

# 641: Advance Programming Skills

## Project 3

Chirag Kular

The project "LibrarySystem" has following Nuget packages dependencies:

- Npgsql(3.2.2)
- Dapper(1.50.2)
- Swashbuckle.AspNetCore(1.0.0)

ReadMe.txt:



ReadMe.txt

### Database Setup:

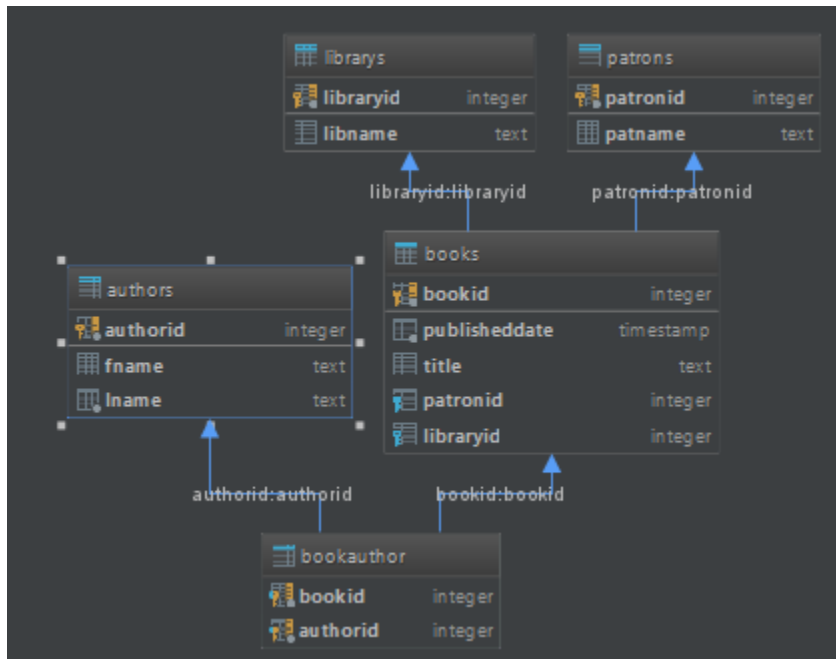
I am using Postgresql (9.6.2)

The table structure is defined in DDL.txt file and its contains the following tables:



DDL.txt

- authors
- books
- bookauthor
- librarys
- patrons



I have configured the connection string in appsettings.json as

```
"DbConn": {  
  "Name": "postgres",  
  "ConnectionString": "UserID=postgres;Password=chirag;Host=localhost;Port=5432;Database=postgres;Pooling=true;"  
}
```

In **Startup.cs** inside **ConfigureServices** method, added the following snippet:

```
//Database Configuration  
services.AddSingleton<IConfiguration>(Configuration);
```

### API documentation using Swagger:

Install the package “Swashbuckle.AspNetCore”

In **Startup.cs**

- Inside **ConfigureServices** method, added the following snippet:

```
// Register the Swagger generator, defining one or more Swagger documents  
services.AddSwaggerGen(c =>  
{  
  c.SwaggerDoc("v1", new Info { Title = "My API", Version = "v1" });  
});
```

- Inside **Configure** Method, added the following snippet:

```
app.UseSwagger();  
  
// Enable middleware to serve swagger-ui (HTML, JS, CSS etc.), specifying the Swagger JSON endpoint.  
app.UseSwaggerUI(c =>  
{  
  c.SwaggerEndpoint("/swagger/v1/swagger.json", "My API V1");  
});
```

### Controllers, Repositories and Interfaces:

All the interface, repositories and controller follow similar structures

- **BookController, BookProvider and IBookRepository**

```
public interface IBookRepository  
{  
  Book Add(Book book);  
  IEnumerable<Book> GetAll();  
  Book GetById(int id);  
  void Delete(Book book);  
  void Update(Book book);  
}
```

- **AuthorController, AuthorRepository and IAuthorRepository**

```
public interface IAuthorRepository
{
    Author Add(Author author);
    IEnumerable<Author> GetAll();
    Author GetById(int id);
    Author GetByName(string authorName);
    void Delete(Author author);
    void Update(Author author);
    void AddBooktoAuthor(Author author);
}
```

- **LibraryController, LibraryRepository and ILibraryRepository**

```
public interface ILibraryRepository
{
    Library Add(Library library);
    IEnumerable<Library> GetAll();
    Library GetById(int id);
    void Delete(Library library);
    void Update(Library library);
}
```

- **PatronController, PatronRepository and IPatronRepository**

```
public interface IPatronRepository
{
    Patron Add(Patron patron);
    IEnumerable<Patron> GetAll();
    Patron GetById(int id);
    void Delete(Patron patron);
    void Update(Patron patron);
    void Checkout(int patronid, int bookid);
    void BookReturn(int patronid, int bookid);
}
```

## API EndPoints:

Swagger Link: [http://localhost:\(portNumber\)/swagger/#/](http://localhost:(portNumber)/swagger/#/)

### 1. Author

a. -----Get all the Authors-----

**GET:** /Author

Request URL: <http://localhost:58177/Author>

Response Body:

```
[
  {
    "authorId": 1,
    "fName": "JRR",
    "lName": "Tolkien",
    "bookAuthors": []
  },
]
```

```

{
  "authorId": 2,
  "fName": "George",
  "lName": "Orwell",
  "bookAuthors": []
},
{
  "authorId": 5,
  "fName": "Harper",
  "lName": "Lee",
  "bookAuthors": [
    {
      "bookId": 7,
      "book": {
        "bookId": 7,
        "title": "The Da Vinci Code",
        "publishedDate": "2014-04-30T00:00:00",
        "bookAuthors": [],
        "patronId": 1
      },
      "authorId": 5,
      "author": null
    }
  ]
}
]

```

b. -----Get Author by specifying ID-----

**GET:** /Author/{id}

Parameters:

Parameter	Value
ID	1

Request URL: <http://localhost:58177/Author/1>

Response Body:

```

{
  "authorId": 1,
  "fName": "JRR",
  "lName": "Tolkien",
  "bookAuthors": []
}

```

c. -----Add Book to an Author-----

**PUT:** /Author/AddBooktoAuthor

Parameters:

Parameter	Value
Value	<pre> {   "authorId": 8,   "bookAuthors": [     {       "bookId": 4     }   ] } </pre>

	}
--	---

Request URL: <http://localhost:58177/Author/AddBooktoAuthor>

Response Body:

```
{
  "authorId": 8,
  "fName": "Suzanne",
  "lName": "Collins",
  "bookAuthors": [
    {
      "bookId": 4,
      "book": {
        "bookId": 4,
        "title": "Hobbit",
        "publishedDate": "2009-07-04T00:00:00",
        "bookAuthors": [],
        "patronId": 0,
        "libraryId": 2
      },
      "authorId": 8,
      "author": null
    }
  ]
}
```

Database Results:

	bookid	authorid
1	7	5
2	6	2
3	7	3
4	4	8

d. -----Get author by specifying Author name-----

GET: /Author/AuthorName/{authorName}

Parameters:

Parameter	Value
authorName	Da

Request URL: <http://localhost:58177/Author/AuthorName/Da>

Response Body:

```
{
  "authorId": 4,
  "fName": "Dan",
  "lName": "Brown",
  "bookAuthors": []
}
```

e. -----Update the author by id-----

**PUT:** /Author/{id}

**Parameters:**

Parameter	Value
Id	1
Value	{ "fName": "John", "lName": "Tolkien", }

**Request URL:** <http://localhost:58177/Author/1>

**Response Body:**

```
{  
  "messageBody": "Successfully Updated"  
}
```

**Database Results:**

	authorid	fname	lname
1	1	John	Tolkien

f. -----**Add Author**-----

**POST:** /Author

**Parameters:**

Parameter	Value
Value	{ "fName": " Suzanne", "lName": " Collins" }

**Request URL:** <http://localhost:58177/Author>

**Response Body:**

```
{  
  "authorId": 8,  
  "fName": "Suzanne",  
  "lName": "Collins",  
  "bookAuthors": []  
}
```

**Database Results:**

	authorid	fname	lname
1	2	George	Orwell
2	3	Jane	Austen
3	4	Dan	Brown
4	5	Harper	Lee
5	1	John	Tolkien
6	6	Scott	Fitzgerald
7	8	Suzanne	Collins

g. -----Delete Author-----

**DELETE:** /Author/{id}

Parameters:

Parameter	Value
id	6

Request URL: <http://localhost:58177/Author/6>

Response Body:

```
{
  "messageBody": "Successfully Deleted"
}
```

\*\*\*\*\*

## 2. Book

h. -----Get all the Books-----

**GET:** /Book

Request URL: <http://localhost:58177/Book>

Response Body:

```
[
  {
    "bookId": 4,
    "title": "Hobbit",
    "publishedDate": "2009-07-04T00:00:00",
    "bookAuthors": [],
    "patronId": 0
  },
  {
    "bookId": 7,
    "title": "The Da Vinci Code",
    "publishedDate": "2014-04-30T00:00:00",
    "bookAuthors": [
      {
        "bookId": 7,
        "book": null,
        "authorId": 5,
        "author": {
          "authorId": 5,
          "fName": "Harper",
          "lName": "Lee",
          "bookAuthors": []
        }
      }
    ]
  }
]
```

```

    }
  ],
  "patronId": 1
}]

```

i. -----Get Book by specifying ID-----

**GET**: /Book/{id}

Parameters:

Parameter	Value
ID	7

Request URL: <http://localhost:58177/Book/7>

Response Body:

```

{
  "bookId": 7,
  "title": "The Da Vinci Code",
  "publishedDate": "2014-04-30T00:00:00",
  "bookAuthors": [
    {
      "bookId": 7,
      "book": null,
      "authorId": 5,
      "author": {
        "authorId": 5,
        "fName": "Harper",
        "lName": "Lee",
        "bookAuthors": []
      }
    }
  ],
  "patronId": 1
}

```

j. -----Update the book by id-----

**PUT**: /Book /{id}

Parameters:

Parameter	Value
Id	7
Value	{         "title": "To Kill a Mockingbird",         "publishedDate": "2014-05-06T00:58:38.986Z"       }

Request URL: <http://localhost:58177/Book/7>

Response Body:

```

{

```



```
"messageBody": "Successfully Updated"
}
```

#### Database Results:

	bookid	publisheddate	title	patronid	libraryid
1	7	2014-05-06 00:58:38.986000	To Kill a Mockingbird	<null>	<null>

#### k. -----Add Book-----

**POST:** /Book

#### Parameters:

Parameter	Value
Value	{ "title": "Twilight", "publishedDate": "2017-05-06T01:14:37.869Z", }

**Request URL:** <http://localhost:58177/Book>

#### Response Body:

```
{
  "bookId": 8,
  "title": "Twilight",
  "publishedDate": "2017-05-06T01:14:37.869",
  "bookAuthors": [],
  "patronId": 0,
  "libraryId": 0
}
```

#### Database Results:

	bookid	publisheddate	title	patronid	libraryid
1	4	2009-07-04 00:00:00.000000	Hobbit	<null>	<null>
2	6	2009-07-04 00:00:00.000000	1984	<null>	<null>
3	7	2014-05-06 00:58:38.986000	To Kill a Mockingbird	<null>	<null>
4	8	2017-05-06 01:14:37.869000	Twilight	<null>	<null>

#### l. -----Delete Book-----

**DELETE:** /Book/{id}

Parameters:

Parameter	Value
id	8

Request URL: <http://localhost:58177/Book/6>

Response Body:

```
{
  "messageBody": "Successfully Deleted"
}
```

---

### 3. Library

a. -----Get all the Library's Details-----

**GET:** /Library

Request URL: <http://localhost:58177/Library>

Response Body:

```
[
  {
    "libraryId": 1,
    "libName": "Bavarian Library",
    "libBooks": [
      {
        "bookId": 8,
        "title": "Twilight",
        "publishedDate": "2017-05-06T01:14:37.869",
        "bookAuthors": [],
        "patronId": 0,
        "libraryId": 1
      }
    ]
  },
  {
    "libraryId": 2,
    "libName": "British Library",
    "libBooks": [
      {
        "bookId": 4,
        "title": "Hobbit",
        "publishedDate": "2009-07-04T00:00:00",
        "bookAuthors": [],
        "patronId": 0,
        "libraryId": 2
      }
    ]
  },
  {
    "libraryId": 3,
    "libName": "Wren Library",
    "libBooks": []
  }
]
```

```
}  
]
```

b. -----Get Library by specifying ID-----

**GET:** /Library/{id}

**Parameters:**

Parameter	Value
ID	2

**Request URL:** <http://localhost:58177/Library/2>

**Response Body:**

```
{  
  "libraryId": 2,  
  "libName": "British Library",  
  "libBooks": [  
    {  
      "bookId": 4,  
      "title": "Hobbit",  
      "publishedDate": "2009-07-04T00:00:00",  
      "bookAuthors": [],  
      "patronId": 0,  
      "libraryId": 2  
    }  
  ]  
}
```

c. -----Update the Library by id-----

**PUT:** /Library/{id}

**Parameters:**

Parameter	Value
Id	1
Value	{ "libName": "Henrietta Library" }

**Request URL:** <http://localhost:58177/Library/7>

**Response Body:**

```
{  
  "messageBody": "Successfully Updated"  
}
```

**Database Results:**

	libraryid	libname
1	2	British Library
2	3	Wren Library
3	4	Central Library
4	5	Wallace Library
5	1	Henrietta Library

d. -----Add Library-----

**POST:** /Library

Parameters:

Parameter	Value
Value	{ "libName": "Carlson Library" }

Request URL: <http://localhost:58177/Library>

Response Body:

```
{
  "libraryId": 6,
  "libName": "Carlson Library",
  "libBooks": []
}
```

Database Results:

	libraryid	libname
1	2	British Library
2	3	Wren Library
3	4	Central Library
4	5	Wallace Library
5	1	Henrietta Library
6	6	Carlson Library

e. -----Delete Library-----

**DELETE:** /Library/{id}

Parameters:

Parameter	Value
id	6

Request URL: <http://localhost:58177/Library/6>

Response Body:

```
{
```

```
"messageBody": "Successfully Deleted"
}
```

## 4. Patron

a. -----Get all the Patron's Details-----

GET: /Patron

Request URL: <http://localhost:58177/Patron>

Response Body:

```
[
  {
    "patronId": 1,
    "patName": "Dylan",
    "patronsBooks": [
      {
        "bookId": 8,
        "title": "Twilight",
        "publishedDate": "2017-05-06T01:14:37.869",
        "bookAuthors": [],
        "patronId": 1,
        "libraryId": 1
      }
    ]
  },
  {
    "patronId": 2,
    "patName": "Amparo\n",
    "patronsBooks": []
  }
]
```

b. -----Get Patron by specifying ID-----

GET: /Patron/{id}

Parameters:

Parameter	Value
ID	1

Request URL: <http://localhost:58177/Patron/1>

Response Body:

```
{
  "patronId": 1,
  "patName": "Dylan",
  "patronsBooks": [
    {
      "bookId": 8,
      "title": "Twilight",
      "publishedDate": "2017-05-06T01:14:37.869",
      "bookAuthors": [],
      "patronId": 1,
      "libraryId": 1
    }
  ]
}
```

c. -----Update the Patron by id-----

**PUT:** /Patron/{id}

Parameters:

Parameter	Value
Id	2
Value	{ "patName": "Bob", }

Request URL: <http://localhost:58177/Patron/2>

Response Body:

```
{  
  "messageBody": "Successfully Updated"  
}
```

Database Results:

	patronid	patname
1	1	Dylan
2	3	Cristal
3	4	Karina
4	5	Margurite
5	6	Amal
6	2	Bob

d. -----Add Patron-----

**POST:** /Patron

Parameters:

Parameter	Value
Value	{ "patName": "Alex", }

Request URL: <http://localhost:58177/Patron>

Response Body:

```
{  
  "patronId": 7,  
  "patName": "Alex",  
  "patronsBooks": []  
}
```

Database Results:

	patronid	patname
1	1	Dylan
2	3	Cristal
3	4	Karina
4	5	Margurite
5	6	Amal
6	2	Bob
7	7	Alex

e. -----Delete Patron-----

**DELETE:** /Patron/{id}

Parameters:

Parameter	Value
id	7

Request URL: <http://localhost:58177/Patron/7>

Response Body:

```
{
  "messageBody": "Successfully Deleted"
}
```

f. -----Patron Checkouts a book-----

**PUT:** /Patron /Checkout

Parameters:

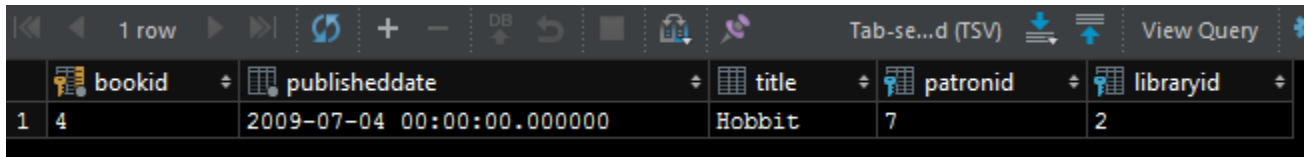
Parameter	Value
patron	{ "patronId": 7 }
bookid	4

Request URL: <http://localhost:58177/Patron/Checkout?bookid=4>

Response Body:

```
{
  "patronId": 7,
  "patName": "Alex",
  "patronsBooks": [
    {
      "bookId": 4,
      "title": "Hobbit",
      "publishedDate": "2009-07-04T00:00:00",
      "bookAuthors": [],
      "patronId": 7,
      "libraryId": 2
    }
  ]
}
```

## Database Results:



A screenshot of a database client interface. The top toolbar shows navigation and query execution icons. The main table has columns: bookid, publisheddate, title, patronid, and libraryid. The first row contains the values: 4, 2009-07-04 00:00:00.000000, Hobbit, 7, and 2.

	bookid	publisheddate	title	patronid	libraryid
1	4	2009-07-04 00:00:00.000000	Hobbit	7	2

## g. -----Patron Returns a book-----

**PUT:** /Patron /Return

### Parameters:

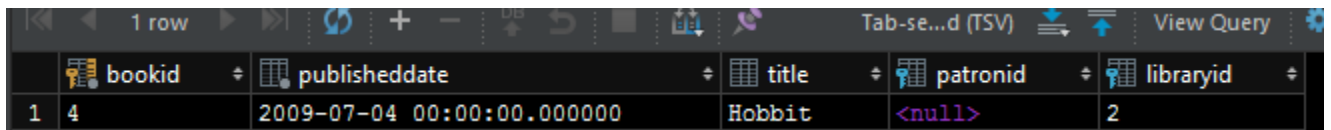
Parameter	Value
patron	{ "patronId": 7 }
bookid	4

**Request URL:** http://localhost:58177/Patron/Return?bookid=4

### Response Body:

```
{  
  "patronId": 7,  
  "patName": "Alex",  
  "patronsBooks": []  
}
```

## Database Results:



A screenshot of a database client interface. The top toolbar shows navigation and query execution icons. The main table has columns: bookid, publisheddate, title, patronid, and libraryid. The first row contains the values: 4, 2009-07-04 00:00:00.000000, Hobbit, <null>, and 2.

	bookid	publisheddate	title	patronid	libraryid
1	4	2009-07-04 00:00:00.000000	Hobbit	<null>	2

## References:

<https://docs.microsoft.com/en-us/aspnet/core/tutorials/web-api-help-pages-using-swagger>

<http://techbrij.com/asp-net-core-postgresql-dapper-crud>