

Express - 6

Controllers and Express Validator

What are controller methods ?

Controller methods are functions within controllers that handle various actions in response to HTTP requests.

They encapsulate the business logic for different operations related to the resource they manage

i.e we separate the logic from the routes and write them in separate files. We will move all the logic from the routes file to a controller file. We will then import these logics in the route file and use them.

Benefits of using controller methods

Benefits of Using Controller Methods

1. **Organization:** Keeps related operations grouped together, making the codebase easier to manage.
2. **Reusability:** Allows reuse of business logic across different parts of the application.
3. **Maintainability:** Simplifies updates and debugging by isolating logic within dedicated functions.
4. **Clarity:** Provides clear, named functions that describe the operations being performed.

Express Validator

What is express-validator ?

Express Validator is a set of middleware functions for validating and sanitizing user input in Express.js applications.

Role of Express Validator:

- **Validation:** Ensures that the data received from clients meets specified criteria.
- **Sanitization:** Cleans the data to prevent SQL injection, XSS attacks, and other vulnerabilities.
- **Error Handling:** Provides detailed error messages for invalid inputs.

Validating query params using express-validator

notesValidator.js

```
const notesValidation = {  
  limit: {  
    in: ['query'],  
    isInt: {  
      options: {min: 1},  
      errorMessage: "Limit must be an positive integer"  
    },  
    optional: true  
  },  
  
  sort: {  
    in: ['query'],  
    isIn: {  
      options: [['asc', 'desc']],  
      errorMessage: "Sort must have a value of asc or desc"  
    },  
    optional: true  
  },  
}  
  
export default notesValidation
```

Validating query params using express-validator

notesRouter.js

```
router.get('/notes', checkSchema(notesValidation) , getNotes);
```

notesController.js

```
function getNotes(req, res){  
  const error = validationResult(req);  
  
  if(!error.isEmpty()){  
    return res.status(400).json({message: error.array()});  
  }  
}
```