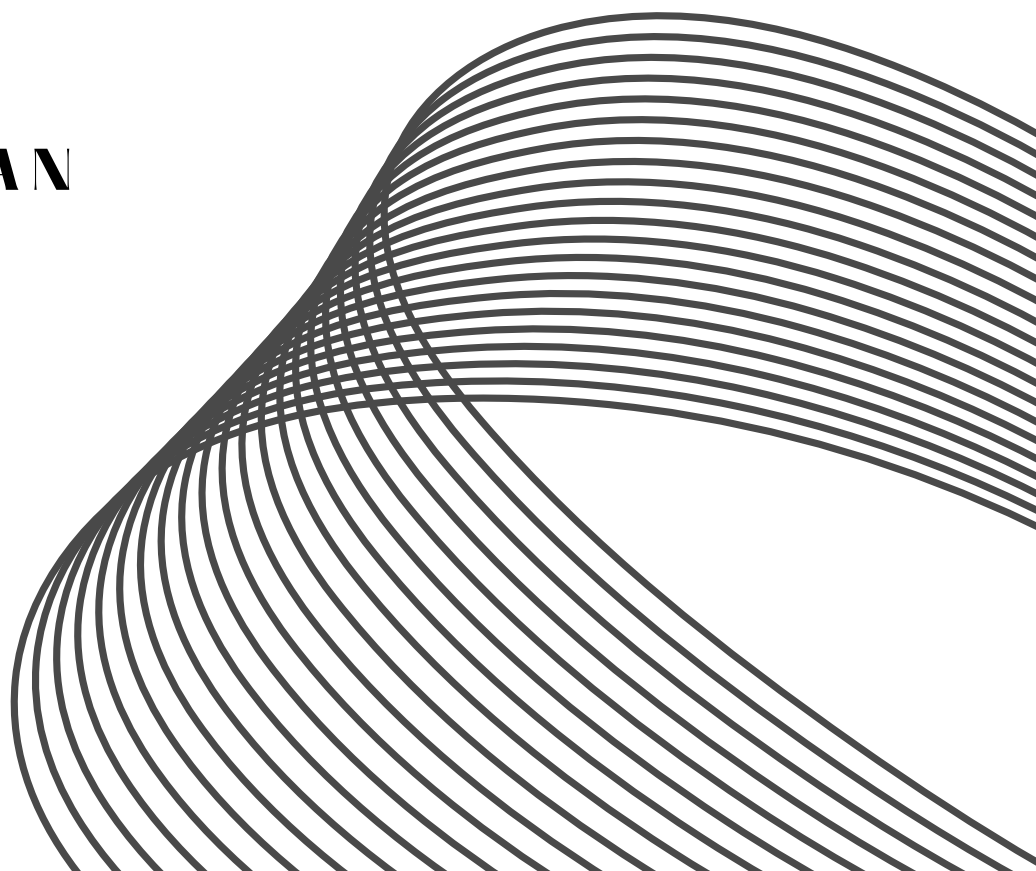


Rapport SAE 2.04

Exploitation d'une base de données
VAE

AUTEUR
**GNANESWARAN
ROSHAN**

PARTIE
SECTION 3,4
SYNTHESE
ANALYSE
DEMONSTRATION



Section 3

Liste des objets volées correspondant à la description

- L'objet a été mis en vente en début de mois (avant le 15)
- Il a été mis en vente à une somme inférieure à 500€.
- Il a été vendu à plus de 10 fois le prix de vente initial.

CREATE or REPLACE VIEW Enquete **as**

```
select distinct idOb,nomOb, prixBase prix, max(montant) montant, e.idUt, u.pseudout
from OBJET natural join VENTE v natural join STATUT join ENCHERIR e ON e.idve = v.idve join
where DAY(debutVe) <15 and prixBase <500 and idSt = 4
group by idOb
having montant > 10*prixBase;
```

J'ai donc créé une vue qui me permet de récupérer un tableau avec les informations suivantes :

- idOb : l'identifiant de l'objet
- nomOb : le nom de l'objet
- prix : le prix de base de l'objet
- montant : le montant de l'enchère
- idUt : l'identifiant de l'utilisateur
- pseudout : le pseudo de l'utilisateur

Afin d'avoir uniquement la liste des objets avec l'idOb et le nomOb, j'ai utilisé la commande suivante :

```
select idOb, nomOb
from Enquete;
```

Pour avoir le nom des suspects, j'ai utilisé la commande suivante :

```
select idUt, pseudout
from Enquete;
```

Liste des objets correspondant à la description :

Liste des suspects :

Section 4

Ecrire les instructions SQL permettant de décrire que l'utilisation IUTO d'adresse mail iuto@ info.univ-orleans.fr qui est actif et dont le mot de passe est IUTO met en vente un canapé clic-clac "très beau et ayant peu servi" à partir du 23 mars 2023 jusqu'au 30 mars 2024 pour un prix de base de 40€ et un prix minimum de 80€. Cet article n'a pas de photo associée.

Nous commençons déjà par voir pour chaque table UTILISATEUR, OBJET et VENTE. Le dernier identifiant en effectuant un select max sur chaque table.

idOb	nomOb
1	Pantalon de qualité
40	Commode jamais servie
50	T-shirt jaune
54	Armoire comme neuve
59	T-shirt jaune
84	Perceuse jamais servie
92	Scie verte
112	Pantalon bleu
133	T-shirt comme neuf
141	Robot ménager de qualité
146	Pantalon vert
159	Chapeau bleu
161	Armoire verte
163	Marteau comme neuf
173	Chemise bleue
183	Marteau jaune
215	Scie jamais servie
241	T-shirt rouge
261	Marteau bleu
268	Chemise comme neuve
270	Casserole rouge
277	Commode jamais servie
285	Marteau bleu
300	T-shirt vert
315	Chaise rouge
325	Buffet rouge
330	Marteau vert
333	Casserole rouge
335	Chaise rouge
337	Armoire bleue
362	Chemise de qualité
382	Marteau jaune
386	Commode de qualité
388	Tournevis comme neuf
390	Robot ménager rouge
398	Chaise jaune
404	Cocotte verte
412	T-shirt de qualité
415	Pantalon comme neuf
417	Buffet jamais servi
433	Cocotte de qualité
446	Robe de qualité
512	Pantalon jaune

43 rows in set (0,133 sec)

Figure 1: Liste des objets

idUt	pseudout
39	votante35
8	ridou
101	bentaupon
349	vauduvu39
293	dounoipau3
45	foiton23
11	goiti065
62	nivon01
182	foncytu7
281	koududon575
196	jebena77
60	pesoi73
421	camy03
128	joroman01
101	bentaupon
55	daubi
170	nirato911
566	bebon4
62	nivon01
128	joroman01
63	lubontau865
55	daubi
62	nivon01
19	poiculo
49	koururen930
75	jongen47
77	tondouca223
195	jugu42
44	meveca958
226	loide440
126	fubaurou
36	janjencan913
45	foiton23
46	farensoi651
221	jenen43
72	roraulu18
124	vusiti94
333	bato581
64	moican944
105	poimoijen212
84	patoi2
255	couva
117	nansaujou647

43 rows in set (0,131 sec)

Figure 2: Liste des suspects

```
select max(idUt) from UTILISATEUR;
select max(idOb) from OBJET;
select max(idVe) from VENTE;
```

Nous avons donc les identifiants suivants en ajoutant 1 à chaque résultat :

- idUt : 1002
- idOb : 515
- idVe : 515

Nous pouvons donc insérer les données dans les tables UTILISATEUR, OBJET et VENTE.

```
insert into UTILISATEUR values (1002,'IUTO','iuto@info.univ-orleans.fr','IUTO','0',2);

insert into OBJET(idOb,nomOb,descriptionOb,idCat,idUt) values
(515,'canape clic-clac','très beau et ayant peu servi',3,1002);

insert into VENTE(idVe,prixBase,prixMin,debutVe,finVe,idSt,idOb) values
(515,40,80, STR_TO_DATE('23/03/2023:10:00:00','%d/%m/%Y:%h:%i:%s'),STR_TO_DATE('30/03/2024:10:00:00','%d/%m/%Y:%h:%i:%s'),515);
```

Synthèse

Lors de cette SAE, j'ai réussi tout ce qui était demandé dans le sujet. J'ai pu exploiter une base de données, insérer des données, faire des requêtes simples et des requêtes plus complexes. J'ai pu également créer des vues afin d'avoir des requêtes bien plus optimisées et efficace.

Analyse

J'ai rencontré des difficultés lors de la création de la vue Enquete. J'ai eu des problèmes avec les jointures et les conditions. Notamment à cause des jointures naturelles qui ne fonctionnent pas toujours. C'est pour cela que j'ai du faire des jointures explicites avec la JOIN et ON. J'ai également eu des difficultés à comprendre comment fonctionnait la commande STR_TO_DATE dans le cadre de cette base de données, j'ai donc du chercher dans le jeu de données pour trouver un exemple de date afin de pouvoir la reproduire.

Dans la section 5, j'ai eu du mal avec la dernière requête pour le top des vendeurs toujours à cause des jointures naturelles. J'ai donc appliqué la même méthode que pour la vue Enquete.

N'ayant pas pu lancer le programme de conversion de sql à csv. En effet le script a été conçu pour fonctionner avec le serveur mariaDB, servinfo et par conséquent ne fonctionne pas avec le serveur personnel. Du moins, je n'ai pas réussi à le faire fonctionner. J'ai donc du faire des recherches sur internet afin de trouver une solution. J'ai trouvé une méthode dans la requête permettant de convertir le résultat d'une requête en csv. J'ai donc utilisé cette méthode pour convertir les résultats des requêtes 8 et 9 en csv.

Ce qui donne les requêtes suivantes :

```
select idCat, nomCat, count(idOb) nb_objets
from CATEGORIE natural join OBJET natural join VENTE
where YEAR(finVe) = 2022 and idSt = 4
group by idCat
INTO OUTFILE './ObjetsVendu.csv'
FIELDS TERMINATED BY ';'
ENCLOSED BY '"'
LINES TERMINATED BY '\n';

SELECT u.idUt, u.pseudout, SUM(pv.montant) montant
FROM PRIXVENTE pv
JOIN VENTE v ON pv.idve = v. idve
JOIN OBJET o ON v.idob = o. idob
JOIN UTILISATEUR u ON o.idut = u. idut
WHERE MONTH(finVe) = 01 and YEAR(finVe) = 2023
GROUP BY u. idut
ORDER BY montant DESC
LIMIT 10;
INTO OUTFILE './vendeurs.csv'
FIELDS TERMINATED BY ';'
ENCLOSED BY '"'
LINES TERMINATED BY '\n';
```

Grace à ces requêtes, j'ai pu obtenir les fichiers csv et créer les graphiques demandés.

Démonstration de compétences

Les compétences que j'ai pu développer lors de cette SAE sont les suivantes :

- AC14.01 : Mettre à jour et interroger une base de données relationnelle (en requêtes directes ou à travers une application)

J'ai donc pu mettre à jour la base de données à l'aide des insertions UTILISATEUR, VENTE et OBJET. J'ai également pu interroger la base de données à l'aide des requêtes afin d'obtenir les résultats demandés.

- AC14.02 : Visualiser des données

J'ai pu visualiser les données à l'aide des requêtes 8 et 9. J'ai également pu visualiser les données à l'aide des graphiques en les ayant convertis en csv. J'ai également pu exploiter les données avec la section 7. En effet, j'ai pu des statistiques sur les ventes et le nombre d'enchères. Ce qui m'a permis d'avoir un nuage de points et de pouvoir visualiser les données.