

Ejercicio Adicional – Clase 4

Cree un nuevo proyecto en Netbeans y resuelva:

I) Representar dos tipos de urnas electrónicas: **electorales** (usada en elecciones donde se presentan varias listas) y de **referéndum** (usada para saber si la población está a favor/en contra de la aprobación de una ley).

- Cualquier **urna** se caracteriza por su número de urna, un contador de votos en blanco y la información de la **zona** a la que pertenece (localidad, partido, provincia).
- Una **urna electoral** es una urna que se caracteriza además por la cantidad de listas existentes y por mantener un contador de votos para cada lista (son “L” listas, y están numeradas desde 0).
- Una **urna de referéndum** es una urna que se caracteriza además por mantener un contador de votos a favor y un contador de votos en contra.

A) Genere las clases necesarias.

B) Implemente constructores. Las urnas electorales deben iniciarse a partir de un número de urna “U”, una cantidad de listas a “L”, una zona “Z”. Las urnas de referéndum deben iniciarse a partir de un número de urna “U” y una zona “Z”. Ambas urnas deben inicialmente poseer los contadores de votos en 0.

C) Implemente métodos para obtener/modificar (get/set) aquellos atributos que considere adecuado.

D) Incorpore los métodos listados a continuación:

i. Cualquier urna debe saber responder al mensaje **votarEnBlanco** que incrementa en 1 el contador de votos en blanco.

ii. Las urnas electorales deben responder a los mensajes:

- **validarNumeroDeLista:** recibe un nro. “N” y devuelve un boolean indicando si N es un nro. de lista válido.
- **votarPorLista:** recibe un nro. de lista válido “I” y debe incrementar en 1 el contador de votos de esa lista.
- **devolverVotosPorLista:** recibe un nro de lista válido “I” y devuelve la cantidad de votos para esa lista.

iii. Las urnas de referéndum deben responder a los mensajes:

- **votarAFavor:** incrementa en 1 el contador de votos a favor.
- **votarEnContra:** incrementa en 1 el contador de votos en contra.

iv. Ambos tipos de urnas deben saber responder al mensaje **calcularGanador** pero de manera diferente:

- Para las urnas electorales debe devolver el nro. de la lista ganadora. En caso de empate entre 2 o más listas, se elegirá una como ganadora – indistintamente.
- Para las urnas de referéndum debe devolver un nro. **0** si hubo empate, **-1** si ganó la opción En Contra, **1** si ganó la opción A Favor.

v. Ambos tipos de urnas deben saber responder al mensaje **calcularTotalVotos** pero de manera diferente:

- Para las urnas electorales debe devolver la suma de votos en blanco y votos a favor de cada lista.
- Para las urnas de referéndum debe devolver la suma de votos en blanco, a favor y en contra.

vi. Cualquier urna debe saber responder al mensaje **toString** que retorna un String del estilo:

“Urna: nro. Zona: loc-partido-prov Total Votos: cant. total Ganador: nro. de la opción ganadora”

II) Realice un programa que instancie una urna electoral para la mesa “203” con 5 listas y una urna de referéndum para la mesa “204”. Luego, lea DNIs de personas que llegan a votar a la mesa, hasta que se ingresa el DNI 0 (el cual no debe procesarse). Cada persona vota en ambas urnas de la siguiente manera:

- Para la urna electoral, la persona ingresa un nro. “N”: si “N” corresponde a una lista válida se debe votar por la lista “N” y en caso contrario se debe votar en blanco.
- Para la urna de referéndum, la persona ingresa un número “M”: si “M” es positivo se debe votar a favor, si es negativo se debe votar en contra, y si es 0 debe votar en blanco.

Al finalizar la votación, muestre en consola la representación String de cada urna.