



09/11/2024

PROJET 5CLOUD

Master of engineering II

Etudiant(e)s :

Elisabeth NOKAM DASSI TAGUEMNE

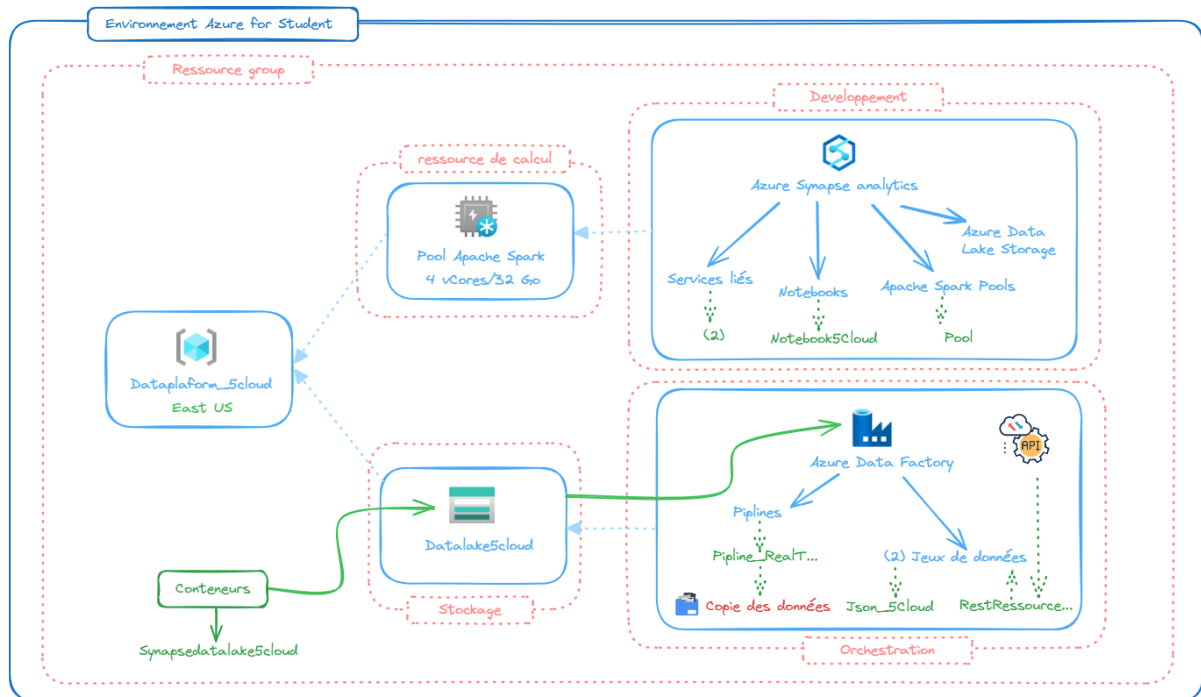
Alexandre MOINDROT

Ibrar HAMOUDA

Présentation du Projet :

Objectif : Créons une plateforme de données unifiée pour l'analyse d'événements en temps réel à l'aide de Microsoft Azure (ou d'outils locaux alternatifs).

Architecture Azure :



Accès au repo Github : <https://github.com/deszr/5CLOU.git>

Étape 1 : Choix de la source de données

Comme source de données en temps réel, nous avons opté pour **Yahoo Finance** qui est un flux boursier.

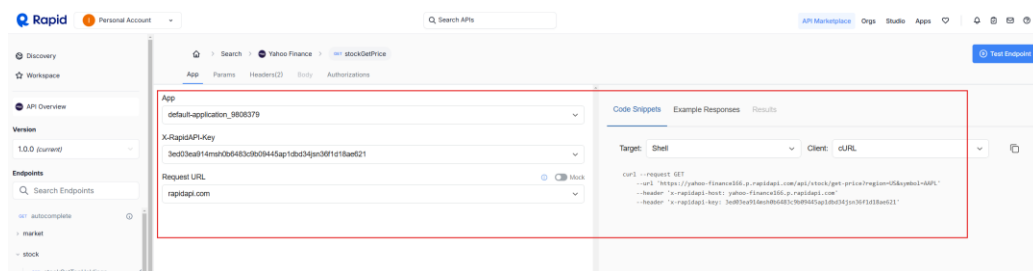
Pour accéder aux données de Yahoo Finance, nous le ferons via une API **RapidAPI** :

Étapes pour Accéder à l'API Yahoo Finance via RapidAPI

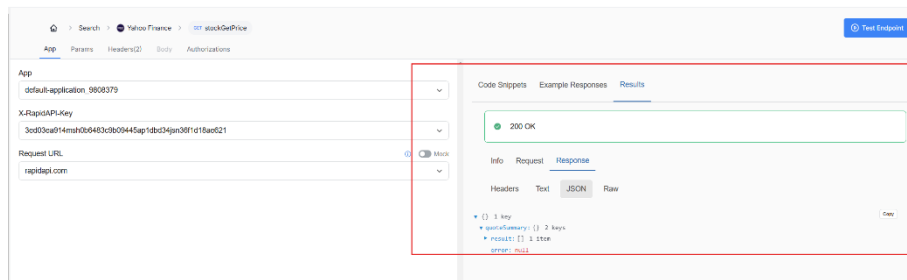
- Création d'un compte sur RapidAPI
- Sélectionner l'API Yahoo Finance sur RapidAPI
- S'abonner à un Plan de l'API

Basic	Pro	Ultra	Mega ★ Recommended
\$0.00 /mo	\$5.00 /mo	\$15.00 /mo	\$20.00 /mo
Requests 🕒 300 / Month Hard Limit	Requests 🕒 5000 / Month + \$0.0005	Requests 🕒 20 000 / Month + \$0.0004	Requests 🕒 100 000 / Month + \$0.0003
Rate Limit 1000 requests per hour	Rate Limit 5 requests per second	Rate Limit 5 requests per second	Rate Limit 15 requests per second
Start Free Plan	Choose This Plan	Choose This Plan	Choose This Plan

- Obtention de la clé donnant accès à l'API



- Testons un Endpoint de l'API Yahoo Finance avec RapidAPI

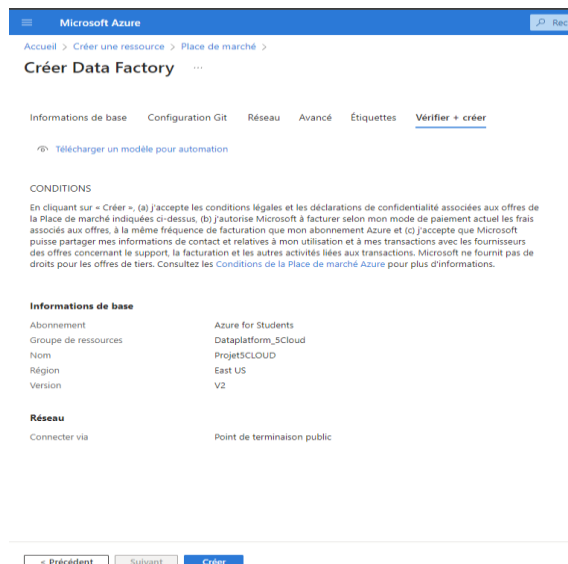


Étape 2 : Configuration du Pipeline d'Ingestion de Données

Objectif : Configurons un pipeline d'ingestion capable de récupérer les données en continu et de les stocker pour effectuer traitement.

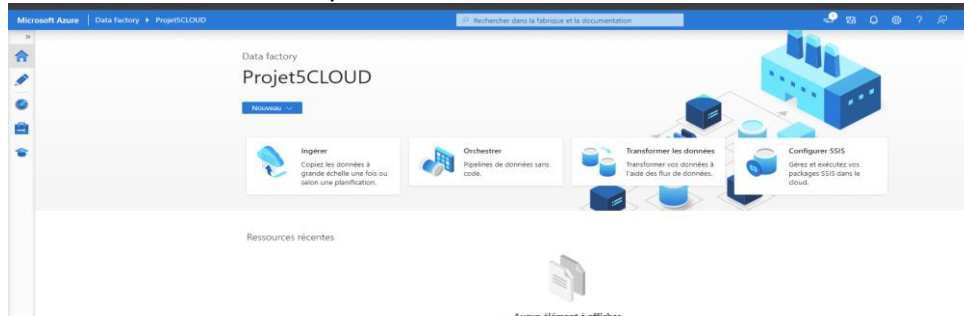
Créons un service **Data Factory** dans le portail Azure :

- Recherchons et créons une instance d'**Azure Data Factory**.
- Assignons un nom et un groupe de ressources au service ensuite cliquons sur « **Créer** »

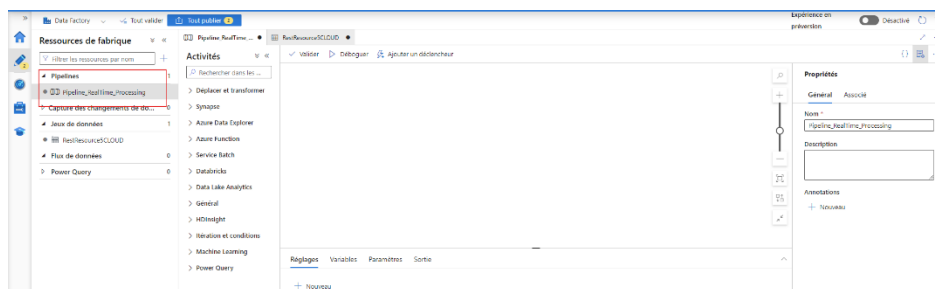


Configurons le pipeline d'ingestion

Ouvrons Azure Data Factory Studio.

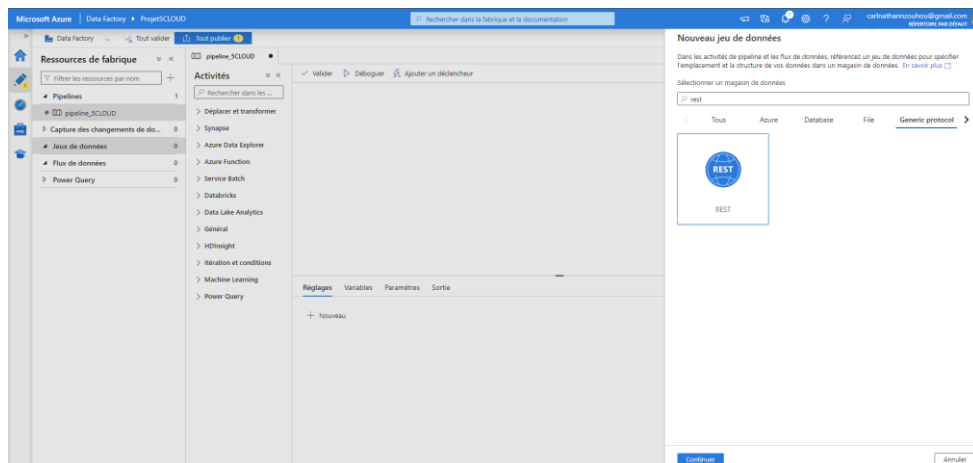


Créons un nouveau pipeline



Ajoutons notre jeu de données

Pour cela, nous avons configuré la source en sélectionnant “REST”(generic protocol) :



Ensuite nous renseignons L’URL de base (celle de l’API), ainsi que les paramètres d’authentification. Une fois cela fait, on se rassure que la connexion a bien été établie. Ainsi on peut créer notre connexion.

Nouveau service lié
 REST En savoir plus

Nom *
 Yahoo_Finance

Description

Se connecter via un runtime d'intégration * ⓘ
 AutoResolveIntegrationRuntime

URL de base *
 https://rapidapi.com
 ⚠ Les informations seront envoyées à l'URL spécifiée. Vérifiez que vous faites confiance à l'URL entrée.

Type d'authentification *
 Anonyme

Validation de certificat de serveur ⓘ
☒ Activer ☐ Désactiver

En-têtes d'authentification ⓘ
 + Nouveau | Supprimer

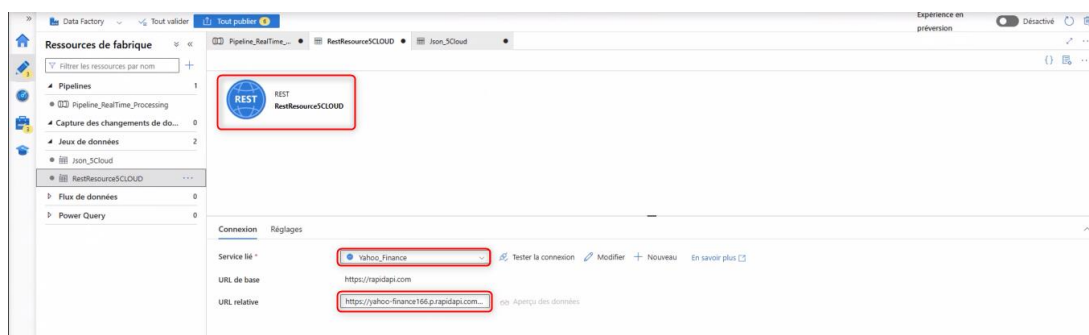
<input type="checkbox"/> Nom	Valeur
<input type="checkbox"/> X-RapidAPI-Key
<input type="checkbox"/> X-RapidAPI-Host

Annotations
 + Nouveau

▼ Réglages

Créer **Annuler**

✓ Connexion établie
 Tester la connexion

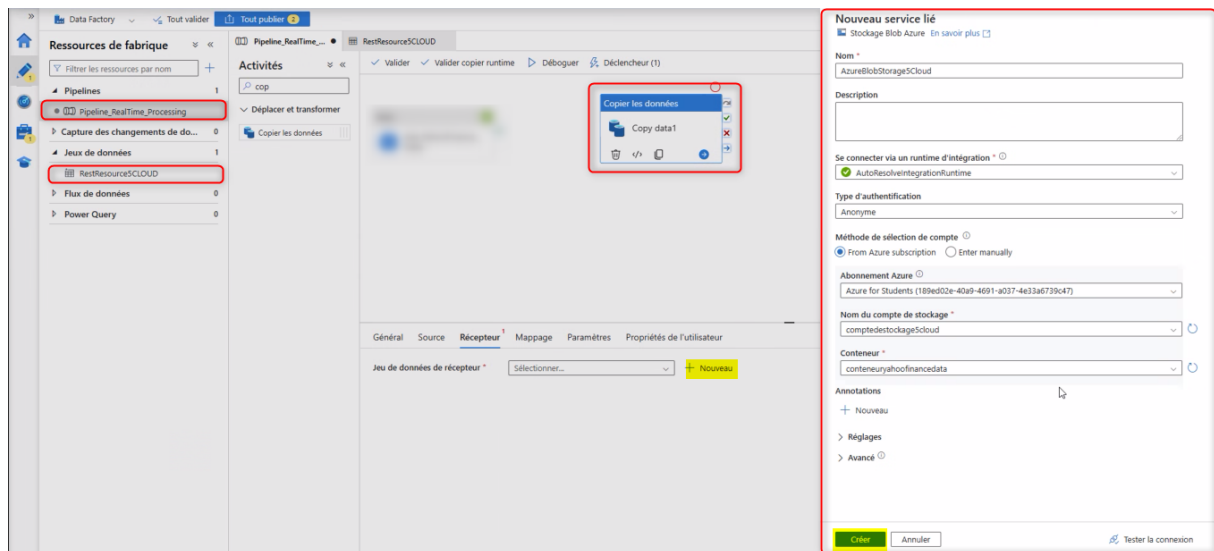


Revenons à notre Pipeline « **Pipeline_RealTime_Processing** » créé au départ :

Configurons l'Activité Copy Data :

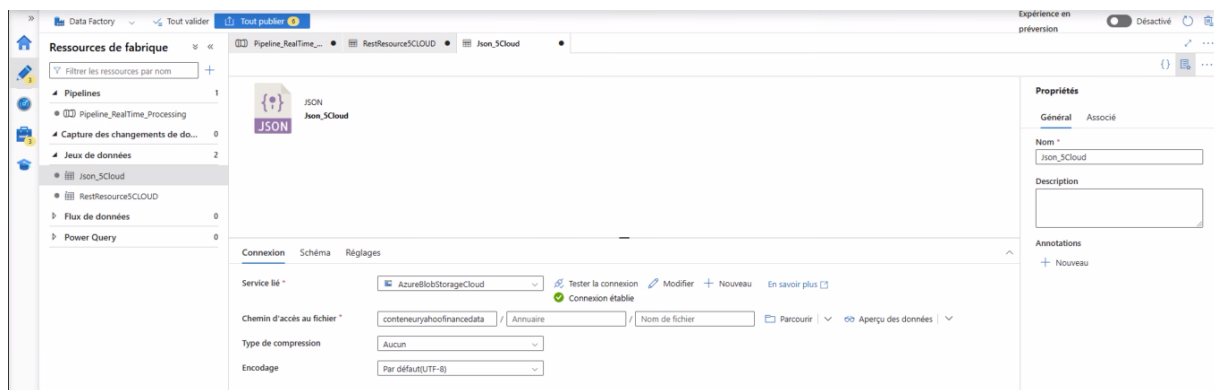
A partir du « **Pipeline_RealTime_Processing** » Dans l'onglet **Source** de l'activité "Copy Data", on configure la connexion vers la source de données depuis un Blob Storage avec du « CSV » comme format de données.

Dans les paramètres de configuration du « copy data » on configure notre jeu de données de récepteur en cliquant sur « Nouveau »

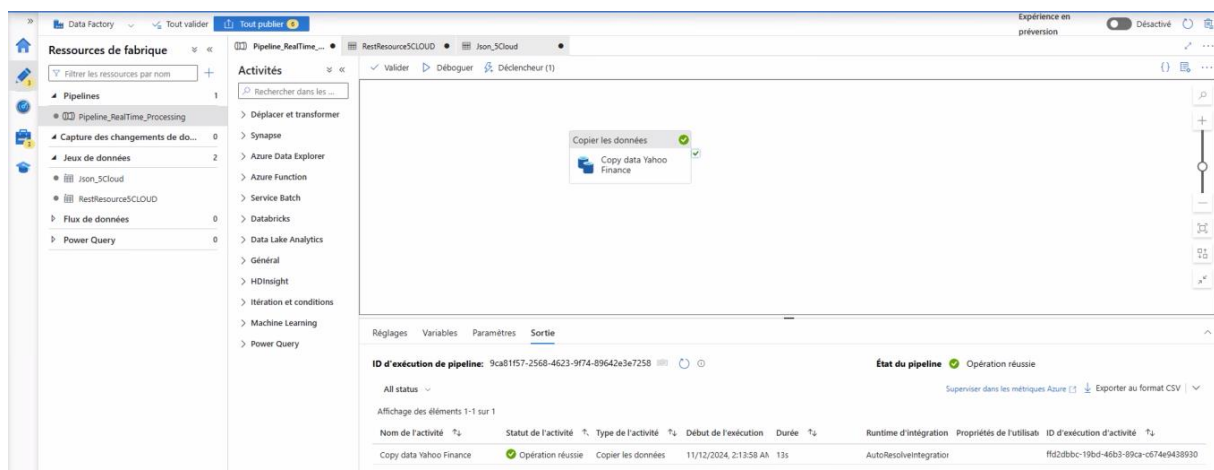


Testons l'Activité de notre jeu de données :

A partir du jeu de données « Json_5cloud » on clique sur tester la connexion :



Ensuite on effectue un test de debug :



On peut constater que l'opération a réussie.

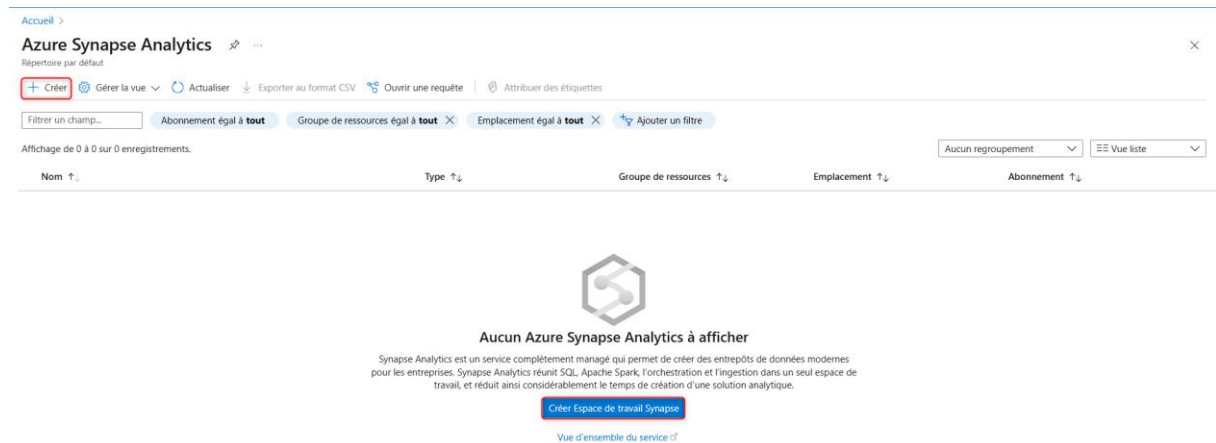
Vérification des données dans Azure Blob Storage :

Etape 3 : Traitement des données en temps réel avec Synapse Analytics

3.1 : Configuration Synapse Workspace

On crée un espace de travail dans Synapse :

Dans Azure Sélectionne Azure Synapse Analytics puis on clique sur créer :



On renseigne ensuite les informations de la synapse :

Créer un espace de travail Synapse

* Informations de base Sécurité Réseau Balises Vérifier + créer

Créez un espace de travail Synapse pour développer une solution analytique d'entreprise en quelques clics.

Détails du projet

Sélectionnez l'abonnement pour gérer les coûts et les ressources déployées. Utilisez les groupes de ressources comme les dossiers pour organiser et gérer toutes vos ressources.

Abonnement * Azure for Students

Groupe de ressources * Dataplatform_SCloud

Groupe de ressources managé * Entrez le nom du groupe de ressources managé

Détails de l'espace de travail

Nommez votre espace de travail, sélectionnez un emplacement et choisissez un système de fichiers Data Lake Storage Gen2 principal comme emplacement par défaut des journaux et de la sortie du travail.

Nom de l'espace de travail * synapseanalyticscloud

Région * East US

Sélectionner Data Lake Storage Gen2 * À partir de l'abonnement Manuellement via l'URL

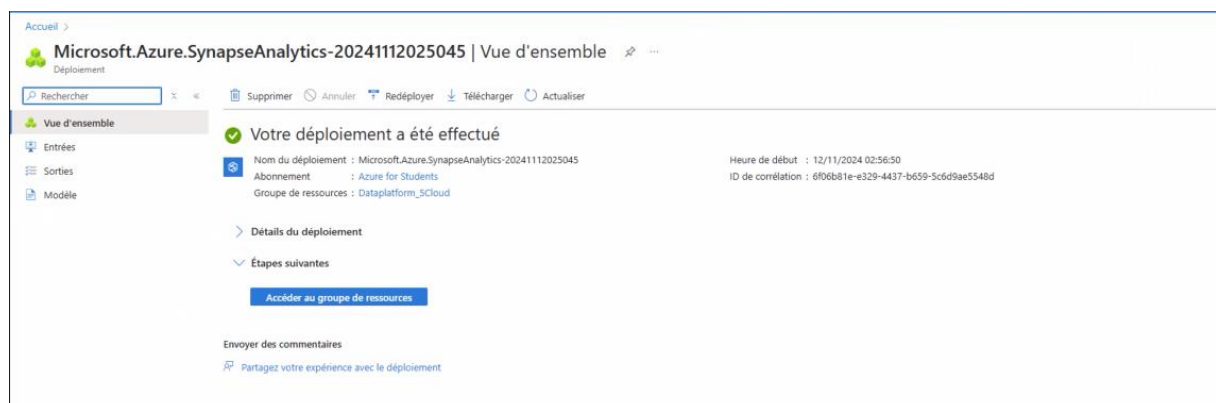
Nom du compte * (Nouveau) datalakecloud

Nom du système de fichiers * (Nouveau) synapsedatalakecloud

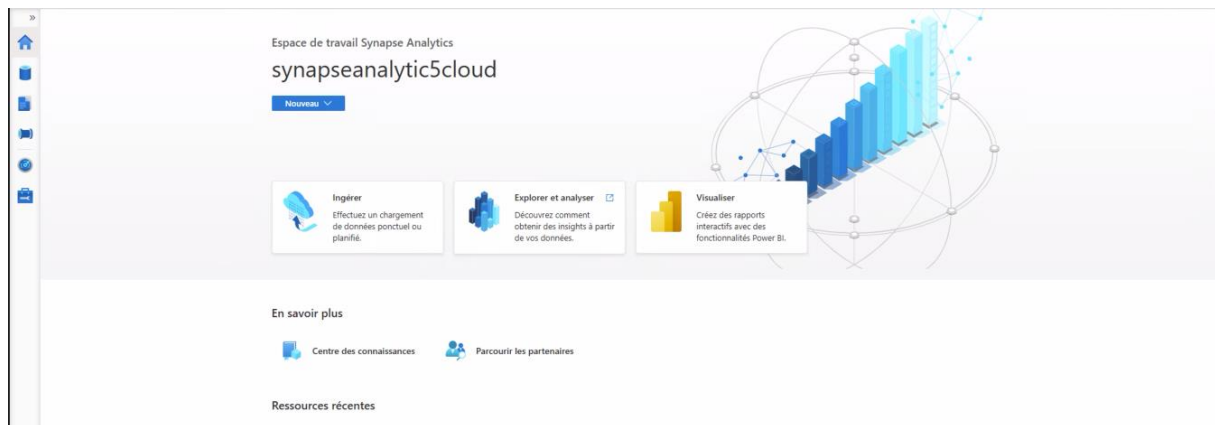
M'attribuer le rôle Contributeur de données b10h de stockage sur le compte

[Vérifier + créer](#) [< Précédent](#) [Suivant : Sécurité >](#)

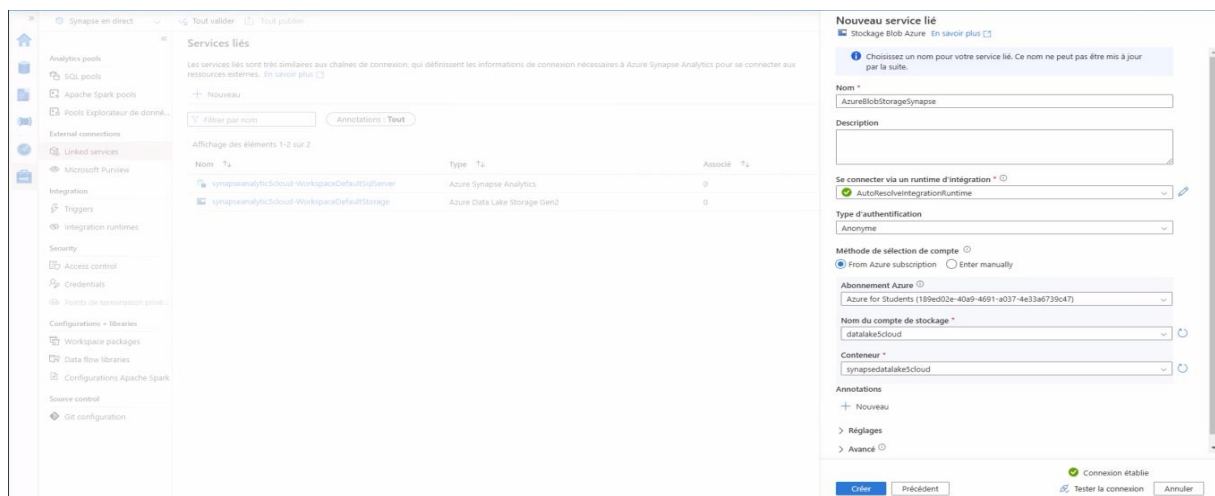
Déploiement de la ressource :



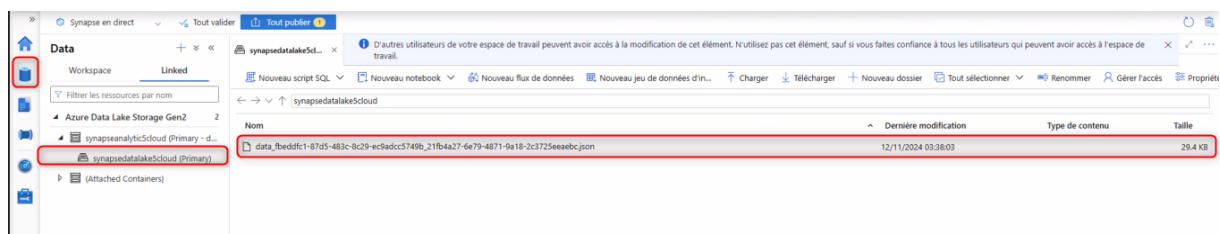
Visualisation de l'environnement Azure Synapse Studio :



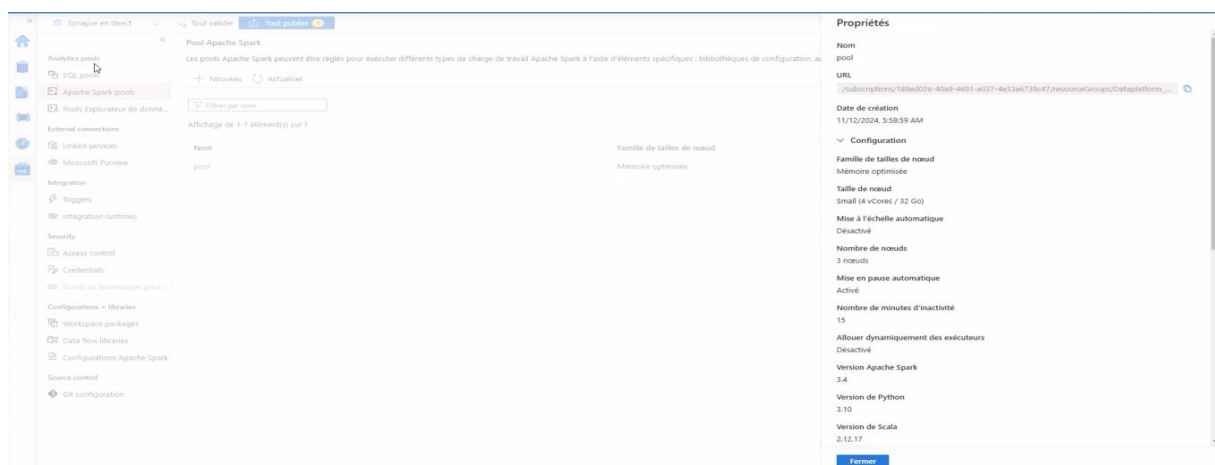
Connection de Synapse Workspace à notre Blob Storage :



Vérification des données :



3.2 : Configuration d'un pool Spark pour le traitement des données avec python



Chargement du notebook :

The screenshot shows a Synapse notebook titled 'Notebook 1'. The left sidebar displays the workspace with 'Azure Data Lake Storage Gen2' and 'synapsedatalake5cloud (Primary)'. The main area shows a code cell with the following PySpark code:

```
1 %%pyspark
2 df = spark.read.load('abfss://synapsedatalake5cloud@datalake5cloud.dfs.core.windows.net/data_bc5eb12-70d5-4bd9-95b8-ae43e4b69d5_fc31f9f-dcfc-49b5-b129-8b0df8d7cd72.json', format=
3 display(df.limit(10))
```

The execution status is 'Prêt' (Ready). Below the code, a table shows the execution details:

ID	Description	État	Phases	Tâches	Heure d'envoi	Durée
Travail 0	load at NativeMethodAccessorImpl.java	Opération réussie	1/1	1/1 réussites	6:08:25 AM, 11/12/24	9 s
Travail 1	getFlowToJsonString at Display.scala:474	Opération réussie	1/1	1/1 réussites	6:08:36 AM, 11/12/24	8 s

The output of the code is displayed as a JSON string:

```
quoteResponse
{"result": "15thDayUsageChange", "result": "15thDayUsageChange"}
```

Chargement du data Frame Spark :

The screenshot shows a Synapse notebook titled 'Notebook5Cloud'. The left sidebar displays the workspace with 'Azure Data Lake Storage Gen2'. The main area shows a code cell with the following PySpark code:

```
1 %%pyspark
2 df = spark.read.load('abfss://synapsedatalake5cloud@datalake5cloud.dfs.core.windows.net/data_877b0874-f33d-4885-a7fc-0f7e759a1ea5_3
3 display(df.limit(10))
```

The execution status is 'Non démarré' (Not started). Below the code, a table shows the execution details:

data	message	success
["userPhotoUrl": "rank": 29, "nick"	undefined	true

Rédaction des requêtes de transformation qui nettoient et formatent les données pour la visualisation :

3.3 : Traitement des données avec python

Extraction et Flattening des Données

Le JSON contient une structure imbriquée, donc nous devons extraire les données de l'objet data

The screenshot shows a Synapse notebook titled 'Notebook5Cloud'. The left sidebar displays the workspace with 'Azure Data Lake Storage Gen2'. The main area shows a code cell with the following PySpark code:

```
1 # Extraction des informations de "data"
2 df = df_raw.selectExpr("explode(data) as data").select("data.*")
3
```

The execution status is 'Prêt' (Ready). Below the code, a table shows the execution details:

data	message	success
["userPhotoUrl": "rank": 29, "nick"	undefined	true

Nettoyage et Transformation des Données

Cette partie se fera en trois étapes :

- Sélection des colonnes pertinentes

- Conversion de l'update Time (en millisecondes depuis l'époque Unix) en date
- Gestion des valeurs nulles

```

1 from pyspark.sql.functions import col, from_unixtime, when
2
3 # Sélection des colonnes et nettoyage des valeurs nulles
4 df_cleaned = df.select(
5     col("encryptedId").alias("user_id"),
6     col("nickName").alias("nickname"),
7     col("userPhotoUrl").alias("photo_url"),
8     col("rank"),
9     col("followerCount").alias("follower_count"),
10    col("leaderboardUrl").alias("leaderboard_url"),
11    from_unixtime(col("updateTime") / 1000).cast("timestamp").alias("update_time")
12).fillna({"nickname": "Unknown", "photo_url": "", "follower_count": 0})
13
14 # Affichage des premières lignes après nettoyage
15 df_cleaned.show(5)
16
17

```

Aperçu des données des 5 premières lignes uniquement

user_id	nickname	photo_url	rank	follower_count	leaderboard_url	update_time
E4C2BCB6FDF2A2A7A...	Ethcoin Founder		29	6174	https://www.binan...	2024-11-12 00:00:00
26D230E2086395890...	Anonymous User-ea...		144	900	https://www.binan...	2024-11-12 00:00:00
8FEB3EA2D767A2732...	Dumbass1		183	2158	https://www.binan...	2024-11-12 00:00:00
61FBD1ADC31E3483...	musthave1uck		231	489	https://www.binan...	2024-11-12 00:00:00
988EE5573E692A34D...	Miracle55555		254	0	https://www.binan...	2024-11-12 00:00:00

Ajout de Colonnes Dérivées

Pour cette partie il nous a fallu ajouter des colonnes dérivées pour faciliter la visualisation, comme une colonne **rank_category** pour catégoriser les utilisateurs en fonction de leur **rank**

```

1 # Ajout d'une colonne de catégorie de rang (par exemple, Top 50, 100, 500, etc.)
2 df_transformed = df_cleaned.withColumn(
3     "rank_category",
4     when(col("rank") <= 50, "Top 50")
5     .when(col("rank") <= 100, "Top 100")
6     .when(col("rank") <= 500, "Top 500")
7     .otherwise("500+")
8 )
9
10 # Affichage des premières lignes après nettoyage
11 df_transformed.show(5)
12

```

Aperçu des données des 5 premières uniquement

user_id	nickname	photo_url	follower_count	leaderboard_url	update_time	rank_category
E4C2BC6FDF2A2A7A...	Ethcoin Founder	29	6174	https://www.binan...	2024-11-12 00:00:00	Top 500
260230E2086395890...	Anonymous User-ea...	144	900	https://www.binan...	2024-11-12 00:00:00	Top 500
8FEB3EA2D767A2732...	Dumbass1	183	2158	https://www.binan...	2024-11-12 00:00:00	Top 500
61FB1D1ADC31E3483...	musthaveluck	231	489	https://www.binan...	2024-11-12 00:00:00	Top 500
988EE5573E692A34D...	Miracle55555	254	0	https://www.binan...	2024-11-12 00:00:00	Top 500

Écriture des Données dans Azure Synapse pour la Visualisation :

Après nettoyage et transformation des données, nous écrivons les données nettoyées dans une table **Synapse**

5. Écriture des Données dans Azure Synapse pour la Visualisation

Après le nettoyage et les transformations, nous pouvons écrire les données nettoyées dans un fichier dans le Data Lake pour la visualisation.

```
1 # Écriture des données transformées dans un fichier CSV
2 df_transformed.write.mode("overwrite").option("header", "true").csv("abfss://synapsedatalake5cloud@datalake5cloud.dfs.core.windows.
3
```

Exécution du travail Opération réussie Spark 2 exécuteurs, 8 cœurs

ID	Description	État	Phases	Tâches	Heure d'envoi	Durée
Travail 8	csv at NativeMethodAccessorimpl.java0	Opération réussie	1/1	1/1 réussites	10:43:30 AM, 11/12/24	2 s

Le résultat peut être visible dans notre espace de stockage « **Azure Data Lake Storage Gen2** »

Nom	Dernière modification	Type de contenu	Taille
_SUCCESS	12/11/2024 10:41:26		
part-00000-878217a0-6507-4178-897d-9a5f0620441f-c000.csv	12/11/2024 10:41:25		25.8 KB

Étape 4 : Visualisation des données avec Power BI

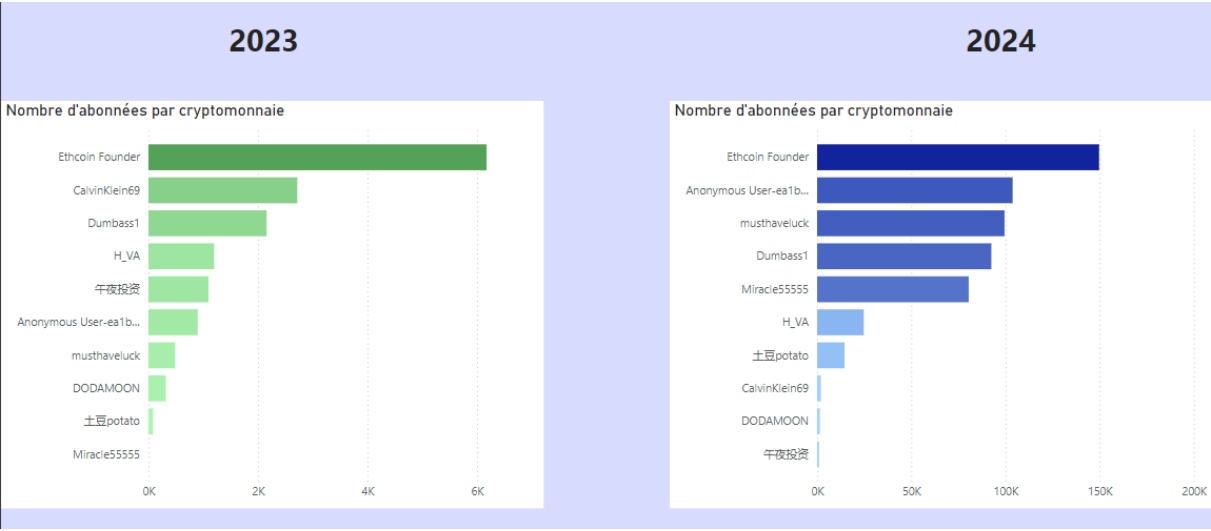
Pour cette partie visualisation des données, nous allons utiliser Power BI.

Voici notre rapport :

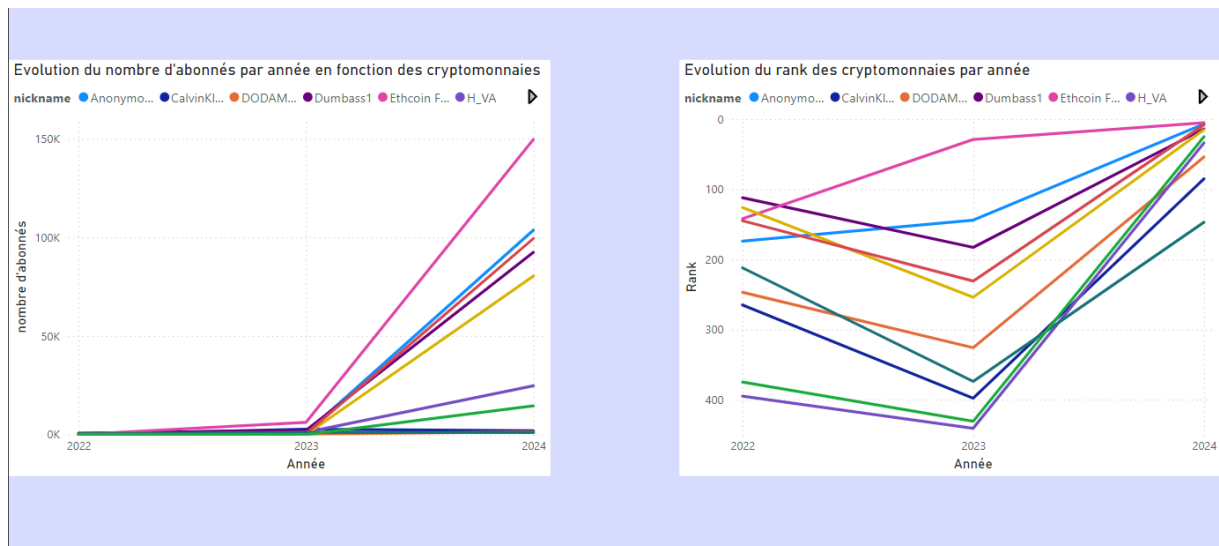
Nous avons créé des TOP pour identifier les cryptomonnaies les mieux classées.



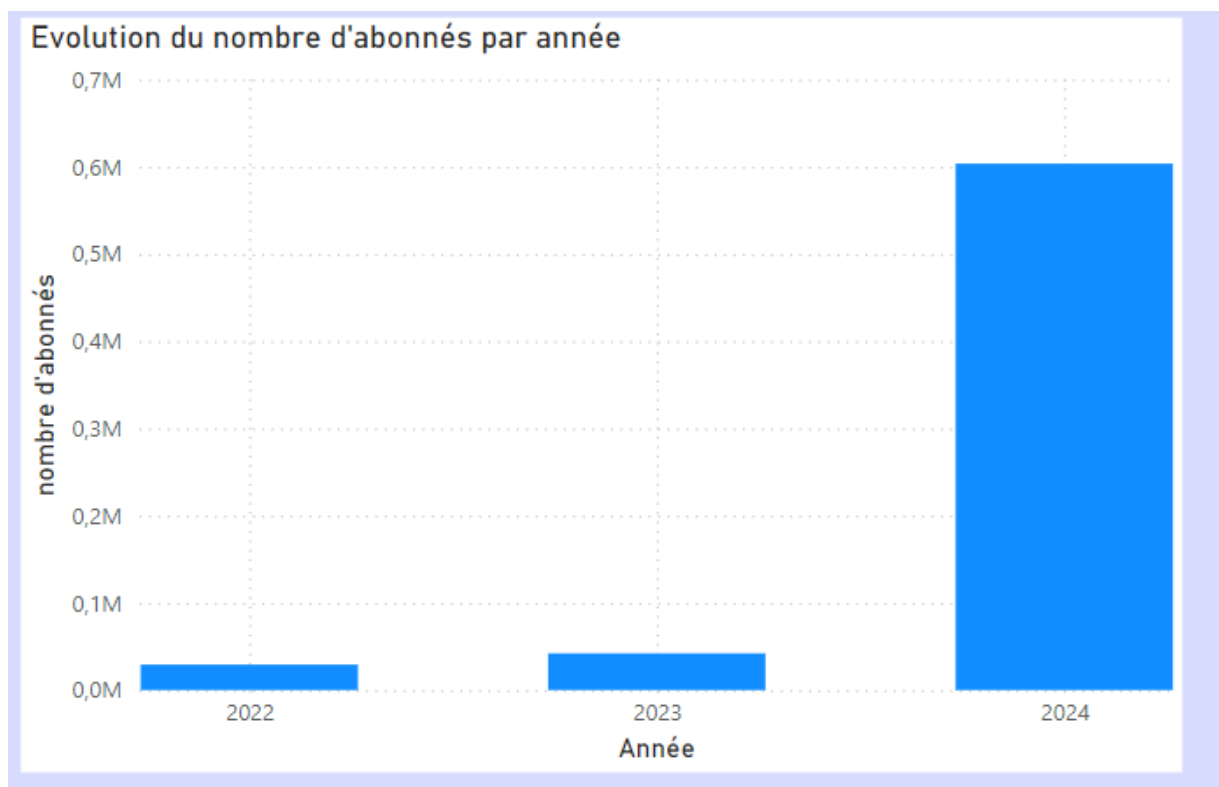
Nous avons fait la même chose pour le nombre d'abonnés.



Nous avons réalisé des graphiques sur l'évolution des cryptomonnaies en fonction du rank et du nombre d'abonnés par années.



Nous avons créé un histogramme pour représenter l'évolution du nombre d'abonnés total par années.



C'est le même graphique que le précédent mais sous forme différente avec des indicateurs.

	2022	2023	2024
nombre d'abonnés	29K	42K	604K

Enfin nous avons les données sous forme tabulaire.

Données de 2022,2023 et 2024			
Année	nickname	Somme de rank	Somme de follower_count
2024	Ethcoin Founder	5	149687
2024	Anonymous User-ea1b015	7	103786
2024	musthaveluck	8	99478
2024	Dumbass1	14	92484
2024	Miracle55555	15	80475
2024	土豆potato	25	14563
2023	Ethcoin Founder	29	6174
2024	H_VA	34	24761
2024	DODAMOON	54	1456
2024	CalvinKlein69	85	2007
2022	Dumbass1	112	743
2022	Miracle55555	126	42
2022	Ethcoin Founder	142	123
2023	Anonymous User-ea1b015	144	900
2022	musthaveluck	145	105
2024	午夜投资	147	1025
2024	Ohtanishohei	148	6142
2024	Anonymous User-f0f6b22	158	2164
2022	Anonymous User-ea1b015	174	426
2023	Dumbass1	183	2158
2022	午夜投资	212	452
2023	musthaveluck	231	489
2022	DODAMOON	247	14
2023	Miracle55555	254	0
2022	CalvinKlein69	265	102
2023	DODAMOON	326	318
2023	午夜投资	374	1095
2022	土豆potato	375	149
2022	H_VA	395	126
2022	CalvinKlein50	398	3717
Total		1095164	675149

Après analyse des données nous pouvons conclure qu'il y a une énorme augmentation du nombre d'abonnés sur les cryptomonnaie que nous avons étudié.

En 2023 on remarque une tendance générale en diminution pour les rank des cryptomonnaies sélectionné puis une augmentation en 2024.