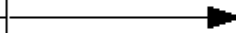


ATELIER 3 :

| Warehouses | | |
|------------|----------|---------|
| PK | Code | integer |
| | Location | text |
| | Capacity | integer |



| Boxes | | |
|-------|-----------|---------|
| PK | Code | text |
| | Contents | text |
| | Value | real |
| FK | Warehouse | integer |

--3.1 Sélectionner tous les entrepôts.

```
SELECT * FROM `warehouses`
```

--3.2 Sélectionnez toutes les cases dont la valeur est supérieure à 150 \$.

```
SELECT `Contents`, `Value` FROM `boxes` WHERE VALUE > 150
```

```
select * from boxes where Value>150;
```

--3.3 Sélectionner tous les contenus distincts dans toutes les cases.

```
SELECT DISTINCT Contents FROM `boxes`
```

--3.4 Sélectionner la valeur moyenne de toutes les boîtes.

```
SELECT AVG(Value) FROM `boxes`
```

--3.5 Sélectionner le code de l'entrepôt et la valeur moyenne des boîtes dans chaque entrepôt.

```
SELECT AVG(Value), warehouse FROM boxes GROUP BY warehouse
```

--3.6 Identique à l'exercice précédent, mais ne sélectionnez que les entrepôts où la valeur moyenne des boîtes est supérieure à 150.

```
SELECT AVG(Value), warehouse FROM boxes GROUP BY warehouse HAVING AVG(Value) > 150
```

--3.7 Sélectionnez le code de chaque boîte, ainsi que le nom de la ville dans laquelle la boîte est située.

```
SELECT boxes.Code , warehouses.Location FROM boxes INNER JOIN warehouses ON  
boxes.Warehouse = warehouses.Code
```

```
select boxes.code, warehouses.location from boxes join warehouses on
```

```
boxes.Warehouse = Warehouses.Code;
```

```
SELECT Boxes.Code, Location FROM Warehouses INNER JOIN Boxes ON Warehouses.Code =  
Boxes.Warehouse;
```

--3.8 Sélectionnez les codes des entrepôts, ainsi que le nombre de boîtes dans chaque entrepôt.

-- Facultativement, tenez compte du fait que certains entrepôts sont vides (c'est-à-dire que le nombre de boîtes devrait apparaître comme zéro, au lieu d'omettre l'entrepôt du résultat).

```
SELECT Warehouse , COUNT(*) FROM boxes GROUP BY Warehouse
```

--3.9 Sélectionnez les codes de tous les entrepôts qui sont saturés (un entrepôt est saturé si le nombre de boîtes qu'il contient est supérieur à la capacité de l'entrepôt).

```
SELECT Code FROM warehouses WHERE Capacity < ( SELECT COUNT(*) FROM boxes WHERE Warehouse = warehouses.Code )
```

--3.10 Sélectionnez les codes de toutes les boîtes situées à Chicago.

```
SELECT Code FROM `boxes` WHERE Warehouse IN ( SELECT Code FROM warehouses WHERE Location = ('Chicago') )
```

--3.11 Créer un nouvel entrepôt à New York avec une capacité de 3 boîtes.

```
INSERT INTO `warehouses`(`Code`, `Location`, `Capacity`) VALUES (6, 'New York' ,3)
```

--3.12 Créer une nouvelle boîte, avec le code "H5RT", contenant des "Papers" d'une valeur de 200 \$, et située dans l'entrepôt 2.

```
INSERT INTO `boxes`(`Code`, `Contents`, `Value`, `Warehouse`) VALUES ("H5RT","Papers", 200 ,2 )
```

--3.13 Réduire la valeur de toutes les boîtes de 15 %.

```
UPDATE `boxes` SET `Value`= Value*0.85
```

--3.14 Retirer toutes les boîtes d'une valeur inférieure à 100 \$.

```
DELETE FROM `boxes` WHERE Value < 100
```

-- 3.15 Retirer toutes les boîtes des entrepôts saturés.

```
DELETE FROM `boxes` WHERE Warehouse IN ( SELECT Code FROM warehouses WHERE Capacity < ( SELECT COUNT(*) FROM boxes WHERE Warehouse = warehouses.Code ) )
```

-- 3.16 Ajouter un indice pour la colonne "Entrepôt" dans le tableau "boîtes".

-- !!!NOTE!! : l'index ne doit PAS être utilisé sur des petites tables dans la pratique

```
CREATE INDEX `index_nom` ON `table` (`colonne1`);  
CREATE INDEX `index_Warehouse` ON boxes (Warehouse);
```

-- 3.17 Imprimer tous les index existants

-- !!!NOTE!! : l'index ne doit PAS être utilisé sur des petites tables dans la pratique

```
SHOW INDEX FROM `boxes`
```

NB : Entrepôt c'est warehouse // les cases : boxes