

ANALYSE DE TWEETS LIÉS AUX APP COVID EN UTILISANT TALEND

Afin de pouvoir exécuter le projet "**ANALYSE DE TWEETS LIÉS AUX APP COVID**", il faut suivre les instructions suivantes, si vous rencontrez quelques difficultés, vous pourriez écrire à savacano_28@live.com.

Récupération des sources

1. Pour récupérer les sources, il faut aller sur : <https://github.com/savacano28/analysis-twitter>, et télécharger en .zip le projet.
Dans ce repository vous trouverez :
 - + code source du projet talend,
 - + les .jar du projet pour l'exécuter sans avoir besoin de talend,
 - + `dump_12-12-2020.dmp` pour mettre en place la base de données postgres,
 - + des historiques des données .json déjà traités depuis du 15 novembre jusqu'au 09 décembre 2020,
 - + fichier de description du projet,
 - + ce fichier lisez-moi,
 - + dashboard généré en tableau avec les données traitées sur talend

Mis en place de la base de données sur postgres

1. Nous avons travaillé avec postgres pour sauvegarder les données et les résultats des analyses, c'est pour cela qu'il est nécessaire d'installer un postgres version > 12.0.

Gardez les identifiants de : user, password et port.

2. Après de l'installation, il faut télécharger et restaurer la base de données "talend" du projet dans votre postgres. Le fichier à télécharger se trouve dans le dossier du projet que vous venez de télécharger depuis github : .../[documentations/ressources/](#), le fichier à utiliser est : **"dump_12-12-20.dmp"**. Pour ce faire on peut utiliser la commande suivante : **pg_restore talend < dump_12-12-2020.dmp** et si vous avez une interface graphique, il faudrait juste créer la bd talend, et chercher l'option : restore en utilisant le fichier dump_12-12-20.dmp.

3. Une fois le point 2 effectué vous pourrez accéder à votre bdd avec un usager (par défaut postgres) pour valider le fonctionnement. Vous pouvez alors faire une série de requêtes très simples depuis de votre ligne de commandes comme :

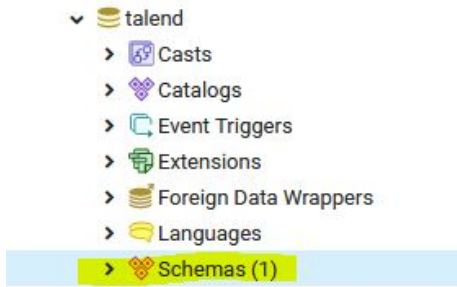
psql talend;

\d fait_tweet;

et voilà, vous trouverez une scheme comme :

```
talend=# \d fait_tweet;
              Table "public.fait_tweet"
  Column      |          Type          | Collation | Nullable | Default
-----|-----|-----|-----|-----
 hashtag      | character varying(500) |           | not null |
 country      | character varying(500) |           | not null |
 date         | date                  |           | not null |
 no_ht        | integer               |           |          |
 no_regions   | integer               |           |          |
 no_users     | integer               |           |          |
 s_positivos  | integer               |           |          |
 s_negativos  | integer               |           |          |
 t_reception  | integer               |           |          |
Indexes:
    "fait_tweet_pkey" PRIMARY KEY, btree (hashtag, country, date)
Foreign-key constraints:
    "fait_tweet_country_fkey" FOREIGN KEY (country) REFERENCES dim_country (country)
    "fait_tweet_date_fkey" FOREIGN KEY (date) REFERENCES dim_date(date_description)
    "fait_tweet_hashtag_fkey" FOREIGN KEY (hashtag) REFERENCES dim_hashtag (hashtag)
```

Par contre, si vous utilisez une interface graphique comme pgAdmin, dBeaver ou autre, il faut juste aller vers votre db talend est chercher les tableaux.



Mis en place d'un exécutable talend de projet

1. Allez dans le dossier .../documentations/executables/analysis twitter dw pj/workflow final, vous trouverez plusieurs fichiers d'exécution comme *.bat*, *.sh* et *.jar*.
2. Ouvrez le fichier *.sh* ou *.bat*, et ajoutez les paramètres suivants avec vos valeurs de connexion et de path file après le **--context=dev :**

```
--context_param
path_files_global="/votre_dossier_où_vous_avez_telechargé_le_projet_jusqua_un_dossier_avant_de_documentations/" --context_param connection_postgres_Port="votre_port" --context_param
connection_postgres_Password="votre_pwd" --context_param connection_postgres_Login="votre_login"
--context_param connection_postgres_Server="votre_server" --context_param
connection_postgres_Database="talend"
```

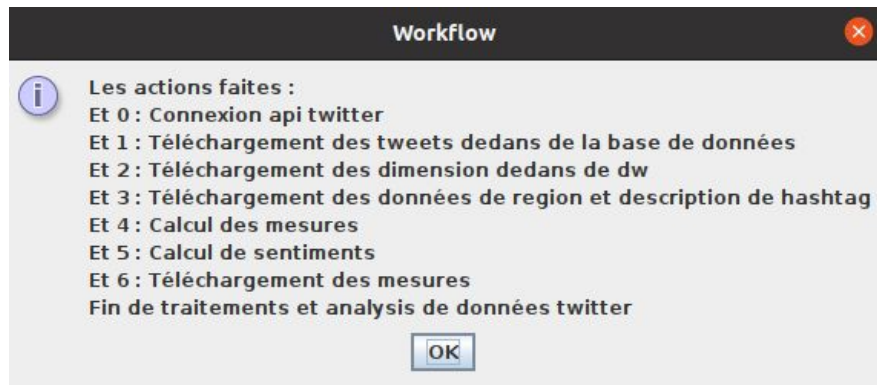
Par exemple :

```
--context_param
path_files_global="/home/scasanova/Downloads/TOS_BD-20200219_1130-V7.3.1/workspace/ANALYSIS_TWITTER/" --context_param connection_postgres_Port="5432" --context_param
connection_postgres_Password="admin" --context_param connection_postgres_Login="postgres"
--context_param connection_postgres_Server="localhost" --context_param
connection_postgres_Database="talend"
```

3. Lancer l'exécution, par exemple avec double click sur *projet.sh* ou dans une console de commandes *./projet.sh*, si vous ne réussissez pas à le lancer, regardez s'il faut le donner de droit d'accès avec **chmod +x** ou exécuter en tant qu'administrateur. Si vous réussissez à le lancer, le message suivant sera affiché :



4. Après l'exécution, et s'il n'y a pas d'erreurs, un message avec les étapes du projet sera affiché et vous pourrez vérifier aussi si dans le dossier */documentations/mesures/* et */documentations/apitwitter/* il y a des nouveaux fichiers qui correspondent à l'exécution du projet.



Edition du projet Talend

Si vous souhaitez inspecter les jobs développés du projet et ensuite le lancer sur talend, il faudra **IMPÉRATIVEMENT remplir les variables de contexte**, tels que : paramètres de connexion à la bdd et le chemin du dossier du projet (avant du dossier de documentations).

1. Après d'avoir importé le projet dans Talend, vous aurez la configuration suivante :

	Name	Type	Comment
1	connection_postgres (from repository context)		
2	connection_postgres_Server	String	
3	connection_postgres_Login	String	
4	connection_postgres_Port	String	
5	connection_postgres_Password	String	admin
6	connection_postgres_Database	String	talend
7	connection_postgres_Schema	String	

Dans le projet, vous pourrez exécuter chacun des jobs de manière individuel ou dans leur ensemble dans le job "workflow_final".