

/expired\_invoices

1. IInvoices issued after their due date. Return all attributes.

/wrong\_date\_invoices

1. Invoices that were issued before the date in which the order they refer to was placed.  
   Return the ID of the invoice, the date it was issued, the ID of the order associated with it  
   and the date the order was placed.

/orders\_without\_details

1. Orders that do not have a detail and were placed before 6 September 2016. Return all  
   attributes.

/customers\_without\_orders

1. Customers who have not placed any orders in 2016. Return all attributes.

/customers\_last\_orders

1. ID and name of customers and the date of their last order. For customers who did not  
   place any orders, no rows must be returned. For each customer who placed more than  
   one order on the date of their most recent order, only one row must be returned.

/overpaid\_invoices

1. Invoices that have been overpaid. Observe that there may be more than one payment  
   referring to the same invoice. Return the invoice number and the amount that should be  
   reimbursed.

/high\_demand\_products

1. Products that were ordered more than 10 times in total, by taking into account the  
   quantities in which they appear in the order details. Return the product code and the  
   total number of times it was ordered.

/bulk\_products

1. Products that are usually ordered in bulk: whenever one of these products is ordered, it  
   is ordered in a quantity that on average is equal to or greater than 8. For each such  
   product, return product code and price.

/number\_of\_products\_in\_year

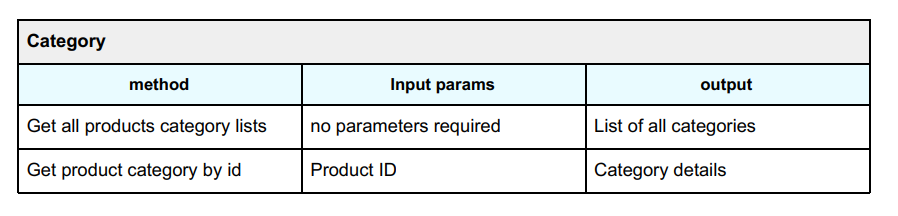
1. Total number of orders placed in 2016 by customers of each country. If all customers  
   from a specific country did not place any orders in 2016, the country will not appear in  
   the output.

/orders\_without\_invoices

1. For each order without invoice, list its ID, the date it was placed and the total price of the  
   products in its detail, taking into account the quantity of each ordered product and its unit  
   price. Orders without detail must not be included in the answers.

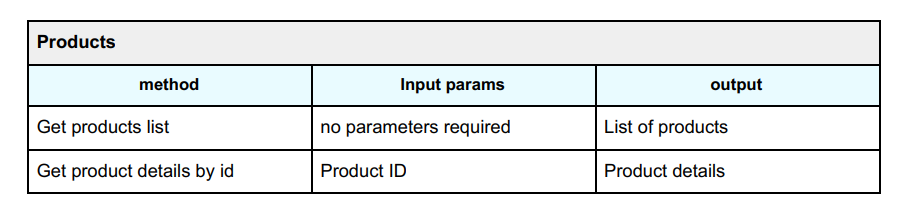
All methods are GET, without params

Create next api methods



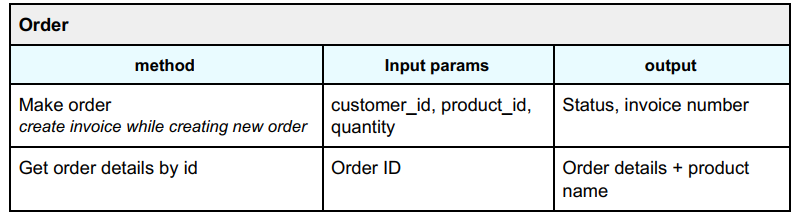
GET /category/list

GET /category?product\_id={product\_id}



GET /product/list

GET /product/details?product\_id={product\_id}



POST /order

{

customer\_id : c\_id,

product\_id: p\_id,

quantity: q

}

Response

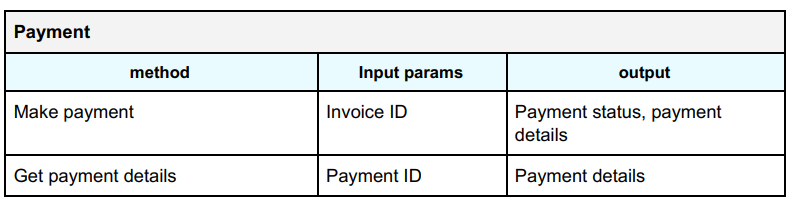
{

status : “status\_string” – “SUCCESS” or “FAILED” string,

invoice\_number: created\_invoice\_id

}

GET order/details?order\_id={order\_id}



POST /payment

{

Invoice\_id: invoice\_id

}

Response

{

Payment\_status: “status\_string” – “SUCCESS” or “FAILED” string,

Payment\_details: {

All payments table rows of created payment

}

}

GET /payment/details?id={payment\_details\_id}

Return all rows