

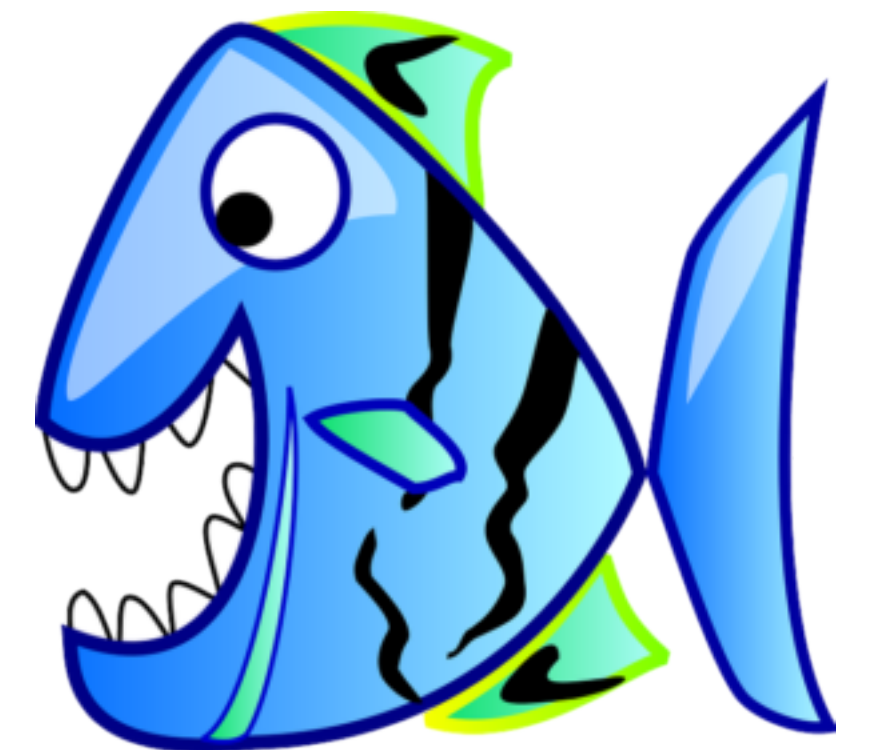
# SQL et Bases de données

## Cours 4

Jean-Jacques Lévy

`jean-jacques.levy@inria.fr`

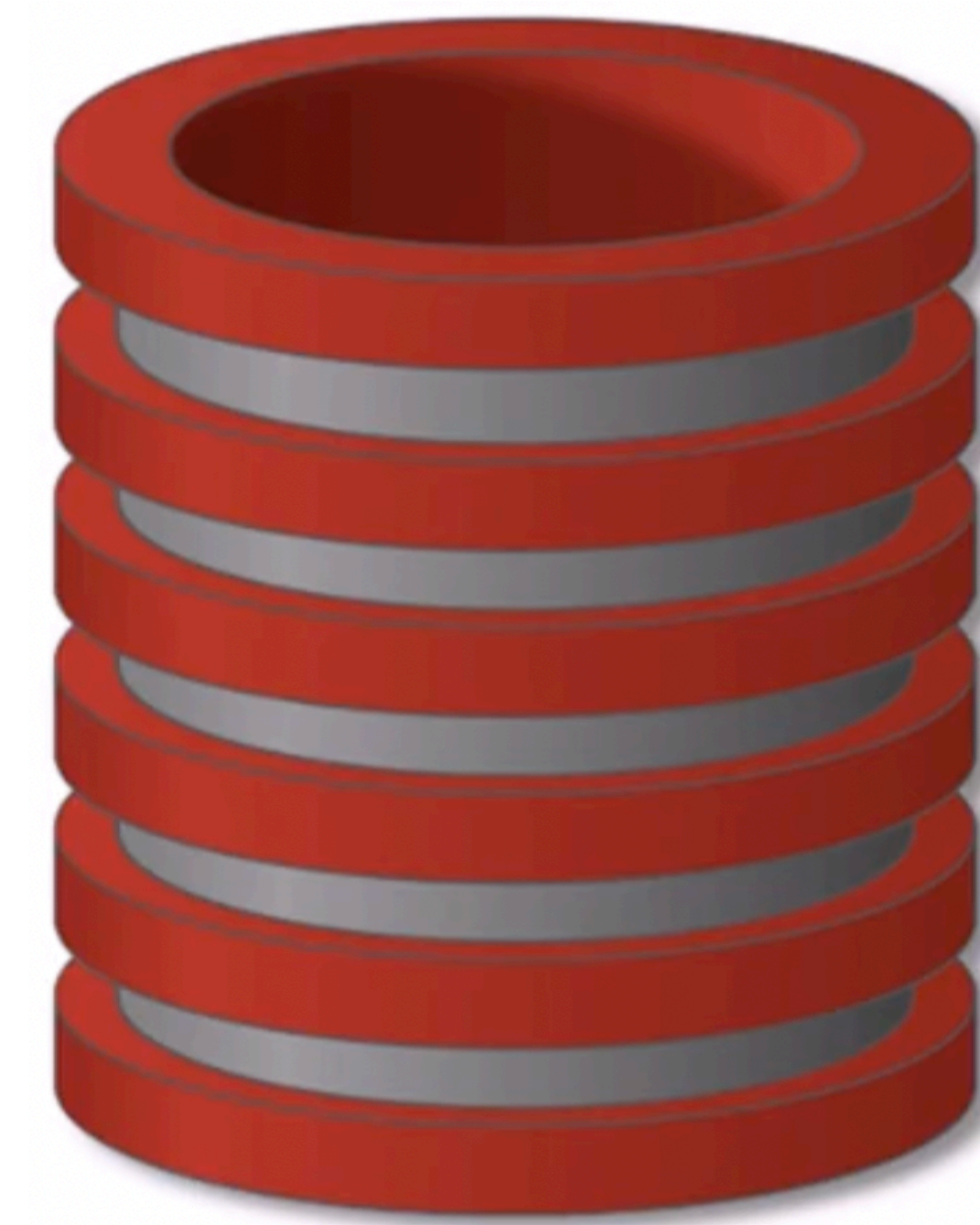
`http://jeanjacqueslevy.net/lp-sql`



1 avril 2022

# Plan

- modifications
- attributs de groupe
- valeurs NULL
- jokers et filtrage
- join et contraintes
- indexation (début)



- deux bons tutoriels

<http://www.w3schools.com/sql/default.asp>

<http://www.programiz.com/sql>



## client

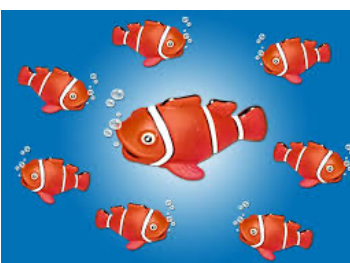
cID	nom	actif	ville
1	Tom	23000	Bordeaux
2	Jean-Jacques	38000	Paris
3	Martin	51000	Nice
4	Kiki	54000	Pekin
5	Iteki	84000	Tokyo
6	Bob	6100	Nice
7	Albert	12000	Bordeaux
8	Manu	8150	Paris
9	Valou	10300	Bordeaux
10	Joe	32500	Nice
11	Helmut	8150	Paris
12	Martine	11200	Bordeaux
13	Marina	9150	Nice
14	Masha	10290	Nice
15	Julia	32000	Paris
17	Bob	38000	Nice

## produit

pID	pNom	pCat	pVille	prix
1	clio	auto	Paris	13000
2	audi	auto	Paris	45000
3	tesla	auto	Pekin	70000
4	tesla	auto	Nice	40000
5	yamaha	moto	Tokyo	8000
6	kawasaki	moto	Tokyo	8000
7	megamo	velo	Paris	3240
8	shimano	velo	Paris	1900
9	btwin	velo	Nice	990
10	triban	velo	Nice	690
11	peugeot	velo	Paris	750
12	bertin	eVelo	Paris	1190
13	trek	eVelo	Bordeaux	1390
14	trek	eVelo	Paris	1350

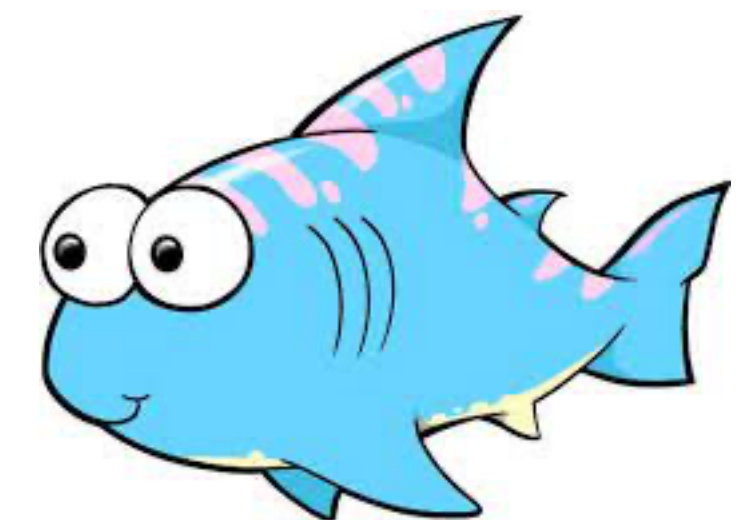
## devis

cID	pNom	dCat	commande
2	peugeot	velo	0
7	NULL	velo	0
6	trek	eVelo	0
8	NULL	auto	0
8	NULL	velo	0
8	NULL	eVelo	0
9	honda	auto	0
11	NULL	auto	0
4	honda	moto	0
5	NULL	moto	0
8	triban	velo	0
13	NULL	velo	0
11	yaris	auto	0
12	NULL	velo	0
1	NULL	eVelo	0



# Modifications

- jour de chance: tous les clients reçoivent 10000€ -> modifier la table client.
- spécial Bordeaux: tous les clients reçoivent 10000€ -> modifier la table client.
- créer la table des clients de Bordeaux
- retirer Helmut et Masha de la table des clients



# Attributs de groupe

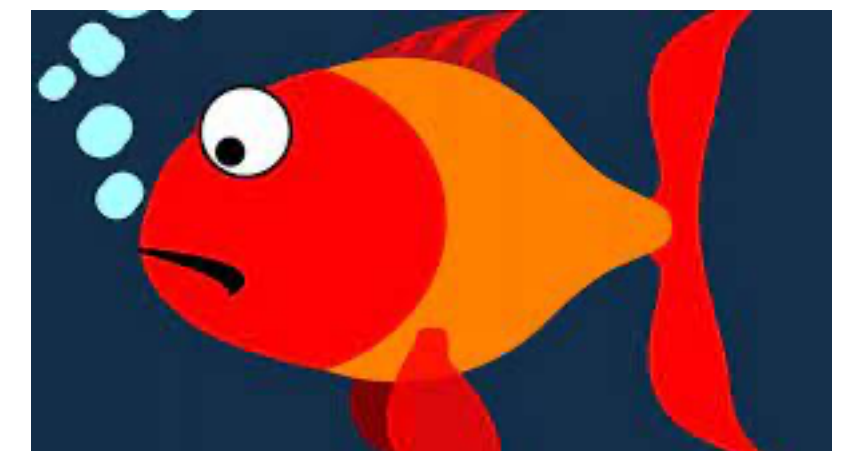
- on peut regrouper les n-uplets avec **GROUP** selon des attributs et utiliser une commande de groupe

```
SELECT column_name(s)
FROM table_name
WHERE condition
GROUP BY column_name(s)
HAVING condition
ORDER BY column_name(s);
```

- commandes de groupe **MIN**, **MAX**, **COUNT**, **AVG**, **SUM**

```
select ville, count(*)
from client
group by
ville;
```

[ ne pas oublier **DISTINCT** ]



# Attributs de groupe

- trouver l'actif moyen des acheteurs de motos
- lister tous les clients qui veulent acheter une moto
- compter les acheteurs de motos
- compter les acheteurs de motos par ville
- lister les clients qui ont le même actif à 10000€ près
- compter le nombre de clients qui ont le même actif à 10000€ près
- lister les actifs moyens des acheteurs de velos, de eVelos, de motos, d'autos
- compter les acheteurs par catégorie de produits
- lister les clients ordonnés par nom de produits et actifs



# Attributs de groupe

- lister les cID des clients avec le nombre de produits qu'ils veulent acheter et lister les clients qui ne veulent rien acheter
- lister les produits avec plus que 3 devis
- lister les produits avec plus que 3 acheteurs
- lister les catégories de produits avec des devis dont les actifs des acheteurs sont inférieurs à la moyenne des actifs  
[ « les les catégories de produits achetés par les plus pauvres » ]



# Attributs NULL

- les attributs inexistant ont la valeur **NULL**

```
select * from devis  
where pNom is NULL;
```

[ remarque: ne pas utiliser =, mais utiliser **IS** ]

- le résultat de tests avec des attributs nuls peut être aléatoire





# Prochains cours

- indexation
- introduction au requêtes concurrentes
- transactions
- correspondance avec la logique du 1er ordre
- interface avec Python, HTML et Javascript



# Solutions des exercices

- lister tous les clients qui veulent acheter un velo

```
select client.cId, nom, actif  
from client, devis  
where client.cID = devis.cID  
and devis.dCat = "velo";
```

cID	nom	actif
2	Jean-Jacques	68000
7	Albert	22000
8	Manu	18150
8	Manu	18150
13	Marina	19150
12	Martine	21200

- trouver l'actif moyen des acheteurs de velos

```
select avg (actif)  
from client, devis  
where client.cID = devis.cID  
and devis.dCat = "velo";
```

avg (actif)
27775.0

- compter les acheteurs de velos

```
select count (*)  
from client, devis  
where client.cID = devis.cID  
and devis.dCat = "velo";
```

count (*)
6

- compter les acheteurs de velos par ville

```
select ville, count (*)  
from client, devis  
where client.cID = devis.cID  
and devis.dCat = "velo"  
group by ville;
```

ville	count (*)
Bordeaux	2
Nice	1
Paris	3



# Solutions des exercices

- lister les clients qui ont le même actif à 1000€ près

```
select C1.cID, C1.nom, C1.actif, C2.cID, C2.nom, C2.actif
from client as C1, client as C2
where C1.cID < C2.cID
and abs (C2.actif - C1.actif) <= 1000
```

cID	nom	actif	cID	nom	actif
7	Albert	22000	12	Martine	21200
8	Manu	18150	13	Marina	19150
9	Valou	20300	12	Martine	21200
10	Joe	42500	15	Julia	42000

- compter le nombre de clients qui ont le même actif à 1000€ près

```
select count (distinct C1.nom)
from client as C1, client as C2
where C1.cID <> C2.cID
and abs (C2.actif - C1.actif) <= 1000 ;
```

count (distinct C1.nom)
7



# Solutions des exercices

- compter les acheteurs par catégorie de produits

```
select * from client, devis
where client.cID = devis.cID
order by dCat;
```

```
select count(distinct nom) from client, devis
where client.cID = devis.cID
group by dCat
order by dCat;
```

count(distinct nom)
2
3
2
5

- lister les actifs moyens des acheteurs de velos, de eVelos, de motos, d'autos

(idem en changeant **COUNT** par **AVG**)

- lister les clients ordonnés par nom de produits et actifs

```
select nom, pNom, dCat from client, devis
where devis.cID = client.cID
order by pNom, actif;
```

cID	nom	actif	ville	cID	pNom	dCat	commande
8	Manu	18150	Paris	8	NULL	auto	0
9	Valou	20300	Bordeaux	9	honda	auto	0
6	Bob	16100	Nice	6	trek	eVelo	0
8	Manu	18150	Paris	8	NULL	eVelo	0
1	Tom	33000	Bordeaux	1	NULL	eVelo	0
4	Kiki	64000	Pekin	4	honda	moto	0
5	Iteki	94000	Tokyo	5	NULL	moto	0
2	Jean-Jacques	68000	Paris	2	peugeot	velo	0
7	Albert	22000	Bordeaux	7	NULL	velo	0
8	Manu	18150	Paris	8	NULL	velo	0
8	Manu	18150	Paris	8	triban	velo	0
13	Marina	19150	Nice	13	NULL	velo	0
12	Martine	21200	Bordeaux	12	NULL	velo	0



# Solutions des exercices

- lister les cID des clients avec le nombre de produits qu'ils veulent acheter

```
select client.cID, nom, count(devis.cID)
from client, devis
where client.cID = devis.cID
group by devis.cID;
```

- lister les clients qui ne veulent rien acheter

```
select distinct client.cID, nom
from client, devis
where client.cID not in
(select devis.cID from devis);
```

cID	nom
3	Martin
10	Joe
15	Julia
17	Bob
18	Tim
19	Al

cID	nom	count(devis.cID)
1	Tom	1
2	Jean-Jacques	1
4	Kiki	1
5	Iteki	1
6	Bob	1
7	Albert	1
8	Manu	4
9	Valou	1
12	Martine	1
13	Marina	1



# Solutions des exercices

- lister les catégories de produits avec des devis dont les actifs des acheteurs sont inférieurs à la moyenne des actifs diminuée de 20000€

[ « les les catégories de produits achetés par les plus pauvres » ]

```
select avg (actif) from client;
```

```
select dCat, avg(actif) as actifCatMoyen
from
(select distinct dCat, actif, client.cID
from devis, client
where client.cID = devis.cID
order by dCat, client.cID)
group by dCat;
```

avg (actif)
43712.5

dCat	actifCatMoyen
auto	19225.0
eVelo	22416.6666666667
moto	79000.0
velo	31700.0

```
select dCat, actifCatMoyen
from
(select dCat, avg(actif) as actifCatMoyen
from
(select distinct dCat, actif, client.cID
from devis, client
where client.cID = devis.cID
order by dCat, client.cID)
group by dCat)
where actifCatMoyen >
(select avg(actif) from client
as actifMoyen) - 20000;
```



dCat	actifCatMoyen
moto	79000.0
velo	29700.0