## **系统必备**

* [.NET Core SDK 2.2 或更高版本](https://www.microsoft.com/net/download/all)
* 已安装“ASP.NET 和 Web 开发”工作负载的 [Visual Studio 2017 版本 15.9 或更高版本](https://www.visualstudio.com/downloads/?utm_medium=microsoft&utm_source=docs.microsoft.com&utm_campaign=button+cta&utm_content=download+vs2017)
* [MongoDB](https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows/)

## **配置 MongoDB**

## **创建 ASP.NET Core Web API 项目**

Install-Package MongoDB.Driver -Version 2.7.2

## **添加模型**

using MongoDB.Bson;using MongoDB.Bson.Serialization.Attributes;

namespace BooksApi.Models

{

public class Book

{

public ObjectId Id { get; set; }

[BsonElement("Name")]

public string BookName { get; set; }

[BsonElement("Price")]

public decimal Price { get; set; }

[BsonElement("Category")]

public string Category { get; set; }

[BsonElement("Author")]

public string Author { get; set; }

}

}

## **添加 CRUD 操作类**

using System.Collections.Generic;

using System.Linq;

using BooksApi.Models;

using Microsoft.Extensions.Configuration;

using MongoDB.Bson;

using MongoDB.Driver;

namespace BooksApi.Services

{

public class BookService

{

private readonly IMongoCollection<Book> \_books;

public BookService(IConfiguration config)

{

var client = new MongoClient(config.GetConnectionString("BookstoreDb"));

var database = client.GetDatabase("BookstoreDb");

\_books = database.GetCollection<Book>("Books");

}

public List<Book> Get()

{

return \_books.Find(book => true).ToList();

}

public Book Get(string id)

{

var docId = new ObjectId(id);

return \_books.Find<Book>(book => book.Id == docId).FirstOrDefault();

}

public Book Create(Book book)

{

\_books.InsertOne(book);

return book;

}

public void Update(string id, Book bookIn)

{

var docId = new ObjectId(id);

\_books.ReplaceOne(book => book.Id == docId, bookIn);

}

public void Remove(Book bookIn)

{

\_books.DeleteOne(book => book.Id == bookIn.Id);

}

public void Remove(ObjectId id)

{

\_books.DeleteOne(book => book.Id == id);

}

}

}

将 MongoDB 连接字符串添加到 appsettings.json：

{

"ConnectionStrings": {

"BookstoreDb": "mongodb://localhost:27017" },

"Logging": {

"IncludeScopes": false,

"Debug": {

"LogLevel": {

"Default": "Warning"

}

},

"Console": {

"LogLevel": {

"Default": "Warning"

}

}

}

}

在 Startup.ConfigureServices 中，向依赖关系注入系统注册 BookService 类：

public void ConfigureServices(IServiceCollection services){

services.AddScoped<BookService>();

services.AddMvc().SetCompatibilityVersion(CompatibilityVersion.Version\_2\_2);

}

MongoClient – 读取执行数据库操作的服务器实例。 此类的构造函数提供了 MongoDB 连接字符串：

public BookService(IConfiguration config){

var client = new MongoClient(config.GetConnectionString("BookstoreDb"));

var database = client.GetDatabase("BookstoreDb");

\_books = database.GetCollection<Book>("Books");

}