**Book API Documentation**

Base URL: /api/books

**Get All Books**

* **Endpoint**: GET /api/books
* **Description**: Retrieves a list of all books.
* **Response**:
  + **200 OK**: Returns an array of books.
* **Response Example**:

json

[

{

"id": 1,

"title": "Book Title",

"author": "Author Name",

"publicationYear": 2023,

"isbn": "ISBN123456"

},

...

]

**Get Book by ID**

* **Endpoint**: GET /api/books/{id}
* **Description**: Retrieves details of a specific book by its ID.
* **Parameters**:
  + **Path Variable**: id (Long) - ID of the book to retrieve.
* **Response**:
  + **200 OK**: Returns the details of the book.
  + **404 Not Found**: Returns an error if the book with the specified ID is not found.
* **Response Example**:

json

{

"resultCode": 200,

"resultDescription": "Success",

"message": "Book details retrieved successfully",

"data": {

"id": 1,

"title": "Book Title",

"author": "Author Name",

"publicationYear": 2023,

"isbn": "ISBN123456"

}

}

**Add a New Book**

* **Endpoint**: POST /api/books
* **Description**: Adds a new book to the collection.
* **Request Body**:

json

{

"title": "New Book Title",

"author": "New Author",

"publicationYear": 2023,

"isbn": "ISBN123456"

}

* **Response**:
  + **201 Created**: Returns the newly created book.
* **Response Example**:

json

{

"resultCode": 201,

"resultDescription": "Success",

"message": "Book created successfully",

"data": {

"id": 28,

"title": "New Book Title",

"author": "New Author",

"publicationYear": 2023,

"isbn": "ISBN123456"

}

}

**Update Book by ID**

* **Endpoint**: PUT /api/books/{id}
* **Description**: Updates details of an existing book by its ID.
* **Parameters**:
  + **Path Variable**: id (Long) - ID of the book to update.
* **Request Body**:

json

{

"title": "Updated Book Title",

"author": "Updated Author",

"publicationYear": 2023,

"isbn": "ISBN123456"

}

* **Response**:
  + **200 OK**: Returns the updated book details.
  + **404 Not Found**: Returns an error if the book with the specified ID is not found.
* **Response Example**:

json

{

"resultCode": 200,

"resultDescription": "Success",

"message": "Book updated successfully",

"data": {

"id": 1,

"title": "Updated Book Title",

"author": "Updated Author",

"publicationYear": 2023,

"isbn": "ISBN123456"

}

}

**Delete Book by ID**

* **Endpoint**: DELETE /api/books/{id}
* **Description**: Deletes a book by its ID.
* **Parameters**:
  + **Path Variable**: id (Long) - ID of the book to delete.
* **Response**:
  + **200 OK**: Indicates successful deletion with no data.
  + **404 Not Found**: Returns an error if the book with the specified ID is not found.
  + **500 Internal Server Error**: Returns an error if there was an issue deleting the book.
* **Response Example (404 Error)**:

json

{

"resultCode": 404,

"resultDescription": "Error",

"message": "Error deleting Book not found"

}