**Today’s Agenda**

1. **Improving The ElectionResult Class**
2. **Defining The ElectionResultService Class**
3. **Defining The ElectionResultDTO Class**
4. **Defining The ElectionResultController Class**
5. **Testing The API**
6. **Improving The ElectionResult Class**
7. With **ElectionResult** class also we will do the same as we did with **Vote** class.
8. **That is instead** of **ignoring** the **winner** completely, we can **store only the winner’s ID** in the **response**.
9. **This can be done** by using **@JsonProperty** annotation which **allows us** to **return selected field** from an **entity**.



**Now** the **response** would be like:

**{**

**"id": 1,**

**"electionName": "Delhi Elections 2025",**

**"totalVotes": 1000,**

**"winnerId": 5**

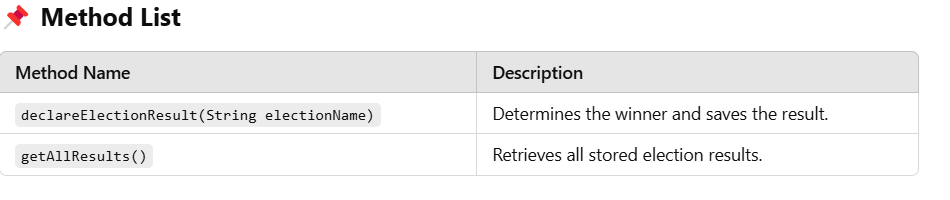
**}**

1. **Defining The ElectionResultService Class**

**📌 Explanation**

**Purpose:**

* **Determines** and **stores** the **election results**.
* **Finds** the **candidate** with the **highest votes**.
* Handles **edge cases** like **no candidates available**.

**\**

**Code For ElectionResultService Class**

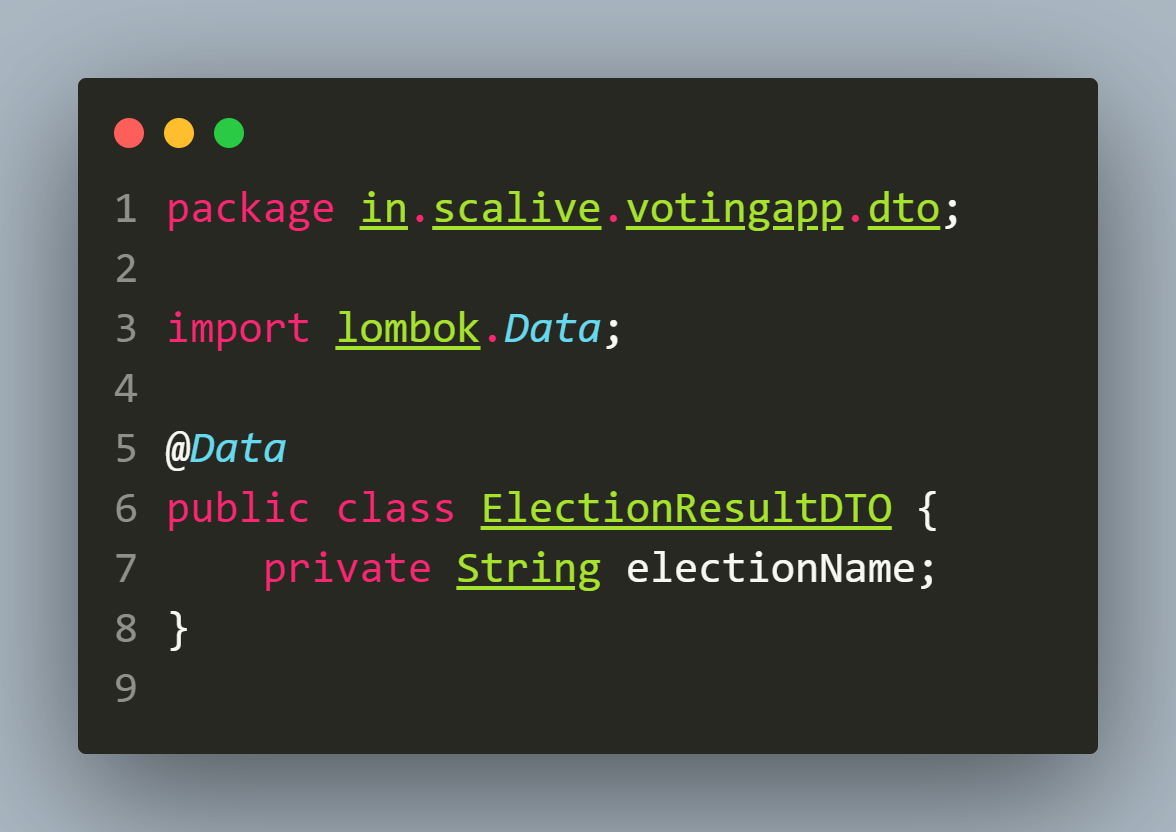


1. **Defining The ElectionResultDTO Class**

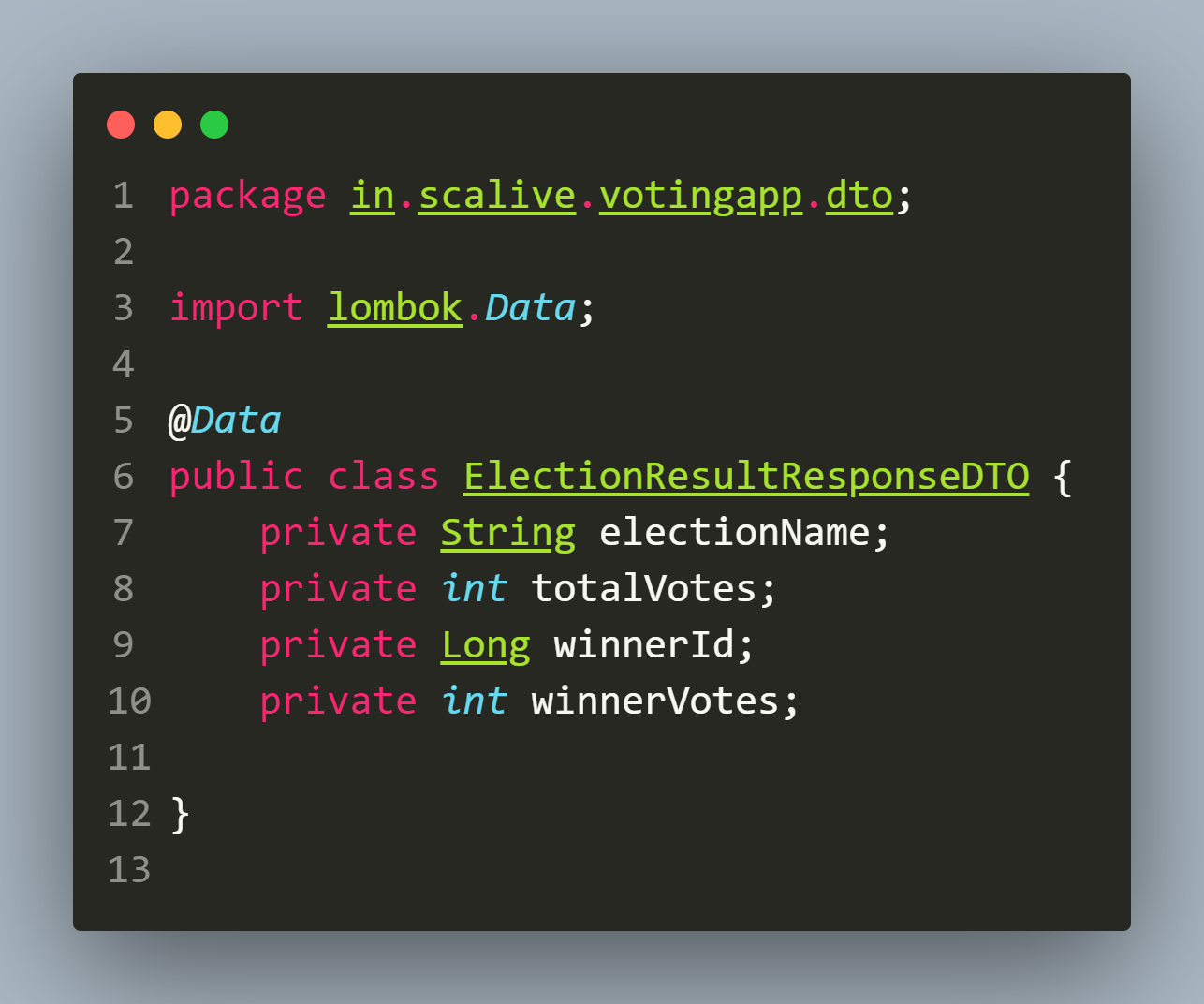
1. This **class** is **needed** because while **generating request** for **result declaration** we will have to **pass** the **election name** also.

2. Now since **result declaration**  will be a **POST request** so it is **recommended practice** to **pass this data** as **object** in **request body**.

3. This **object** will be of **ElectionResultDTO** class and will have just **2 fields** : **voterId** and **candidateId**.



1. **Defining The ElectionResultResponseDTO Class**



1. **Defining The ElectionResultController Class**

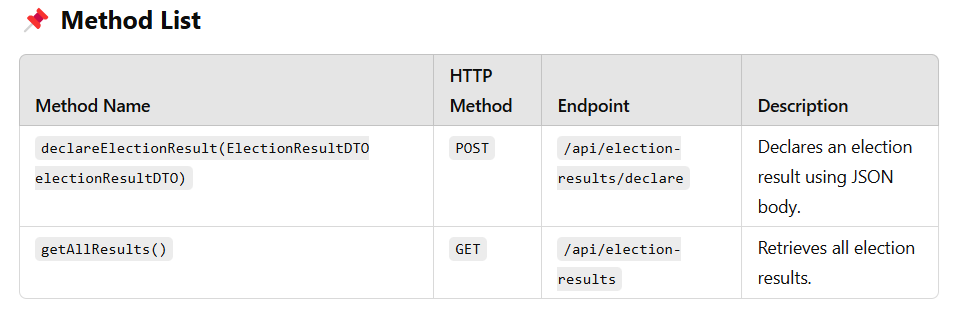
**📌 Explanation**

**Purpose:**

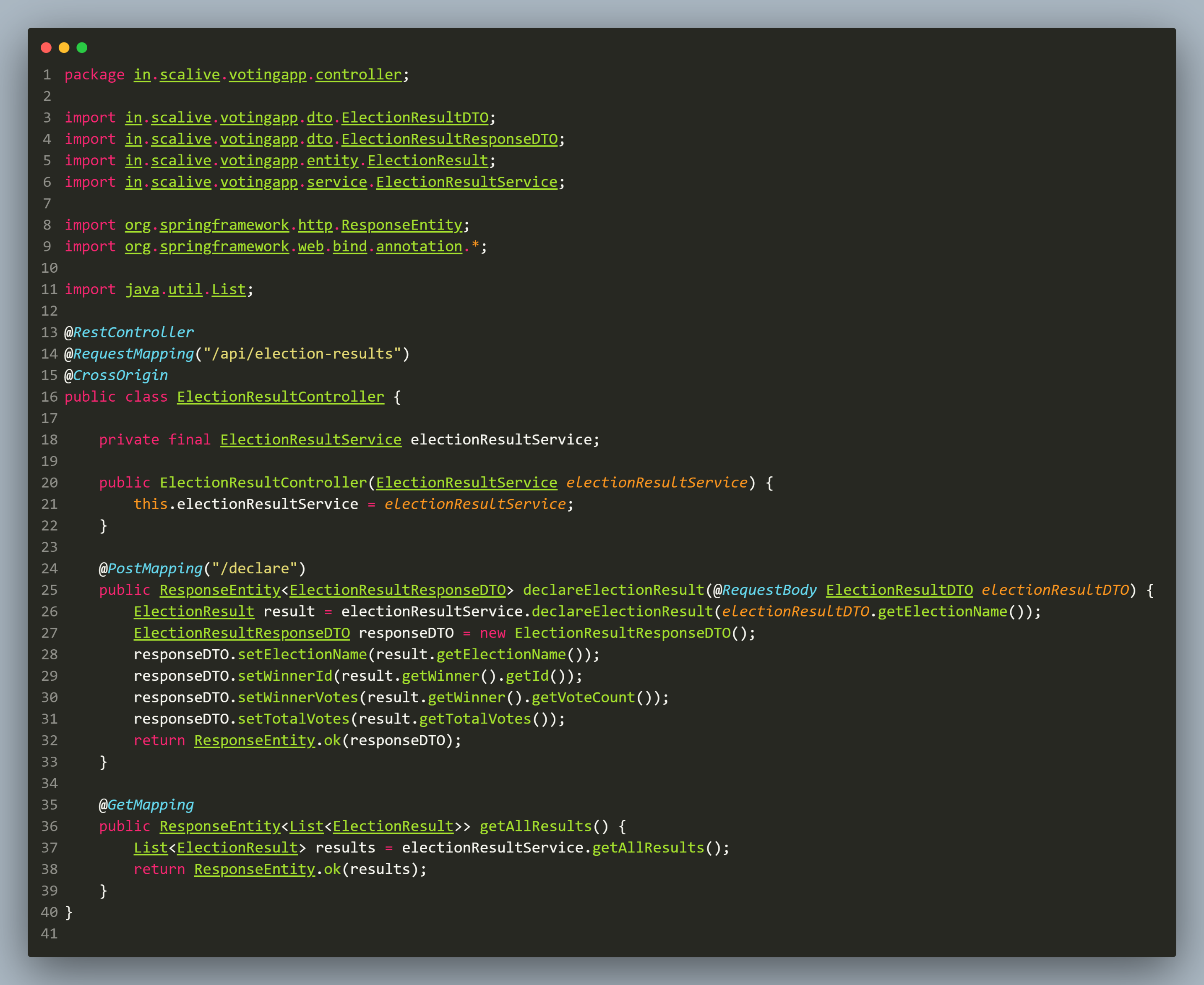
* **Exposes REST API endpoints** for **election result operations**.
* Allows **declaring election results** and **retrieving all stored results**.

**📌 Annotations Used**

* **@RestController** → **Marks this class** as a **REST API controller**.
* **@RequestMapping("/api/election-results")** → **Base path** for **election results APIs**.
* **@PostMapping, @GetMapping** → **Defines HTTP endpoints**.
* **@RequestBody** → Accepts **JSON input** in the **object** of **ElectionResultDTO.**

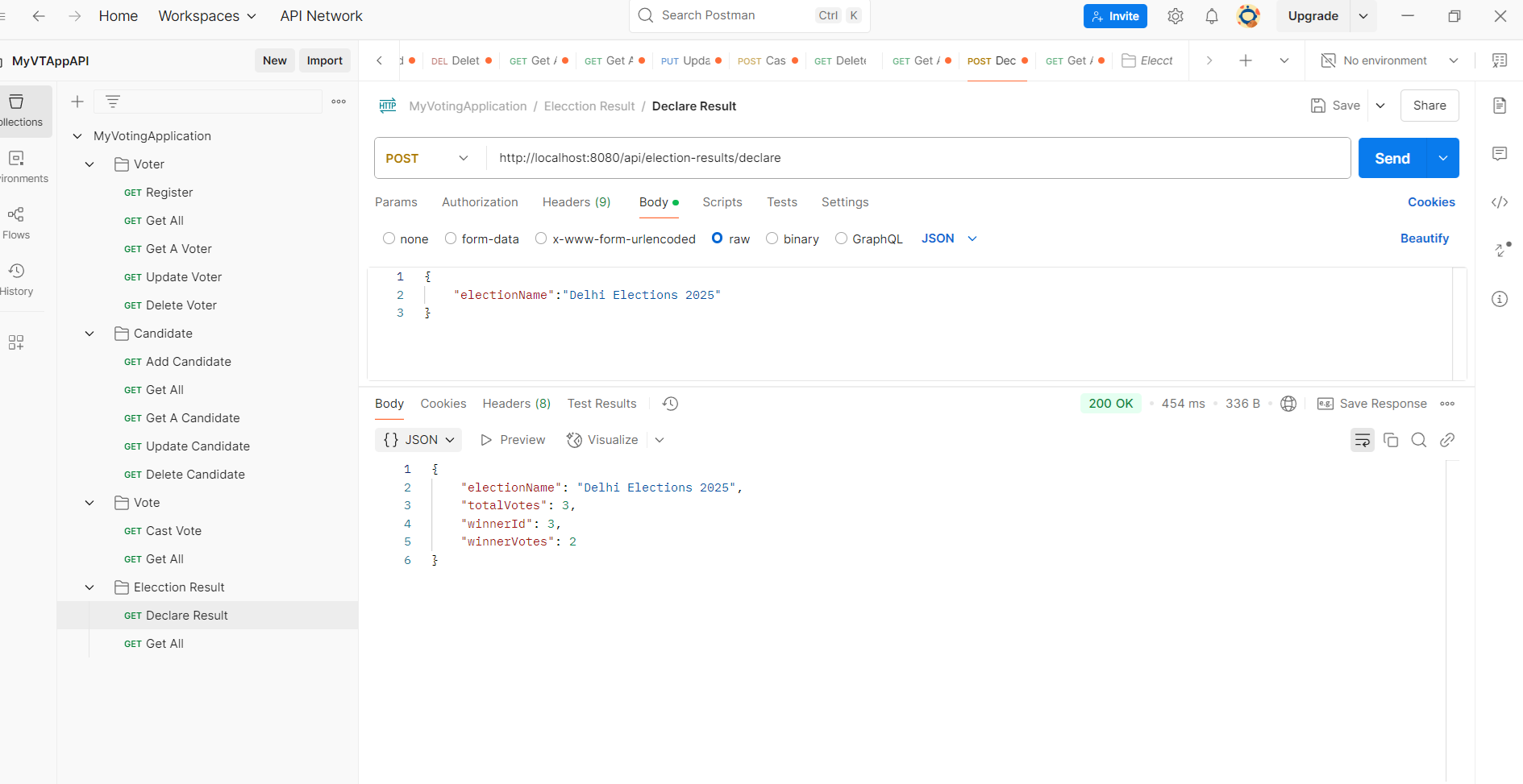


**Code For ElectionResultController Class**



**5.Testing The ElectionResult API**

1. **Declare Result:** [***http://localhost:8080/api/election-results/declare***](http://localhost:8080/api/election-results/declare)

******