

Module 15

What to learn

Models and Data Transfer

- Creating models in Web API

- Data annotations and validations

- Model binding and validation

- Custom Validation in .NET Core

- Hands-on: Enhance the "Books" API with model validation.

- Hands-on Custom Validation

Practice Exercise

Practice 1

Repository Pattern

Practice 2

Creating Models in Web API

Design a model class named **Book** with properties: **Id** (int), **Title** (string), **Author** (string), **PublishedYear** (int), and **Genre** (string). Ensure that the **Id** is an auto-incrementing primary key.

Practice 3

Data Annotations and Validations

Enhance the **Book** model by adding data annotations for validation. For example, make the **Title** property required and ensure that **PublishedYear** is a valid year (greater than 0).

Practice 4

Model Binding and Validation

Implement model binding in a controller action that accepts a **Book** object. Ensure that the model is validated and return appropriate error messages if validation fails.

Practice 5

Enhancing the 'Books' API with Model Validation

Modify the existing **BooksController** to include model validation. If a **Book** object fails validation, return a 400 Bad Request response with the validation errors.

Practice 6

Creating a Custom Validation Attribute

Create a custom validation attribute named **ValidGenreAttribute** that checks if the **Genre** property of the **Book** model is one of the predefined genres (e.g., Fiction, Non-Fiction, Science, History). Apply this attribute to the **Genre** property.

Assignment Exercise

Assignment 1

Assignment: Building a Comprehensive 'Books' API

Functional Flow: Create a RESTful API for managing a collection of books. The API should support the following operations:

- GET all books
- GET a book by ID
- POST a new book
- PUT to update an existing book
- DELETE a book

Ensure that all operations validate the **Book** model using data annotations and return appropriate HTTP status codes.

Business Logic: Implement the following rules:

- The **Title** must be unique.
- The **PublishedYear** must be a valid year (greater than 0).
- The **Genre** must be one of the predefined genres.

Use in-memory storage for simplicity, and ensure that all outputs are displayed on the user interface (UI) only, not in the console.

Online Reference

No online Reference

.NET Core Web API

WEB API (old)

Authentication And Authorization (WEBAPI)(old)

FullStackDevelopment_With_Dotnet_AND_Angular