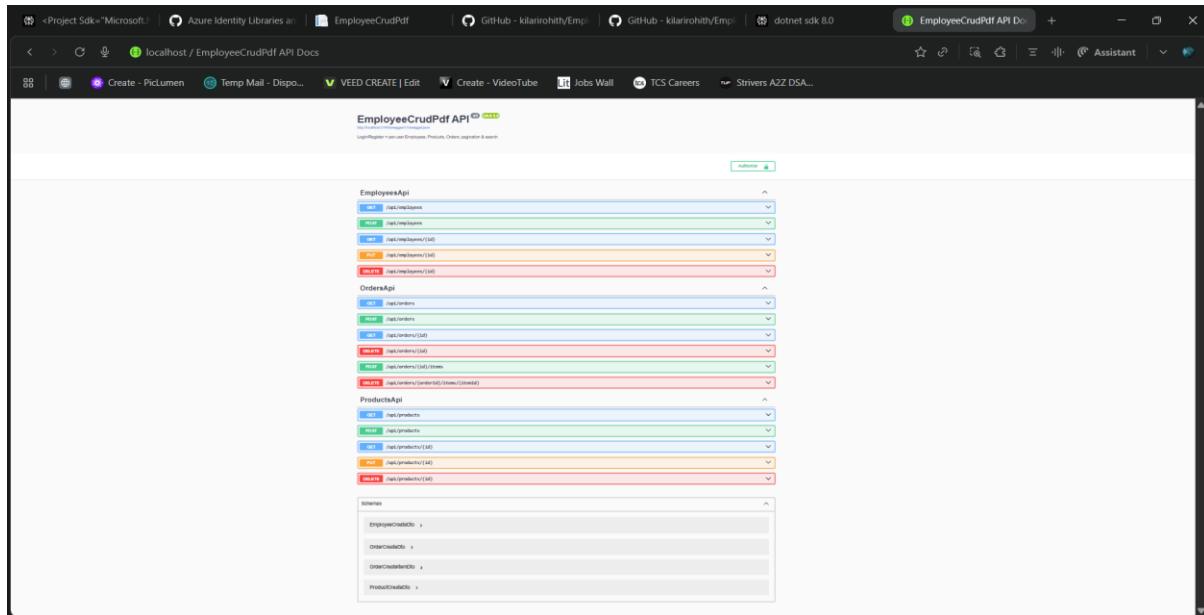


## EmployeeCrudPdf

Per-user Employees / Products / Orders demo using **ASP.NET Core MVC + Web API, Dapper (SQL Server)**, **Cookie auth for MVC, JWT (Bearer) for APIs**, **Rotativa** for PDF export, and **Swagger** for API exploration.

- MVC pages (Cookies): /Employees, /Products, /Orders
- APIs (JWT Bearer): /api/employees, /api/products, /api/orders
- Auth UI: /AccountMvc/Login, /AccountMvc/Register
- Swagger: /swagger



### 1. View all users

The screenshot shows the Object Explorer and a query editor in SSMS. In the Object Explorer, the database "EmployeeCrudDb" is selected, showing tables like "Users", "Orders", "Products", etc. In the query editor, a script is run to insert three new users:

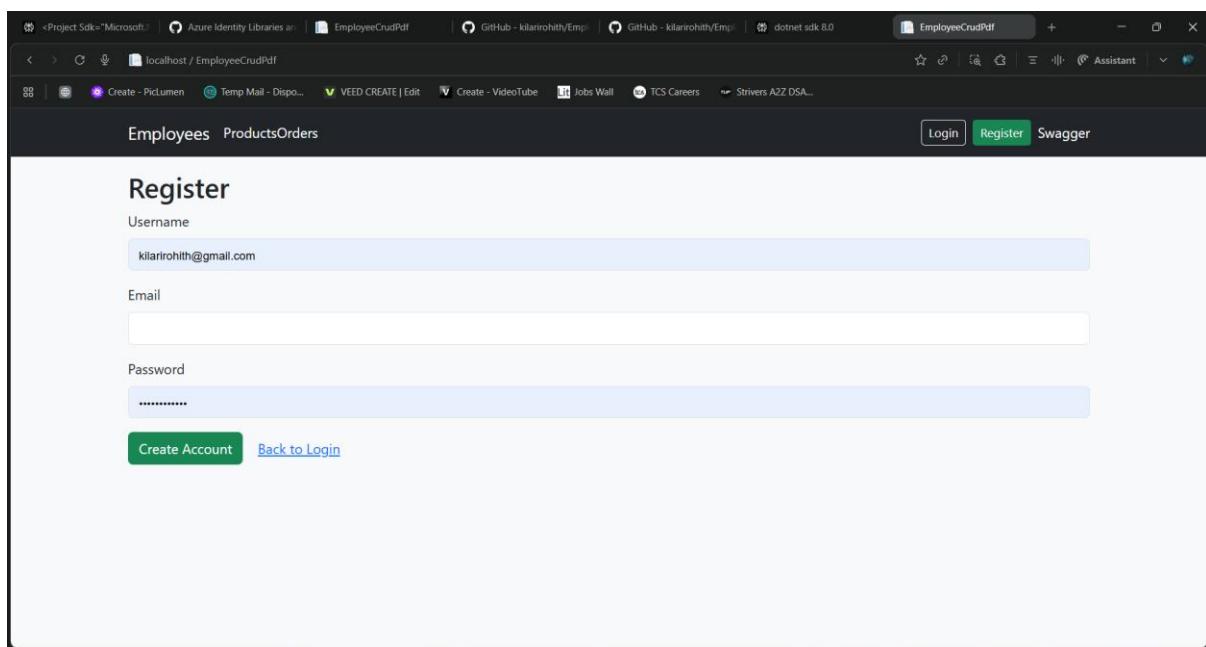
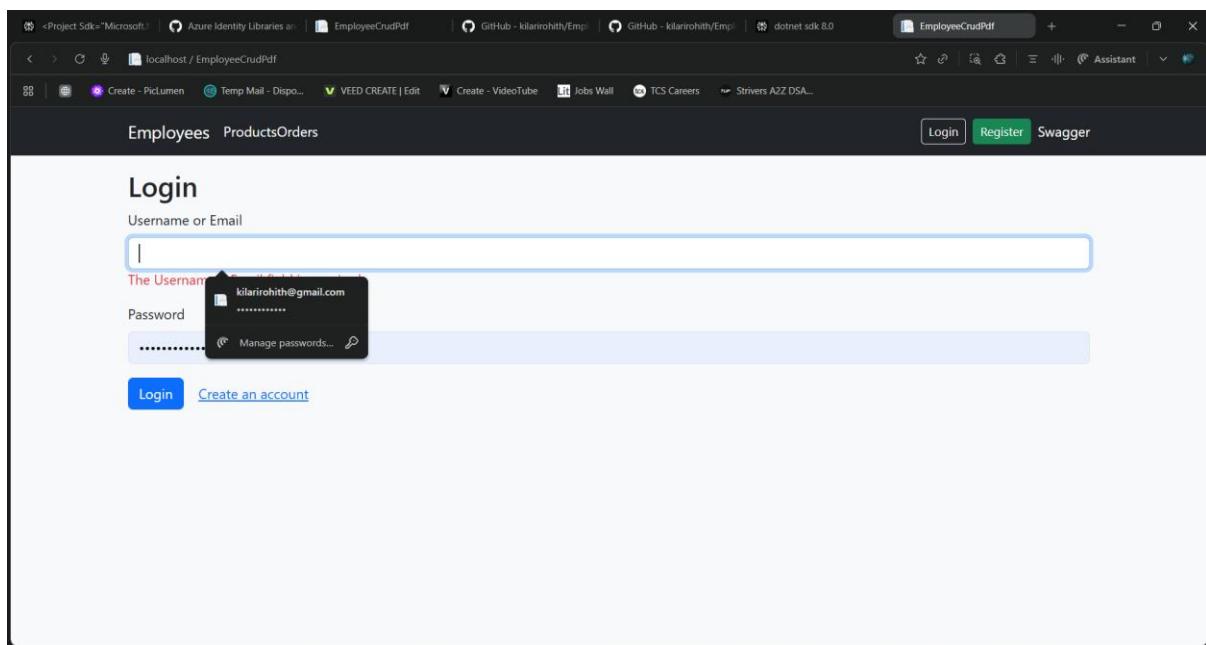
```
VALUES ('admin','admin@example.com','admin'); -- for demo only
END
USE EmployeeCrudDB;
GO

SELECT
    Id,
    Username,
    Email,
    CreatedAt
FROM dbo.Users
ORDER BY Id;
```

The results pane shows the following data:

	Id	Username	Email	CreatedAt
1	1	admin	admin@example.com	2025-10-31 17:30:21.4079207
2	2	kilarirohith	kilarirohith@gmail.com	2025-11-01 03:59:40.5914849
3	3	hemanth	hemanth@gmail.com	2025-11-01 04:56:19.4768325

A message at the bottom of the screen says "Query executed successfully."



## 2. View all products (with user info)

- GET /api/products?q=&page=1&pageSize=10
- POST /api/products { name, price, category }
- GET /api/products/{id} PUT /api/products/{id} { name, price, category }
- PUT /api/products/{id} { name, price, category }
- DELETE /api/products/{id}

```

bsykl0j.sq_MIS0386 (62)*
JOIN dbo.Users AS u ON e.user_id = u.id
ORDER BY e.id DESC;

SELECT
    p.id AS ProductId,
    p.name AS ProductName,
    p.category,
    p.price,
    u.Username AS CreatedByUser
FROM dbo.products AS p
JOIN dbo.Users AS u ON p.user_id = u.Id
ORDER BY p.id DESC;

```

ProductID	ProductName	Category	Price	CreatedByUser
1	bricks	1	30.00	hemanth
2	detergents	1	68.00	hemanth
3	pringle	1	90.00	hemanth
4	bottle	1	125.00	hemanth
5	charger	1	1.00	hemanth
6	Tv	1	500000.00	hemanth
7	Ac	1	450000.00	hemanth
8	Sprite	1	55.00	hemanth
9	Soap	1	10.00	hemanth
10	Oil	1	130.00	hemanth
11	Rice	1	53.00	hemanth
12	Rice	2	100000.00	kilaroithi

Query executed successfully.

ID	Name	Price	Category	Action
12	bricks	₹ 30.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
11	detergents	₹ 68.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
10	pringle	₹ 90.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
9	bottle	₹ 125.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
8	charger	₹ 1.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
7	Tv	₹ 5,000,00.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
6	Ac	₹ 4,50,000.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
5	Sprite	₹ 55.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
4	Soap	₹ 10.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
3	Oil	₹ 130.00	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>

1 2

### 3. View all order items (each product per order)

- GET /api/orders/{id}
- POST /api/orders/{id}/items { "productId": 5, "qty": 2, "price": 0 }
- DELETE /api/orders/{id}
- DELETE /api/orders/{orderId}/items/{itemId}

SQLEditor

```

File Edit View Query Git Project Tools Extensions Window Help Search Solution1
localhost EmployeeCrudDb Execute
bcskyk0jJ.sq...MS10386 (62)* = X
136     u.username AS CreatedByUser
137     FROM dbo.orders AS o
138     JOIN dbo.Users AS u ON o.user_id = u.Id
139     ORDER BY o.id DESC;
140
141
142     SELECT
143         o.id AS OrderId,
144         o.order_no AS OrderNumber,
145         p.name AS ProductName,
146         oi.qty AS Quantity,
147         oi.price AS UnitPrice,
148         (oi.qty * oi.price) AS LineTotal,
149         u.Username AS CreatedByUser
150     FROM dbo.order_items AS oi
151     JOIN dbo.orders AS o ON oi.order_id = o.id
152     JOIN dbo.products AS p ON oi.product_id = p.id
153     JOIN dbo.Users AS u ON oi.user_id = u.Id
154     ORDER BY o.id DESC, oi.id;

```

Object Explorer

Results Messages

OrderItemId	OrderNumber	ProductName	Quantity	UnitPrice	LineTotal	CreatedByUser
1	6	bricks	1	30.00	30.00	hemanth
2	3	Soap	1	10.00	10.00	hemanth
3	4	Oil	1	130.00	130.00	hemanth
4	5	Rice	1	53.00	53.00	hemanth
5	1	Rice	1	1000.00	1000.00	kilarothin
6	2	Rice	1	1000.00	1000.00	kilarothin

Query executed successfully.

EmployeeCrudPtf

localhost / EmployeeCrudPtf

Employees ProductsOrders

Hello, hemanth Logout Swagger

Add to Order

Product: bricks

Qty: 1

Add to existing order (optional):

- Create New Order --
- Create New Order --
- 2 - ORD-20251101050738-9fa142

Add Cancel

Employees ProductsOrders

Hello, hemanth Logout Swagger

Order ORD-20251101050738-9fa142

Created 01-11-2025 10:37:38 AM

#	Product	Category	Qty	Price	Line Total
3	Soap (4)	1	1	₹ 10.00	₹ 10.00
4	Oil (3)	1	1	₹ 130.00	₹ 130.00
5	Rice (2)	1	1	₹ 53.00	₹ 53.00
				<b>TOTAL</b>	<b>₹ 193.00</b>

#### 4. View all orders (with user info)

```

SELECT
    o.id AS OrderId,
    o.order_no AS OrderNumber,
    u.Username AS CreatedByUser,
    o.created_at AS CreatedAt,
    SUM(oi.qty * oi.price) AS TotalAmount,
    COUNT(oi.id) AS TotalItems
FROM dbo.orders AS o
LEFT JOIN dbo.order_items AS oi ON o.id = oi.order_id
JOIN dbo.Users AS u ON o.user_id = u.Id
GROUP BY o.id, o.order_no, o.created_at, u.Username
ORDER BY o.id DESC;
  
```

OrderId	OrderNumber	CreatedByUser	CreatedAt	TotalAmount	TotalItems
1	ORD-202511010503854-8d004c	hemanth	2025-11-01 09:38:54.4109933	30.00	1
2	ORD-20251101050738-9fa142	hemanth	2025-11-01 05:07:38.6470390	193.00	3
3	ORD-20251101045200-55dcf7	hemanth	2025-11-01 04:52:00.6267838	2000.00	2

## 5. View all employees (with user info)

### Employees

- GET /api/employees?q=&page=1&pageSize=10 → PagedList<EmployeeReadDto>
- POST /api/employees { name, department, email, salary }
- GET /api/employees/{id}
- PUT /api/employees/{id} { name, department, email, salary }
- DELETE /api/employees/{id}

EmployeeId	EmployeeName	department	EmployeeEmail	salary	CreatedByUser
1	kiran	ceo	kilarrohit@gmail.com	795764.00	hemanth
2	rahul	Electrical	rahul@gmail.com	20000.00	hemanth
3	rupesh	biology	rupesq23432@gmail.com	1000000.00	hemanth
4	jashwanth	software tester	jashwanth@gmail.com	232432.00	hemanth
5	sunanda	English	sunanda@gmail.com	29940.00	hemanth
6	ramesh	ComputerScience	Ramesh@gmail.com	2636.00	hemanth
7	valence	Information Mch	valence@gmail.com	120000.00	hemanth
8	B Lohith	Biology	Lohith@gmail.com	564777.00	hemanth
9	sunu	ceo	jdcok@gmail.com	123434.00	hemanth
10	K Hemanth	Electrical	Hemanth@gmail.com	37578.00	hemanth
11	KiranRohit	HoD	kilarrohit@gmail.com	150000.00	hemanth
12	ramesh	Physics	Ramesqa@gmail.com	120000.00	hemanth
13	charan	Chemistry	charan@gmail.com	150000.00	kilarrohit
14	chandru	agriculture	chandru@gmail.com	200000.00	kilarrohit
15	K Hemanth	Biology	jdcok@gmail.com	20000.00	kilarrohit
16	sunu	Physics	seenu@gmail.com	200000.00	kilarrohit
17	K Hemanth	Physics	Hemanth@gmail.com	100000.00	kilarrohit

Employees				
Search name/department				Search
<b>Id</b>	<b>Name</b>	<b>Department</b>	<b>Email</b>	<b>Salary</b>
18	rahul	Electrical	rahul@gmail.com	₹ 20,000.00
17	rupeesh	biology	rupesg2343@gmail.com	₹ 10,00,000.00
16	jashwanth	software tester	jashwanth@gmail.com	₹ 2,32,432.00
15	sunanda	English	sunanda@gmail.com	₹ 2,99,040.00
14	ramesh	ComputerScience	Ramesroja@gmail.com	₹ 43,636.00
13	rakesh	integrated M.tech	rakesh@gmail.com	₹ 1,20,000.00
12	B Lohith	Biology	Lohith@gmail.com	₹ 5,64,777.00
11	senu	ceo	jdscjk@gmail.com	₹ 1,23,434.00
10	K Hemanth	Electrical	Hemanth@gmail.com	₹ 3,75,776.00
9	Kilarirohith	HoD	kilarirohith@gmail.com	₹ 1,50,000.00

1 2

Employees				
Search name/department				Search
<b>Id</b>	<b>Name</b>	<b>Department</b>	<b>Email</b>	<b>Salary</b>
19	kilari	ceo	kilarirohith@gmail.com	₹ 7,65,764.00
9	Kilarirohith	HoD	kilarirohith@gmail.com	₹ 1,50,000.00