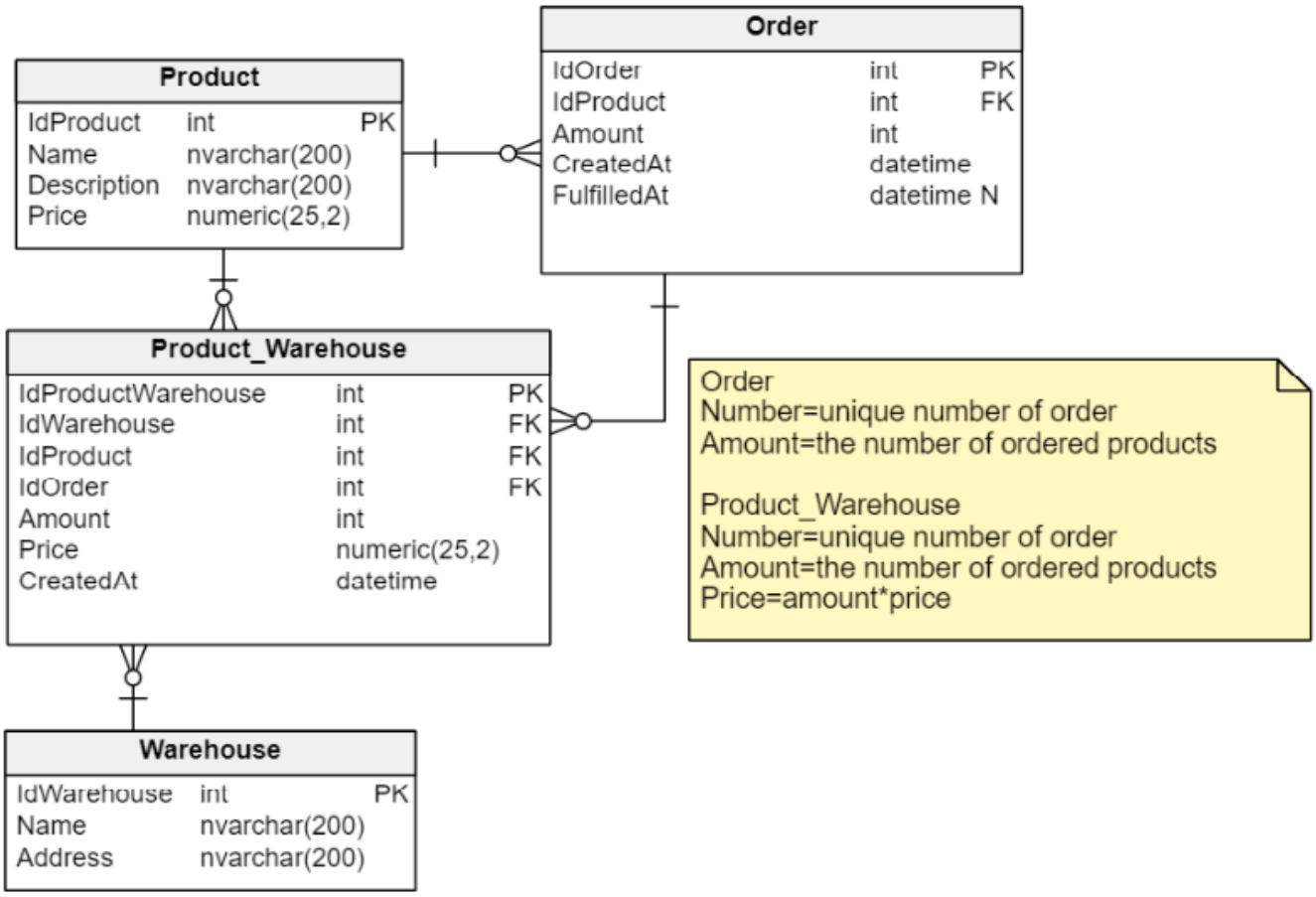

In the course of this tutorial, we will again use the SqlConnection and SqlCommand classes. This time around, the logic involved in interacting with our database will be a bit more complex. We create an application for a company that manages the warehouse. The database we use is presented below. In addition, in the create.sql file you will find a script that creates tables and fills them with data.

Task 1

Create WarehouseController and endpoint which will accept the data in the following format.

```
{
    "IdProduct": 1,
    "IdWarehouse": 2,
    "Amount": 20,
    "CreatedAt": "2012-04-23T18:25:43.511Z"
}
```

All the fields are required.
Then implement the following scenario.

Main scenario:

1. We check if the product with the given id exists. Then we check if the warehouse with the given id exists. The amount value passed in the request should be greater than 0.
2. We can add a product to the warehouse only if there is a product purchase order in the Order table. Therefore, we check if there is a record in the Order table with IdProduct and Amount that matches our request. The CreatedAt of the order should be lower than the CreatedAt in the request.
3. We check whether this order has been completed by any chance. We check if there is no row with the given IdOrder in the Product_Warehouse table.
4. We update the FullfilledAt column of the order with the current date and time. (UPDATE)
5. We insert a record into the Product_Warehouse table. The Price column should corresponds to the price of the product multiplied by amount value from our request. Moreover, we insert the CreatedAt value according to the current time. (INSERT)
6. As a result of the operation, we return the value of the primary key generated for the record inserted into the Product_Warehouse table.

Task 2*
Then add second endpoint to WarhouseController and try to implement the same logic using stored procedure. In proc.sql you will find a stored procedure prepared by the database administrator.
Check work of the database administrator and make sure that the stored procedure is correct (it may contain errors).
Then execute it from your endpoint.

Remember:

- in case of any error return a proper HTTP code