



Projet Neo4j

Exoplanètes

Equipe :

Paul Lelégard

Nicolas Morand

Encadré par

Jérôme Bâton

Année académique

2023/2024

SOMMAIRE

I.	Contexte	3
II.	Données Brutes Sources	3
III.	Pré-traitement avant ingestion dans Neo	3
IV.	Structure Neo4j.....	6
V.	Script d'Ingestion	7
VI.	Scripts de Requêtes.....	10
VII.	Visualisation – Fichier HTML	19

I. Contexte

Nous avons choisi pour ce projet Neo4j d'utiliser les données publiques de la NASA sur les exoplanètes découvertes (confirmées). Il y en a 5539 officielles à ce jour.

II. Données Brutes Sources

Les fichiers sources sont stockés dans le répertoire ./Tables_brutes

A. Site NASA avec tables sur les exoplanètes

Lien :	Fichier brut :
https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=PSCompPars	PSCompPars_2023.11.08_08.32.47.csv

Pour plus de détail sur la définition des variables, se rapporter à ce lien

https://exoplanetarchive.ipac.caltech.edu/docs/API_PS_columns.html

B. Librairie Python Astropy

Lien :
https://github.com/astropy/astropy

C. Liste des constellations & visibilité

Lien :	Fichier brut :
https://www.constellation-guide.com/constellation-list/	List_of_constellations.csv

D. Liste des observatoires

Lien :	Fichier brut :
https://airmass.org/observatories	list_observatories.csv

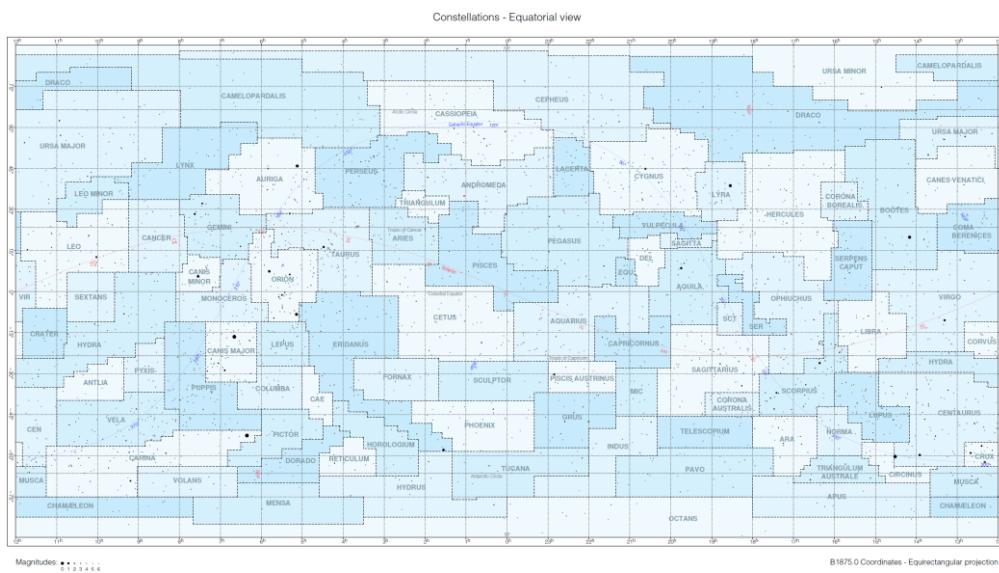
III. Pré-traitement avant ingestion dans Neo

Toutes les données ayant subi un pré-traitement sont stockées dans le répertoire « ./Tables_traitées »

Il est nécessaire d'enregistrer **5 fichiers** dans le répertoire import pour pouvoir charger le modèle complet Neo4j.

A. Liste des exoplanètes et système stellaires

- Ajout des coordonnées x0,y0,z0 : Définition des coordonnées de chaque système solaire (hostname) par rapport à notre système solaire dans un repère x0,y0,z0 à partir des coordonnées polaire ('ra' et 'dec') et de la distance avec notre système solaire 'sy_dist'. Ces 3 variables sont disponibles dans le fichier original de la Nasa.
- Séparation des informations du type spectral : Séparation des informations de rayonnement suivant les règles de Morgan-Keenan en 3 items : st_specclass, st_specsdiv et st_lumclass.
- Détermination des constellations à partir des coordonnées : Détermination des constellation désignée 'con', pour cela nous utiliserons la librairie **Astropy**, ainsi que les coordonnées 'ra' et 'dec'. On vérifiera sur quelques exemples que les étoiles 'hostname' appartiennent bien aux constellations telles que décrit dans le graphe ci-dessous (Eugène Delporte)



Fichier :
planetes_avec_reperes_cartesien_et_constellations.csv

Code de traitement :
Planètes_Pré-traitement.ipynb

B. Voisinage des constellations

Construction d'une table définissant pour chaque constellations ses voisines. Cela permettra de rentrer la map de Eugène Delporte ci-dessus dans Neo4j et de reconstituer cette carte.

Fichier :
Constellations_voisines_une_seule_relation.csv

Code de traitement :
Planètes_Pré-traitement.ipynb

C. Préparation de la Visibilité et alignment des noms des Constellations

Eclatement de la colonne visibilité en 2 variables pour les séparer entre hémisphère Nord et hémisphère Sud. Puis alignment du nom des constellations avec la librairie Astropy ci-dessus. En effet il y a de petits écarts de noms de constellation entre les 2 fichiers.

<u>Fichier :</u> Visibilité_Constellations.csv	<u>Code de traitement :</u> Planètes_Pré-traitement.ipynb
---	--

D. Observatoires astronomiques

Conversion des latitude, longitude en décimal. Recherche des observatoires dans OpenStreetMap grâce à l'API et la librairie Nominatim. Puis recouplement manuel des noms des observatoires entre ce fichier et le fichier de la NASA 'disc_facility', qui sera utile pour faire le lien entre les 2 bases de données.

<u>Fichier :</u> Observatoires_consolidés.csv	<u>Code de traitement :</u> Planètes_Pré-traitement.ipynb
--	--

E. Calcul des distances inter-système stellaires

Calcul des distances inter-étoiles, à partir de la distance avec notre système solaire (x_0, y_0, z_0). Pour des raisons de taille de modèle, nous ne garderons que les distances inférieures à 250 parsecs.

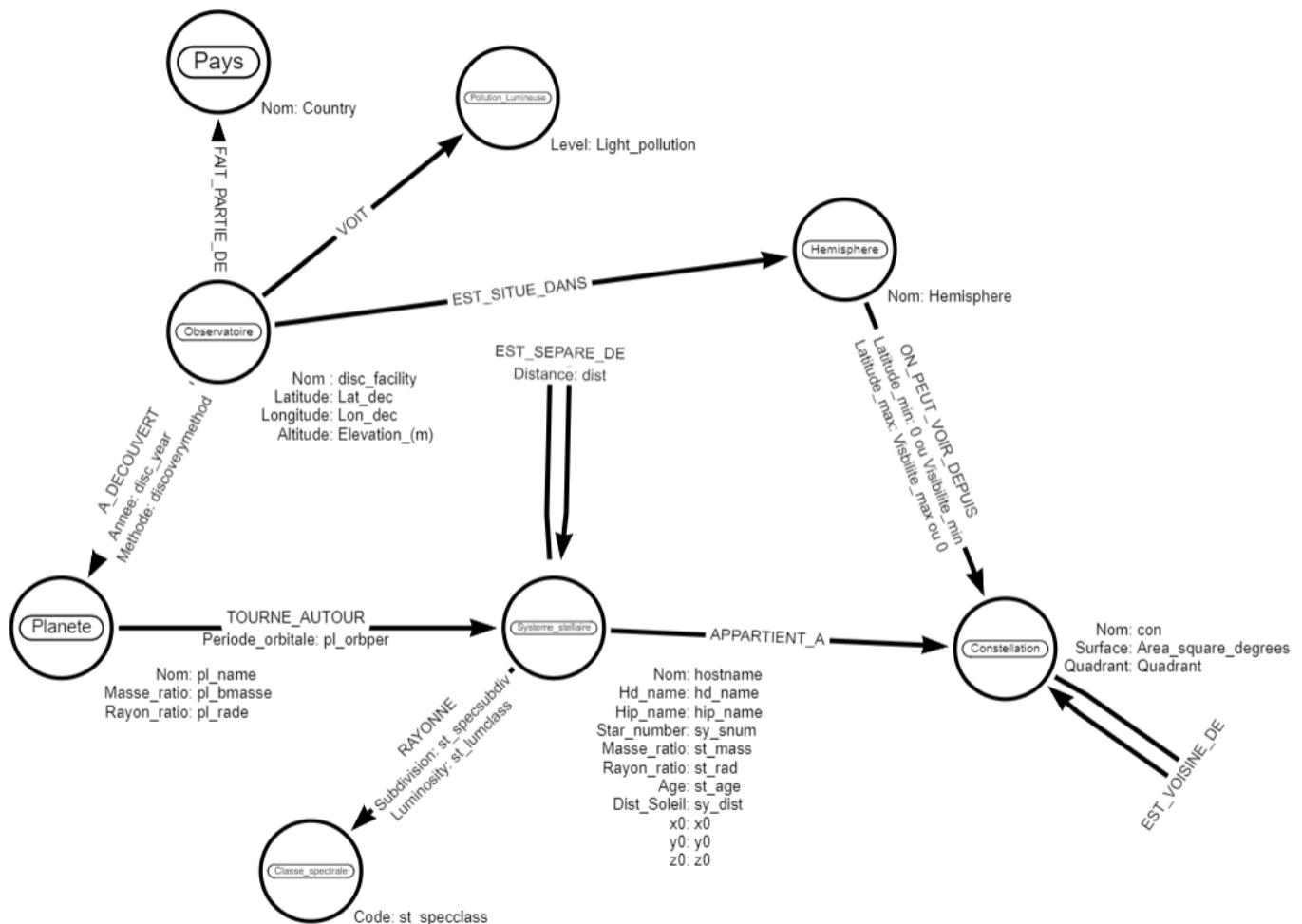
<u>Fichier :</u> Distances_systemes.csv	<u>Code de traitement :</u> Planètes_Pré-traitement.ipynb
--	--

IV. Structure Neo4j

A. Structure des données pour l'ingestion dans Neo4j

Défini avec l'application arrows-app, disponible sous le lien : https://drive.google.com/file/d/1HB_FeWkfFR-vw1zkiryZdsMeOz6Gm8Ma/view?usp=sharing

Le nom à gauche des ‘:’ est le nom de la variable Neo4j, à droite du : est le nom de la variable dans les fichiers csv.



B. Définition des Variables

Nous avons pour la plupart des noms de variables ingérés dans Neo4j mis des noms explicites. Dans le tableau ci-dessous, nous précisons quelques éléments pour certaines d'entre elles. Si besoin plus de détail est disponible, sur le site de la NASA [Planetary Systems and Planetary Systems Composite Parameters Data Column Definitions \(caltech.edu\)](https://exoplanetarchive.ipac.caltech.edu/doc/PSDCOLUMNS.html)

Nom fichier brut	Nom Neo4j	Description
pl_name	Nom	Nom de la planète
pl_bmasse	Masse_ratio	Masse de la planète (par rapport à la Terre)
pl_rade	Rayon_ration	Rayon de la planète (par rapport à la Terre)
pl_orbper	Période_orbitale	Temps (en jours sur Terre) que met la planète pour faire un tour complet autour de son étoile
st_specsubdiv	Subdivision	Classe spectrale Morgan-Keenan
st_lumclass	Luminosity	Classe spectrale Morgan-Keenan
st_specclass	Code	Classe spectrale Morgan-Keenan
hostname	Nom	Nom de système stellaire (étoile)
hd_name	Hd_name	Nom de l'étoile dans le catalogue Henry Draper
hip_name	Hip_name	Nom de l'étoile dans le catalogue Hipparcos
NA	x0,y0,z0	Coordonnées cartésiennes de l'étoile par rapport au soleil
disc_facility	Nom	Nom de l'observatoire ayant découvert la planète

V. Script d'Ingestion

Ci-joint le script d'ingestion pour bâtir le modèle, utilisant les 5 fichiers prétraités.

```
-----
// CREATION DES CONTRAINTES
-----
CREATE CONSTRAINT IF NOT EXISTS
FOR (st:Systeme_stellaire)
REQUIRE st.Nom IS UNIQUE;

CREATE CONSTRAINT IF NOT EXISTS
FOR (pl:Planete)
REQUIRE pl.Nom IS UNIQUE;

CREATE CONSTRAINT IF NOT EXISTS
FOR (con:Constellation)
REQUIRE con.Nom IS UNIQUE;

CREATE CONSTRAINT IF NOT EXISTS
FOR (obs:Observatoire)
REQUIRE obs.Nom IS UNIQUE;

CREATE CONSTRAINT IF NOT EXISTS
FOR (spe:Classe_spectrale)
REQUIRE spe.Code IS UNIQUE;

CREATE CONSTRAINT IF NOT EXISTS
FOR (hem:Hemisphere)
REQUIRE hem.Nom IS UNIQUE;

CREATE CONSTRAINT IF NOT EXISTS
FOR (pay:Pays)
```

```

REQUIRE pay.Nom IS UNIQUE;

CREATE CONSTRAINT IF NOT EXISTS
FOR (pol:Pollution_lumineuse)
REQUIRE pol.Level IS UNIQUE;

//-----
// FICHIER PLANETES + SYSTEME STELLAIRE + CONSTELLATION -----
//-----

LOAD CSV WITH HEADERS FROM 'file:///planetes_avec_reperes_cartesien_et_constellations.csv' AS line FIELDTERMINATOR ','
//
WITH line
//
MERGE (st:Systeme_stellaire{Nom:line.hostname})
SET      st.Hd_name = COALESCE(line.hd_name, 'inconnu'),
         st.Hip_name = COALESCE(line.hip_name, 'inconnu'),
         st.Star_number = toInteger(line.sy_snum),
         st.Masse_ratio = COALESCEtoFloat(line.st_mass), 'inconnu'),
         st.Rayon_ratio = COALESCEtoFloat(line.st_rad), 'inconnu'),
         st.Age = COALESCEtoFloat(line.st_age), 'inconnu'),
         st.Dist_Soleil = COALESCEtoFloat(line.sy_dist), 'inconnu'),
st.x0 = COALESCE(ROUNDtoFloat(line.x0), 2), 'inconnu'),
st.y0 = COALESCE(ROUNDtoFloat(line.y0), 2), 'inconnu'),
st.z0 = COALESCE(ROUNDtoFloat(line.z0), 2), 'inconnu')
MERGE (pl:Planete{Nom:line.pl_name})
SET      pl.Masse_ratio = COALESCEtoFloat(line.pl_bmasse), 'inconnu'),
         pl.Rayon_ratio = COALESCEtoFloat(line.pl_rade), 'inconnu')
MERGE (con:Constellation{Nom:line.con})
MERGE (obs:Observatoire{Nom:line.disc_facility})
MERGE (spe:Classe_spectrale{Code:COALESCE(line.st_specclass, 'inconnu')})
//
MERGE (pl)-[:TOURNE_AUTOUR {Periode_orbitale:COALESCEtoFloat(line.pl_orbper), 'inconnu'}]->(st)
MERGE (st)-[:APPARTIENT_A]->(con)
MERGE (obs)-[:A_DECOUVERT {Année:COALESCE(toInteger(line.disc_year), 'inconnu'), Méthode:COALESCE(line.discoverymethod, 'inconnu')}}]->(pl)
MERGE (st)-[:RAYONNE]->(spe)
SET      ray.Subdivision = COALESCEtoFloat(line.st_specsdiv), 'inconnu'),
         ray.Luminosity = COALESCE(line.st_lumclass, 'inconnu');
//
//-----
// FICHIER OBSERVATOIRES -----
//-----

LOAD CSV WITH HEADERS FROM 'file:///Observatoires_consolidés.csv' AS line FIELDTERMINATOR ','
//
WITH line
//
MERGE (obs:Observatoire{Nom:line.Observatoire})
SET      obs.Latitude = COALESCEtoFloat(line.Lat_dec), 'inconnu'),
         obs.Longitude = COALESCEtoFloat(line.Lon_dec), 'inconnu'),
         obs.Altitude = COALESCEtoFloat(line.Elevation), 'inconnu')
MERGE (hem:Hemisphere{Nom:COALESCE(line.Hemisphere, 'inconnu'))}
MERGE (pay:Pays{Nom:COALESCE(line.Country, 'inconnu'))}
MERGE (pol:Pollution_lumineuse{Level:COALESCE(line.Light_pollution, 'inconnu'))})
//
MERGE (obs)-[:FAIT_PARTIE_DE]->(pay)
MERGE (obs)-[:VOIT]->(pol)
MERGE (obs)-[:EST_SITUE_DANS]->(hem);

//-----
// FICHIER VISIBILITE CONSTELLATIONS-----
//-----

LOAD CSV WITH HEADERS FROM 'file:///Visibilité_Constellations.csv' AS line FIELDTERMINATOR ','
//
WITH line
//
MERGE (hemN:Hemisphere{Nom:"Nord"})
MERGE (hemS:Hemisphere{Nom:"Sud"})
MERGE (con:Constellation{Nom:line.Constellation})
MERGE (con)<-[:ON_Peut_VoirDepuis {Latitude_max:COALESCE(toInteger(line.Visibilite_Nord), 'inconnu'))}-(hemN)

```

```

MERGE (con)<[:ON_PUET_VOIR_DEPUIS {Latitude_min:COALESCEtoInteger(line.Visibilite_Sud), 'inconnu')}]>-(hemS);
//-----
//FICHIER CONSTELLATION VOISINES -----1 RELATION PAR COUPLE DE CONSTELLATION -----
//-----
LOAD CSV WITH HEADERS FROM 'file:///Constellations_voisines_une_seule_relation.csv' AS line FIELDTERMINATOR ','
//
WITH line
MATCH (con1:Constellation{Nom:line.con1})
MATCH (con2:Constellation{Nom:line.con2})
MERGE (con1)-[:EST_VOISINE_DE]->(con2);

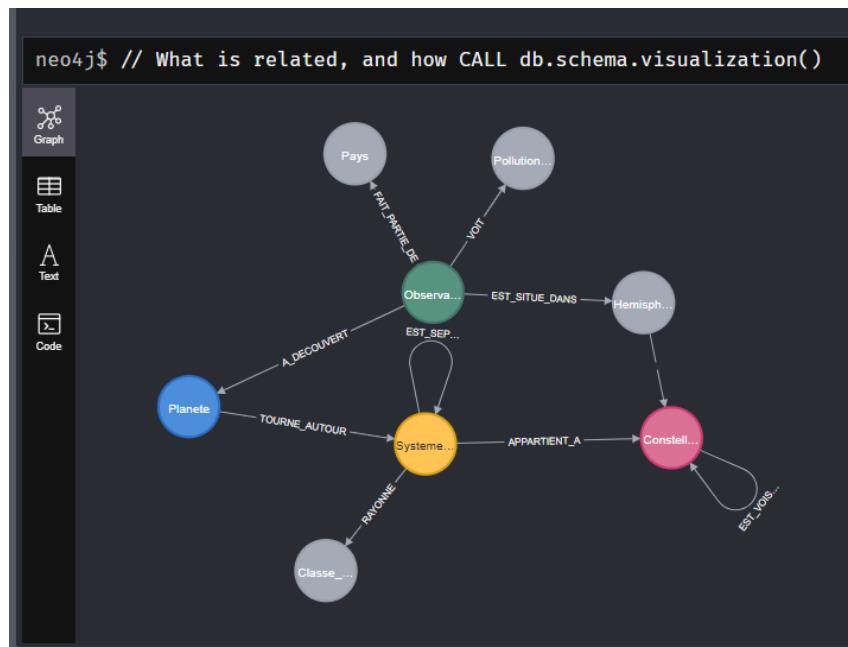
//-----
//-----
// FICHIER DISTANCE ENTRE LES SYSTEMES STELLAIRES -----
//-----
:auto LOAD CSV WITH HEADERS FROM 'file:///Distances_systemes.csv' AS line FIELDTERMINATOR '|'
CALL {
    WITH line
    MATCH (s1:Systeme_stellaire{Nom:line.syst1})
    MATCH (s2:Systeme_stellaire{Nom:line.syst2})
    MERGE (s1)-[dist:EST_SEPARE_DE]->(s2)
    SET dist.Distance =toFloat(line.dist)
} IN TRANSACTIONS OF 50000 ROWS
//-

```

Bilan du modèle :

5 fichiers csv après pré-traitement sont ingérés dans Neo4j pour bâtir le modèle.

Noeuds	Relations
Nb : 9855	Nb : 1 635 303
Systeme_stellaire	TOURNE_AUTOEUR
Planete	EST_VOISINE_DE
Constellation	APPARTIENT_A
Observatoire	A_DECOUVERT
Classe_spectrale	RAYONNE
Hemisphere	VOIT
Pays	EST_SITUE_DANS
Pollution_lumineuse	ON_PUET_VOIR_DEPUIS
	FAIT_PARTIE_DE
	EST_SEPARE_DE



VI. Scripts de Requêtes

Les requêtes sont écrites dans le fichier Requetes_Neo4j.txt

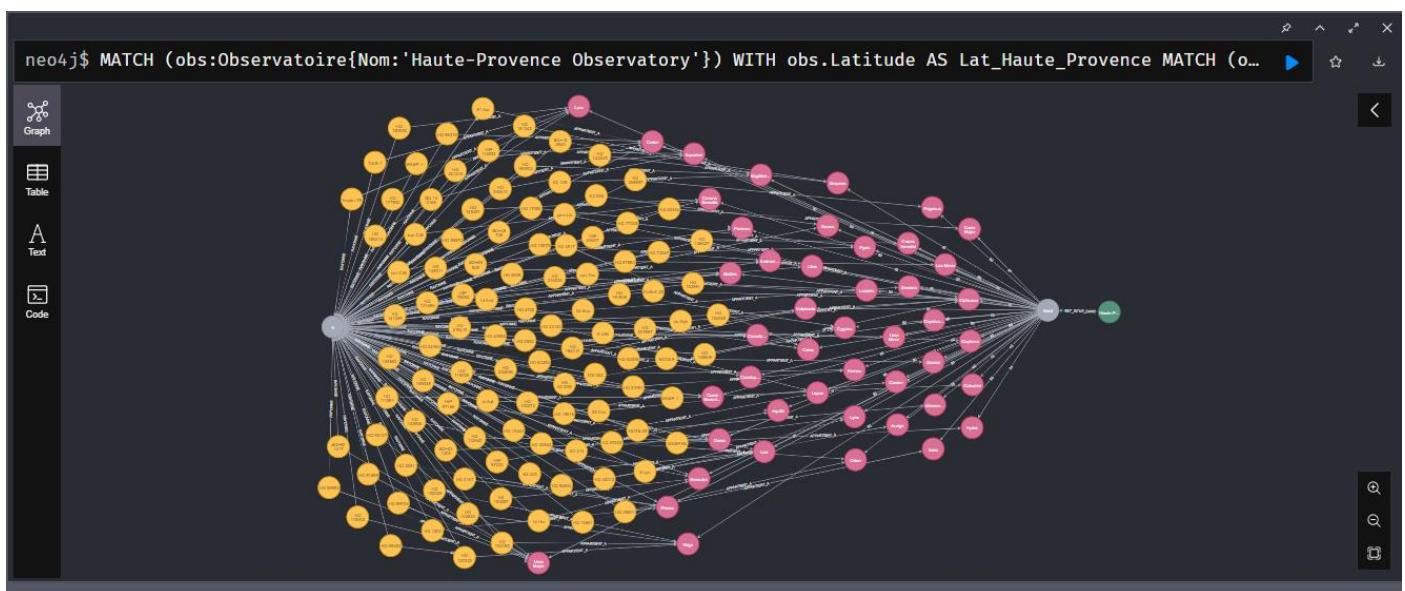
Les requêtes ont été testées dans Neo4j browser et avec le code Python Exoplanet_viz.ipynb qui permet de faire l'interface entre python et la base de données Neo4j, puis créer un fichier html par requête.

Remarque : Nous avons observé que les paramètres de mémoire alloués au projet impactent légèrement le résultat de certaines requêtes. De plus nous avons aussi observé de légers écarts entre les différents PC (par exemple pour la requête_3, on peut trouver une total_distance de 662 ou 679 (soit 2.2%).

A. Requête 1 :

« Un concours est lancé, le prochain observatoire qui trouvera une nouvelle planète autour d'une étoile de classe spectrale « K0 » gagnera 1 million de dollars ! Quels sont les systèmes que l'observatoire de Haute-Provence doit surveiller pour avoir une chance de gagner ? »

Résultat :



Traduction de la requête en code Cypher :

```
MATCH (obs:Observatoire{Nom:'Haute-Provence Observatory'})  
WITH obs.Latitude AS Lat_Haute_Provence  
MATCH (obs:Observatoire)-[r1:EST_SITUÉ_DANS]->(hem:Hémisphère)-[r2:ON_PEUUT_VOIR_DEPUIS]->(con:Constellation)<-[r3:APPARTIENT_A]-  
(st:Système_stellaire)-[r4:RAYONNE]->(sp:Classe_spectrale)  
WHERE (obs.Nom = 'Haute-Provence Observatory') AND (sp.Code = 'K') AND (r4.Subdivision = 0.0) AND (r2.Latitude_max > Lat_Haute_Provence)  
RETURN obs,r1,hem,r2,con,r3,st,r4,sp
```

B. Requête 2 :

« La course à l'espace est rude parmi les habitants de la Terre, chaque pays cherche à prouver qu'il est le meilleur. Quel pays peut se vanter d'avoir l'observatoire ayant découvert le plus de planètes dans des systèmes stellaires à plus de 2 étoiles ? »

Résultat :

obs.Nom	pay.Nom	nb
"La Silla Observatory"	"Chile"	11
"Kepler"	"inconnu"	8
"K2"	"inconnu"	7
"Transiting Exoplanet Survey Satellite (TESS)"	"inconnu"	6
"W. M. Keck Observatory"	"United States"	5
"SuperWASP"	"inconnu"	5
"HATNet"	"inconnu"	4
"Multiple Observatories"	"United States"	3
"Lick Observatory"	"United States"	2
"McDonald Observatory"	"United States"	1

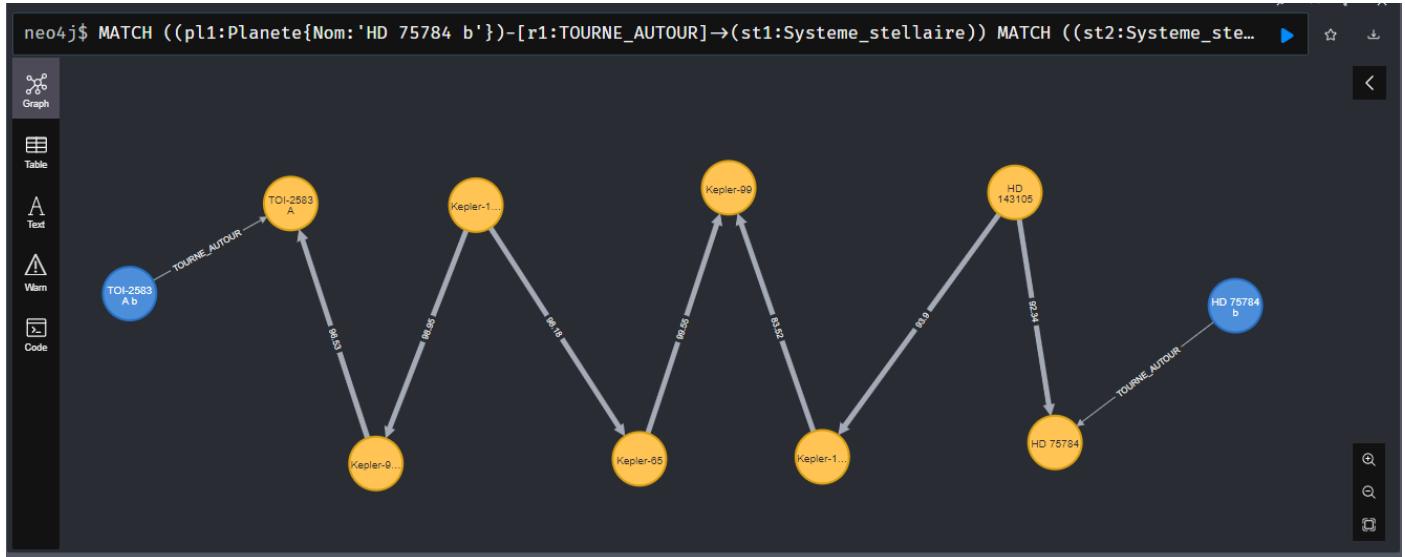
Traduction de la requête en code Cypher :

```
MATCH (pay:Pays)-<[FAIT_PARTIE_DE]-(obs:Observatoire)-[r:A_DECOUVERT]-(pl:Planete)-[:TOURNE_AUTOUR]->(st:Systeme_stellaire)
WHERE (st.Star_number > 2)
RETURN obs.Nom, pay.Nom, count(pl) AS nb
ORDER BY nb DESC
LIMIT 10
```

C. Requête 3 :

« Le peuple de planète Nolagua (HD 75784 b) a découvert qu'il y avait de la vie intelligente sur Chetune (TOI-2583 A b). Par peur de perdre leur suprématie militaire sur leur voisinage, ils préfèrent détruire tout ce qui se trouve sur cette malheureuse planète. Cependant, leurs vaisseaux armés sont obligés de faire un plein de carburant tous les 100 parsecs, par quels systèmes doivent il passer pour atteindre leur destination ? »

Résultat :



pl1	r1	path	pl2	r2	totaldistance
(:Planete {Masse_ratio: 3} [:TOURNE_AUTOUR {Periode_17.83,Rayon_ratio: 13.8,N_orbitale: 341.2}] nom: "HD 75784 b")			(:Planete {Masse_ratio: 7} [:TOURNE_AUTOUR {Periode_1.26,z0: 19.56,Dist_1.26,x0: -56.82,Rayon_ratio: 3.4,Star_number: 1,Hd_name: "HD 75784",Age: 5.36})<-[EST_SEPARER_DE {Distance: 92.34}]-(:Systeme_stellaire {Hip_name: "HIP 77838",Masse_ratio: 1.51,z0: 43.59,Dist_Soleil: 46.7789,y0: -14.46,x0: -8.9,Rayon_ratio: 1.65,Star_number: 1,Hd_name: "HD 143105",Age: "inconnu"})-[EST_SEPARER_DE {Distance: 93.9}]->(:Systeme_stellaire {Hip_name: "HIP 43569",Masse_ratio: 1.4571,Rayon_ratio: 14.46,orbitale: 4.5207265})		662.97

La distance totale qu'ils auront à parcourir sera de 662.97 parsecs, par étape inférieure à 100 parsecs.

Traduction de la requête en code Cypher :

```

MATCH ((pl1:Planete{Nom:'HD 75784 b'})-[r1:TOURNE_AUTOUR]->(st1:Systeme_stellaire))
MATCH ((st2:Systeme_stellaire)<-[r2:TOURNE_AUTOUR]-(pl2:Planete{Nom:'TOI-2583 A b'}))
MATCH path = shortestPath((st1)-[r3:EST_SEPARER_DE*]-(st2))
WHERE all(rel in relationships(path) WHERE rel.Distance < 100.0)
RETURN pl1,r1,path,pl2,r2,REDUCE(s = 0, rel IN relationships(path) | s + rel.Distance) AS totaldistance
    
```

D. Requête 4 :

« Une secte de fanatiques d'astrologie qui ne comporte que des membres de signe Vierge cherche un endroit pour vivre. Il est hors de question d'aller autre-part que sur une planète présente dans la constellation de la Vierge ! Mais est-ce qu'il est possible d'en trouver une qui ne soit pas trop lourde, ni trop légère ? Une gravité proche de la Terre est essentielle pour vivre à la surface. Il faudrait que l'étoile ne soit pas trop lourde non plus, les étoiles légères vivent plus longtemps. »

Résultat :

	syst.Nom	syst.Masse_ratio	p.Nom	p.Masse_ratio	tour.Periode_orbitale	con.Nom
	"Ross 128"	0.17	"Ross 128 b"	1.4	9.8658	"Virgo"
	"EPIC 228836835"	0.42	"EPIC 228836835 b"	1.37	0.728113	"Virgo"
	"K2-156"	0.66	"K2-156 b"	1.37	0.813143	"Virgo"
	"K2-16"	0.68	"EPIC 201754305 d"	1.08	2.71578	"Virgo"
	"K2-228"	0.71	"K2-228 b"	1.92	2.69828	"Virgo"
	"K2-279"	0.79	"K2-279 b"	1.92	7.122605	"Virgo"
	"EPIC 201841433"	0.8	"EPIC 201841433 b"	1.12	4.1698	"Virgo"
	"K2-190"	0.9	"K2-190 c"	1.34	21.574023	"Virgo"
	"K2-157"	0.94	"K2-157 b"	0.972	0.365261	"Virgo"
	"EPIC 212624936"	1.0	"EPIC 212624936 b"	1.87	0.578657	"Virgo"

Moins de 1 jour pour faire le tour de son étoile ? Cette planète est vraiment proche de son étoile ! Vous êtes sûr qu'il fera bon vivre là-bas ?

Traduction de la requête en code Cypher :

```

MATCH (con:Constellation{Nom:'Virgo'})-[:APPARTIENT_A]-(syst:Systeme_stellaire)-[tour:TOURNE_AUTOEUR]->(p:Planete)
WHERE p.Masse_ratio <= 2 AND p.Masse_ratio > 0.5
Return syst.Nom, syst.Masse_ratio, p.Nom, p.Masse_ratio, tour.Periode_orbitale, con.Nom
ORDER BY syst.Masse_ratio ASC, p.Masse_ratio ASC

```

E. Requête 5 :

« Les habitants du système stellaire Koter voient grand. Ils ont besoin de beaucoup d'énergie à cause de la consommation de leurs mineurs de KotCoin. Ils cherchent une planète tournant autour d'une étoile de luminosité V, où il fera bon vivre, avec la contrainte qu'il y ait une géante ou une supergéante dans le voisinage (I, II ou III). Cela permettrait d'y installer une sphère de Dyson suffisamment puissante. Tant qu'à faire, il faudrait que l'étoile en question soit plutôt jeune, pour que ce soit rentable. Les méga-corporations ne voudront jamais payer une sphère pour une étoile en fin de vie. »

Résultat :

s1.Nom	s2.Nom	ray2.Luminosity	d.Distance	s2.Age
"HD 207496"	"nu Oct A"	"III"	2.42	2.8
"HD 27894"	"HD 29399"	"III"	2.89	6.2
"HD 260655"	"HD 62509"	"III"	3.35	0.98
"HD 199509"	"nu Oct A"	"III"	3.65	2.8
"55 Cnc"	"HD 62509"	"III"	3.7	0.98
"HD 45184"	"7 CMa"	"III"	4.19	4.3
"GJ 3021"	"nu Oct A"	"III"	4.5	2.8
"HD 100777"	"ups Leo"	"III"	4.66	"inconnu"
"HR 810"	"HD 27442"	"III"	4.98	10.0
"GJ 251"	"HD 62509"	"III"	5.02	0.98

Traduction de la requête en code Cypher :

```
MATCH (:Classe_spectrale)-<[ray1:RAYONNE{Luminosity:'V'}]->(s1:Systeme_stellaire)-[d:EST_SEPARE_DE]->(s2:Systeme_stellaire)-[ray2:RAYONNE]->(:Classe_spectrale)
WHERE ray2.Luminosity = 'I' OR ray2.Luminosity = 'III' OR ray2.Luminosity = 'II'
RETURN s1.Nom, s2.Nom, ray2.Luminosity, d.Distance, s2.Age
ORDER BY d.Distance ASC
LIMIT 10
```

F. Requête 6 :

« Le peuple terrien cherche à coloniser un nouveau système stellaire. Faisant constat des dégâts qu'il a pu faire dans le système solaire en exploitant toutes ses ressources, la communauté scientifique cherche un système comportant le plus de planètes possibles. Epuiser toutes les ressources présentes prendra plus de temps. Il serait idéal d'avoir une étoile semblable au soleil en terme de taille, une luminosité de V est non négociable pour qu'il y fasse bon vivre. Il faudrait également que l'étoile ne soit pas trop vieille, une géante rouge ne serait pas bon pour le tourisme interstellaire. Est-ce qu'on peut en trouver une qui ne soit pas trop loin de la Terre, pour que le voyage puisse se faire sur un week-end ? »

Résultat :

neo4j\$ MATCH (:Classe_spectrale)->[ray:RAYONNE]-(syst:Systeme_stellaire)->[tour:TOURNE_AUTOUR]-(p:Planete)

syst.Nom	syst.Age	syst.Dist_Soleil	ray.Luminosity	nb_p
"K2-138"	2.8	1202.585	"V"	6
"HD 10180"	4.3	38.9607	"V"	6
"HD 34445"	8.5	46.0907	"V"	6
"HD 191939"	8.7	53.6089	"V"	6
"HD 219134"	12.46	6.53127	"V"	6
"GJ 667 C"	2.0	7.24396	"V"	5
"Kepler-32"	2.69	323.847	"V"	5
"Kepler-296"	4.2	167.0	"V"	5
"HD 40307"	4.5	12.9363	"V"	5
"HD 108236"	6.7	64.5978	"V"	5

MAX COLUMN WIDTH:

Traduction de la requête en code Cypher :

```

MATCH (:Classe_spectrale)->[ray:RAYONNE]-(syst:Systeme_stellaire)->[tour:TOURNE_AUTOUR]-(p:Planete)
WITH syst, COUNT(tour) AS nb_p, ray
WHERE nb_p >= 3 AND syst.Age <> 'inconnu' AND ray.Luminosity = "V"
RETURN syst.Nom, syst.Age, syst.Dist_Soleil, ray.Luminosity, nb_p
ORDER BY nb_p DESC, syst.Age ASC, syst.Dist_Soleil ASC
LIMIT 10

```

G. Requête 7 :

« La communauté scientifique terrienne observe un phénomène étrange. Une grande masse d'objets mobiles semble se déplacer depuis Nolagua (HD 75784 b). Par quelles constellations les télescopes humains verront passer les vaisseaux armés se dirigeants vers Chetune (TOI-2583 A b) ? »

Résultat :



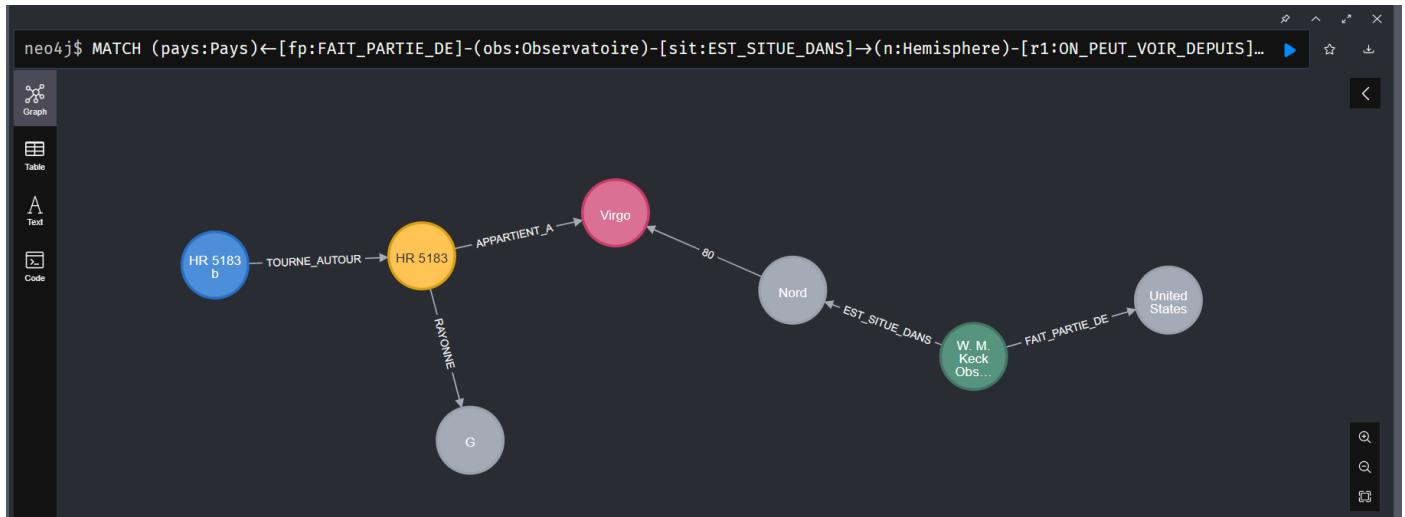
Traduction de la requête en code Cypher :

```
MATCH ((p1:Planete{Nom:'HD 75784 b'})-[r1:TOURNE_AUTOOUR]-(st1:Systeme_stellaire)-[r11:APPARTIENT_A]-(con1:Constellation))
MATCH ((con2:Constellation)-[r21:APPARTIENT_A]-(st2:Systeme_stellaire)-[r22:TOURNE_AUTOOUR]-(p2:Planete{Nom:'TOI-2583 A b'}))
RETURN p1,r1,st1,r11,p2,r21,st2,r22,shortestPath((con1)-[:EST_VOISINE_DE*]-(con2))
```

H. Requête 8 :

«Les élèves du Mastère Spécialisé EBDE à l'UTT souhaitent passer plus de temps sur leurs projets école. Ils ont entendu dire que des fanatiques cherchent à rejoindre la constellation de la Vierge. Est-ce qu'il n'y aurait pas une planète avec une très longue période orbitale autour d'une étoile semblable au Soleil (luminosité V) ? Un mastère Spécialisé d'un an là-bas permettrait de faire beaucoup plus de chose ! Les États-Unis commencent à voir d'un mauvais œil le projet farfelu de la secte et cherchent un observatoire qui pourra surveiller leurs activités. »

Résultat :



pays.Nom	obs.Nom	n.Nom	r1.Latitude_max	con.Nom	s.Nom	tour.Periode_orbitale	p.Nom	ray1.Luminosity
"United States"	"W. M. Keck Observatory"	"Nord"	80	"Virgo"	"HR 5183"	27000.0	"HR 5183 b"	"v"

Les années de travail sur HR 5183 b vont être de 27 000 jours Terriens, bon courage ! 😊

Traduction de la requête en code Cypher :

```

MATCH (pays: Pays) <-[fp: FAIT_PARTIE_DE]-(obs: Observatoire)-[sit: EST_SITUE_DANS]-(n: Hemisphere)-[r1: ON_PEUt_VoIR_DEPUIS]->(con: Constellation{Nom: 'Virgo'}) <-[app: APPARTIENT_A]-(s: Système_stellaire)-<-[tour: TOURNE_AUTOEUR]-(p: Planète)
MATCH (cl: Classe_spectrale) <- [ray1: RAYONNE{Luminosity: 'V'}] -> (s: Système_stellaire)
WHERE pays.Nom = "United States"
RETURN cl, pays, fp, obs, sit, n, r1, con, app, s, tour, p, ray1
//RETURN pays.Nom, obs.Nom, n.Nom, r1.Latitude_max, con.Nom, s.Nom, tour.Periode_orbitale, p.Nom, ray1.Luminosity
ORDER BY tour.Periode_orbitale DESC
LIMIT 1

```

Remarque : Pour afficher la version graphe, il faut utiliser le premier RETURN, pour la version Text, utiliser le deuxième RETURN.

I. Requête 9 :

« La course à l'espace continue sur Terre. Le président du Chili qui a prouvé sa valeur avec son nombre de planètes dans des systèmes de plus de 2 étoiles, cherche à savoir s'il peut ajouter des arguments supplémentaires. Est-ce qu'il est le premier à avoir découvert une planète dans la constellation visible par le plus de monde sur Terre ? »

Résultat :

p.Nom	obs.Nom	pays.Nom	syst.Nom	con.Nom	d.Annee
"HD 200964 c"	"Lick Observatory"	"United States"	"HD 200964"	"Equuleus"	2010
"HD 200964 b"	"Lick Observatory"	"United States"	"HD 200964"	"Equuleus"	2010
"HAT-P-65 b"	"HATNet"	"inconnu"	"HAT-P-65"	"Equuleus"	2016
"WASP-90 b"	"SuperWASP"	"inconnu"	"WASP-90"	"Equuleus"	2016
"MASCARA-1 b"	"Multiple Observatories"	"United States"	"MASCARA-1"	"Equuleus"	2017
"LSPM J2116+0234 b"	"Calar Alto Observatory"	"Spain"	"LSPM J2116+0234"	"Equuleus"	2019

Non le Chili n'est pas le premier pays, ce sont les États-Unis.

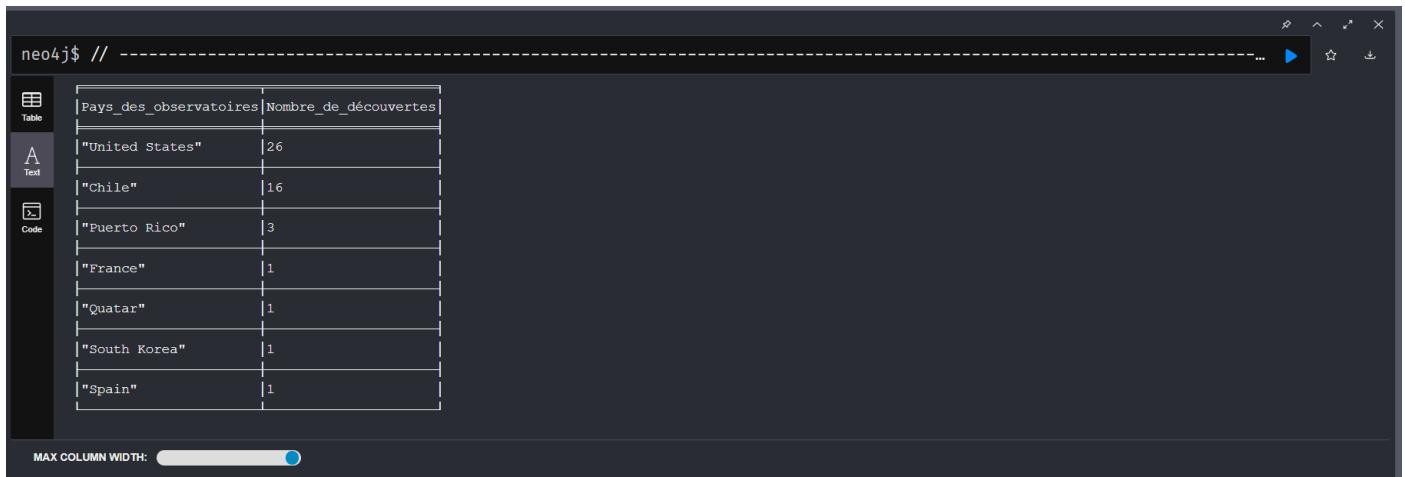
Traduction de la requête en code Cypher :

```
MATCH (n:Hemisphere{Nom:"Nord"})-[r1:ON_PEUT_VOIR_DEPUIS]->(con:Constellation)<-[r2:ON_PEUT_VOIR_DEPUIS]-(s:Hemisphere{Nom:'Sud'})  
WITH con, n, r1, s, r2  
ORDER BY r1.Latitude_max DESC, r2.Latitude_min ASC  
LIMIT 1  
MATCH (con)<-[:APPARTIENT_A]-(syst:Systeme_stellaire)<-[TOURNE_AUTOEUR]-(p:Planete)<-[d:A_DECOUVERT]-(obs:Observatoire)-[:FAIT_PARTIE_DE]->(pays:Pays)  
RETURN p.Nom, obs.Nom, pays.Nom, syst.Nom, con.Nom, d.Annee  
ORDER BY d.Annee ASC
```

J. Requête 10 :

« Faute d'avoir trouvé une planète à leur convenance dans la constellation de la Vierge pour y vivre, la secte de fanatiques revoit ses projets fortement à la baisse (ils ont perdu beaucoup de donateurs après leur dernière mésaventure il faut dire, les finances sont au plus bas). Ne pourraient-ils pas s'installer dans le pays qui a découvert le plus de planètes appartenant à la constellation de la Vierge ? »

Résultat :



The screenshot shows the Neo4j browser interface with a dark theme. On the left, there are three tabs: 'Table' (selected), 'Text', and 'Code'. The main area displays a table with two columns: 'Pays_des_observatoires' and 'Nombre_de_découvertes'. The data is as follows:

Pays_des_observatoires	Nombre_de_découvertes
"United States"	26
"Chile"	16
"Puerto Rico"	3
"France"	1
"Quatar"	1
"South Korea"	1
"Spain"	1

At the bottom left, there is a slider labeled 'MAX COLUMN WIDTH:'.

Traduction de la requête en code Cypher :

```
MATCH (pay:Pays)-[FAIT_PARTIE_DE]-(obs:Observatoire)-[r:A_DECOUVERT]-(pl:Planete)-[:TOURNE_AUTOUR]->(st:Système_stellaire)-[:APPARTIENT_A]->(con:Constellation)
WHERE (con.Nom = "Virgo") AND (pay.Nom <> "inconnu")
RETURN pay.Nom AS Pays_des_observatoires, count(pl) as Nombre_de_découvertes
ORDER BY Nombre_de_découvertes DESC
```

VII. Visualisation – Fichier HTML

Nous pouvons visualiser directement le résultat des requêtes dans Neo4j, toutefois comme tous les utilisateurs n'auront pas accès à Neo4j browser, nous avons fait un notebook Exoplanet_Viz.ipynb pour interfaçer avec la base de données neo4j.

2 types de requêtes :

- ⇒ Requêtes avec résultat de type graphe
- ⇒ Requêtes avec résultat de type dataframe (tableau)

Afin de faciliter la visualisation pour l'utilisateur, le résultat des requêtes est visible dans le fichier « Requête_x.html »