

1 The Northwind database is a sample database provided by Microsoft for learning and demonstration purposes. It's a commonly used database in tutorials and examples. To access or download the Northwind database, you can follow these steps:

Download the instnwnd.sql script file.

Restore Database using SQL Server Management Studio (SSMS):

Open SQL Server Management Studio.

Connect to your SQL Server instance.

Open a new query window.

Open the instnwnd.sql script file that you downloaded.

Execute the script by clicking the "Execute" button or pressing F5.

This script will create the Northwind database along with its tables and data.

Open a new query window.

Copy and Run the following script to create the stored procedure

```
CREATE PROCEDURE pr_GetOrderSummary
```

```
    @StartDate DATE,
```

```
    @EndDate DATE,
```

```
    @EmployeeID INT = NULL,
```

```
    @CustomerID NVARCHAR(5) = NULL
```

```
AS
```

```
BEGIN
```

```
    SELECT
```

```
        (e.TitleOfCourtesy + ' ' + e.FirstName + ' ' + e.LastName) AS EmployeeFullName,
```

```
        s.CompanyName AS ShipperCompanyName,
```

```
        c.CompanyName AS CustomerCompanyName,
```

```
        COUNT(o.OrderID) AS NumberOfOrders,
```

```
        o.OrderDate AS [Date],
```

```
        SUM(o.Freight) AS TotalFreightCost,
```

```
        COUNT(DISTINCT od.ProductID) AS NumberOfDifferentProducts,
```

```
        SUM(od.UnitPrice * od.Quantity) AS TotalOrderValue
```

```
FROM Orders o
```

```
INNER JOIN Employees e ON o.EmployeeID = e.EmployeeID
```

```
INNER JOIN Shippers s ON o.ShipVia = s.ShipperID
```

```
INNER JOIN Customers c ON o.CustomerID = c.CustomerID
INNER JOIN [Order Details] od ON o.OrderID = od.OrderID
WHERE
    o.OrderDate BETWEEN @StartDate AND @EndDate
    AND (@EmployeeID IS NULL OR o.EmployeeID = @EmployeeID)
    AND (@CustomerID IS NULL OR o.CustomerID = @CustomerID)
GROUP BY
    o.OrderDate,
    e.TitleOfCourtesy, e.FirstName, e.LastName,
    s.CompanyName,
    c.CompanyName;
END
```

On a new query window run

```
EXEC pr_GetOrderSummary
    @StartDate = '1996-01-01',
    @EndDate = '1996-08-31',
    @EmployeeID = NULL,
    @CustomerID = NULL;
```

To test

2. API

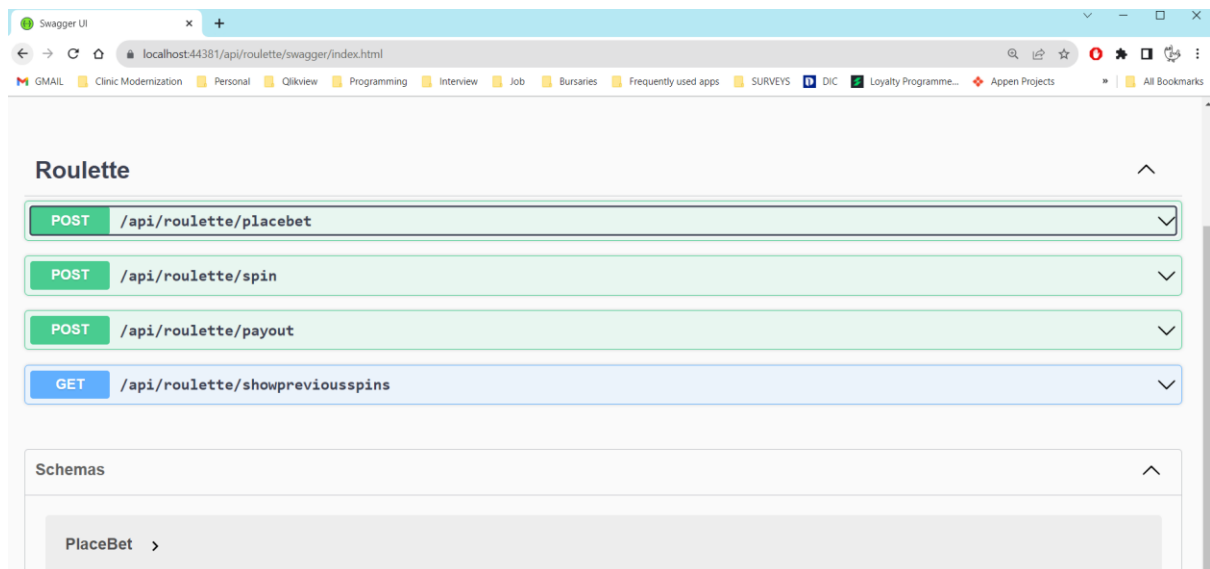
Download all the files in <https://github.com/Asamzilikazi/N2---SQL-C->

Run the Roulette.sln to open up the source code

Change the ConnectionString on the appsettings.json file to point to the location of your **roulette.db** that you have just downloaded

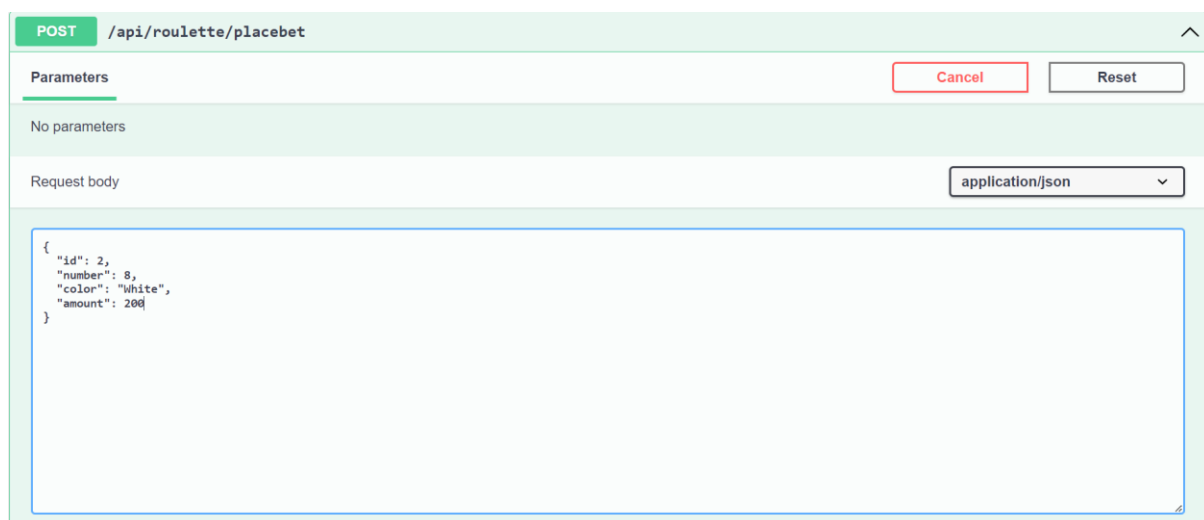
Run IIS (Landing will be Localhost:44381/)

Navigate to <https://localhost:44381/api/roulette/swagger> as shown below



Collapse a control eg. /api/roulette/placebet and click on Try it out

Populate it with a desired bet as shown below



Click on Execute

A successful response will be as shown below

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:44381/api/roulette/placebet' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 2,
    "number": 8,
    "color": "White",
    "amount": 200
  }'
```

Request URL

https://localhost:44381/api/roulette/placebet

Server response

Code	Details
200	<p>Response body</p> <pre>{ "success": true, "message": "Bet placed successfully" }</pre> <p>Response headers</p> <pre>content-type: application/json; charset=utf-8 date: Wed, 06 Dec 2023 11:31:47 GMT server: Microsoft-IIS/10.0 x-powered-by: ASP.NET</pre>

An unsuccessful response will be as shown below

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:44381/api/roulette/placebet' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 2,
    "number": 11,
    "color": "Red",
    "amount": 150
  }'
```

Request URL

https://localhost:44381/api/roulette/placebet

Server response

Code	Details
500 <i>Undocumented</i>	<p>Error: response status is 500</p> <p>Response body</p> <pre>An unexpected error occurred. Please try again later.</pre>

Bets are stored in the SQLite table as shown below

```
sqlite> .tables
Bets
sqlite> SELECT * FROM Bets;
1|5|Red|100
2|8|White|200
sqlite>
```