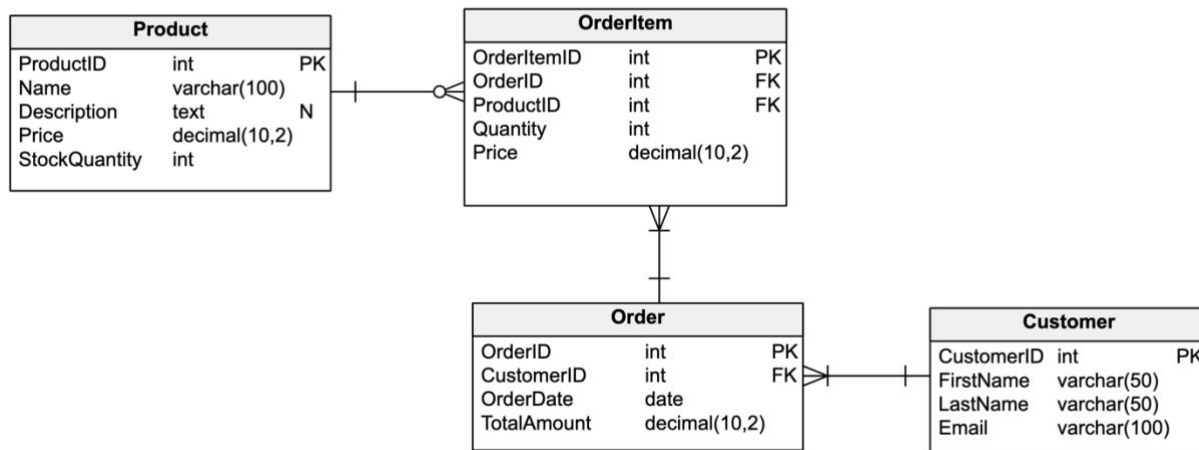


## Test 1

In this task you have to design a web API application that will support the database presented in the diagram. We create an e-commerce application.

The script to create database can be found in the file **script.sql**



1. **Retrieve Out-of-Stock Products:** Create an endpoint that retrieves a list of products that are currently out of stock. These are products with a stock quantity of zero. The endpoint should return the product name, description, and the last time it was in stock (i.e., the order date of the latest order containing this product). If the product has never been ordered, indicate this in the response. The endpoint should accept an optional parameter which will sort the list by product name or last order date (product name is a default value).

Example of the response:

```
1 [
2   {
3     "product_name": "Tablet",
4     "description": "High-performance tablet with HD display",
5     "last_in_stock_date": "2024-03-15"
6   },
7   {
8     "product_name": "Smartwatch",
9     "description": "Fitness tracker with heart rate monitor",
10    "last_in_stock_date": "2024-04-20"
11  }
12 ]
```

2. **Delete order.** Create an endpoint that allows an administrator to delete a order from the database. The endpoint should accept the OrderId as a parameter. In case if has any order items, then we also want to delete the data about those items. Don't forget that endpoint should use the transaction to maintain database consistency.

Remember about:

- a. Properly name the controller and variables
- b. Remember to maintain proper code structure.
- c. Remember to inject dependencies.
- d. Remember to handle errors.
- e. The date format provided in the requests may be different from the one presented in the examples.

### Scoring

Each endpoint is calculated at **10 points**. That is, **20 points** can be earned for the whole.

Points may be deducted for:

- Incorrect or illegible variable names: **-2 pts**
- Incorrect HTTP codes: **-4 pts**
- Failure to separate database communication into a different file/service/repository: **-10 pts**
- No dependency injection: **-8 pts**

### Notes

- A program that does not compile - **0 pts**
- A program that is plagiarized – **2 as the final grade for the subject**