



Ejercicios – Lista Enlazada Simple

Ejercicio 1 – Crear un nodo

Escribe una clase `Nodo` que almacene un número entero y un puntero al siguiente nodo.

- Implementa un programa que cree tres nodos y los enlace manualmente.
 - Imprime la lista completa.
-

Ejercicio 2 – Insertar al inicio

Implementa el método `insertarInicio(int dato)` en la clase `ListaEnlazada`.

- Prueba insertando los valores: `10`, `20`, `30`.
 - Imprime la lista para verificar que el orden sea correcto.
-

Ejercicio 3 – Insertar al final

Agrega a la clase `ListaEnlazada` el método `insertarFinal(int dato)`.

- Inserta los valores `1`, `2`, `3`.
 - Imprime la lista y verifica que se agregan en orden.
-

Ejercicio 4 – Eliminar por valor

Implementa el método `eliminar(int valor)` que elimine el primer nodo que contenga ese valor.

- Prueba con la lista `[10 -> 20 -> 30 -> 40]` eliminando el `30`.
- Verifica el resultado: `[10 -> 20 -> 40]`.

Ejercicio 5 – Buscar un valor

Crea un método `buscar(int valor)` que recorra la lista y devuelva `true` si encuentra el nodo.

- Prueba con la lista `[5 -> 15 -> 25 -> 35]`.
- Busca el `25` (debe devolver `true`) y el `100` (debe devolver `false`).

Ejercicio 6 – Contar elementos

Implementa el método `contar()` que devuelva la cantidad de nodos en la lista.

- Para `[1 -> 2 -> 3 -> 4 -> 5]`, el resultado debe ser `5`.

Ejercicio 7 – Invertir la lista

Escribe un método `invertir()` que invierta el orden de los nodos en la lista.

- Ejemplo: `[10 -> 20 -> 30 -> 40]` debe transformarse en `[40 -> 30 -> 20 -> 10]`.

Ejercicio 8 – Insertar en posición

Crea un método `insertarEn(int pos, int valor)` que inserte un nodo en la posición indicada (0 = inicio).

- Ejemplo: en `[1 -> 2 -> 4]`, al insertar `3` en la posición 2, debe quedar `[1 -> 2 -> 3 -> 4]`.
-

Ejercicio 9 – Eliminar duplicados

Implementa un método `eliminarDuplicados()` que recorra la lista y elimine los nodos repetidos.

- Ejemplo: `[1 -> 2 -> 2 -> 3 -> 1]` debe quedar `[1 -> 2 -> 3]`.
-

Ejercicio 10 – Aplicación práctica (registro de alumnos)

Crea una lista enlazada que almacene alumnos con nombre y legajo.

- Métodos: `agregarAlumno(nombre, legajo)`, `buscarAlumno(legajo)`, `eliminarAlumno(legajo)`.
- Simula un registro de tres alumnos y prueba las operaciones.