

LABORATORIO 6.

ELEMENTOS DEL LENGUAJE DE PROGRAMACIÓN Y ESTRUCTURAS DE DATOS EN JAVA

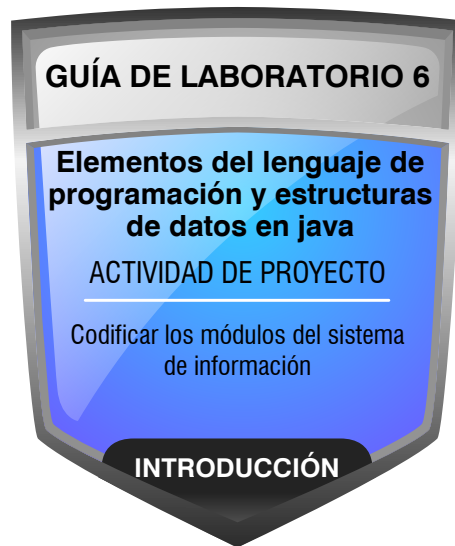
GUÍA DE LABORATORIO N° 6

Actividad de Proyecto: CODIFICAR LOS MÓDULOS DEL SISTEMA DE INFORMACIÓN

Estructura de contenidos.

1. Introducción.	2
2. Objetivos.	2
3. Consideraciones.	3
4. Procedimiento.	4
5. Evidencias a entregar.	10





1. INTRODUCCIÓN

El lenguaje de programación Java posee un conjunto de características, palabras reservadas, sintaxis y estructuras particulares que lo convierten en uno de los lenguajes de programación orientada a objetos más completos en la actualidad.

Como desarrolladores de sistemas de información, es necesario conocer las características particulares de este lenguaje para poder sacarle el máximo provecho en el desarrollo de soluciones de software.

2. OBJETIVOS

- Usar adecuadamente los tipos de datos, identificadores y palabras reservadas en Java.
- Emplear correctamente las expresiones y estructuras de control de flujo en el desarrollo de aplicaciones en Java.
- Crear aplicaciones orientadas a objetos en Java, mediante la implementación de clases con atributos, constructores y métodos.
- Emplear arreglos y colecciones en el desarrollo de aplicaciones en Java.

3. CONSIDERACIONES

Para el desarrollo del laboratorio es importante que tenga en cuenta los siguientes aspectos relacionados en la tabla:

Ítem	Descripción
Soporte Teórico	<p>Para el desarrollo del laboratorio se recomienda la lectura y seguimiento de los objetos de aprendizaje:</p> <ul style="list-style-type: none">• Fundamentos de Java.• Arreglos y Colecciones en Java.
Productos requeridos	<ul style="list-style-type: none">• Enunciados de los ejercicios a desarrollar disponibles en este mismo documento.• API de Java.
Productos requeridos	<p>El laboratorio puede realizarse en cualquier Entorno de Desarrollo Integrado (IDE por sus siglas en inglés) para el desarrollo de aplicaciones en Java, se presentan en el siguiente listado algunos de ellos y se indica el sitio de descarga. Tenga en cuenta que los anteriores laboratorios de Java se desarrollaron en Netbeans.</p> <ul style="list-style-type: none">• Netbeans disponible en https://netbeans.org/• JCreator disponible en http://www.jcreator.com/download.htm• Eclipse disponible en http://www.eclipse.org/downloads/• BlueJ disponible en http://www.bluej.org/download/download.html

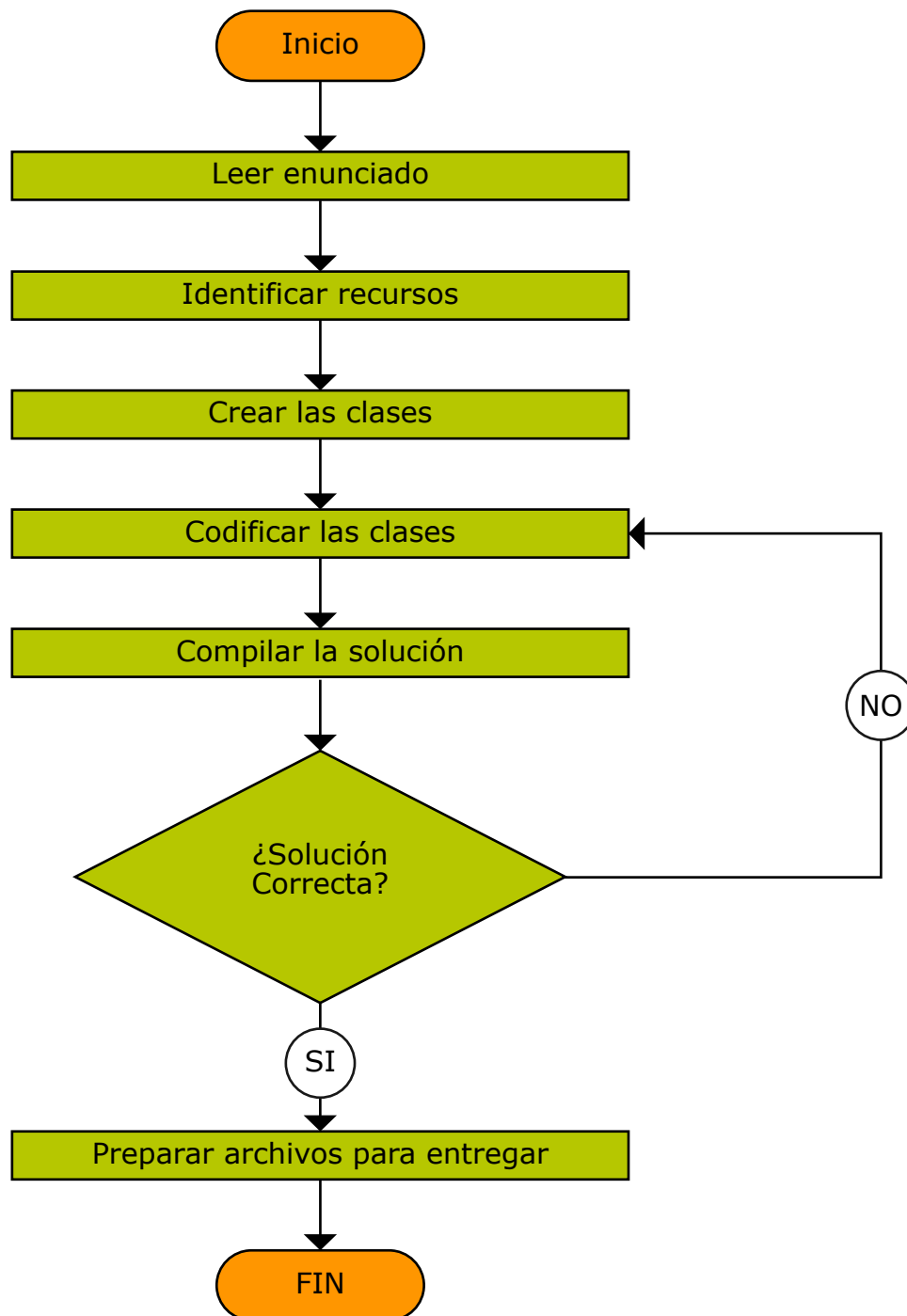
4. PROCEDIMIENTO

Para el Desarrollo de cada uno de los ejercicios tenga en cuenta el siguiente procedimiento:

1. Leer detenidamente cada enunciado.
2. Identificar y/o descargar los recursos necesarios para la solución del enunciado.
3. Crear las clases requeridas como parte de la solución al enunciado.
4. Codificar las clases de acuerdo con lo requerido para dar solución al enunciado.
5. Incluir comentarios que describan los elementos incorporados al código.
6. Compilar la solución.
7. Comprobar la solución de cada ejercicio.
8. Si está correcto continuar con punto 9, si no volver a paso 4.
9. Preparar los archivos entregables como evidencia.



EL SIGUIENTE DIAGRAMA REPRESENTA
EL PROCEDIMIENTO GENERAL DEL LABORATORIO



EJERCICIOS:

1. En una competencia participan N atletas de los cuales es importante conocer su nombre, nacionalidad y su tiempo. Desarrolle una aplicación que presente el siguiente menú:

Competencia
1. Registrar Atleta
2. Datos del campeón.
3. Atletas por país
4. Tiempo promedio de todos los atletas
5. Salir

- Registrar Atleta: Debe solicitar el nombre, la nacionalidad y el tiempo del atleta
- Datos Campeón: Muestra el nombre, la nacionalidad y el tiempo del atleta cuyo tiempo es el menor de todos.
- Atletas por país: Solicita el nombre del país y muestra los nombres de todos los atletas de ese país.
- Tiempo promedio de todos los atletas: Muestra el tiempo promedio de los atletas registrados.

Características de la aplicación:

- La aplicación debe desarrollarse de manera orientada a objetos, incluyendo las clases Atleta, Competencia y Principal.
- Debe emplear arreglos para su solución.

2. Desarrolle una aplicación utilizando colecciones, la cual permita tener las siguientes funcionalidades:

- Insertar elementos (al inicio, al final).
- Eliminar elementos.
- Obtener el valor de una posición determinada.
- Asignar valor a una posición determinada.
- Calcular la sumatoria de los valores de todos los elementos.
- Calcular el promedio de los valores de todos los elementos.
- Obtener el número de elementos.

- Contar el número de valores igual a cero (0).
- Calcular la suma de los pares.
- Contar el número de repeticiones dado un número.
- Modificar el valor de una posición por otro valor suministrado.
- Cambiar todo el contenido.
- Invertir el arreglo.
- Obtener el mayor valor.
- Obtener el menor valor.
- Obtener los primeros n elementos del arreglo.
- Obtener los últimos n elementos del arreglo.
- Indicar si es palíndromo.
- Indicar si un número se encuentra dentro del arreglo.
- Mostrar el contenido del arreglo.

Para conseguir el objetivo, se le suministra la clase principal y la clase de prueba, tenga en cuenta los nombres de los métodos, argumentos y tipos de datos para la codificación.

Cree y Compile la siguiente clase:

MiColeccion
- ArrayList números
<pre>MiColeccion() + void insertarAlFinal (int valor) + void insertarAlPrincipio (int valor) + void agregarAlFinal (List l) + void eliminar (int posicion) + void setValor (int posicion, int valor) + int getValor (int posición) + int getSumatoria () + double getPromedio () + int cuentaElementos () + int cuentaCeros() + int sumaPares () + int cuentaRepeticiones (int numero) + void reemplazarValor (int antiguoValor, int nuevoValor) + void reemplazarTodo (List a) + ArrayList invertir () + int getMayor () + int getMenor () + ArrayList getPrimeros (int cantidad) + ArrayList getUltimos (int cantidad)</pre>



```
+ boolean esPalindrome ( )  
+ boolean buscar (int numero)  
+ void mostrarContenido ( )
```

Descargar, compilar y ejecutar la clase TestMiColeccion.java, la cual contiene el siguiente código:

```
import java.util.*;  
public class TestMiColeccion{  
    public static void main(String arg[]){  
        System.out.println("Mi Coleccion 1");  
        MiColeccion mc1=new MiColeccion();  
        mc1.insertarAlFinal(1);  
        mc1.insertarAlFinal(2);  
        mc1.insertarAlFinal(3);  
        mc1.insertarAlPrincipio(4);  
        mc1.mostrarContenido();  
        ArrayList al=new ArrayList();  
        al.add(9);  
        al.add(10);  
        al.add(11);  
        mc1.agregarAlFinal(al);  
        mc1.mostrarContenido();  
        mc1.eliminar(0);  
        mc1.eliminar(5);  
        mc1.mostrarContenido();  
        mc1.setValor(0,0);  
        mc1.setValor(6,0);  
        mc1.mostrarContenido();  
        System.out.println("En la posicion 4 esta el "+mc1.getValor(4));  
        System.out.println("La sumatoria de sus elementos es:  
"+mc1.getSumatoria());  
        System.out.println("El promedio de sus elementos es:  
"+mc1.getPromedio());  
        System.out.println("En la coleccion hay "+mc1.cuentaElementos()+"  
elementos");  
        System.out.println("Y hay "+mc1.cuentaCeros()+ " ceros");  
        System.out.println("La sumatoria de los pares es: "+mc1.sumaPares());  
        System.out.println("El numero 2 esta "+mc1.cuentaRepeticiones(2)+ "  
veces");  
        mc1.reemplazarValor(0,9);  
        mc1.mostrarContenido();  
        mc1.reemplazarTodo(al);  
        mc1.mostrarContenido();  
        System.out.println(mc1.invertir());  
    }  
}
```



```
System.out.println("El mayor elemento es: "+mc1.getMayor());  
System.out.println("El menor elemento es: "+mc1.getMenor());  
mc1.insertarAlFinal(1);  
mc1.insertarAlFinal(2);  
mc1.insertarAlFinal(3);  
mc1.mostrarContenido();  
System.out.println(mc1.getPrimeros(2));  
System.out.println(mc1.getUltimos(2));  
mc1.mostrarContenido();  
System.out.println("Palindrome="+mc1.esPalindrome());  
System.out.println("Esta el 3 en la lista? "+mc1.buscar(3));  
}  
}
```

Al ejecutar la clase TestMiColeccion.java, esta deberá presentar el siguiente resultado si la clase MiColeccion.java desarrollada por usted fue codificada correctamente:

```
Mi Coleccion 1  
[4, 1, 2, 3]  
[4, 1, 2, 3, 9, 10, 11]  
[1, 2, 3, 9, 10]  
[0, 1, 2, 3, 9, 10, 0]  
En la posicion 4 esta el 9  
La sumatoria de sus elementos es: 25  
El promedio de sus elementos es: 3.5714285714285716  
En la coleccion hay 7 elementos  
Y hay 2 ceros  
La sumatoria de los pares es: 12  
El numero 2 esta 1 veces  
[9, 1, 2, 3, 9, 10, 9]  
[9, 10, 11]  
[11, 10, 9]  
El mayor elemento es: 11  
El menor elemento es: 9  
[9, 10, 11, 1, 2, 3]  
[9, 10]  
[2, 3]  
[9, 10, 11, 1, 2, 3]  
Palindrome=false  
Esta el 3 en la lista? true  
Press any key to continue..._
```

5. EVIDENCIAS

Carpeta comprimida con el nombre laboratorio6 y dentro de ella dos subcarpetas, una con los archivos requeridos para el desarrollo del ejercicio 1 y la otra subcarpeta con los archivos requeridos para el desarrollo del ejercicio 2.

LABORATORIO 6

Elementos del lenguaje de programación
y estructuras de datos en java

Desarrollador de contenido
Experto temático

Andrés Julián Valencia Osorio

Asesor Pedagógico

**Rafael Neftalí Lizcano Reyes
Claudia Milena Hernández Naranjo**

Productor Multimedia

Luis Gabriel Urueta Alvarez

Líder expertos temáticos

Ana Yaqueline Chavarro Parra

Líder línea de producción

Santiago Lozada Garcés



Atribución, no comercial, compartir igual

Este material puede ser distribuido, copiado y exhibido por terceros si se muestra en los créditos. No se puede obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.



Creative Commons



RECURSOS BIBLIOGRÁFICOS

Oracle (2013). Java 2 Platform Standard Edition 5.0 API Specification, consultada en junio de 2013 y disponible en:
<http://docs.oracle.com/javase/1.5.0/docs/api/>

