

## **Enunciado do Exercício: Simulação de Urna Eletrônica**

Você deve criar um programa em Java que simule uma urna eletrônica simplificada, com leitura de votos pelo console.

### **Requisitos:**

1. **Candidatos Pré-Configurados:** A urna deve conter 5 candidatos fictícios, com nome e número pré-definidos. Sugestão de candidatos (inspirados em figuras da ciência e música):
  - 01 - Ada Lovelace
  - 02 - Alan Turing
  - 03 - Marie Curie
  - 04 - Albert Einstein
  - 05 - Ludwig van Beethoven
2. **Entrada de Votos:** O programa deve solicitar ao usuário a entrada de 10 votos, um por vez, pelo console. O usuário deve digitar o número do candidato.
3. **Validação de Votos:** O programa deve validar se o número digitado corresponde a um candidato válido (entre 01 e 05). Caso o número seja inválido, o programa deve exibir uma mensagem de erro e solicitar a entrada de um novo voto. Votos nulos (qualquer entrada que não seja um número entre 01 e 05) devem ser contabilizados separadamente.
4. **Apuração dos Votos:** O programa deve contabilizar os votos de cada candidato e os votos nulos.
5. **Exibição dos Resultados:** Ao final da votação (após receber 10 votos válidos), o programa deve exibir:
  - O número de votos recebidos por cada candidato.
  - O percentual de votos de cada candidato em relação ao total de votos válidos.
  - O número de votos nulos.
  - O nome do candidato vencedor (aquele com o maior número de votos válidos). Em caso de empate, informar que houve empate.

### **Sugestão de Estrutura de Classes:**

- Candidato:
  - Atributos: nome (String), numero (int), votos (int).
  - Construtor: Recebe nome e número.
  - Métodos: getNome(), getNumero(), getVotos(), incrementarVotos().
- UrnaEletronicaJava:

- Atributos: candidatos (um array ou ArrayList de objetos Candidato), votosNulos (int).
- Construtor: Inicializa os candidatos pré-configurados.
- Métodos: receberVoto(int numero), apurarResultados(), exibirResultados().
- Método main() para executar a simulação.

### **Exemplo de Interação com o Usuário:**

Bem-vindo à Urna Eletrônica!

Candidatos:

01 - Ada Lovelace  
02 - Alan Turing  
03 - Marie Curie  
04 - Albert Einstein  
05 - Ludwig van Beethoven

Digite o número do seu candidato: 02  
Digite o número do seu candidato: 01  
Digite o número do seu candidato: 06  
Digite o número do seu candidato: 03  
... (continua até 10 votos válidos)

Resultado da Votação:

Ada Lovelace: 3 votos (30.0%)  
Alan Turing: 2 votos (20.0%)  
Marie Curie: 4 votos (40.0%)  
Albert Einstein: 0 votos (0.0%)  
Ludwig van Beethoven: 1 voto (10.0%)  
Votos Nulos: 0

Vencedor: Marie Curie

### **Dicas e Observações:**

- Utilize Scanner para ler a entrada do usuário pelo console.
- Utilize estruturas de dados como arrays ou ArrayList para armazenar os candidatos.
- Implemente a lógica de validação de votos e apuração dos resultados de forma clara e organizada.
- Formate a saída dos resultados de forma legível, mostrando o número de votos, o percentual e o vencedor.

Este exercício aborda conceitos importantes de POO, como classes, objetos, atributos, métodos, construtores, encapsulamento e manipulação de coleções. Além disso, exercita a lógica de programação e a interação com o usuário via console. É uma excelente forma de consolidar o aprendizado da POO em Java.