te permite elegir entre diferentes aspectos (look & feel), y usa arquitectura Modelo-Vista-Controlador (MVC).

### 3. Java en Visual Code

Para programar Java con Visual Code es recomendado usar el OpenJDK v17 y los siguientes *plugins*:

- Extension Pack for Java: para apoyo en la programación Java
- Debugger for Java: para apoyo en el debug de Java
- Test Runner for Java: para realizar pruebas con JUnit en Java
- Project Manager for Java: apoyo adicional para trabajar en proyectos Java

### 4. Ejercicios

a) Sean X un arreglo unidimensional con n elementos enteros, z un número entero.

1. Genere un programa en Java que determine si un entero z pertenece al arreglo X utilizando una estructura de control *while*.
2. Modifique el programa que obtuvo en la parte (1) de manera que el valor de la dimensión, los elementos del arreglo X y el valor z se ingresen por consola. Muestre el resultado de determinar si z pertenece o no al arreglo X por consola.
3. Modifique el programa que obtuvo en la parte (2) para obtener un programa equivalente utilizando una estructura de control *for* en lugar de la estructura de control *while*.
4. Modifique el programa que obtuvo en la parte (3) especificando el arreglo X y el entero z como argumentos de un método llamado "isElementOf" que retorna un booleano como resultado de verificar si el entero z pertenece al arreglo X. Defina este método en una clase que no contenga al método *main*. Por lo tanto, traslade hacia este método las instrucciones que determinan si z pertenece a X.
5. Incorpore el manejo de excepciones para el caso de que el arreglo X sea null
6. Ejecute el programa con los siguientes datos:  
Primera ejecución X = {-1, 0, 2, 7, 9}; z = 10  
Segunda ejecución X = {-2, -1, 0, 1, 2}; z = 2  
Tercera ejecución X = {null}; z = -3

b) Un centro de investigación requiere el registro de los equipos adquiridos, el costo de adquisición y el responsable del equipo, contabilizando el número total de equipos y el monto total en bolívares de los equipos del centro. Se define el siguiente formato para registrar la data de los equipos y del responsable: `desc#ct#mu#"dd"/"mm"/"aaaa"#nf#ci;.`

De este formato se desprende que:

- **desc** es una cadena de caracteres que corresponde a la descripción del equipo (CPU, monitor, mouse, teclado, UPS, pen-drive, cámara, impresora, fotocopidora)
- **ct** representa la cantidad de equipos y es un número entero
- **mu** representa el costo unitario de adquisición del equipo y es un número real
- **dd** es el día de la fecha de la adquisición, número entero de dos dígitos entre 01 y 31
- **mm** es el mes de la fecha de la adquisición, número entero de dos dígitos entre 01 y 12

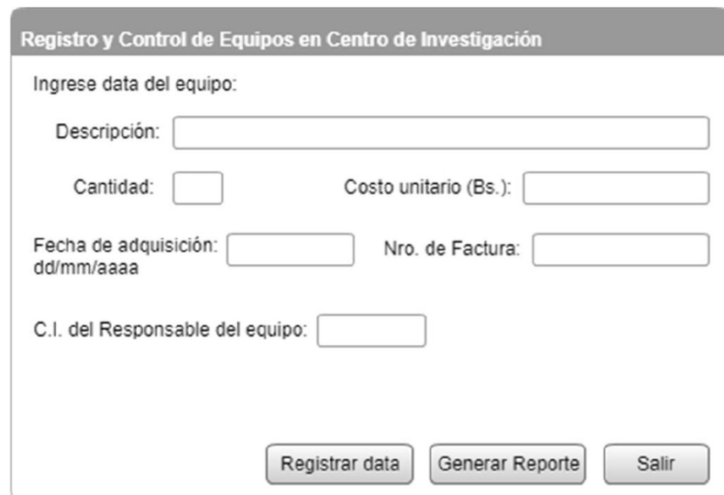
- **aaaa** es el año de la fecha de la adquisición, número entero de cuatro dígitos empezando en 1968
- **nf** es una cadena de caracteres que corresponde al número de la factura
- **ci** es una cadena de caracteres que corresponde a la CI del profesor responsable del equipo

Se requiere un programa en Java que:

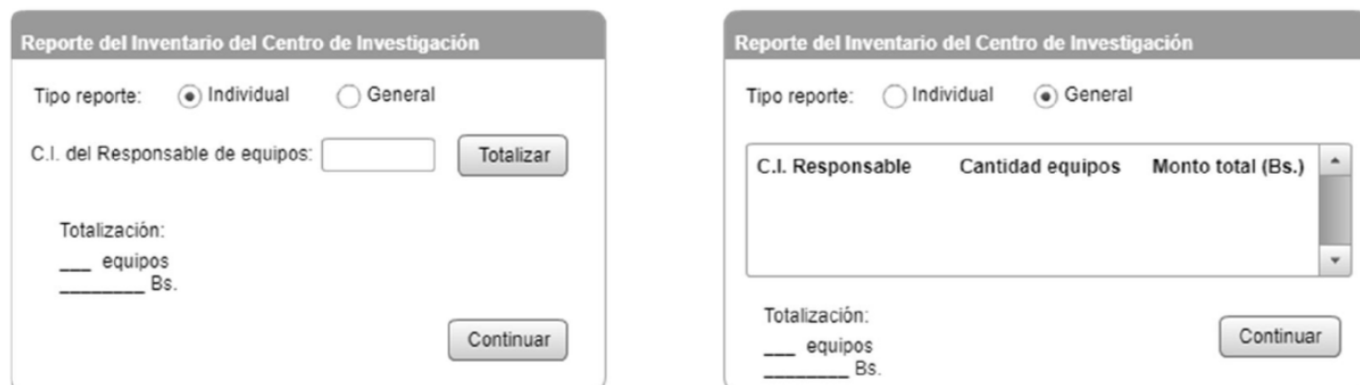
- Lea de consola la data de los equipos y la CI del correspondiente responsable del equipo
- Incorpore instrucciones en el programa para registrar en un archivo inventario.txt la data que leyó por consola según el formato especificado
- Incorpore en el programa instrucciones para generar por consola un reporte en el cual para cada profesor se indique el número total de los equipos y el monto total en bolívares que tiene bajo su responsabilidad
- Incorpore manejo de excepciones para las operaciones con el archivo inventario.txt

c) Considere el programa generado en el ejercicio b. Se requiere sustituir la entrada de la data de los equipos y el responsable de los equipos y la contabilización de los equipos y el monto total en bolívares de los equipos del centro de investigación, definiendo una clase interfaz Gráfica *ICentro* que implemente el prototipo que se muestra en la Figura 1. Para implementar el prototipo, utilice la biblioteca Swing.

Sustituya la generación del reporte por consola para cada uno de los profesores con el número de equipos y el monto total en bolívares que tiene bajo su responsabilidad, definiendo una clase interfaz *IReporte* que implemente el prototipo que se muestra en la Figura 2. Observe que se incorpora la facilidad de generar un reporte individual, es decir para un cierto responsable de los equipos, o un reporte general, para cada uno de los profesores del centro de investigación.



**Figura 1. Prototipo de Interfaz *ICentro*.**



**Reporte del Inventario del Centro de Investigación**

Tipo reporte: ☒ Individual ☐ General

C.I. del Responsable de equipos:

Totalización:  
\_\_\_ equipos  
\_\_\_\_\_ Bs.

**Reporte del Inventario del Centro de Investigación**

Tipo reporte: ☐ Individual ☒ General

C.I. Responsable	Cantidad equipos	Monto total (Bs.)
------------------	------------------	-------------------

Totalización:  
\_\_\_ equipos  
\_\_\_\_\_ Bs.

**Figura 2. Prototipo de Interfaz *IReporte*.**

**Al terminar, colocar en el reto del Aula Virtual el url del repositorio de GitHub que uso para hacer este ejercicio c) (el último).**