Following this course:  
seems good: <https://learn.deeplearning.ai/courses/knowledge-graphs-rag/>

<https://chatgpt.com/c/673dd871-db08-8009-bb67-e5bacc458191?model=gpt-4o>

L5 NOTEBOOK IS IMPORTANT, ALWAYS CONSULT THAT BEFORE WORKING ON KNOWLEDGE GRAPH

KG: a database which has relationship between data as nodes and edges, each of them have labels and can have properties.  
  
  
Cypher query language:

Note\* converting tables/columns and csv rows into knowledge graph data modeling

Setting up:

Add apoc plugin in the db from neo4j desktop:

Then add this to config file located inside .Neo4jdesktop iside related data, <database\_path>/conf/neo4j.conf  
  
dbms.security.procedures.unrestricted=apoc.\*

dbms.security.procedures.allowlist=apoc.\*

and APOC plugin can do these:  
 **Data import/export** helpers (e.g., load JSON or CSV files).

 **Graph algorithms** for advanced computations.

 **Metadata querying** to analyze your database schema (like apoc.meta.data()).

 Utility functions for string handling, date calculations, etc.

\*\*neo4j can be used to store vectors, like it can be used as a vector store.

Pross, all data in one place, emddings attached to some nodes…MAYBE

Cons, increases complexity, …..

Need to see an example how vectors enhance, KG, like methodlogly…pipeline, maybe I can improve it.

\*\*\*best pros, you can find relationship between nodes and create relationship on that spot.

\*use case, rag with kg: first find the relavnt chunk with vector search then, use that chunk id to backtrace the relationship it has, or perform complex query….brillient, as those relationship can be missed from vector search…. my idea.

With Neo4j v5, APOC has been split into Core and Extended editions. If you are using procedures from the Core edition, nothing has changed in the installation process. However, if you are using any procedures from the Extended edition, you need to manually download and copy the Extended edition from the GitHub release page.

CALL apoc.meta.graph()

\*so vector support came in 5.13, able to create vector indexes,   
but some commands are not working

Vector indexes only supported in enterprise

SUCSSESSSSSSSSSSSSSSSSSS…………. 5.16 VERSION IS IDEAL FOR VECTORS

So so here is how vectors work  
you create an index and associate it with a node and specially mention where the embeddings resides

Next if that embedding property doesn’t exist then you create that property using db.create.setNodeVectorProperty, and then create mebddings for that vector

For genai vector encode, you need another plugin

dbms.security.procedures.unrestricted=genai.\*…..dont need this

I needed gen ai jar, for vector function to work  
so you can find it here, it cam as default but finding it was difficult  
C:\Users\Black Mamba\AppData\Local\Neo4j\Relate\Cache\dbmss….\plugin-resources

Then copy it and past it in other folder of plugin, you can get its location from neo4j desktop app

**Scaling**: If the Chunk data grows in size, consider how your graph will scale (e.g., sharding or partitioning data across multiple databases, optimizing for write-heavy or read-heavy operations, etc.)????????????????????????????????????????????????????????

Relationships:

Path: path is defined as a window of path between nodes, length of the path defined by number of relationships between nodes

Or

Matched patterns of nodes and relationships in a graph are called **\*\*paths\*\***

- The length of a path is equal to the number of relationships in the path

- Paths can be captured as variables and used elsewhere in queries

Variable length path: specify a range of relationship to match in a path

DIRECTION OF RELATIONSHIP MATTERS ONLY IN SMEMANTIC MEANING OF THINGS

Full text index can be created as well for string matching search.

L6

Can be used to create new data