

# TD Base de données

## **SOLUTIONS :**    **Corrigé du TD sur la base de données W3school**

**Q1 Trouver tous les clients situés à Paris :** Il y en a deux -> 57 et 74

```
SELECT * FROM Customers  
WHERE City=='Paris'
```

**Q2 Trouver combien de clients sont situés en France :** il y en a 11

```
SELECT count(*) FROM Customers  
WHERE Country=='France'
```

### **Q3 PROJECTION**

Trouver la liste donnant uniquement le nom et le numéro (et dans cet ordre !) de ces clients.

```
SELECT CustomerName, CustomerID FROM Customers  
WHERE Country=='France'
```

**Q4 Fonction d'agrégation : LIKE :** il y en a 2 -> 18 et 74

Au sein de cette liste, qui sont des restaurants ou des marchands tous dans l'alimentaire, on souhaite savoir lesquels ont un nom qui contient le mot «monde» car ils seront sans doute plus ouverts d'esprit ...

La commande **like** **'%monde%'** permet de trouver une chaîne « du genre » 'monde' mais avec quoi que ce soit (%) avant ou après : ceci est indiqué par % . A vous de la faire fonctionner !

Essayer ensuite avec like 'monde' pour bien comprendre.

```
SELECT CustomerName, CustomerID FROM Customers  
WHERE Country=='France' AND CustomerName like '%monde%'
```

**Q5 Trouver un produit :** réponse identifiant 38

Trouver l'identifiant du produit «Côte de Blaye» attention à l'orthographe.

Le résultat de la requête doit être un entier.

```
SELECT ProductID FROM Products  
WHERE ProductName = "Côte de Blaye »
```

### **Q6 Moyenne avg() & Comparaison à la moyenne**

Obtenir la moyenne des prix à l'aide d'une requête

Trouver tous les produits dont le prix est inférieur à la moyenne des prix

```
SELECT AVG(Price) from Products  
SELECT * from Products WHERE price < (SELECT AVG(price) from Products)
```

### **Q7 EXCEPT versus NOT IN**

Trouver le ou les employé(s) qui n'ont pas servi de commande.

Trouver ensuite le nom et prénom du ou des employé(s)

#ne permet pas d'avoir des champs d'attributs différents

```
SELECT EmployeeID FROM Employees  
EXCEPT  
SELECT EmployeeID FROM Orders
```

```
SELECT LastName, FirstName FROM Employees  
WHERE EmployeeID NOT IN (SELECT EmployeeID FROM Orders)
```

**Q8 JOINTURE : jointure entre produits et catégories.**

Trouver tous les produits qui appartiennent à la catégorie de nom «Seafood».  
N'y a-t-il pas des produits étranges dans cette catégorie ?

Il y en a 12 :

```
SELECT * FROM Categories INNER JOIN Products  
on Categories.CategoryID=Products.CategoryID  
WHERE Categories.CategoryName=«Seafood»
```

On pourra trouver étonnant d'y découvrir nos fameux escargots de Bourgogne !!!  
Ceux-ci ne savent pas nager...a priori et encore moins dans la mer !

**Q9 JOINTURE : jointure entre fournisseur et produits**

Justement nous avons une folle envie d'Escargots et de Bourgogne ! Ça tombe bien !  
A l'aide d'une requête, trouver le nom du fournisseur (supplier) des «Escargots de Bourgogne» :  
Nom de la boutique, son adresse, code postal, ville et pays.

```
SELECT SupplierName, Address, PostalCode, City, Country FROM Suppliers INNER JOIN  
Products on Suppliers.SupplierID=Products.SupplierID  
WHERE Products.ProductName="Escargots de Bourgogne"
```

**Q10 JOINTURE : jointure entre Clients et commandes**

Trouver toutes les commandes passées en France en les triant par ordre chronologique.  
En déduire la dernière commande passée en France en modifiant votre requête.  
il y en a 18.

```
SELECT * FROM Orders INNER JOIN Customers on  
ORDERS.CustomerID=CUSTOMERS.CUSTOMERID  
WHERE country='France' order by ORDERDATE
```

```
SELECT *, max(OrderDate) FROM Orders INNER JOIN Customers on  
ORDERS.CustomerID=CUSTOMERS.CUSTOMERID  
WHERE country='France'
```

**Q11 TRIPLE JOINTURE :**

On veut savoir précisément ce qui a été commandé par le restaurant «Blondel père et fils» sur la  
place Kleber à Strasbourg ! Trouver le nom de produit, l'identifiant produit, le prix et la quantité,  
l'adresse, la ville, le pays, et l'identifiant du livreur à l'aide d'une seule requête  
On utilisera le nom «Blondel père et fils». En déduire le coût total de ses achats.

```
SELECT ProductName, Products.ProductID, Price, Address, City, Country, ShipperID FROM  
Orders INNER JOIN Customers on ORDERS.CustomerID=CUSTOMERS.CUSTOMERID INNER  
JOIN ORDERDetails on OrderDetails.OrderID=ORDERS.OrderID INNER JOIN Products on  
Products.ProductID=OrderDetails.ProductID  
WHERE CustomerName ='Blondel père et fils'
```

```
SELECT sum(price) FROM Orders INNER JOIN Customers on  
ORDERS.CustomerID=CUSTOMERS.CUSTOMERID INNER JOIN ORDERDetails on  
OrderDetails.OrderID=ORDERS.OrderID INNER JOIN Products on  
Products.ProductID=OrderDetails.ProductID  
WHERE CustomerName ='Blondel père et fils'
```

Il y en a 13 et un total pour : 658.14 € [www.jeSuisNaïf.com](http://www.jeSuisNaïf.com)

## Il faut aussi tenir compte de la quantité des produits achetés :

```
SELECT ProductName, Products.ProductID, quantity, Price, price*quantity, Address, City, Country, ShipperID FROM Orders INNER JOIN Customers on ORDERS.CustomerID=CUSTOMERS.CUSTOMERID INNER JOIN ORDERDetails on OrderDetails.OrderID=ORDERS.OrderID INNER JOIN Products on Products.ProductID=OrderDetails.ProductID WHERE CustomerName ='Blondel père et fils'
```

soit un total de : **15253.75€** !!!

```
SELECT sum(price*quantity) FROM Orders INNER JOIN Customers on ORDERS.CustomerID=CUSTOMERS.CUSTOMERID INNER JOIN ORDERDetails on OrderDetails.OrderID=ORDERS.OrderID INNER JOIN Products on Products.ProductID=OrderDetails.ProductID WHERE CustomerName ='Blondel père et fils'
```

### Q12 Classement

Faire la liste (pour le même client) des noms de produits et de leurs prix par ordre décroissant.  
Obtenir par requête la plus grosse commande en terme de prix.  
Trouver le produit le plus commandé (en fait il y en a plusieurs)

```
SELECT ProductName, price FROM Orders INNER JOIN Customers on ORDERS.CustomerID=CUSTOMERS.CUSTOMERID INNER JOIN ORDERDetails on OrderDetails.OrderID=ORDERS.OrderID INNER JOIN Products on Products.ProductID=OrderDetails.ProductID WHERE CustomerName ='Blondel père et fils' ORDER BY price DESC
```

```
SELECT ProductName, max(price) FROM Orders INNER JOIN Customers on ORDERS.CustomerID=CUSTOMERS.CUSTOMERID INNER JOIN ORDERDetails on OrderDetails.OrderID=ORDERS.OrderID INNER JOIN Products on Products.ProductID=OrderDetails.ProductID WHERE CustomerName ='Blondel père et fils'
```

Le produit le plus cher est le fameux «Côte de Blaye» pas très alsacien ...

### Q13 CLASSEMENT : requêtes imbriquées complexes avec agrégation

Trouver le produit le plus commandé par le client (en fait il y en a plusieurs)

```
SELECT productName, max(Count) FROM (SELECT Products.ProductID, Products.productName, count(Products.ProductID) as Count FROM [Products] INNER JOIN OrderDetails on Products.ProductID=OrderDetails.ProductID group by Products.ProductID)
```

mais on obtient qu'un seul... souvent il n'y a qu'un seul max  
exemple mines 2014 : données numériques

Il est peu probable que deux valeurs numériques apparaissent autant de fois l'une que l'autre surtout si il y a beaucoup de données (encore que.... )

Ici il y en a 3 : **Mozzarella di Giovanni, Raclette Courdavault, et Gorgonzola Telino**

Bref #attrapezLesTous avec :

```
SELECT productName, Count FROM (  
SELECT Products.ProductID, Products.productName, count(Products.ProductID) as Count FROM  
[Products] INNER JOIN OrderDetails on Products.ProductID=OrderDetails.ProductID group by  
Products.ProductID)
```

```
WHERE Count=(
```

```
SELECT max(Count) FROM (SELECT Products.ProductID, Products.productName,  
count(Products.ProductID) as Count FROM [Products] INNER JOIN OrderDetails on  
Products.ProductID=OrderDetails.ProductID group by Products.ProductID)  
)
```

youpiiii ... sans copier/coller c'est assez peu digeste.