

UNIVERSIDAD DEL VALLE DE GUATEMALA

1CC2003 - Proyecto 2

Sección 10

MSc. Douglas Leonel Barrios Gonzalez



Manual de Usuario

**JUAN CARLOS BAJAN CASTRO
JOSE DANIEL GONZALEZ CARRILLO
SEBASTIAN ARISTONDO PEREZ**

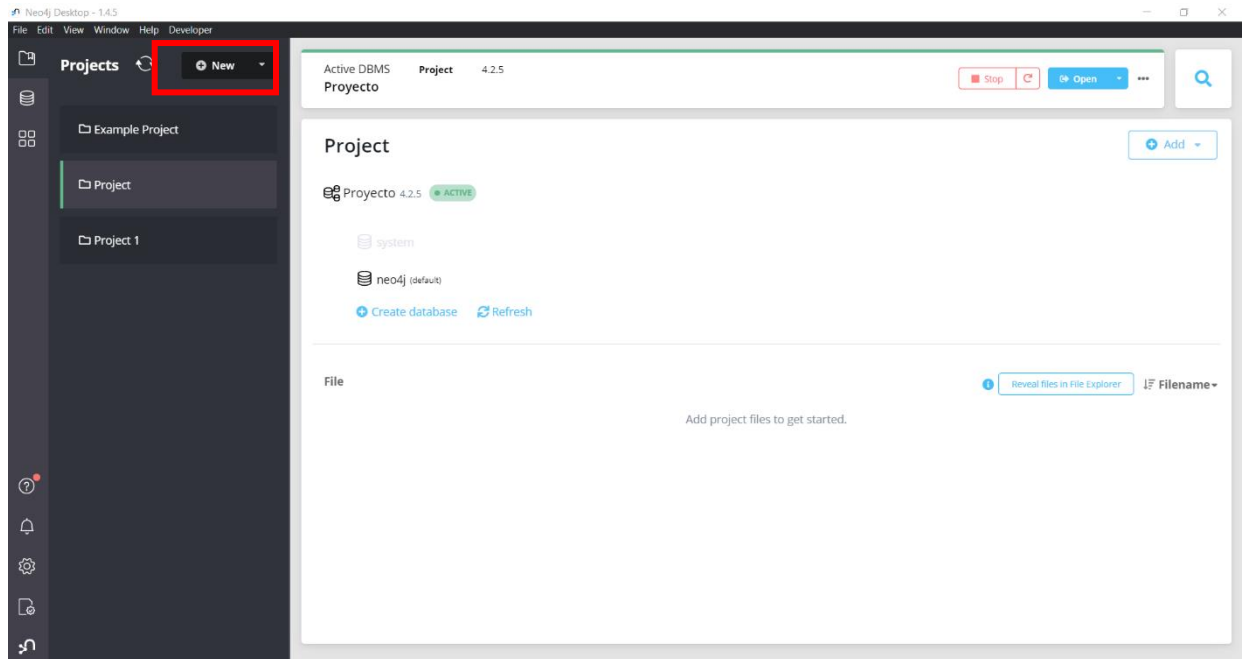
GUATEMALA, 02 de mayo de 2021

Pasos para la instalación y uso de la base de datos para algoritmo de recomendación.

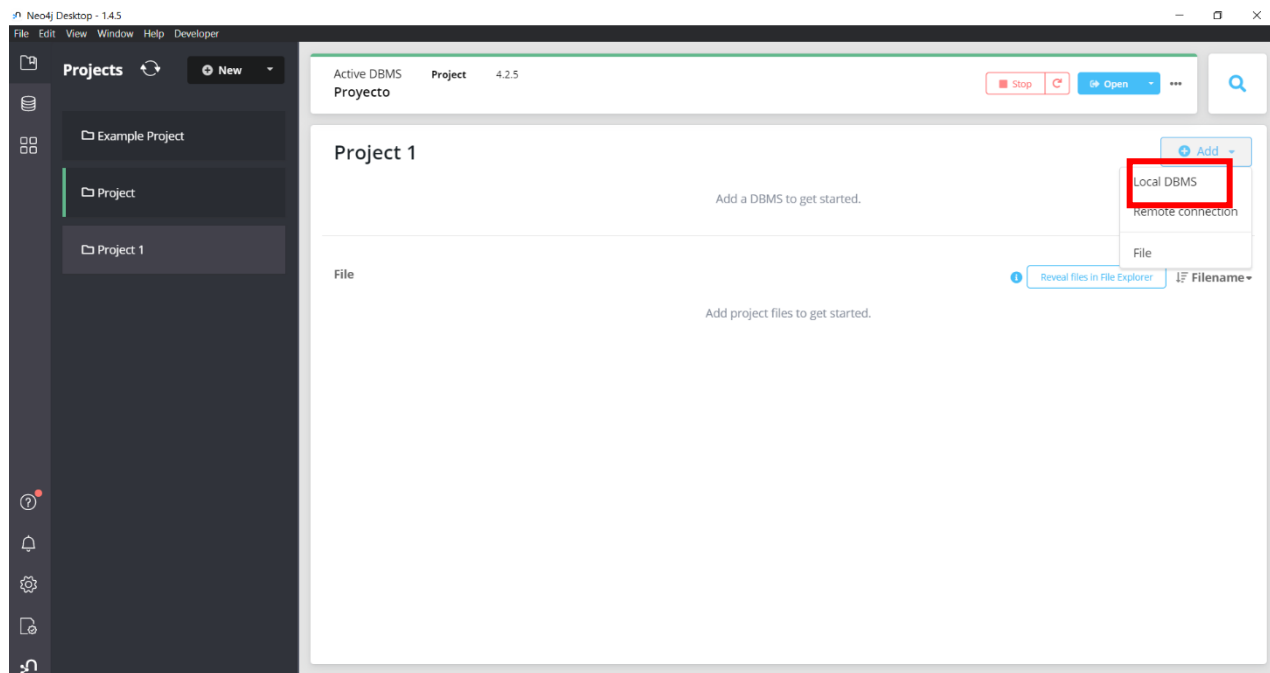
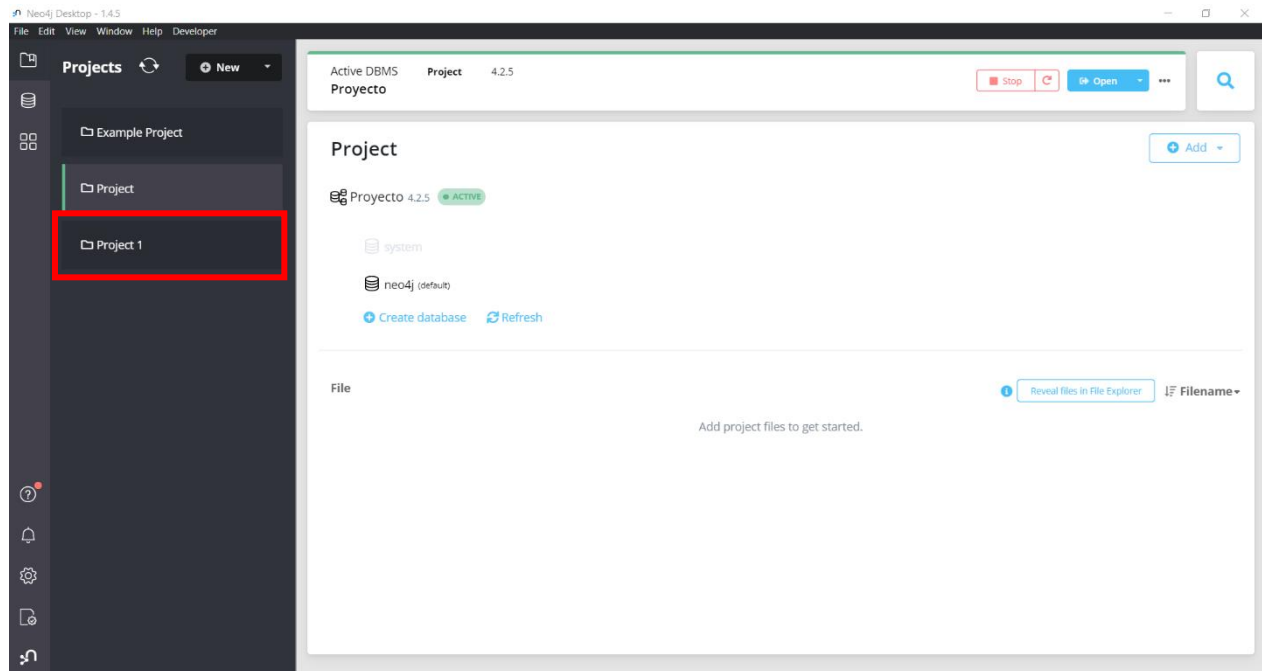
1. Se debe instalar Neo4j desktop, si aún no se tiene, usando el siguiente enlace:

<https://neo4j.com/download/>

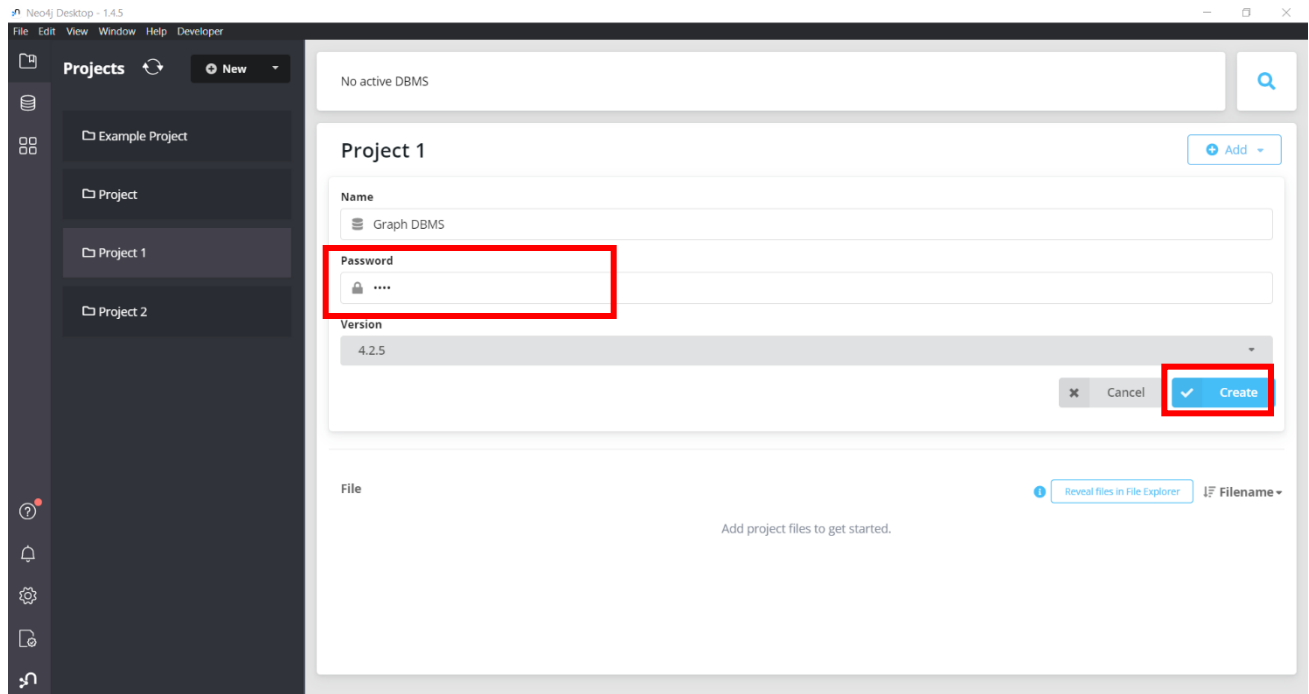
2. Una vez instala Neo4j se debe abrir y crear un nuevo proyecto, como se observa en la imagen.



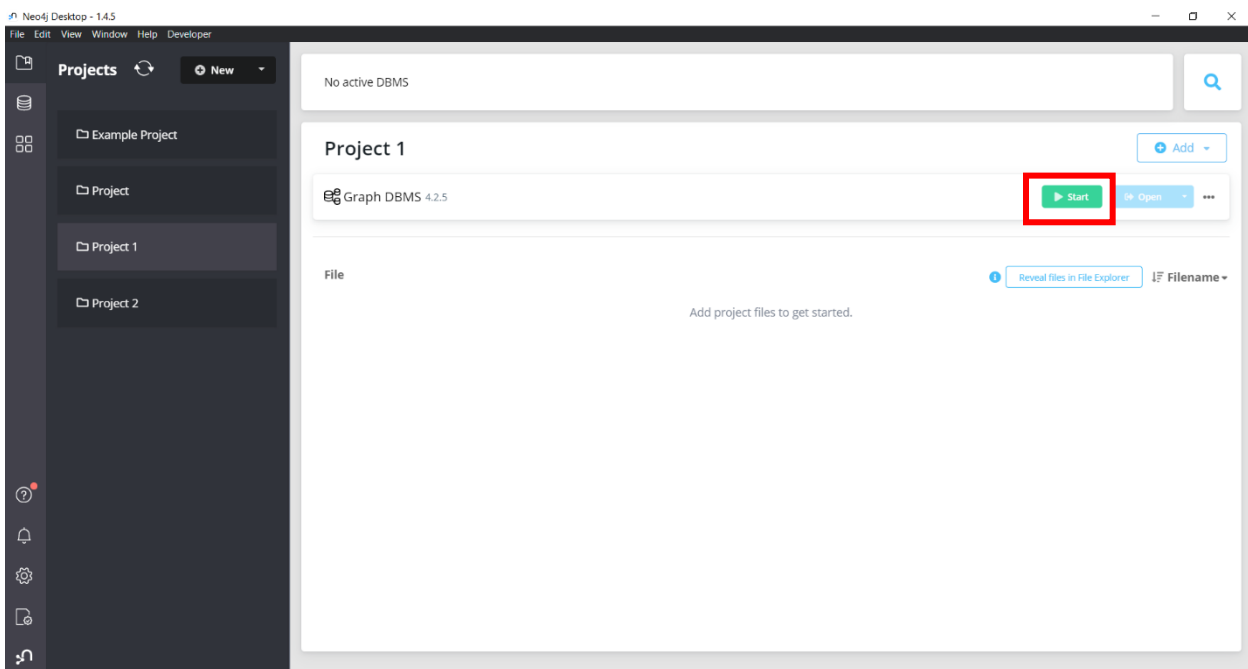
3. Se creará un nuevo proyecto en la interfaz. Se ingresa a él con un click y luego se crea una nueva base de datos en add -> local DBMS.

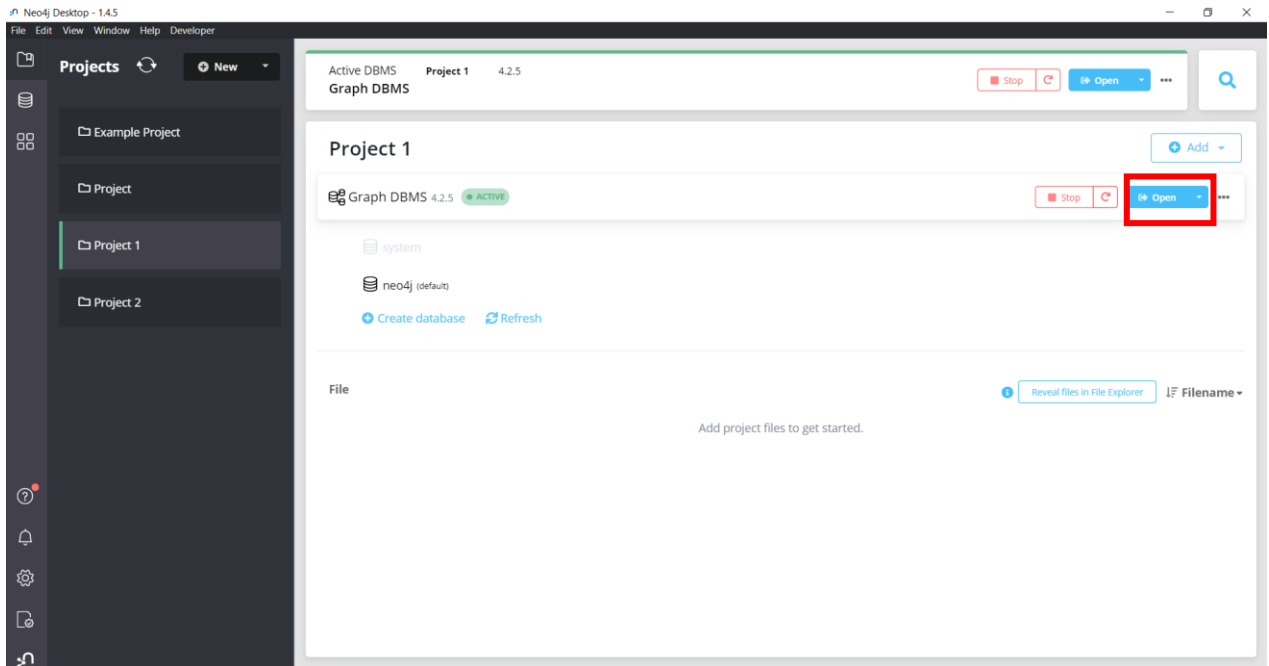


- Se ingresa la contraseña en la creación de la base de datos. La contraseña debe ser 1234. Se presiona al botón crear.

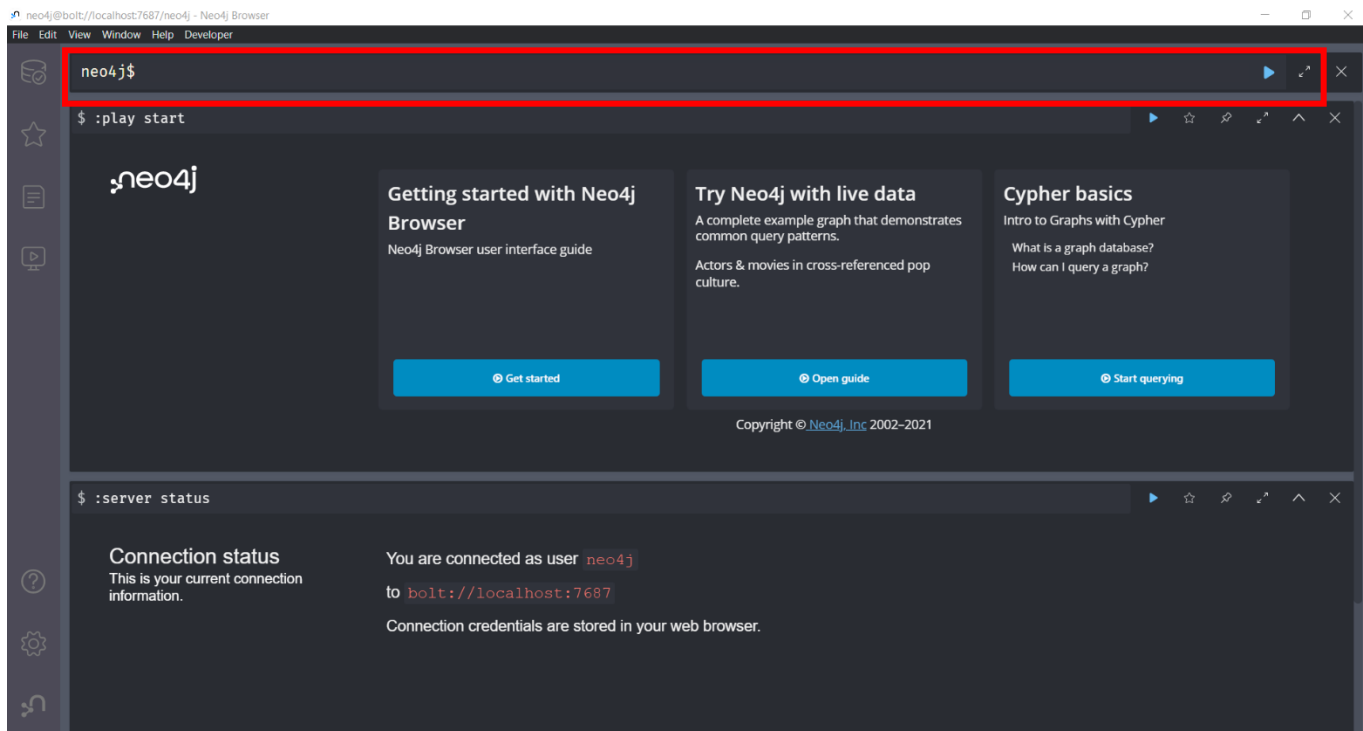


- Se inicia la base de datos y luego se abre.





6. En la terminal de la base de datos se debe pegar el siguiente código y correrlo con el botón de play.



LOAD CSV WITH HEADERS FROM

"https://raw.githubusercontent.com/JuanCarlosBajan/CSVs/main/Perros.csv" AS
row

MERGE (p:Perro{raza:row.Raza, ranking:row.Ranking})

MERGE (c:Comportamiento{comportamiento:row.Comportamiento1})

MERGE (c2:Comportamiento{comportamiento:row.Comportamiento2})

MERGE (c3:Comportamiento{comportamiento:row.Comportamiento3})

MERGE (e:Espacio{espacio:row.EspacioNecesario})

MERGE (tp:TipoPelo{tipoPelo:row.TipoPelo})

MERGE (t:Tamano{tamano:row.Tamano})

MERGE

(cc:ComplexionCorporal{complexionCorporal:row.ComplexionCorporal})

MERGE (af:ActividadFisica{actividadFisica:row.Actividadfisica})

MERGE (ev:ExpectativaDeVida{expectativaDeVida:row.ExpectativadeVida})

MERGE (h:Hocico{hocico:row.Hocico})

MERGE (o:Orejas{orejas:row.Orejas})

CREATE (p)-[:TipoPelo]->(tp)

CREATE (p)-[:Comportamiento]->(c)

CREATE (p)-[:Comportamiento]->(c2)

CREATE (p)-[:Comportamiento]->(c3)

CREATE (p)-[:Tamano]->(t)

CREATE (p)-[:Espacio]->(e)

CREATE (p)-[:ComplexionCorporal]->(cc)

CREATE (p)-[:ActividadFisica]->(af)

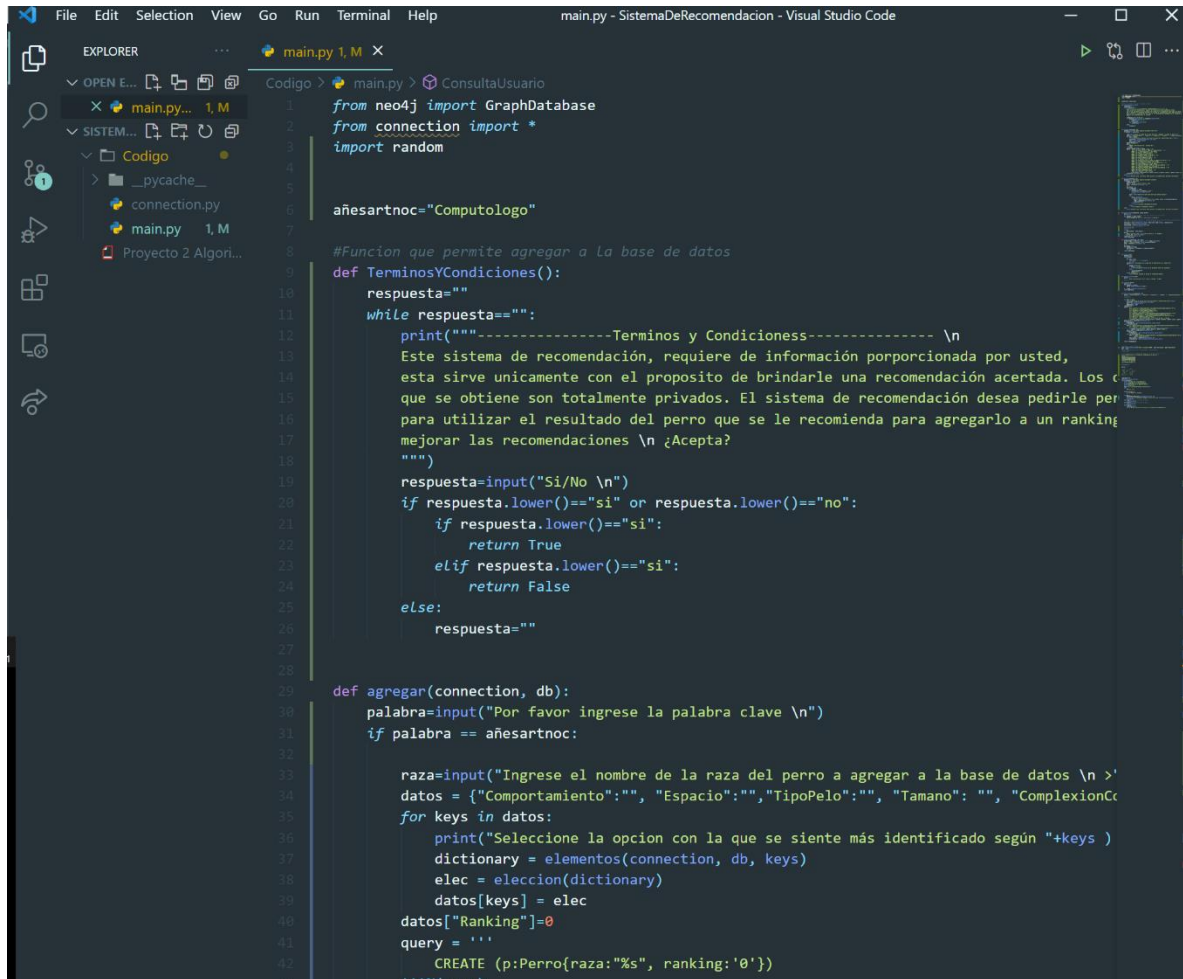
CREATE (p)-[:ExpectativaDeVida]->(ev)

CREATE (p)-[:Hocico]->(h)

CREATE (p)-[:Orejas]->(o)

RETURN p

7. Se debe correr el archivo main.py proporcionado en la carpeta subida del proyecto, sin apagar la base de datos.



```
File Edit Selection View Go Run Terminal Help
main.py - SistemaDeRecomendacion - Visual Studio Code

EXPLORER
main.py 1, M
Codigo
  __pycache__
  connection.py
  main.py 1, M
Proyecto 2 Algori...

main.py 1, M
1 from neo4j import GraphDatabase
2 from connection import *
3 import random
4
5
6 añesartnoc="Computologo"
7
8 #Funcion que permite agregar a la base de datos
9
10 def TerminosYCondiciones():
11     respuesta=""
12     while respuesta=="":
13         print("-----Terminos y Condicioness----- \n")
14         Este sistema de recomendación, requiere de información porporcionada por usted,
15         esta sirve unicamente con el proposito de brindarle una recomendación acertada. Los c
16         que se obtiene son totalmente privados. El sistema de recomendación desea pedirle per
17         para utilizar el resultado del perro que se le recomienda para agregarlo a un ranking
18         mejorar las recomendaciones \n ¿Acepta?
19         """
20         respuesta=input("Si/No \n")
21         if respuesta.lower()=="si" or respuesta.lower()=="no":
22             if respuesta.lower()=="si":
23                 return True
24             elif respuesta.lower()=="si":
25                 return False
26         else:
27             respuesta=""
28
29 def agregar(connection, db):
30     palabra=input("Por favor ingrese la palabra clave \n")
31     if palabra == añesartnoc:
32
33     raza=input("Ingrese el nombre de la raza del perro a agregar a la base de datos \n >'
34     datos = {"Comportamiento":"","Espacio":"","TipoPelo":"","Tamano":"","ComplexionCo
35     for keys in datos:
36         print("Seleccione la opcion con la que se siente más identificado según "+keys )
37         dictionary = elementos(connection, db, keys)
38         elec = eleccion(dictionary)
39         datos[keys] = elec
40     datos["Ranking"]=0
41     query = '''
42         CREATE (p:Perro{raza:"%s", ranking:'0'})
43         RETURN p
44     '''
```