

This is a Java enum named `Role`, which represents the roles that can be assigned to voters in the application. The enum is annotated with Lombok's `@RequiredArgsConstructor` annotation, which automatically generates a constructor with an argument for each enum value. Let's explain the purpose of this enum:

1. `enum Role`: This declares an enumeration type named `Role`.
2. `User` and `Admin`: These are the two enum constants defined within the `Role` enum. They represent the two possible roles that can be assigned to voters, namely "User" and "Admin".
3. `@RequiredArgsConstructor`: This Lombok annotation automatically generates a constructor with an argument for each enum constant. Since enums are inherently immutable and their instances are predefined, this annotation helps in creating the constructor with a private access modifier to prevent direct instantiation of enum objects from outside the enum class.

The purpose of this enum is to provide a simple and type-safe way to represent and manage user roles in the application. By using an enum, the roles are clearly defined and restricted to specific values ("User" and "Admin" in this case), making the code more robust and readable. For example, this enum can be used in the `Voter` entity class (as shown in the previous code) to store the role of each voter. It can also be used for authorization purposes, where certain actions are allowed only for users with specific roles.