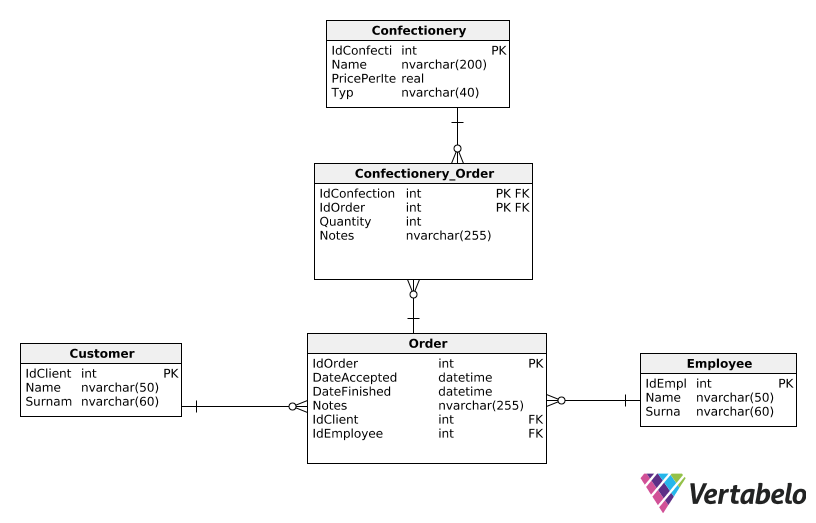
Examplary test from APBD (EF)

Tasks should be completed in the Web API project using the ORM - EntityFramework in the version designed for < .NET Core 3.0.

Task 1.

Design an endpoint that will return a list of orders placed by a customer with the given name. If the customer's name is not sent, a list containing orders of all customers should be returned. The result list does not have to contain information about the employee who was responsible for accepting the order or the customer`s personal data. Instead, it must consider what was included in the order (which confectionery was selected for it). Remember to return the correct error codes.

Example of endpoint structure:

GET: /api/orders

Task 2.

Design an endpoint that will allow you to add a new order. It should receive basic information needed to add the order and a collection of confectionery products that should be included in it. If a product has been provided that is not in the database, stop processing the request and return the appropriate error code. The customer ID making the request should be given as part of the URL segment. The whole task should be completed in one transaction.

Example of endpoint structure:

GET: /api/clients/1/orders

Body:

{

"DateAccepted": "2020-06-01",

"Notes": "Please, prepare for next friday",

"Confectionery": [

{

"Quantity": "1",

"Name": "Cake for birthday",

"Notes": "Write Happy birthday on the cake"

}

]

}

**Pay attention to:**

• Correctly name variables, controllers, services etc.

• Remember to maintain an appropriate code structure (separate controllers, communication with the DB in the service etc.).

• Remember about DI.

• Remember about proper error handling.