RAPPORT du brief bdd-first-S
Automatiser et Contrôler le nombre de commandes

Introduction

Autant que gérant du coffee_shop (l'entracte), j'aimerai bien avoir un système automatisé qui me permet d'enregistrer les clients et leurs commandes pour visualiser la date et le client qui a pris la commande, peut être au futur j'aimerai ajouter une promotion pour les clients fidèles

#2 what's the SQL query that allow us to delete a column in a table

====> Une syntaxe permet également de supprimer une colonne pour une table. Il y a 2 manières totalement équivalente pour supprimer une colonne:

```
# ALTER TABLE nom_table
DROP nom_colonne
```

```
# ALTER TABLE nom_table

DROP COLUMN nom_colonne
```

#3 **'Truncat'** : la commande TRUNCATE permet de supprimer toutes les données d'une table sans supprimer la table en elle-même

////APPLICATION \\\\

1- créer une table (truncat) avec 4 colums.

	#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
☐	1	**colum N1**	int(11)			Non	*Aucun(e)*			🖉 Mod
☐	2	**colum N2**	int(11)			Non	*Aucun(e)*			🖉 Mod
☐	3	**colum N3**	int(11)			Non	*Aucun(e)*			🖉 Mod
☐	4	**colum N4**	int(11)			Non	*Aucun(e)*			🖉 Mod

↑ ☐ Tout cocher   *Avec la sélection :*   📋 Parcourir   🖉 Modifier   ⊖ Supprimer   🔑 Primaire

🖨 Imprimer   📇 Suggérer des optimisations de structure ❓   ▶ Déplacer des colonnes   ✴ Normaliser

Ajouter `1` colonne(s) `après colum N4 ▾` **Exécuter**

2- Il est possible de supprimer toutes les données de cette table en utilisant la requête suivante :

```
TRUNCATE TABLE `truncat`
```

📋 Parcourir   Structure   📄 SQL   🔍 Rechercher   Insérer   📤 Exporter   📥 Importer   Privilèges   Opérations   Déclencheurs

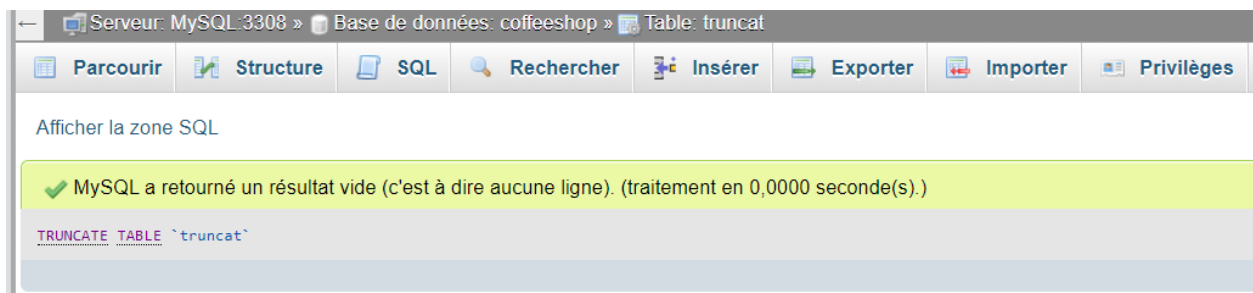Exécuter une ou des requêtes SQL sur la table « coffeeshop.truncat » :

```
1   TRUNCATE TABLE `truncat`
```

Colonnes
colum N1
colum N2
colum N3
colum N4

[SELECT*] [SELECT] [INSERT] [UPDATE] [DELETE] [Effacer] [Format]

Récupérer la requête auto-sauvegardée   <<

☐ Lier les paramètres

résultat :
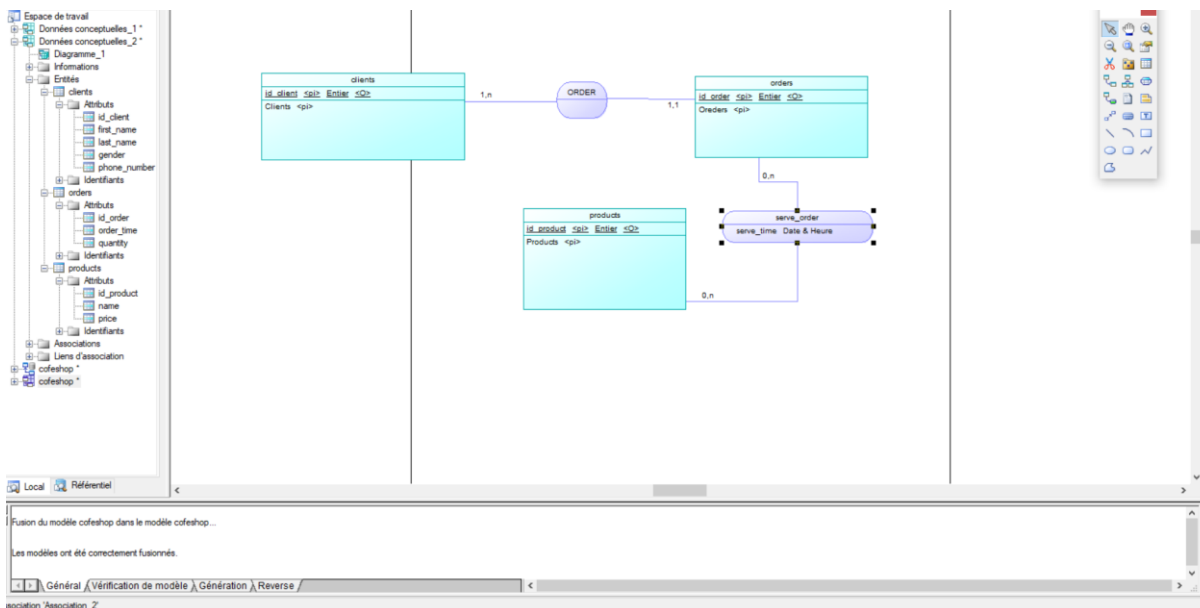
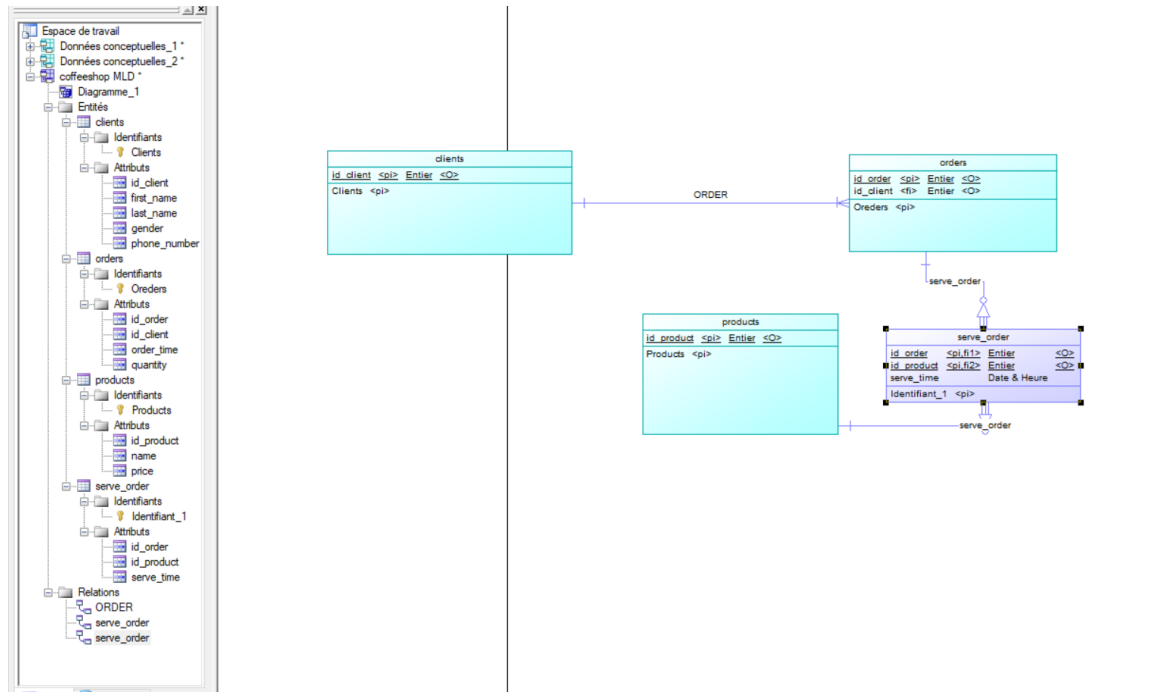

PowerDesigner (anciennement **PowerAMC)** :

est un logiciel de conception créé par la société *SAP*, qui permet de modéliser les traitements informatiques et leurs bases de données associées.
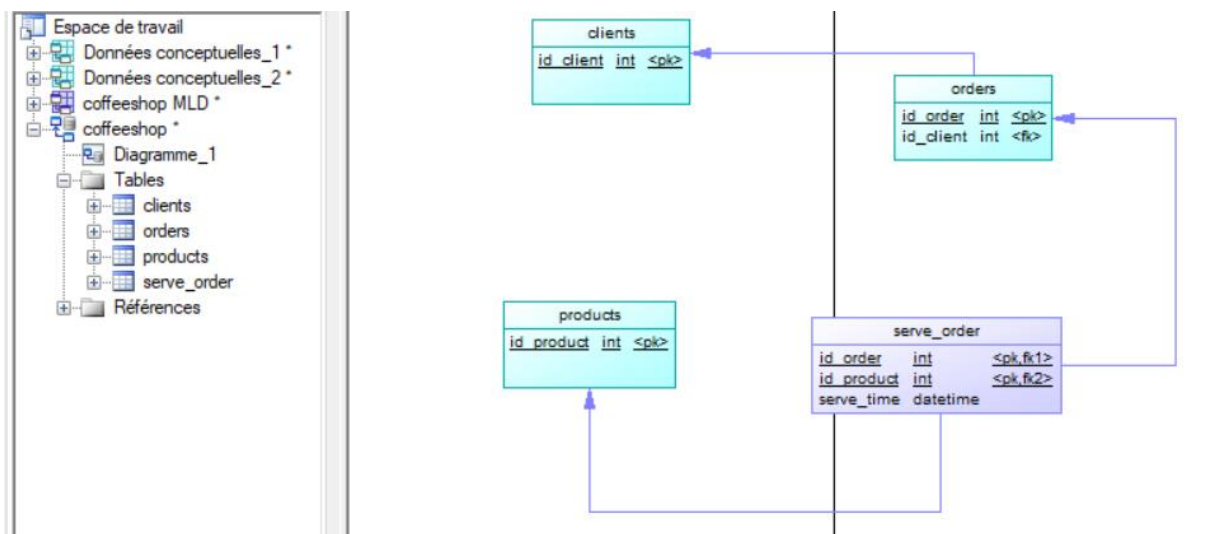
MCD: *Modele Conceptuel Donnees*

MLD: Modele Logique Donnees

MPD: Modele Physique Donnees

création base de donnée et création les tableau DLL :

```
# MySQL a retourné un résultat vide (aucune ligne).

create database coffeeshop;# 1 ligne affectée.

use coffeeshop;# MySQL a retourné un résultat vide (aucune ligne).

create table CLIENTS
(
ID_CLIENT int not null auto_increment,
FIRST_NAME varchar(50),
LAST_NAME varchar(50),
GENDER enum("M","F"),
PHONE_NUMBER int,
primary key (ID_CLIENT)
);# MySQL a retourné un résultat vide (aucune ligne).

create table ORDERS
(
ID_ORDER int not null auto_increment ,
ID_CLIENT int not null,
ORDER_TIME date,
QUANTITY int,
primary key (ID_ORDER)
);# MySQL a retourné un résultat vide (aucune ligne).

alter table ORDERS add constraint FK_ORDER foreign key (ID_CLIENT)
references CLIENTS (ID_CLIENT) on delete restrict on update restrict;# MySQL a retourné un résultat vide (aucune ligne).
```

phpMyAdmin

Serveur: mysql wampserver » Base de données: coffeeshop

	Table ▲	Action							Lignes	Type	Interclassement	Taille	Perte
	clients	Afficher	Structure	Rechercher	Insérer	Vider	Supprimer		~0	InnoDB	latin1_swedish_ci	16 Kio	-
	orders	Afficher	Structure	Rechercher	Insérer	Vider	Supprimer		~0	InnoDB	latin1_swedish_ci	32 Kio	-
	products	Afficher	Structure	Rechercher	Insérer	Vider	Supprimer		~0	InnoDB	latin1_swedish_ci	16 Kio	-
	serve_order	Afficher	Structure	Rechercher	Insérer	Vider	Supprimer		~0	InnoDB	latin1_swedish_ci	32 Kio	-
	4 tables	Somme							0	InnoDB	latin1_swedish_ci	96 Kio	0 o

Tout cocher   Pour la sélection :

add a new column Product_origin varchar(50) :

MySQL a retourné un résultat vide (aucune ligne). (Traitement en 0.0219 sec)

```
ALTER TABLE products ADD product_origin varchar(50)
```

[En ligne] [Modifier] [Créer source PHP]

Montrer zone SQL

Add a unique constraint to the 'first_name' and 'last_name' column in clients table :

MySQL a retourné un résultat vide (aucune ligne). (Traitement en 0.0126 sec)

ALTER TABLE clients ADD constraint full_name UNIQUE (first_name,last_name)

[En ligne] [Modifier] [Créer source PHP]

Montrer zone SQL

change the column name 'order_tim'e to 'Date_order' in the table 'command' and change the type to 'DATETIME'

MySQL a retourné un résultat vide (aucune ligne). (Traitement en 0.0275 sec)

ALTER TABLE orders CHANGE order_time date_order datetime

[En ligne] [Modifier] [Créer source PHP]

Montrer zone SQL

#6 delete the column 'quantity' from the table 'command' :

MySQL a retourné un résultat vide (aucune ligne). (Traitement en 0.0217 sec)

ALTER TABLE orders drop quantity

[En ligne] [Modifier] [Créer source PHP]

Montrer zone SQL

Insert those Data in the table Clients: ('Chris','Martin','M','01123147789'), ('Emma','Law','F','01123439899'), ('Mark','Watkins','M','01174592013'),...

✔ MySQL a retourné un résultat vide (aucune ligne). (Traitement en 0.0047 sec)

INSERT INTO clients (FIRST_NAME,LAST_NAME,GENDER,PHONE_NUMBER) VALUES ('Chris','Martin','M','01123147789'), ('Emma','Law','F','01123439899'), ('Mark','Watkins','M','01174592013'),
('Daniel','Williams','M',NULL), ('Sarah','Taylor','F','01176348290'), ('Katie','Armstrong','F','01145787353'), ('Michael','Bluth','M','01980289282'), ('Kat','Nash','F','01176987789'),
('Buster','Bluth','M','01173456782'), ('Charlie',NULL,'F','01139287883'), ('Lindsay','Bluth','F','01176923804'), ('Harry','Johnson','M',NULL), ('John','Smith','M','01174987221'), ('John','Taylor','M',NULL),
('Emma','Smith','F','01176984116'), ('Gob','Bluth','M','01176985498'), ('George','Bluth','M','01176984303'), ('Lucille','Bluth','F','01198773214'), ('George','Evans','M','01174502933'),
('Emily','Simmonds','F','01899284352'), ('John','Smithe','M','01144473330'), ('Jennifer',NULL,'F',NULL), ('Toby','West','M','01176009822'), ('Paul','Edmonds','M','01966947113')

[En ligne] [Modifier] [Créer source PHP]

Montrer zone SQL

Insert those Data in the table command (1,1,'2017-01-01 08-02-11'), (1,2,'2017-01-01 08-05-16'), (5,12,'2017-01-01 08-44-34'),

✔ MySQL a retourné un résultat vide (aucune ligne). (Traitement en 0.0053 sec)

INSERT INTO products (ID_PRODUCT,NAME,PRICE,product_origin) VALUES (1,'special coffee', 7.5,'Brazil'), (2,'coffee family', 8,'Cuba'), (13,'coffee 2.0', 8.5,'Ethopia'), (3,'animal coffee',25,'Indonesia'),
(22,'coffe plus', 15.5,'Italy'), (14,'coffee ++',30.00,'Maroc')

[En ligne] [Modifier] [Créer source PHP]

Montrer zone SQL

Insert those Data in the table command (1,'2017-01-01 08-02-11'), (2,'2017-01-01 08-05-16'), (12,'2017-01-01 08-44-34'), (4,'2017-01-01 09-20-02'),

✔ MySQL a retourné un résultat vide (aucune ligne). (Traitement en 0.0347 sec)

INSERT INTO orders (ID_CLIENT,date_order) VALUES (1,'2017-01-01 08-02-11'), (2,'2017-01-01 08-05-16'), (12,'2017-01-01 08-44-34'), (4,'2017-01-01 09-20-02'), (9,'2017-01-01 11-51-56'), (22,'2017-01-01 13-07-
10'), (1,'2017-01-02 08-03-41'), (10,'2017-01-02 09-15-22'), (2,'2017-01-02 10-10-10'), (13,'2017-01-02 12-07-23'), (1,'2017-01-03 08-13-50'), (16,'2017-01-03 08-47-09'), (21,'2017-01-03 09-12-11'),
(22,'2017-01-03 11-05-33'), (3,'2017-01-03 11-08-55'), (11,'2017-01-03 12-02-14'), (23,'2017-01-03 13-41-22'), (1,'2017-01-04 08-08-56'), (10,'2017-01-04 11-23-43'), (12,'2017-01-05 08-30-09'), (1,'2017-01-06
09-02-47'), (18,'2017-01-06 13-23-34'), (16,'2017-01-07 09-12-39'), (14,'2017-01-07 11-24-15'), (5,'2017-01-08 08-54-11'), (1,'2017-01-09 08-03-11'), (20,'2017-01-10 10-34-12'), (3,'2017-01-10 11-02-11'),
(24,'2017-01-11 08-39-11'), (8,'2017-01-12 13-20-13'), (1,'2017-01-14 08-27-10'), (15,'2017-01-15 08-30-56'), (7,'2017-01-16 10-02-11')

[En ligne] [Modifier] [Créer source PHP]

Montrer zone SQL

Price * 5 condition (product_name = special coffee)

✔ 1 ligne affectée. (Traitement en 0.0166 sec)

UPDATE products SET product_origin = "EST Europe" WHERE ID_PRODUCT = 22

[En ligne] [Modifier] [Créer source PHP]

Montrer zone SQL

Coffee_orgin = EST Europe where id_product = 22

✔ 0 ligne affectée. (Traitement en 0.0149 sec)

UPDATE products SET product_origin = "EST Europe" WHERE ID_PRODUCT = 22

[En ligne] [Modifier] [Créer source PHP]

Montrer zone SQL

in the table command, Print the recent command ordered

✔ Affichage des lignes 0 - 0 (total de 1, Traitement en 0.0111 sec)

SELECT * FROM orders WHERE date_order IN (SELECT max(date_order) FROM orders)

☐ Profilage [En ligne] [Modifier] [Expliquer SQL] [Créer source PHP] [Actualiser]

Nombre de lignes : 25 ▾

+ Options

←T→			▼ ID_ORDER	ID_CLIENT	date_order
☐	✎ Modifier	᠁ Copier ⊖ Effacer	33	7	2017-01-16 10:02:11

↑ ☐ Tout cocher Pour la sélection : ✎ Modifier ⊖ Effacer 📤 Exporter

Nombre de lignes : 25 ▾

the customer who made the maximum of command

⚠ La sélection courante ne contient pas de colonne unique. Les grilles d'édition, les cases à cocher ainsi que les liens Edition, Copie et Supprimer ne sont pas disponibles.

✔ Affichage des lignes 0 - 0 (total de 1, Traitement en 0.0009 sec)

SELECT id_client,COUNT(id_client) AS most_orders FROM orders GROUP BY id_client ORDER BY most_orders DESC LIMIT 1

☐ Profilage [En ligne] [Modifier] [Expliquer SQL] [Créer source PHP] [Actualiser]

+ Options

id_client	most_orders
1	7

in the table clients print how many man and woman we have

⚠ La sélection courante ne contient pas de colonne unique. Les grilles d'édition, les cases à cocher ainsi que les liens Edition, Copie et Supprimer ne sont pas disponibles.

✔ Affichage des lignes 0 - 1 (total de 2, Traitement en 0.0065 sec)

SELECT gender,COUNT(gender) AS client_gender FROM clients GROUP BY gender ORDER BY client_gender DESC

☐ Profilage [En ligne] [Modifier] [Expliquer SQL] [Créer source PHP] [Actualiser]

Nombre de lignes : 25 ▾

Trier sur l'index: Aucune ▾

+ Options

gender	client_gender ▾
M	14
F	10

#Step 4:

#1 Select from the products Table the following:

- ➢ All the products which have the price above 15:
- • SELECT * from products
  where price > 15;

- ➢ The product name and the price which contains the sentence 'special':
- • SELECT NAME,price from products
  WHERE NAME like '%special%';

- ➢ Product name which price != 8.5:
- • SELECT NAME from products
  WHERE price !=8.5;

➢ the name and price of all products with a coffee origin Equal to Cuba
  or Indonesia. Ordered by name from A-Z:
• select NAME,price from products where product_origin = "Indonesia" or
  product_origin = "Cuba" order by NAME ASC;

> Affichage des lignes 0 - 1 (total de 2, Traitement en 0.0008 sec) [NAME: **ANIMAL COFFEE - COFFEE FAMILY**]

```
select NAME,price from products where product_origin = "Indonesia" or product_origin = "Cuba" order by NAME ASC
```

☐ Profilage [En ligne] [Modifier] [Expliquer SQL] [Créer source PHP] [Actualiser]

Nombre de lignes : 25 ▾

Trier sur l'index: Aucune   ▾

+ Options

NAME ▲	price
animal coffee	25.00
coffee family	8.00

➢ the name, price and coffee origin but rename the price to retail_price
  in the results set:
•   SELECT NAME,product_origin,price as retail_price FROM products;

⚠ La sélection courante ne contient pas de colonne unique. Les grilles d'édition, les cases à cocher ainsi que les liens Edition, Copie et Supprimer ne sont pas disponibles.

> Affichage des lignes 0 - 5 (total de 6, Traitement en 0.0008 sec)

```
SELECT NAME,product_origin,price as retail_price FROM products
```

☐ Profilage [En ligne] [Modifier] [Expliquer SQL] [Créer source PHP] [Actualiser]

Nombre de lignes : 25 ▾

Trier sur l'index: Aucune   ▾

+ Options

NAME	product_origin	retail_price
special coffee	Brazil	7.50
coffee family	Cuba	8.00
animal coffee	Indonesia	25.00
coffee 2.0	Ethopia	8.50
coffee ++	Maroc	30.00
coffe plus	EST Europe	15.50

#2 Select from the table clients the following:

➢ the first name and phone number of all the females who have a last name of Bluth:
•   SELECT FIRST_NAME,PHONE_NUMBER FROM clients
    WHERE GENDER = "F" AND LAST_NAME LIKE "%BLUTH%";

> Affichage des lignes 0 - 1 (total de 2, Traitement en 0.0047 sec)

```
SELECT FIRST_NAME,PHONE_NUMBER FROM clients WHERE GENDER = "F" AND LAST_NAME LIKE "%BLUTH%"
```

☐ Profilage [En ligne] [Modifier] [Expliquer SQL] [Créer source PHP] [Actualiser]

Nombre de lignes : 25 ▾

Trier sur l'index: Aucune   ▾

+ Options

FIRST_NAME	PHONE_NUMBER
Lindsay	1176923804
Lucille	1198773214

➢ How many male customers don't have a phone number entered into the customers table:

- SELECT * FROM clients
  WHERE PHONE_NUMBER is null;

	ID_CLIENT	FIRST_NAME	LAST_NAME	GENDER	PHONE_NUMBER
☐ 🖉 Modifier 📋 Copier ⊖ Effacer	4	Daniel	Williams	M	NULL
☐ 🖉 Modifier 📋 Copier ⊖ Effacer	12	Harry	Johnson	M	NULL
☐ 🖉 Modifier 📋 Copier ⊖ Effacer	14	John	Taylor	M	NULL
☐ 🖉 Modifier 📋 Copier ⊖ Effacer	22	Jennifer	NULL	F	NULL

➢ the first name and phone number of all customers who's last name contains the pattern 'ar':

- SELECT FIRST_NAME,PHONE_NUMBER FROM clients
  WHERE LAST_NAME LIKE "%ar%";

FIRST_NAME	PHONE_NUMBER
Chris	1123147789
Katie	1145787353

➢ select distinct last names and order alphabetically from A-Z:

- SELECT DISTINCT LAST_NAME FROM clients
  order by LAST_NAME ASC;

LAST_NAME ▲
NULL
Armstrong
Bluth
Edmonds
Evans
Johnson
Law
Martin
Nash
Simmonds
Smith
Smithe
Taylor

#3 Select from the table command the following:

➤ all the orders from February 2017 for customers with id's of 2, 4, 6 or 8:
- select * from orders
  where ID_CLIENT = 2 or ID_CLIENT = 4 or (ID_CLIENT = 6 or ID_CLIENT = 8);

➤ the first 3 orders placed by customer with id 1, in january 2017:
- select * from orders
  where ID_CLIENT = 3 limit 3;

#Step 5:

Multiple Selection using Joins:

➤ Select the order id and clients phone number for all orders of
product id 4:
- SELECT Orders.ID_ORDER,clients.PHONE_NUMBER FROM Orders
  INNER JOIN clients
  ON Orders.ID_CLIENT = clients.ID_CLIENT
  AND Orders.ID_PRODUCT = 4;

☐ Show all | Number of rows: 25 ⌄ Filter rows: Search this table Sort by key: None ⌄

+ Options

ID_ORDER	PHONE_NUMBER
15	1174592013
20	NULL
25	1176348290
29	1966947113
30	1176987789
32	1176984116

☐ Show all | Number of rows: 25 ⌄ Filter rows: Search this table Sort by key: None ⌄

➢ Select the product name and order time for filter coffees sold between January 15th 2017 and February 14th 2017.

- SELECT products.NAME,orders.DATE_ORDER FROM orders
  INNER JOIN products
  ON orders.ID_product = products.ID_product;

☐ Show all | Number of rows: 25 ⌄ Filter rows: Search this table Sort by key: None ⌄

+ Options

NAME	PRICE	DATE_ORDER
coffee family	8.00	2017-01-02 10:10:10
coffee family	8.00	2017-01-03 13:41:22
coffee family	8.00	2017-01-07 09:12:39
coffee family	8.00	2017-01-07 11:24:15
coffee niore	8.00	2017-01-01 08:44:34
coffee niore	8.00	2017-01-03 11:05:33
coffe plus	15.50	2017-01-01 13:07:10
coffe plus	15.50	2017-01-03 09:12:11
coffe plus	15.50	2017-01-10 10:34:12

☐ Show all | Number of rows: 25 ⌄ Filter rows: Search this table Sort by key: None ⌄

➢ Select the product name and price and order time for all orders from females in January 2017.

- SELECT products.NAME,products.PRICE,orders.DATE_ORDER FROM orders
  INNER JOIN products
  ON orders.ID_product = products.ID_product
  INNER JOIN clients
  ON orders.ID_product = clients.ID_CLIENT
  AND clients.GENDER = "F";

☐ Show all | Number of rows: 25 ⌄ Filter rows: Search this table Sort by key: None ⌄

+ Options

NAME	PRICE	DATE_ORDER
coffee family	8.00	2017-01-02 10:10:10
coffee family	8.00	2017-01-03 13:41:22
coffee family	8.00	2017-01-07 09:12:39
coffee family	8.00	2017-01-07 11:24:15
coffee niore	8.00	2017-01-01 08:44:34
coffee niore	8.00	2017-01-03 11:05:33
coffe plus	15.50	2017-01-01 13:07:10
coffe plus	15.50	2017-01-03 09:12:11
coffe plus	15.50	2017-01-10 10:34:12

☐ Show all | Number of rows: 25 ⌄ Filter rows: Search this table Sort by key: None ⌄

Realise par : BOUKRIM Abdeladim

#FIN#