

QUESTIONS BUSINESS

INTRODUCTION

Ce fichier est une suite de 12 questions qui vous permettront de prendre en main les données de production et de répondre à des problématiques business.

Chaque question suivra la trame suivante :

- Demande du business
- Requête SQL (vous devez y insérer une capture d'écran)
- Réponse (vous devez y insérer une capture d'écran)

QUESTION 1

Nous avons besoin de connaître la quantité totale produite par variété pour la ligne 1 par ordre décroissant.

REQUETE(S) SQL

```
Requête Historique

1 SELECT Product_ID AS Produit,
2 SUM(NB_Produits) AS Quantité,
3 MIN(DATE_DEBUT) AS DÉbut,
4 MAX(DATE_HEURE) AS Fin
5 FROM L1_Production
6 GROUP BY Product_ID
7 ORDER BY SUM(NB_Produits) DESC;
```

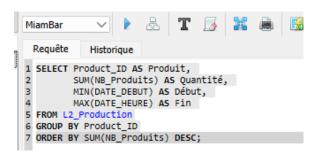
```
💈 🗸 🔣 🕒 1 🗗 🗃 🗎 Nombre de lignes chargées : 30
             Quantité Début
    1932773 16139656 2021-04-01 03:05:25.103 2022-06-01 01:57:01.063
2 1932777 13245824 2021-04-02 15:20:54.237 2022-05-31 23:37:59.540
3 1932789 8520927 2021-04-01 06:49:51.657 2022-05-31 20:59:58.497
4 1932782 7661481 2021-04-06 15:28:17.923 2022-05-30 23:13:07.497
5 1932786 7650328 2021-04-06 14:01:00.657 2022-05-30 20:23:50.283
    1932793 6970277 2021-04-01 00:12:44.683 2022-05-26 05:26:16.793
    1932794 6943390 2021-04-01 04:21:02.723 2022-05-26 06:49:56.543
    1932792 5582038 2021-04-02 16:33:05.507 2022-05-31 22:07:25.243
9 1932797 4015392 2021-04-01 01:04:52.220 2022-05-26 07:58:31.817
10 1932796 4010304 2021-04-01 01:55:24.410 2022-05-26 04:13:21.363
11 1932852 3882185 2021-04-02 04:37:52.770 2022-05-26 17:48:03.587
12 1932853 3881841 2021-04-02 06:08:27.263 2022-05-26 16:38:11.327
13 1962961 3543525 2021-04-20 14:38:55.523 2022-05-20 20:58:47.823
14 1962953 3518071 2021-04-20 20:12:13.690 2022-05-21 01:46:40.957
    1932864 2282084 2021-04-13 00:35:31.040 2022-05-26 15:21:42.867
16 1932862 2272118 2021-04-13 01:56:50.363 2022-05-26 14:06:48.953
   1062054 2070104 2021 04 22 12:05:21 557 2022 05 20 11:22:27 572
```



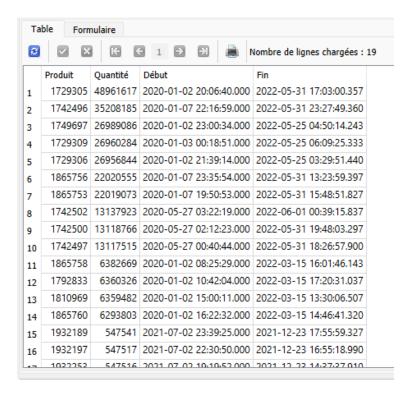
QUESTION 2

Nous avons besoin de connaître la quantité totale produite par variété pour la ligne 2 par ordre décroissant.

REQUETE(S) SQL



RESULTAT(S)

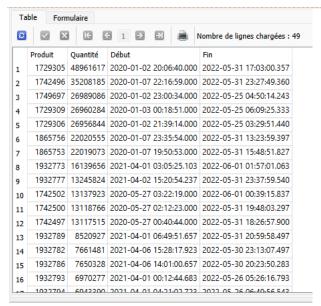


QUESTION 3

Nous avons besoin de connaître la quantité totale produite par variété et par ligne sur les deux lignes, par ordre décroissant.



```
Requête Historique
1 SELECT Product_ID AS Produit,
         SUM(NB_Produits) AS Quantité,
         MIN(DATE_DEBUT) AS Début,
         MAX(DATE HEURE) AS Fin
  FROM L1_Production
6 GROUP BY Product_ID
  UNION ALL
10 SELECT Product_ID AS Produit,
         SUM(NB_Produits) AS Quantité,
12
         MIN(DATE_DEBUT) AS Début,
13
         MAX(DATE_HEURE) AS Fin
15 GROUP BY Product_ID
17 ORDER BY Quantité DESC;
```



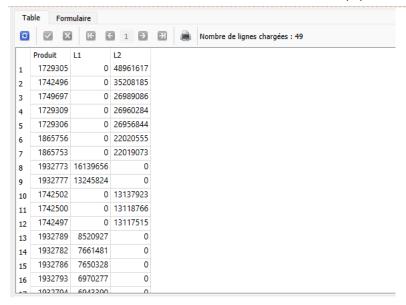
QUESTION 4

Pour chaque variété, nous avons besoin de connaître la quantité produite sur L1 et sur L2 (les lignes L1 et L2 deviennent des colonnes).



```
Requête Historique

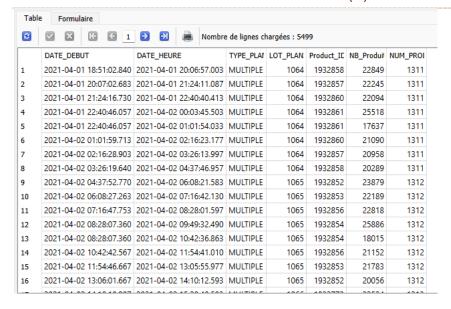
1 SELECT
2 Product_IO AS Produit,
3 SUM(CASE WHEN Ligne = 'L1' THEN Quantité_Totale ELSE 0 END) AS L1,
4 SUM(CASE WHEN Ligne = 'L2' THEN Quantité_Totale ELSE 0 END) AS L2
5 FROM (
6 SELECT
7 Product_ID,
8 SUM(NB_Produits) AS Quantité_Totale,
9 'L1' AS Ligne
10 FROM L1_Production
11 GROUP BY Product_ID
12 UNION ALL
14
15 SELECT
16 Product_ID,
17 SUM(NB_Produits) AS Quantité_Totale,
18 'L2' AS Ligne
19 FROM L2_Production
20 GROUP BY Product_ID
21 ) AS Combined
22 GROUP BY Product_ID
23 ORDER BY (L1 + L2) DESC;
```



QUESTION 5

Créez une requête montrant la table de production L1 avec le num_prod associée pour chaque batch.





Pour les questions suivantes, vous pouvez vous concentrer uniquement sur une ligne, L1 ou L2 au choix.

QUESTION 6

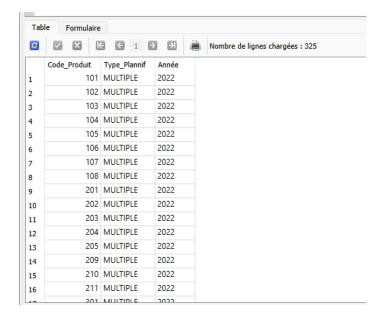
Lorsque la planification est faite par les opérateurs, les données vont apparaître sous forme « MULTIPLE ». Cependant, en cas de problème sur la ligne, il se peut que les opérateurs doivent replanifier une ou plusieurs masses du coffret. Ces lignes apparaîtront en « UNIQUE ». Le numéro de production lui restera inchangé. Lister les num_prod concernant des coffrets multiples.

REQUETE(S) SQL

```
Requête Historique

1 WITH planification AS (
2 SELECT
3 IDENT_LOT,
4 NUM_PROD,
5 FORMAT,
6 MIN(DATE_PLANIF) AS DATE_PLANIF
7 FROM L1_Planif
8 GROUP BY IDENT_LOT, NUM_PROD, FORMAT
9 )
10
11 SELECT DISTINCT
12 p.NUM_PROD AS Code_Produit,
13 pr.YYPE_PLANIF AS Type_Plannif,
14 strftime('%Y', pr.DATE_DEBUT) AS Année
15 FROM L1_Production pr
16 JOIN planification p ON pr.LOT_PLANIF = p.IDENT_LOT
17 WHERE pr.TYPE_PLANIF = 'MULTIPLE'
18
19 ORDER BY Code Produit ASC, Type_Plannif, Année;
```





QUESTION 7

Listez les num_prod concernant des coffrets multiples ayant au moins 1 batch marqué comme « UNIQUE ».

REQUETE(S) SQL

```
Requête Historique

FORMAT,

MIN(DATE_PLANIF) AS DATE_PLANIF

FROM L1_Planif

Requeste Format,

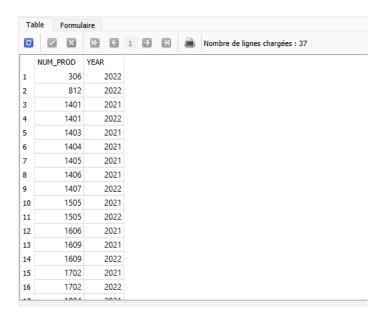
MIN(DATE_PLANIF) AS DATE_PLANIF

FROM L1_Planif

Requeste Format,

In the format for the format for the format for the format format for the format format for the format format format for the format format format for the format format
```





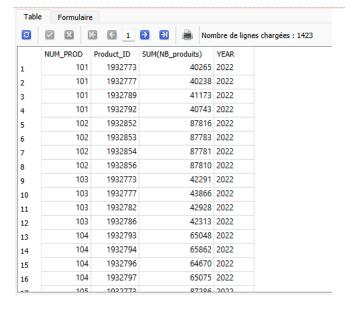


QUESTION 8

Listez la quantité produite par numéro de production et par produit.

REQUETE(S) SQL

RESULTAT(S)



QUESTION 9

Pour chaque numéro de production : listez la date de fin du numéro de production, le numéro de production, le nombre de références produits, la quantité totale produite, ainsi que le déséquilibre total.



```
Requête Historique

1 WITH planification AS (
2 SELECT
3 IDENT_LOT,
4 NUM_PROD,
5 FORWAT,
6 MIN(DATE_PLANIF) AS DATE_PLANIF
7 FROM L1_PLANIF) AS DATE_PLANIF
8 GROUP BY DIENT_LOT, NUM_PROD, FORMAT
9),
10 production AS (
1 SELECT
12 NUM_PROD,
13 ProductIO,
14 SUM(NB_Produits) AS Qté_produite,
15 MAX(DATE_HERR) AS date_fin,
16 strftime('XY', DATE_HEURE) AS YEAR
17 FROM L1_Production
18 LEFT JOIN planification ON IDENT_LOT - LOT_PLANIF
19 WHEE NUM_PROD IS NOT NULL
28 GROUP BY NUM_PROD, Product_ID, strftime('XY', DATE_HEURE)
21 )
21 SELECT
23 MAX(date_fin) AS date_fin,
24 NUM_PROD,
25 COUNT(*) AS nb_product,
26 SUM(Qté_produite) AS Qté_Produite,
27 SUM(Qté_produite) AS Qté_Produite,
28 FROM_production COUNT(*) * MIN(Qté_produite) AS Déséquilibre
29 GROUP BY NUM_PROD, YEAR;
```

Tab	Table Formulaire					
8	▼ X K € 1	> ->	Nombre de lignes chargées : 450			
	date_fin	NUM_PROD	nb_product	Qté_Produite	Déséquilibre	
1	2022-01-04 01:36:40.427	101	4	162419	1467	
2	2022-01-04 22:34:22.560	102	4	351190	66	
3	2022-01-05 08:36:45.417	103	4	171398	2234	
4	2022-01-05 23:23:35.717	104	4	260655	1975	
5	2022-01-06 18:52:31.123	105	4	350722	1578	
6	2022-01-07 16:13:56.393	106	4	347030	1194	
7	2022-01-08 04:25:16.970	107	4	165735	6495	
8	2022-01-11 02:11:02.737	108	4	606334	82	
9	2022-01-11 13:01:23.677	201	4	164458	8630	
10	2022-01-12 03:02:41.130	202	4	260606	1062	
11	2022-01-12 21:26:35.997	203	4	249963	2275	
12	2022-01-13 02:39:36.427	204	4	85649	877	
13	2022-01-13 12:34:47.357	205	4	173509	289	
14	2022-01-13 18:28:44.670	206	1	90034	0	
15	2022-01-14 01:55:10.087	207	1	72930	0	
16	2022-01-14 09:03:30.860	208	1	131599	0	
	2022 01 14 20-27-50 700	200		17/610	101/	

QUESTION 10

Il serait intéressant d'avoir des informations sur les produits utilisés par numéro de production.

À partir de la requête précédente, ajoutez une colonne nommée COFFRET_PROD contenant la liste des Product_ID pour chaque num_prod.



```
Roudaut_Quitte ✓ 🕨 🔠 🕇 👼 🌃 🔒 🗸 🗁 🔑 🗀 📄
  Requête Historique
  1 WITH planification AS (
            SELECT
IDENT_LOT,
                  NUM_PROD,
FORMAT,
MIN(DATE_PLANIF) AS DATE_PLANIF
            FROM L1_Planif
            GROUP BY IDENT_LOT, NUM_PROD, FORMAT
     production AS (
10
11
12
13
14
15
16
17
18
19
20
21
22
23
            SELECT
                  NUM_PROD,
Product_ID,
           SUM(NB_Produits) AS Qté_produite,
MAX(DATE_HEURE) AS date_fin,
strftime('%'', DATE_HEURE) AS YEAR
FROM L1_Production
            LEFT JOIN planification ON IDENT_LOT = LOT_PLANIF
WHERE NUM_PROD IS NOT NULL
            GROUP BY NUM_PROD, Product_ID, strftime('%Y', DATE_HEURE)
     SELECT
            MAX(date_fin) AS date_fin,
    MAX(date_fin) AS date_fin,

NUM_PROD,

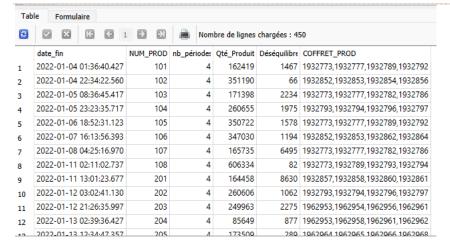
COUNT(*) AS nb_périodes_production,

SUM(Qté_produite) AS Qté_produite,

SUM(Qté_produite) - COUNT(*) * MIN(Qté_produite) AS Déséquilibre,

GROUP_CONCAT(DISTINCT Product_ID) AS COFFRET_PROD

FROM production
24
25
     GROUP BY NUM_PROD, YEAR
    GROUP BY NUM_PROD; ILDN:
HAVING
DATE_FIN IS NOT NULL
AND COFFRET_PROD IS NOT NULL
AND NUM_PROD IS NOT NULL;
```



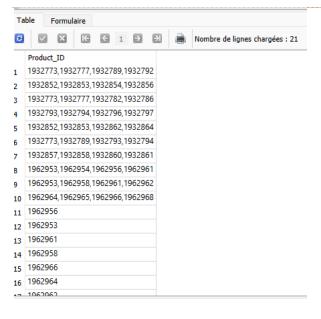
QUESTION 11

Pour avoir une idée de la variété de coffrets existants, exécutez une requête qui extrait la liste de COFFRET_PROD sans doublon.



```
Requête Historique

1 WITH planification AS (
2 SELECT IDENT_LOT, NUM_PROD, FORMAT, MIN(DATE_PLANIF)
3 FROM L1_Planif
4 GROUP BY IDENT_LOT, NUM_PROD, FORMAT
5 ),
6 production AS (
7 SELECT Num_prod, product_ID, strftime('%Y', DATE_HEURE) AS YEAR
8 FROM L1_Production
9 LEFT JOIN planification ON ident_lot = lot_planif
10 WHERE num_prod IS NOT NULL
11 GROUP BY Num_prod, Product_ID, strftime('%Y', DATE_HEURE)
12 ),
13 Product_coffret AS (
14 SELECT GROUP_CONCAT(Product_ID) AS Product_ID
15 FROM production
16 GROUP BY num_prod, YEAR
17 )
18 SELECT DISTINCT(Product_ID) AS Product_ID
19 FROM Product_coffret
20
```



QUESTION 12

D'après cette liste, vous observez qu'un coffret est composé de 1, 2 ou 4 variétés. En réalité, ce sera toujours 1 ou 4. Reprenez la dernière requête afin qu'elle reflète la réalité.



```
Requête Historique

| MITH planification AS (
| SELECT IDENT_LOT, NUM_PROD, FORMAT, MIN(DATE_PLANIF) |
| GROUP BY IDENT_LOT, NUM_PROD, FORMAT |
| GROUP BY IDENT_LOT, NUM_PROD, FORMAT |
| GROUP BY IDENT_LOT, NUM_PROD, FORMAT |
| SPOUNT |
| SPOUNT |
| FROM 1.1 PRODUCTION |
| LEFT JOIN planification ON ident_lot = lot_planif |
| WHERE num_prod IS NOT NULL |
| GROUP BY Num_prod, Product_ID, strftime('%Y', DATE_HEURE) |
| PROMUCT_CONFIRM |
| SELECT GROUP_CONCAT(Product_ID) AS Product_ID, count(*) AS nb_périodes_production |
| FROM production |
| GROUP BY num_prod, YEAR |
| PROMUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| FROM PRODUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| FROM PRODUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| FROM PRODUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| FROM PRODUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| FROM PRODUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| FROM PRODUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| FROM PRODUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| FROM PRODUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| FROM PRODUCT_CONFIRM |
| SELECT DISTINCT(Product_ID) AS Product_ID |
| SELECT DISTINCT(Product_ID) AS
```

