EXERCISE 1 (4 POINTS)

In the folder Q:\Java, you will find the **ProductList.html** file. This file shows the products that are in 4 or more orders.

Product	Product Name	
92218	Hero Factory Weapon - Rubber Blade, Long	Product orders
973psx	Minifig Torso with SW Dark Red Robe Print	Product orders
67c01	Technic Pneumatic Airtank	Product orders
3876	Minifig Shield - Round with Stud	Product orders
32294	Technic Wishbone Suspension Arm	Product orders
30303	Plate Special 6 x 6 x 2/3 Cross with Dome	Product orders
2446p51	Minifig Helmet Standard with Silver and Blue Machinery Print	Product orders

Develop a Servlet named **ProductOrders** that lists all the order lines that include the product. The servlet receives a parameter called **productid**.

The title below the menu must be "List of orders of **productName**".

The line "Developed by **your name,** card id: **your card id**" should appear under the title.

The first row of the table will include the following fields' names : Order Id, Customer Id, Customer Name, Quantity.

For each row, you should show the values of these fields. The row will also have a link to OrderEdit with the parameter id and the value of the Order Id. Example: OrderEdit?id=10317. This servlet already exists and you can use it to test the validity of the link.

You will develop the file ***ProductOrdersData.java***, with the following method:

```
public    static    Vector<ProductOrdersData>    getProductOrdersList(Connection
connection, String productId)
```

Where the *productId* parameter refers to the selected product, and the result will be a vector containing the list of order lines where the product appears.

| | Home | Products | Last product | Orders | Last order | Categories | Category 45 | |

List of Orders of Hero Factory Weapon - Rubber Blade, Long

Developed by Juanfran, card id: A904161

Order Id	Customer Id	Customer Name	Quantity	
10317	LONEP	Lonesome Pine Restaurant	20	Edit
10368	ERNSH	Ernst Handel	35	Edit
10380	HUNGO	Hungry Owl All-Night Grocers	6	Edit
10856	ANTON	Antonio Moreno Taquería	20	Edit
10923	LAMAI	La maison d'Asie	10	Edit

Check the sections you have done:

☐ *The solution shows the name of the Product in the title and your name and card id in the next line (1 point)*

☐ *The solution shows the list of fields in the first row of the table (0,5 points)*

☐ *The solution shows the list of Products with the specified fields and link (1,5 points)*

☐ *The solution shows only the Orders than contains the selected Product (1 point)*

EXERCISE 2 (4 POINTS)

Create a copy of ProductList.html and name it **ProductListJS**.html. In this file you should add, the button "Edit name" (preferably with JavaScript) to one of the products, or better to all the products (**). This button will be placed in the last column, currently empty but with an id attribute. The button adds a form, preferably with JavaScript when it is clicked.

The form will have the field "productId" (not editable) and the field"productName" (preferible initiallised with the current name of the product *).

Product	Product Name		
92218	Hero Factory Weapon - Rubber Blade, Long	Product orders	Edit Name
973psx	Minifig Torso with SW Dark Red Robe Print	Product orders	Edit Name
67c01	Technic Pneumatic Airtank	Product orders	Edit Name
3876	Minifig Shield - Round with Stud	Product orders	Edit Name
32294	Technic Wishbone Suspension Arm	Product orders	Edit Name
30303	Plate Special 6 x 6 x 2/3 Cross with Dome	Product orders	Edit Name
2446p51	Minifig Helmet Standard with Silver and Blue Machinery Print	Product orders	Edit Name

Product ID	92218
Product Name	Hero Factory Weapon - Rubl

Send

The form will call the servlet '**ProductNameEdit**'. The servlet **ProductNameEdit** will be developed in Exercise 3.

☐ *The solution shows the "Edit name" button in the last column (1 point)*

☐ *The solution creates the form with the defined style (1 points)*

☐ *(*) The solution sets the current value of the Product name (0.5 points)*

☐ *The solution creates the button and form with JavaScript (1 point)*

☐ *(**) The solution creates the button for all the rows with JavaScript (0.5 points)*

EXERCISE 3 (2 POINT)

In this exercise, you have to develop a new servlet, called **ProductNameEdit**, that reads the fields from the form created in Exercise 2 (fields: productId, productName) and updates the name of the product in the database.

Just to check if the name has been updated correctly in the database, you could add the following line in the servlet after inserting the record in the database. Don't forget to refresh the page to see the update!! The servlet ProductList already exists.

```
res.sendRedirect("ProductList?id=" + categoryId);
```

☐ *Done (2 point)*

ADITIONAL EXERCISE

In the list of products (ProductListJS.html), the page should show the number of orders that contain the product when the user moves the cursor over the first column. The data will be read from the server with an Ajax call. The JavaScript event is "onmouseover".

☐ *Done (1 point)*