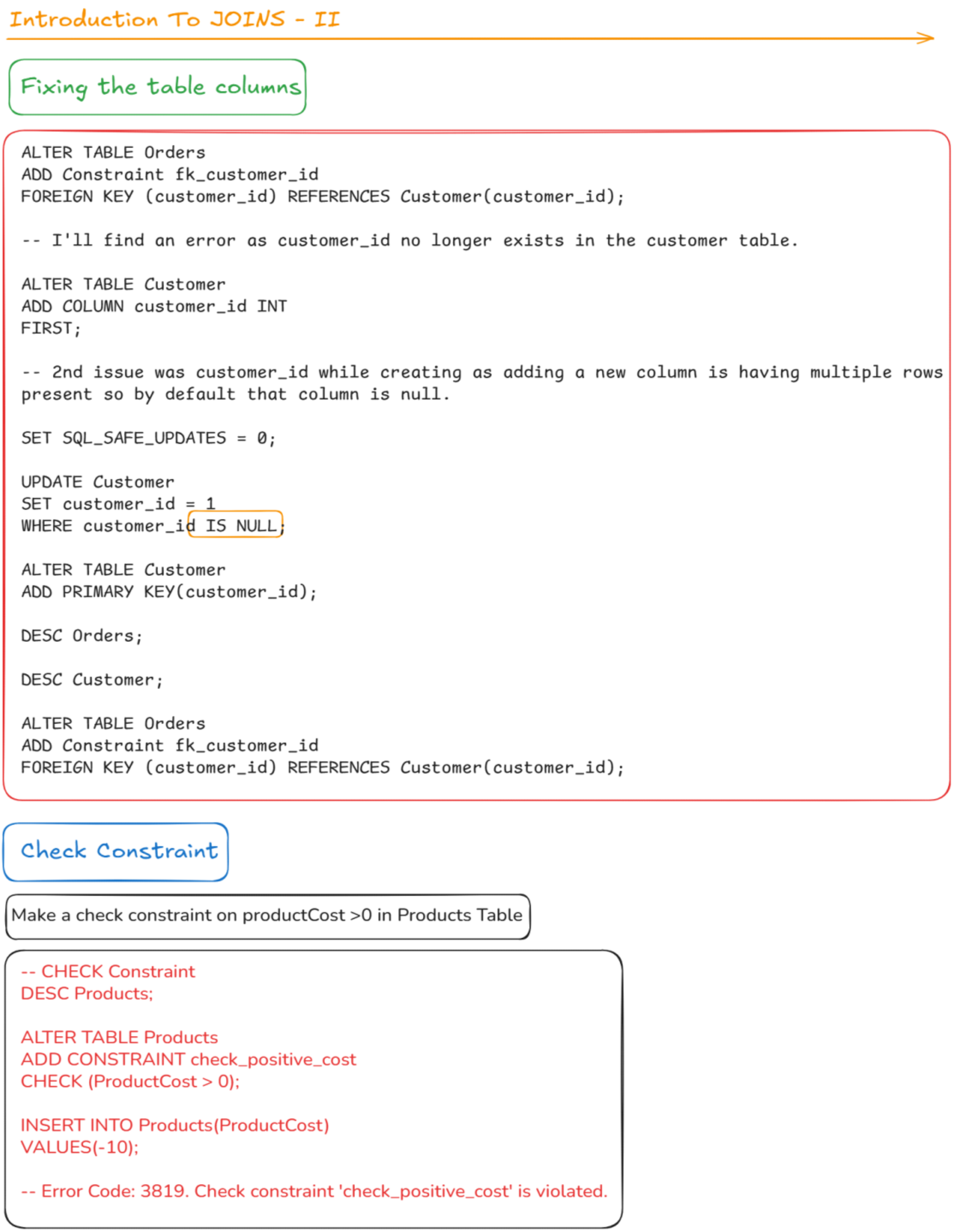
CN-Lec11-Introduction To Joins - II



Introduction To JOINS - II

Fixing the table columns

```
ALTER TABLE Orders
ADD Constraint fk_customer_id
FOREIGN KEY (customer_id) REFERENCES Customer(customer_id);

-- I'll find an error as customer_id no longer exists in the customer table.

ALTER TABLE Customer
ADD COLUMN customer_id INT
FIRST;

-- 2nd issue was customer_id while creating as adding a new column is having multiple rows
present so by default that column is null.

SET SQL_SAFE_UPDATES = 0;

UPDATE Customer
SET customer_id = 1
WHERE customer_id IS NULL;

ALTER TABLE Customer
ADD PRIMARY KEY(customer_id);

DESC Orders;

DESC Customer;

ALTER TABLE Orders
ADD Constraint fk_customer_id
FOREIGN KEY (customer_id) REFERENCES Customer(customer_id);
```

Check Constraint

Make a check constraint on productCost >0 in Products Table

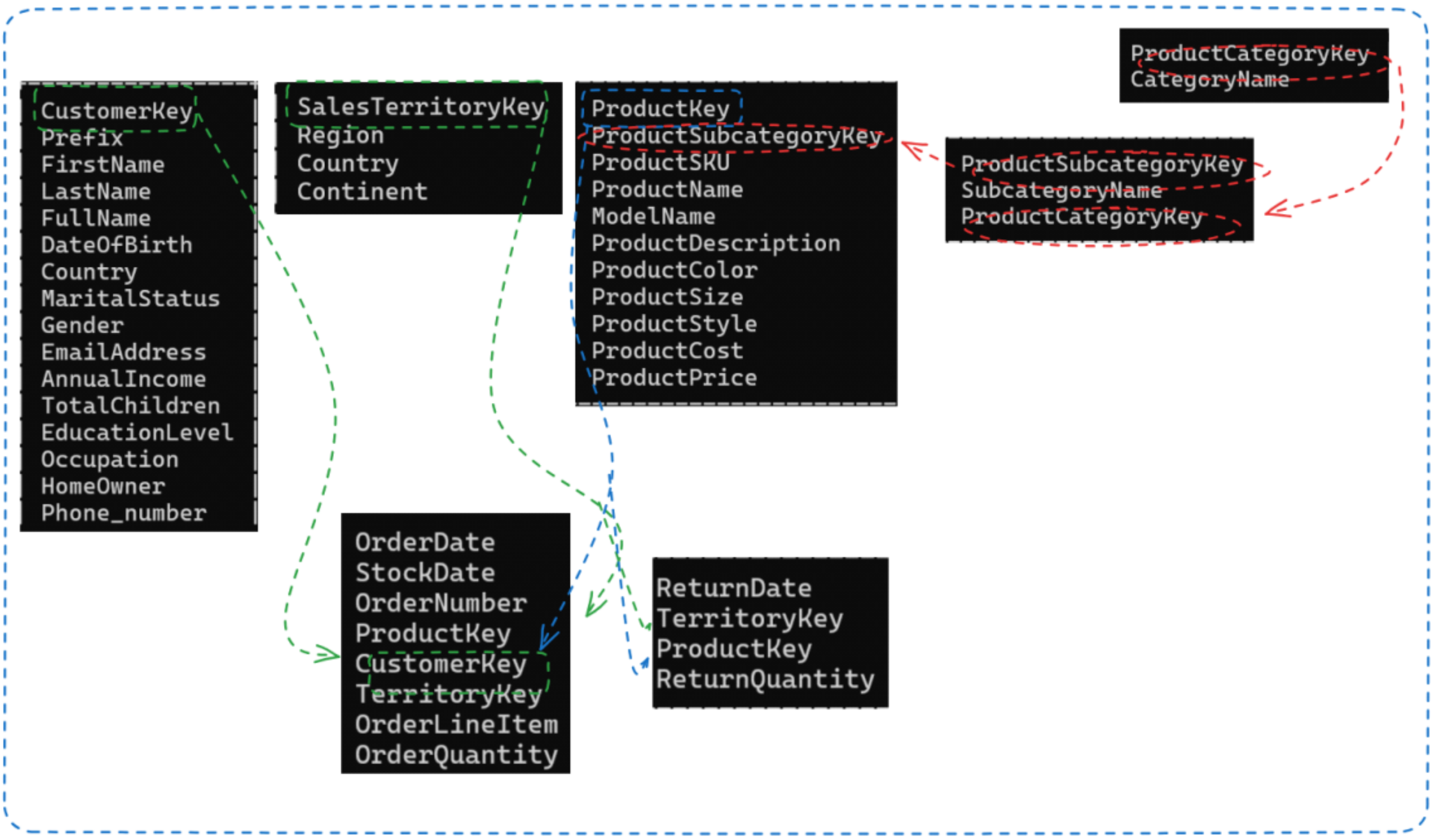
```
-- CHECK Constraint
DESC Products;

ALTER TABLE Products
ADD CONSTRAINT check_positive_cost
CHECK (ProductCost > 0);

INSERT INTO Products(ProductCost)
VALUES(-10);

-- Error Code: 3819. Check constraint 'check_positive_cost' is violated.
```

Make a check constraint on returnQty >0 in returns Table

```
-- Adding Constraint on Return Quantity >0

DESC Returns;

ALTER TABLE Returns
ADD CONSTRAINT check_return_quantity
CHECK (ReturnQuantity > 0);

INSERT INTO Returns(ReturnQuantity)
VALUES(-1);

-- Error Code: 3819. Check constraint 'check_return_quantity' is violated.
```

ERD Diagram

ON table1.key = table2.key



```
CustomerKey
Prefix
FirstName
LastName
FullName
DateOfBirth
Country
MaritalStatus
Gender
EmailAddress
AnnualIncome
TotalChildren
EducationLevel
Occupation
HomeOwner
Phone_number
```

```
SalesTerritoryKey
Region
Country
Continent
```

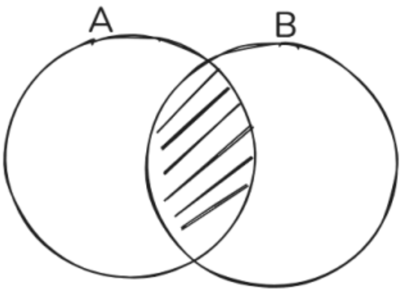
```
ProductKey
ProductSubcategoryKey
ProductSKU
ProductName
ModelName
ProductDescription
ProductColor
ProductSize
ProductStyle
ProductCost
ProductPrice
```

```
ProductCategoryKey
CategoryName
```

```
ProductSubcategoryKey
SubcategoryName
ProductCategoryKey
```

```
OrderDate
StockDate
OrderNumber
ProductKey
CustomerKey
TerritoryKey
OrderLineItem
OrderQuantity
```

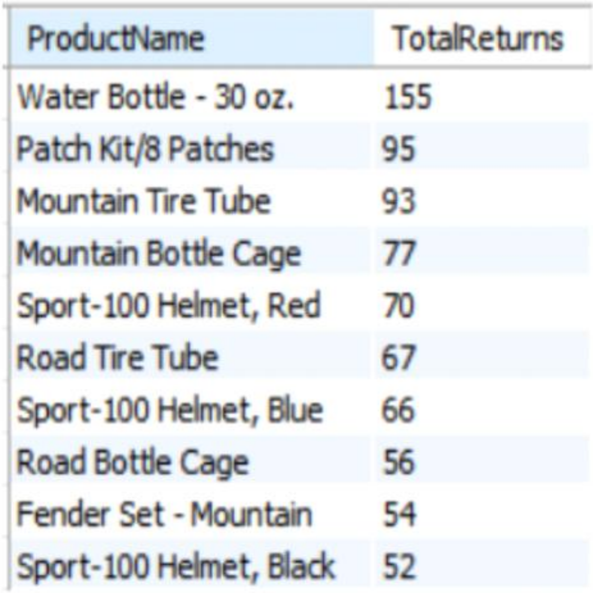
```
ReturnDate
TerritoryKey
ProductKey
ReturnQuantity
```



INNER JOIN/ JOIN

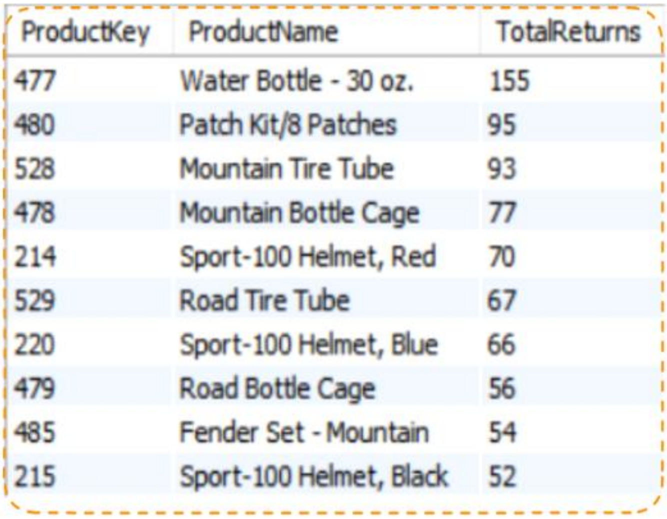
Find the name of the Products that returned

```
SELECT
   ProductName,
   SUM(ReturnQuantity)
FROM Product
INNER JOIN
Returns
ON product.ProductKey = returns.ProductKey
GROUP BY ProductName
```



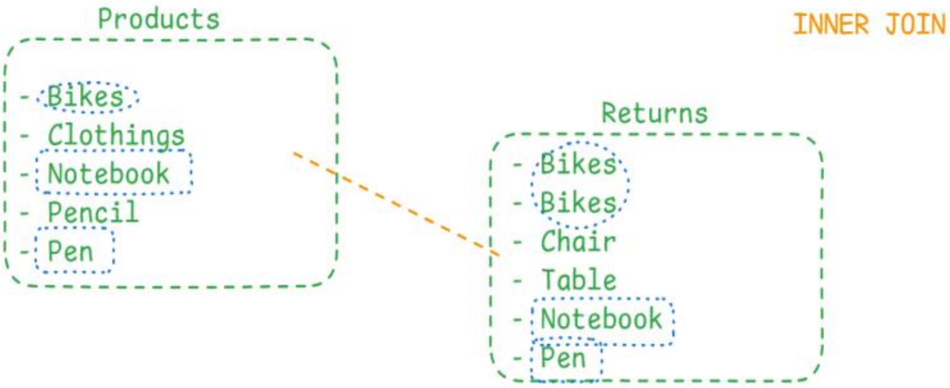
ProductName	TotalReturns
Water Bottle - 30 oz.	155
Patch Kit/8 Patches	95
Mountain Tire Tube	93
Mountain Bottle Cage	77
Sport-100 Helmet, Red	70
Road Tire Tube	67
Sport-100 Helmet, Blue	66
Road Bottle Cage	56
Fender Set - Mountain	54
Sport-100 Helmet, Black	52

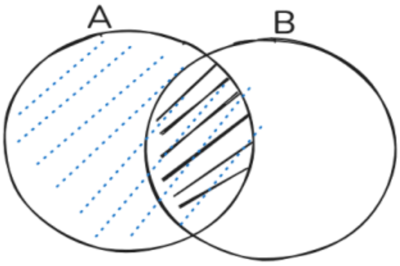
```
SELECT
    ProductName,
    SUM(ReturnQuantity) AS TotalReturns
FROM Products
INNER JOIN Returns
ON Products.ProductKey = returns.ProductKey
GROUP BY 1
ORDER BY 2 DESC
LIMIT 10;
```



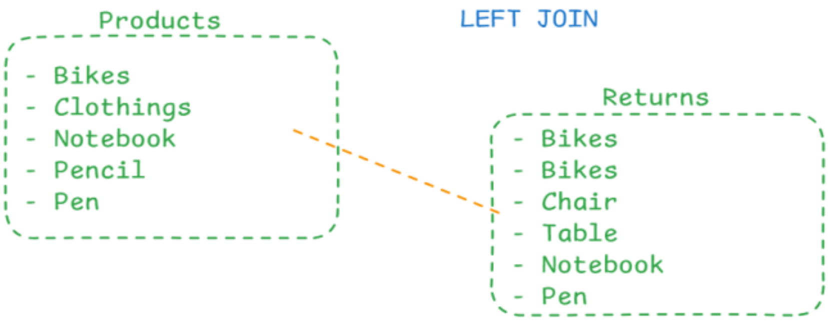
ProductKey	ProductName	TotalReturns
477	Water Bottle - 30 oz.	155
480	Patch Kit/8 Patches	95
528	Mountain Tire Tube	93
478	Mountain Bottle Cage	77
214	Sport-100 Helmet, Red	70
529	Road Tire Tube	67
220	Sport-100 Helmet, Blue	66
479	Road Bottle Cage	56
485	Fender Set - Mountain	54
215	Sport-100 Helmet, Black	52

```
SELECT
    p.ProductKey,
    p.ProductName,
    SUM(r.ReturnQuantity) AS TotalReturns
FROM Products AS p
INNER JOIN Returns AS r
ON p.ProductKey = r.ProductKey
GROUP BY 1,2
ORDER BY 3 DESC
LIMIT 10;
```





LEFT JOIN



Products

- Bikes
- Clothings
- Notebook
- Pencil
- Pen

LEFT JOIN

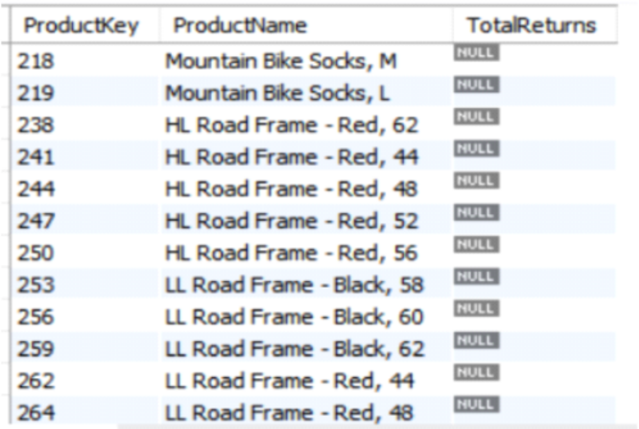
Returns

- Bikes
- Bikes
- Chair
- Table
- Notebook
- Pen

Final Table

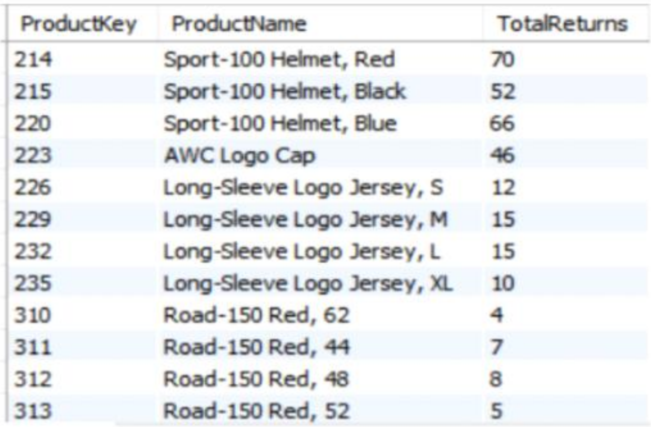
- Bikes - 2
- Clothings - NULL
- Notebook - 1
- Pencil - NULL
- Pen - 1

Find the products who never being returned.



ProductKey	ProductName	TotalReturns
218	Mountain Bike Socks, M	NULL
219	Mountain Bike Socks, L	NULL
238	HL Road Frame - Red, 62	NULL
241	HL Road Frame - Red, 44	NULL
244	HL Road Frame - Red, 48	NULL
247	HL Road Frame - Red, 52	NULL
250	HL Road Frame - Red, 56	NULL
253	LL Road Frame - Black, 58	NULL
256	LL Road Frame - Black, 60	NULL
259	LL Road Frame - Black, 62	NULL
262	LL Road Frame - Red, 44	NULL
264	LL Road Frame - Red, 48	NULL

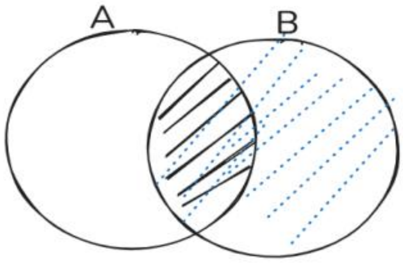
```
-- Find the products who never being returned.

SELECT
        p.ProductKey,
        p.ProductName,
    SUM(r.ReturnQuantity) AS TotalReturns
FROM Products AS p
LEFT JOIN Returns AS r
ON p.ProductKey = r.ProductKey
GROUP BY 1,2
HAVING TotalReturns IS NULL;
```



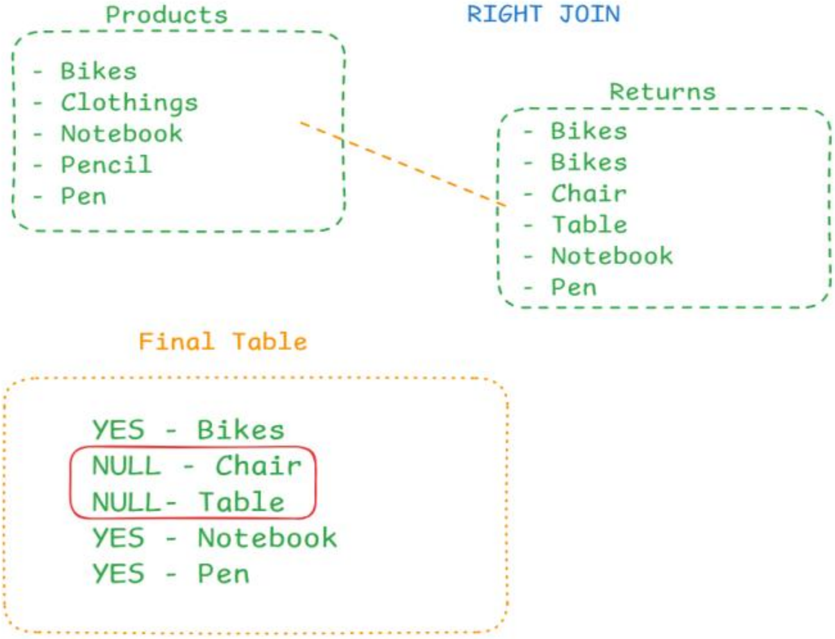
ProductKey	ProductName	TotalReturns
214	Sport-100 Helmet, Red	70
215	Sport-100 Helmet, Black	52
220	Sport-100 Helmet, Blue	66
223	AWC Logo Cap	46
226	Long-Sleeve Logo Jersey, S	12
229	Long-Sleeve Logo Jersey, M	15
232	Long-Sleeve Logo Jersey, L	15
235	Long-Sleeve Logo Jersey, XL	10
310	Road-150 Red, 62	4
311	Road-150 Red, 44	7
312	Road-150 Red, 48	8
313	Road-150 Red, 52	5

```
-- Find the products who are returned.
SELECT
        p.ProductKey,
        p.ProductName,
    SUM(r.ReturnQuantity) AS TotalReturns
FROM Products AS p
LEFT JOIN Returns AS r
ON p.ProductKey = r.ProductKey
GROUP BY 1,2
HAVING TotalReturns IS NOT NULL;
```

Finding the products which are not returned. [via return table]



RIGHT JOIN



Products

- Bikes
- Clothings
- Notebook
- Pencil
- Pen

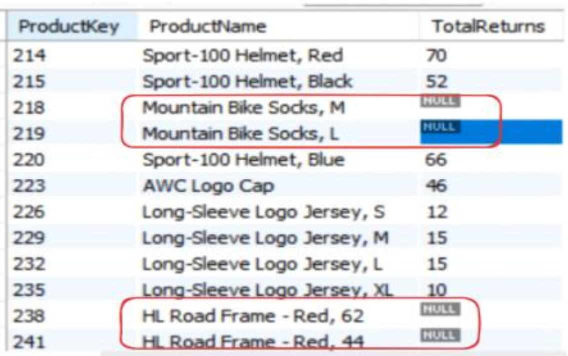
RIGHT JOIN

Returns

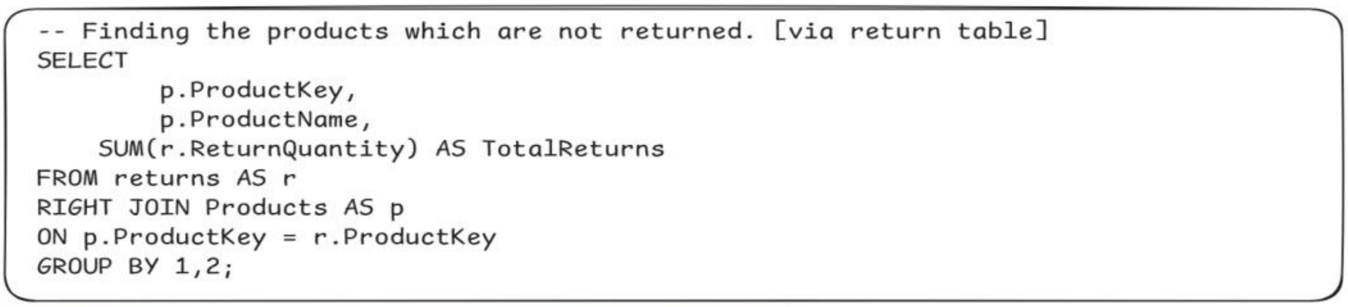
- Bikes
- Bikes
- Chair
- Table
- Notebook
- Pen

Final Table

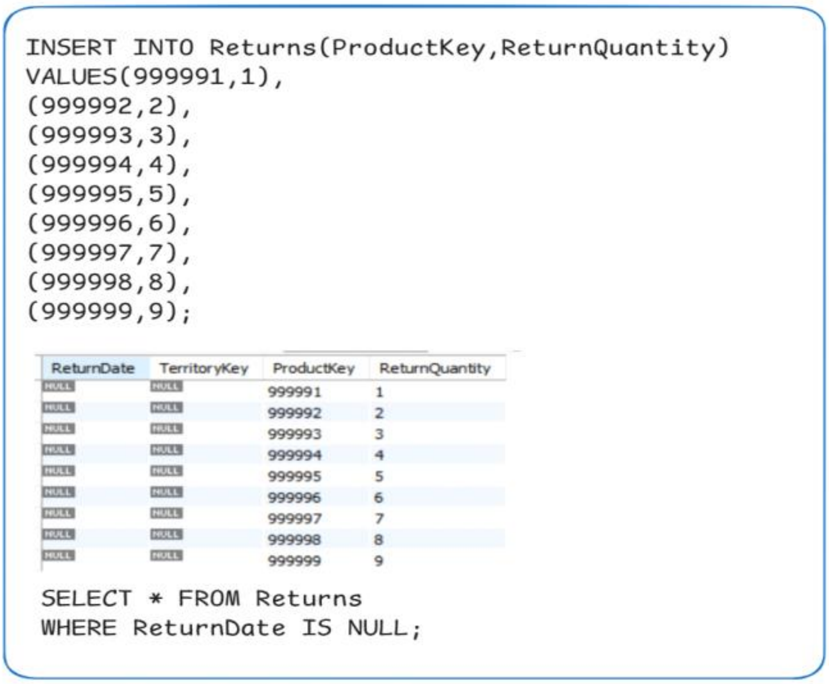
YES - Bikes
NULL - Chair
NULL- Table
YES - Notebook
YES - Pen



ProductKey	ProductName	TotalReturns
214	Sport-100 Helmet, Red	70
215	Sport-100 Helmet, Black	52
218	Mountain Bike Socks, M	NULL
219	Mountain Bike Socks, L	NULL
220	Sport-100 Helmet, Blue	66
223	AWC Logo Cap	46
226	Long-Sleeve Logo Jersey, S	12
229	Long-Sleeve Logo Jersey, M	15
232	Long-Sleeve Logo Jersey, L	15
235	Long-Sleeve Logo Jersey, XL	10
238	HL Road Frame - Red, 62	NULL
241	HL Road Frame - Red, 44	NULL



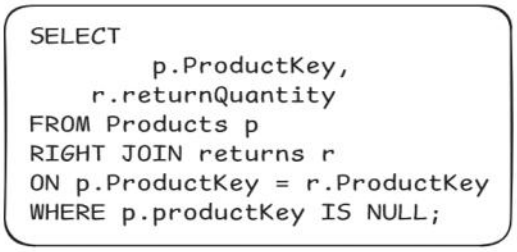
```
-- Finding the products which are not returned. [via return table]
SELECT
        p.ProductKey,
        p.ProductName,
    SUM(r.ReturnQuantity) AS TotalReturns
FROM returns AS r
RIGHT JOIN Products AS p
ON p.ProductKey = r.ProductKey
GROUP BY 1,2;
```



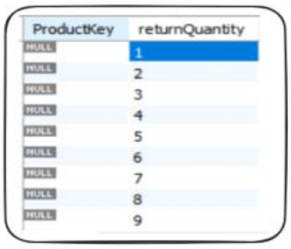
```
INSERT INTO Returns(ProductKey,ReturnQuantity)
VALUES(999991,1),
(999992,2),
(999993,3),
(999994,4),
(999995,5),
(999996,6),
(999997,7),
(999998,8),
(999999,9);
```

ReturnDate	TerritoryKey	ProductKey	ReturnQuantity
NULL	NULL	999991	1
NULL	NULL	999992	2
NULL	NULL	999993	3
NULL	NULL	999994	4
NULL	NULL	999995	5
NULL	NULL	999996	6
NULL	NULL	999997	7
NULL	NULL	999998	8
NULL	NULL	999999	9

```
SELECT * FROM Returns
WHERE ReturnDate IS NULL;
```



```
SELECT
        p.ProductKey,
    r.returnQuantity
FROM Products p
RIGHT JOIN returns r
ON p.ProductKey = r.ProductKey
WHERE p.productKey IS NULL;
```



ProductKey	returnQuantity
NULL	1
NULL	2
NULL	3
NULL	4
NULL	5
NULL	6
NULL	7
NULL	8
NULL	9