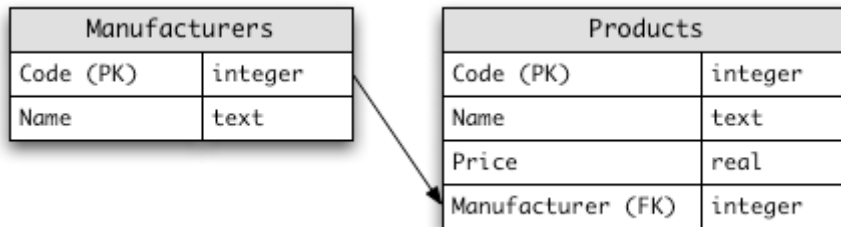


ATELIER 1



```
CREATE TABLE Manufacturers (
    Code INTEGER PRIMARY KEY NOT NULL,
    Name CHAR(50) NOT NULL
);

CREATE TABLE Products (
    Code INTEGER PRIMARY KEY NOT NULL,
    Name CHAR(50) NOT NULL ,
    Price REAL NOT NULL ,
    Manufacturer INTEGER NOT NULL
    CONSTRAINT fk_Manufacturers_Code REFERENCES
Manufacturers(Code)
);
```

```
INSERT INTO Manufacturers(Code,Name) VALUES (1,'Sony');
INSERT INTO Manufacturers(Code,Name) VALUES (2,'Creative Labs');
INSERT INTO Manufacturers(Code,Name) VALUES (3,'Hewlett-Packard');
INSERT INTO Manufacturers(Code,Name) VALUES (4,'Iomega');
INSERT INTO Manufacturers(Code,Name) VALUES (5,'Fujitsu');
INSERT INTO Manufacturers(Code,Name) VALUES (6,'Winchester');

INSERT INTO Products(Code,Name,Price,Manufacturer) VALUES (1,'Hard
drive',240,5);
INSERT INTO Products(Code,Name,Price,Manufacturer)
VALUES (2,'Memory',120,6);
INSERT INTO Products(Code,Name,Price,Manufacturer) VALUES (3,'ZIP
drive',150,4);
INSERT INTO Products(Code,Name,Price,Manufacturer) VALUES (4,'Floppy
disk',5,6);
INSERT INTO Products(Code,Name,Price,Manufacturer)
VALUES (5,'Monitor',240,1);
INSERT INTO Products(Code,Name,Price,Manufacturer) VALUES (6,'DVD
drive',180,2);
INSERT INTO Products(Code,Name,Price,Manufacturer) VALUES (7,'CD
drive',90,2);
INSERT INTO Products(Code,Name,Price,Manufacturer)
VALUES (8,'Printer',270,3);
```

```
INSERT INTO Products (Code, Name, Price, Manufacturer) VALUES (9, 'Toner cartridge', 66, 3);
INSERT INTO Products (Code, Name, Price, Manufacturer) VALUES (10, 'DVD burner', 180, 2);
```

5/ Sélectionnez les noms de tous les produits du magasin.

SELECT Name FROM Products;

6/ Sélectionner les noms et les prix de tous les produits du magasin.

SELECT Name, Price FROM Products;

7/ Sélectionner le nom des produits dont le prix est inférieur ou égal à 200 \$.

SELECT Name FROM Products WHERE Price <= 200;

8/ Sélectionnez tous les produits dont le prix est compris entre 60 et 120 dollars.

SELECT * FROM Products

WHERE Price >= 60 AND Price <= 120;

9/ Sélectionnez le nom et le prix en cents (c'est-à-dire que le prix doit être multiplié par 100).

SELECT Name, Price * 100 FROM Products;

10/ Calculer le prix moyen de tous les produits.

SELECT AVG(Price) FROM Products;

11/ Calculer le prix moyen de tous les produits dont le code fabricant est égal à 2.

SELECT AVG(Price) FROM Products WHERE Manufacturer=2;

12/ Calculer le nombre de produits dont le prix est supérieur ou égal à 180 dollars.

SELECT COUNT(*) FROM Products WHERE Price >= 180;

13/ Sélectionner le nom et le prix de tous les produits dont le prix est supérieur ou égal à 180 dollars, et trier d'abord par prix (par ordre décroissant), puis par nom (par ordre croissant).

SELECT Name, Price FROM Products WHERE Price >= 180 ORDER BY Price DESC, Name;

14/ Sélectionnez toutes les données des produits, y compris toutes les données relatives au fabricant de chaque produit.

SELECT * FROM Products, Manufacturers

WHERE Products.Manufacturer = Manufacturers.Code;

15/ Sélectionnez le nom du produit, le prix et le nom du fabricant de tous les produits.

```
SELECT Products.Name, Price, Manufacturers.Name
```

```
FROM Products, Manufacturers
```

```
WHERE Products.Manufacturer = Manufacturers.Code;
```

16/ Sélectionnez le prix moyen des produits de chaque fabricant, en indiquant uniquement le code du fabricant.

```
SELECT AVG(Price), Manufacturer FROM Products GROUP BY Manufacturer;
```

17/ Sélectionnez le prix moyen des produits de chaque fabricant, en indiquant le nom du fabricant.

```
SELECT AVG(Price), Manufacturers.Name
```

```
FROM Products, Manufacturers
```

```
WHERE Products.Manufacturer = Manufacturers.Code
```

```
GROUP BY Manufacturers.Name;
```

18/ Sélectionnez les noms des fabricants dont les produits ont un prix moyen supérieur ou égal à 150 \$.

```
SELECT AVG(Price), Manufacturers.Name
```

```
FROM Products, Manufacturers
```

```
WHERE Products.Manufacturer = Manufacturers.Code
```

```
GROUP BY Manufacturers.Name
```

```
HAVING AVG(Price) >= 150;
```

19/ Sélectionnez le nom et le prix du produit le moins cher.

```
SELECT Name, Price FROM Products WHERE Price = (SELECT MIN(Price) FROM Products);
```

20/ Sélectionnez le nom de chaque fabricant ainsi que le nom et le prix de son produit le plus cher.

```
SELECT max_price_mapping.name as manu_name, max_price_mapping.price,  
products_with_manu_name.name as product_name FROM
```

```
(SELECT Manufacturers.Name, MAX(Price) price
```

```
FROM Products, Manufacturers
```

```
WHERE Manufacturer = Manufacturers.Code
```

```
GROUP BY Manufacturers.Name)
```

```
as max_price_mapping LEFT JOIN
```

```
(SELECT products.*, manufacturers.name manu_name
```

```
FROM products join manufacturers
```

```
ON (products.manufacturer = manufacturers.code))
```

```
AS products_with_manu_name ON  
(max_price_mapping.name = products_with_manu_name.manu_name  
AND
```

```
max_price_mapping.price = products_with_manu_name.price);
```

21/ Ajouter un nouveau produit : Loudspeakers, 70 \$, manufactur 2

```
INSERT INTO Products( Code, Name , Price , Manufacturer)
```

```
VALUES ( 11, 'Loudspeakers' , 70 , 2 );
```

22/ Mettre à jour le nom du produit 8 en "laser Print".

```
UPDATE Products SET Name = 'Laser Printer' WHERE Code = 8;
```

23/ Appliquer une remise de 10 % à tous les produits.

```
UPDATE Products SET Price = Price - (Price * 0.1);
```

24/ Appliquer une remise de 10 % à tous les produits dont le prix est supérieur ou égal à 120 \$.

```
UPDATE Products SET Price = Price - (Price * 0.1) WHERE Price >= 120;
```