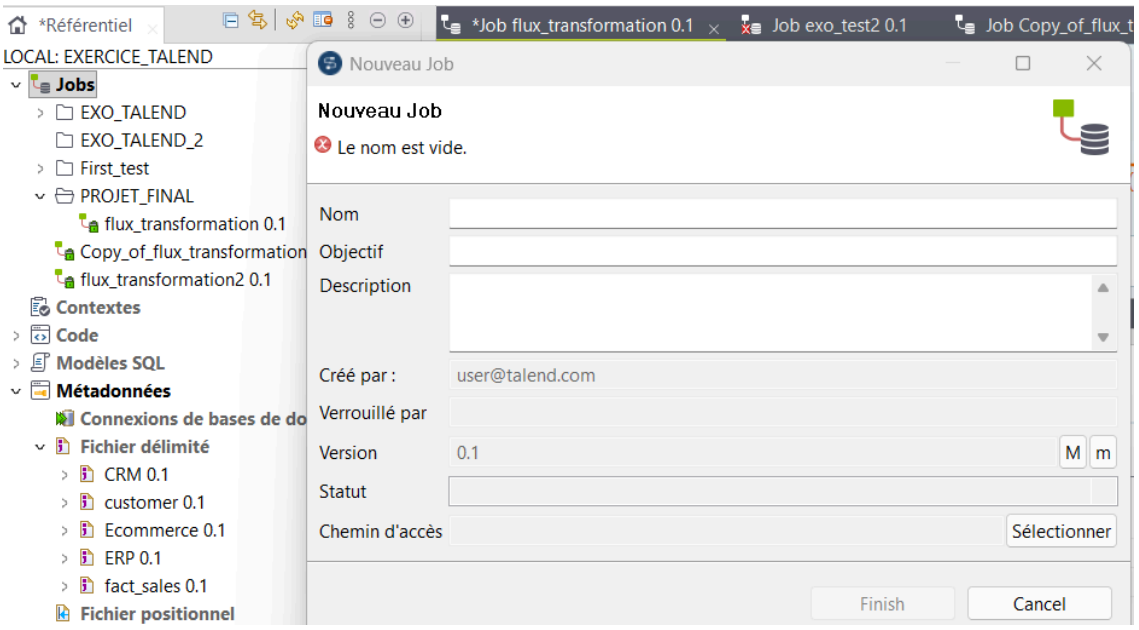
Documentation projet TALEND

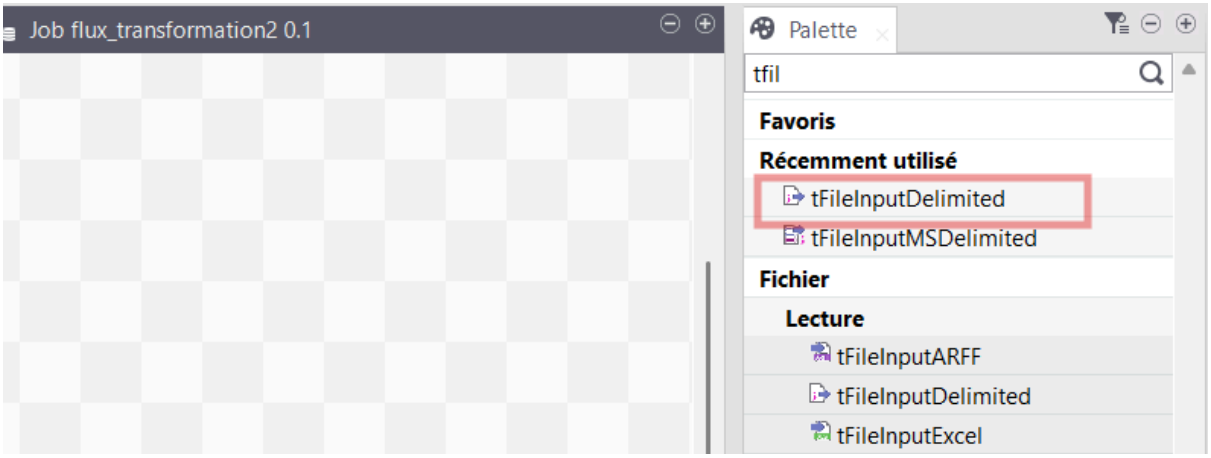
1. Import de nos différentes sources sur TALEND

Charger les fichiers de données venant de plusieurs systèmes (CRM, E-commerce, ERP) dans Talend.

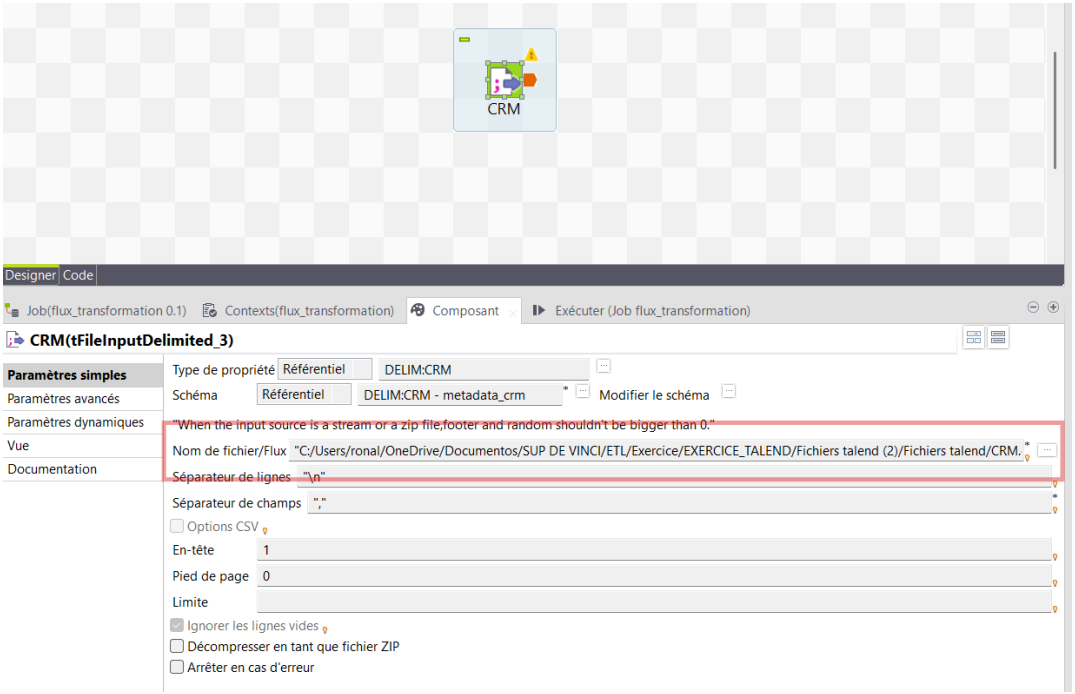
Avant d'importer les fichiers, il faut créer un environnement sur Talend en créant un job.



Après la création de l'environnement, charger les fichiers CSV en utilisant le composant **tFileInputDelimited.** Pour le faire, aller dans "Palette" à droite de et taper le nom du composant et faire glisser dans le job.

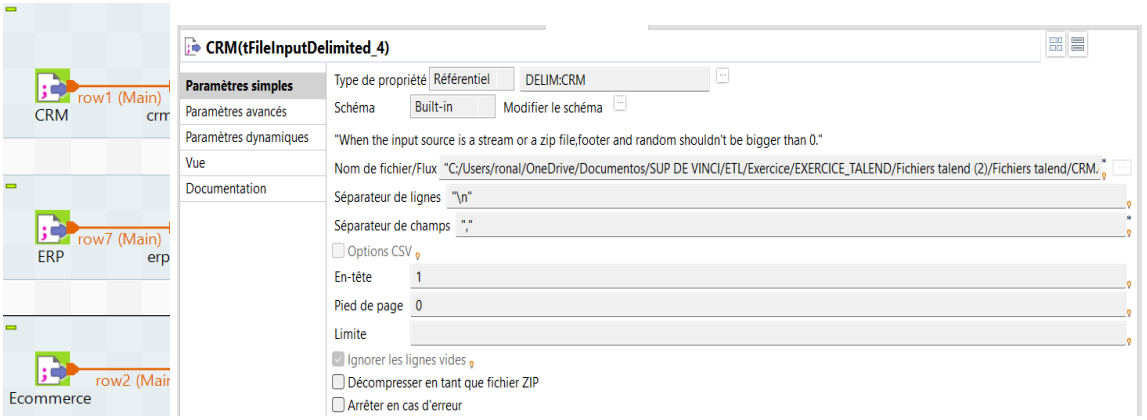


Double cliquer sur le composant, et charger le fichier comme indiqué ci-dessous.



-

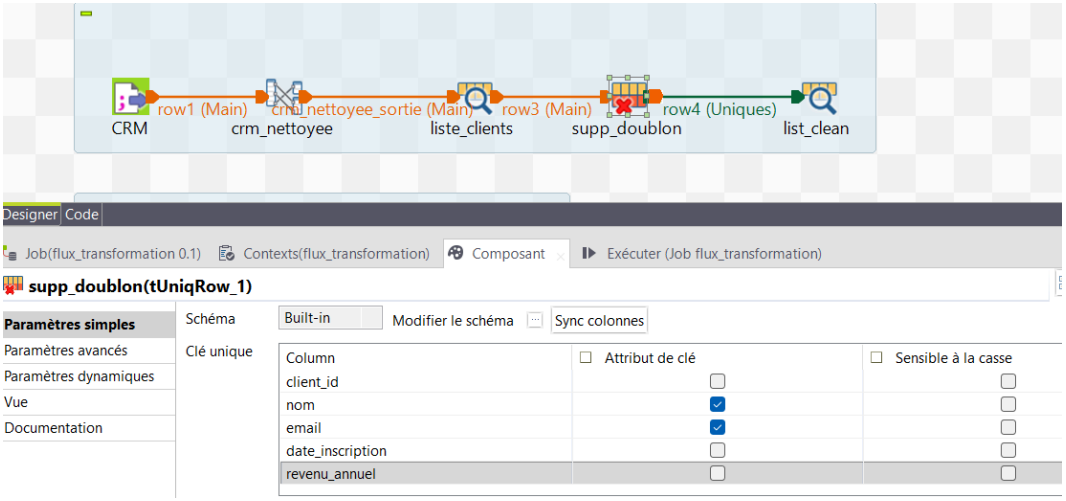
Après chargement des fichiers, il est possible de modifier le schéma en allant dans “Modifier le schéma”.



2. Gestion des doublons

Après le chargement des fichiers CSV dans le job Talend, un profilage de données est effectué pour nous permettre de savoir si les fichiers contiennent des clients ou des transactions en double, basés sur des critères comme l’adresse mail, le nom afin de ne garder qu’un seul enregistrement par client.

Après profilage, nous avons observé que le fichier venant de la source CRM contient des doublons. Et pour les supprimer, nous avons utilisé le composant **tUniqRow** en se basant sur le nom de client et l’adresse mail.



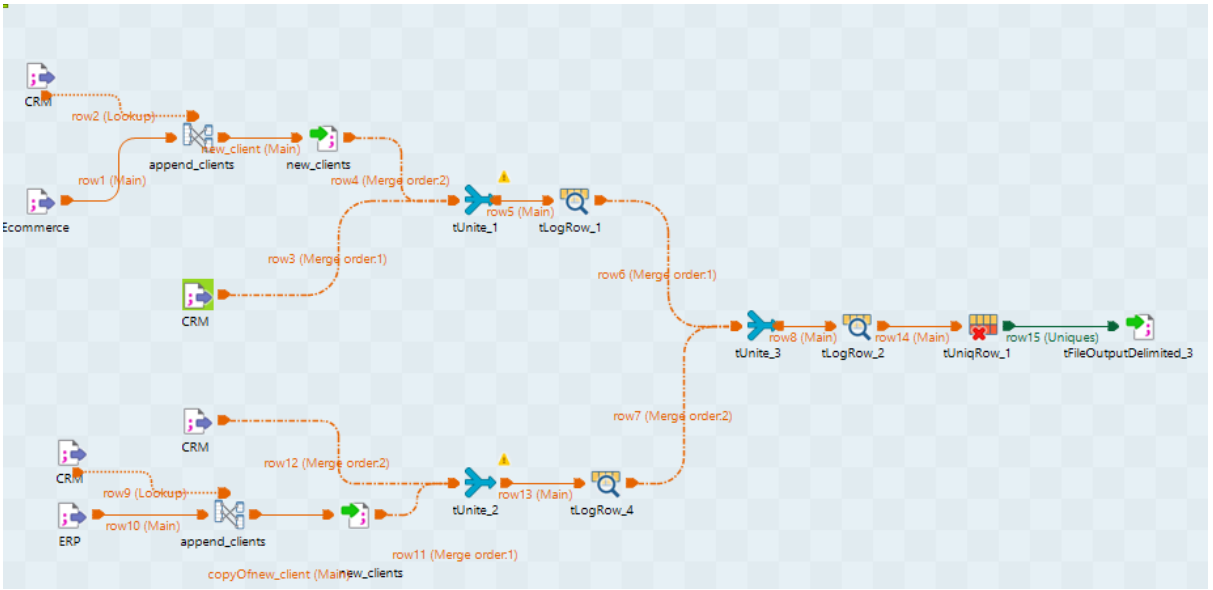
3. Jointures

Concernant les jointures des données clients, nous avons fusionné les fichiers issus de différentes sources, en gérant les cas suivants :

- Les colonnes différentes entre les fichiers,
- Les identifiants clients (client_id) distincts pour chaque système.

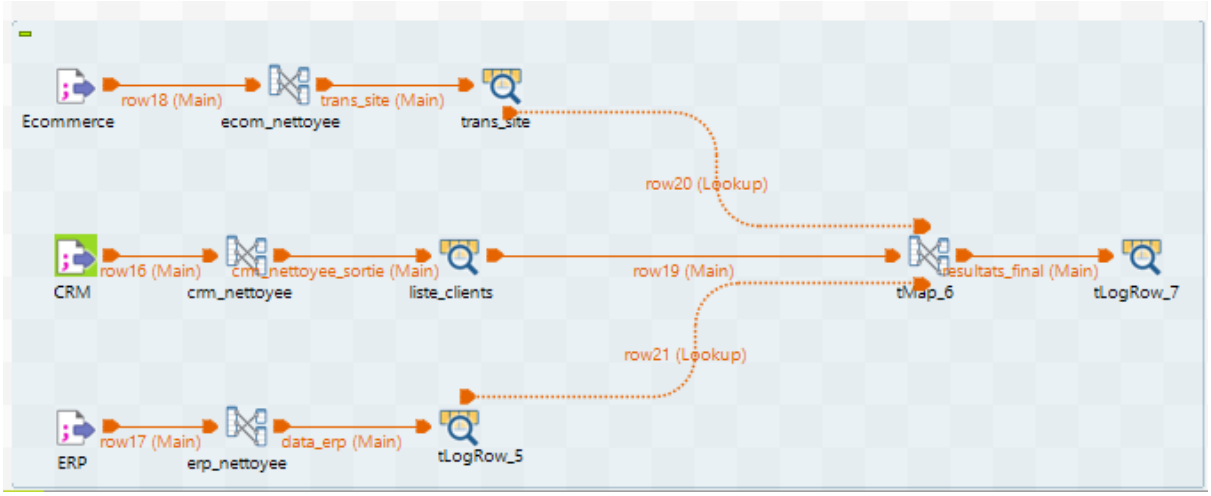
Pour effectuer les jointures, nous avons utilisé les composants **tMap** et **tUnite**.

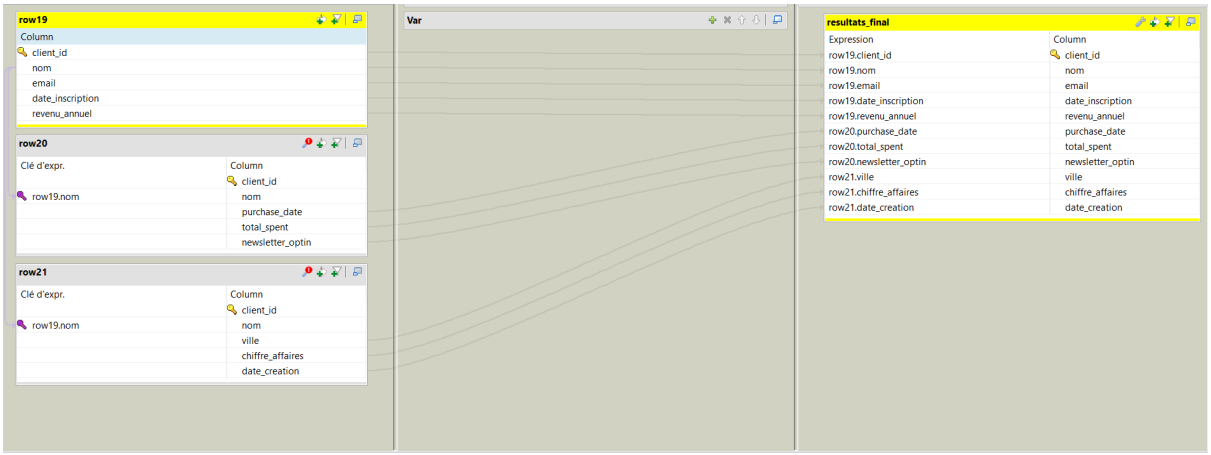
Lors du profilage des données, nous avons également observé que certains clients du fichier E-Commerce et du fichier ERP n'existaient pas dans le CRM.
Nous avons donc utilisé **tMap** pour identifier et filtrer les nouveaux clients, puis un composant **tUnite** afin de les intégrer dans le CRM.



Nous avons ensuite réalisé une jointure, en utilisant à nouveau le composant **tMap**.

Le composant **tMap** nous a permis de fusionner les différentes sources de données et de standardiser les colonnes avant la génération du jeu de données final. Chaque flux d'entrée (CRM, ERP et E-commerce) a été relié à l'aide de clés communes - telles que le nom du client.





4. Normalisation (gestion des valeurs nulles et formatage)

Dans notre composant tMap il nous est aussi possible de normaliser et standardiser à l'aide d'expressions nos données, ces expressions sont faites à partir de fonctions en Java, car c'est aussi le moteur de TALEND, donc il interprète ce langage.
Par exemple en entrée nous avons des données de type String mais nous souhaitons les faire sortir au format Date, pour que nos données soient bien interprétables par la suite depuis notre base de données, voici un exemple:

```
TalendDate.parseDate("dd/MM/yyyy", row5.date_inscription)
```

Cela prend une date au format String et la retourne au format Date.

Il nous est aussi possible de gérer les espaces en trop, ou encore les valeurs nulles, voici un autre exemple d'expressions permettant de le faire:

```
row6.ville != null && !row6.ville.trim().equals("")
? row6.ville
: "Inconnue"
```

Si nous avons une valeur nulle ou vide comme "" nous affichons "Inconnue" sinon nous affichons le nom de la ville.
Cela permet de ne pas se retrouver avec des valeurs non voulues dans notre base de données.
Pour finir voici un dernier exemple montrant la conversion de nos données du type String vers le type Float:

```
row5.revenu_annuel != null && !
row5.revenu_annuel.trim().equals("")
? Float.valueOf(row5.revenu_annuel.replaceAll("[^0-9]",
""))
: 0.0f
```

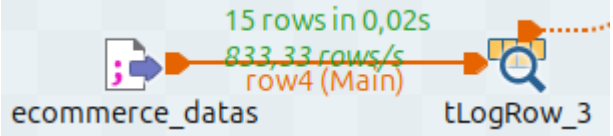
Si le revenu annuel est nul ou vide, on affiche 0.0 au format float, et en utilisant une regex (expression régulière) permettant de ne garder que les chiffres d'une chaîne de caractères et de remplacer tous les autres caractères par du vide "", alors nous obtiendrons la possibilité de ne garder que le chiffre de ce genre de données au format non normé :

```
?,
?,35000
3,35000
?,28000€
?,42 000 €
3,52000
?,52000
?,52000
?,28000€
?023,28000€
?023,42 000 €
?022,35000
?023,
?023,35000
?023,52000
```

5. Affichage

Pour afficher nos données nous utilisons d'abord le composant tLogRow qui permet d'afficher l'état de notre flux de données à des moments précis, on peut donc voir le changement de nos données pendant l'étape de transformation.

Voici à quoi ressemble ce composant et comment il est utilisé simplement dans TALEND:

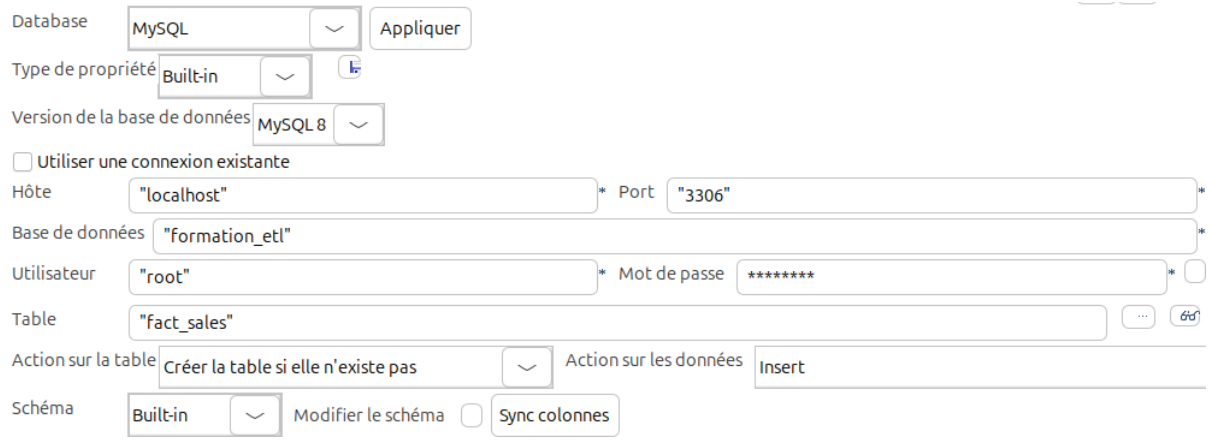


Et voici ce qu'il affiche par exemple, avant la transformation:

```
.---------+---------+-----------------+---------------+-------------+------
|                                        tLogRow_3                            |
|=--------+---------+-----------------+---------------+-------------+------
|client_id|full_name|email            |date_inscription|revenu_annuel|ville
|=--------+---------+-----------------+---------------+-------------+------
|13       |Client_11|null             |null           |0.0          |Paris
|10       |Client_8 |null             |null           |0.0          |Lyon
|null     |Client_18|null             |null           |0.0          |Paris
|null     |Client_9 |null             |null           |0.0          |Toulou
|9        |Client_17|null             |null           |0.0          |Toulou
|null     |Client_12|null             |null           |0.0          |
|12       |Client_16|null             |null           |0.0          |null
|null     |Client_14|null             |null           |0.0          |Marsei
|null     |Client_6 |null             |null           |0.0          |Paris
|14       |Client_24|null             |null           |0.0          |null
|6        |Client_21|null             |null           |0.0          |null
|null     |Client_23|null             |null           |0.0          |null
|5        |Client_19|null             |null           |0.0          |null
|7        |Client_13|null             |null           |0.0          |null
|8        |Client_22|null             |null           |0.0          |null
|1        |Client_1 |client1@gmail.com|06/03/2022     |0.0          |null
|2        |Client_2 |client2@gmail.com|15/03/2022     |35000.0      |null
|3        |Client_3 |client3@gmail.com|08/01/2023     |35000.0      |Marsei
|4        |Client_4 |client4@gmail.com|28/04/2022     |28000.0      |Paris
|5        |Client_5 |client5@gmail.com|25/09/2022     |42000.0      |Toulou
|7        |Client_7 |client7@gmail.com|29/10/2022     |52000.0      |Toulou
|10       |Client_10|client10@gmail.com|21/04/2023    |28000.0      |Toulou
|15       |Client_15|client15@gmail.com|08/11/2023    |52000.0      |Marsei
'---------+---------+-----------------+---------------+-------------+------
```

6. Mise en base de données

Pour mettre nos données transformées dans une base de données, il existe plusieurs composants qui permettent de le faire dans TALEND.
Notre base de données de destination est une base mySQL que nous visionnons sur PhpMyAdmin en local.
Donc dans notre cas utiliser le composant tMySQLOutput est le plus pertinent.
Il nous faut donc paramétrer la connexion vers la base de données pour que TALEND y ait accès et puisse insérer les données dans une table de la base.



Une fois ces informations renseignées, on peut lancer l'exécution du flux et observer (si toutes nos informations sont correctes) que la table a été créée et est peuplée par nos données.

7. Rendu dans la base de données

Pour finir voici le rendu en base de données, une fois que les données ont bien été transférées dans notre base de données :

	client_id	nom ▲ 1	email	date_inscription	revenu_annuel	ville	chiffre_affaires	date_creation	purchase_date	total_spent	newsletter_optin
Éditer Copier Supprimer	1	Client_1	client1@gmail.com	2022-03-06 00:00:00	0	Inconnue	0	NULL	NULL	NULL	0
Éditer Copier Supprimer	10	Client_10	client10@gmail.com	2023-04-21 00:00:00	28000	Toulouse	824209	2022-02-22 00:00:00	NULL	NULL	0
Éditer Copier Supprimer	11	Client_11	client11@gmail.com	2023-08-17 00:00:00	42000	Paris	24841	2022-03-20 00:00:00	2023-10-08 00:00:00	436.86	1
Éditer Copier Supprimer	12	Client_12	client12@gmail.com	2022-05-20 00:00:00	35000	Inconnue	950923	2022-11-30 00:00:00	2023-06-05 00:00:00	364.92	0
Éditer Copier Supprimer	13	Client_13	client13@gmail.com	2023-03-16 00:00:00	0	Inconnue	0	NULL	2023-07-16 00:00:00	265.94	0
Éditer Copier Supprimer	14	Client_14	client14@gmail.com	2023-02-02 00:00:00	35000	Marseille	606234	2023-09-22 00:00:00	2023-08-02 00:00:00	329.87	0
Éditer Copier Supprimer	15	Client_15	client15@gmail.com	2023-11-08 00:00:00	52000	Marseille	256862	2023-08-05 00:00:00	NULL	NULL	0
Éditer Copier Supprimer	2	Client_2	client2@gmail.com	2022-03-15 00:00:00	35000	Inconnue	0	NULL	NULL	NULL	0
Éditer Copier Supprimer	3	Client_3	client3@gmail.com	2023-01-08 00:00:00	35000	Marseille	30228	2021-10-24 00:00:00	NULL	NULL	0
Éditer Copier Supprimer	4	Client_4	client4@gmail.com	2022-04-28 00:00:00	28000	Paris	954721	2023-10-13 00:00:00	NULL	NULL	0
Éditer Copier Supprimer	5	Client_5	client5@gmail.com	2022-09-25 00:00:00	42000	Toulouse	371465	2022-12-27 00:00:00	NULL	NULL	0
Éditer Copier Supprimer	6	Client_6	client6@gmail.com	2023-03-26 00:00:00	52000	Paris	216676	2021-07-07 00:00:00	2023-07-05 00:00:00	230.15	0
Éditer Copier Supprimer	7	Client_7	client7@gmail.com	2022-10-29 00:00:00	52000	Toulouse	59260	2022-04-07 00:00:00	NULL	NULL	0
Éditer Copier Supprimer	8	Client_8	client8@gmail.com	2022-08-22 00:00:00	52000	Lyon	93821	2024-01-08 00:00:00	2023-02-24 00:00:00	238.32	0
Éditer Copier Supprimer	9	Client_9	client9@gmail.com	2022-08-30 00:00:00	28000	Toulouse	162085	2021-10-26 00:00:00	2023-01-07 00:00:00	394.42	0