

In [127]:

```
import pandas as pd
from neo4j import GraphDatabase
import credentials as c
parent = pd.read_csv("DX_Chapters.csv")
children = pd.read_csv("CCSR_Categories.csv")
```

## Import data into neo4j graph database

In [128]:

```
uri = "bolt:http://127.0.0.1:7687"
driver = GraphDatabase.driver(uri, auth = (c.username, c.password))
session = driver.session()
#reset graph database - delete all existing nodes and relationships
session.run("MATCH (n) DETACH DELETE n")
```

Out[128]:

<neo4j.work.result.Result at 0x11c90a7f0>

In [129]:

```
#combine parent and children dataframes

#1.Create new abbreviation column that takes first three letters of disease code
children["Abbreviation"] = children['Code'].apply(lambda x: x[0:3])

#2.Create dictionary from chapter code to chapter description
parent_dictionary = dict(zip(parent.Abbreviation, parent.Chapter))

#3.Create descriptions for the classifications with no parent
parent_dictionary["NoD"] = "Default"
parent_dictionary["Inv"] = "Invalid"

#4. Create new chapter description column that obtains the description by the three letter code from the dictionary
children["Chapter Description"] = children['Abbreviation'].apply(lambda x: parent_dictionary[x[0:3]])

#5. Assign the mutated dataframe to a new dataframe
df = children
df
```

Out[129]:

|     | Code   | Description                                       | Abbreviation | Chapter Description                               |
|-----|--------|---|--------------|---|
| 0   | BLD001 | Nutritional anemia                                | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 1   | BLD002 | Hemolytic anemia                                  | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 2   | BLD003 | Aplastic anemia                                   | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 3   | BLD004 | Acute posthemorrhagic anemia                      | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 4   | BLD005 | Sickle cell trait/anemia                          | BLD          | Diseases of the Blood and Blood Forming Organs... |
| ... | ...    | ...   | ...          | ...   |
| 539 | SYM017 | Abnormal findings without diagnosis               | SYM          | Symptoms, Signs and Abnormal Clinical and Labo... |
| 540 | XXX000 | Code is unacceptable as a principal diagnosis ... | XXX          | Unacceptable principal diagnosis (inpatient da... |
| 541 | XXX111 | Code is unacceptable as a first-listed diagnos... | XXX          | Unacceptable principal diagnosis (inpatient da... |

| 542 | Code<br>NoDX1 | Only used for the default CCSR when the<br>princi... | Description<br>Abbreviation<br>NoD | Chapter Description<br>Default |
|-----|---------------|--|------------------------------------|--------------------------------|
| 543 | InvIDX        | Used to indicate the diagnosis was invalid and...    | Inv                                | Invalid                        |

544 rows x 4 columns

In [130]:

```
bulk_query = '''
UNWIND $rows as row
WITH row
MERGE (chapter:Chapter{ description: row["Chapter Description"], name: row.Abbreviation}
)
MERGE (disease:Disease{ description:row.Description , name:row.Code  } )
MERGE (disease)-[:BELONGS_TO]->(chapter)

'''
```

In [131]:

```
#split into n dataframes
n = 10
dfList = [df[i:i+n] for i in range(0, df.shape[0], n)]
```

In [132]:

```
for rows in dfList:
    session.run(bulk_query, parameters = {'rows': rows.to_dict('records')})
```

Retrieve data from neo4j graph database

In [133]:

```
#create empty output dataframe to store retrieved data from neo4j
column_names = ["Code", "Description", "Abbreviation", "Chapter Description"]

df_output = pd.DataFrame(columns = column_names)
```

In [134]:

```
for i in range(len(dfList)):
    #1. Create query that retrieves 10 items at a time
    skip = 10*i
    retrieve_query = f'MATCH p=()-->() RETURN p SKIP {skip} LIMIT {n}'

    #2. Run query to get all relationships
    paths = list(session.run(retrieve_query))
    for node in paths:
        path = node.values()[0]

        #3. Get properties of starting node of each relationship
        child = list(path.start_node.values())

        #4. Get properties of ending node of each relationship
        parent = list(path.end_node.values())

        #5. Concatenate two lists to obtain the full row to be entered into dataframe. In
        this row, code is matched to its description, abbreviation, and chapter
        relationship = child + parent

        #6. Add relationship to end of dataframe
        df_output.loc[len(df_output)] = relationship
df_output
```

Out[134]:

| Code  | Description | Abbreviation | Chapter Description                     |
|---|-------------|--------------|---|
| Other specified and unspecified hematologic |             |              | Diseases of the Blood and Blood Forming |

|     | Code   | Description                                       | Abbreviation | Chapter Description                               |
|-----|--------|---|--------------|---|
| 1   | BLD009 | Postprocedural or postoperative complications ... | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 2   | BLD008 | Immunity disorders                                | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 3   | BLD007 | Diseases of white blood cells                     | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 4   | BLD006 | Coagulation and hemorrhagic disorders             | BLD          | Diseases of the Blood and Blood Forming Organs... |
| ... | ...    | ...   | ...          | ...   |
| 539 | SYM001 | Syncope   | SYM          | Symptoms, Signs and Abnormal Clinical and Labo... |
| 540 | XXX111 | Code is unacceptable as a first-listed diagnos... | XXX          | Unacceptable principal diagnosis (inpatient da... |
| 541 | XXX000 | Code is unacceptable as a principal diagnosis ... | XXX          | Unacceptable principal diagnosis (inpatient da... |
| 542 | NoDX1  | Only used for the default CCSR when the princi... | NoD          | Default   |
| 543 | InvIDX | Used to indicate the diagnosis was invalid and... | Inv          | Invalid   |

544 rows x 4 columns

In [135]:

```
df_output
```

Out[135]:

|     | Code   | Description                                       | Abbreviation | Chapter Description                               |
|-----|--------|---|--------------|---|
| 0   | BLD010 | Other specified and unspecified hematologic co... | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 1   | BLD009 | Postprocedural or postoperative complications ... | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 2   | BLD008 | Immunity disorders                                | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 3   | BLD007 | Diseases of white blood cells                     | BLD          | Diseases of the Blood and Blood Forming Organs... |
| 4   | BLD006 | Coagulation and hemorrhagic disorders             | BLD          | Diseases of the Blood and Blood Forming Organs... |
| ... | ...    | ...   | ...          | ...   |
| 539 | SYM001 | Syncope   | SYM          | Symptoms, Signs and Abnormal Clinical and Labo... |
| 540 | XXX111 | Code is unacceptable as a first-listed diagnos... | XXX          | Unacceptable principal diagnosis (inpatient da... |
| 541 | XXX000 | Code is unacceptable as a principal diagnosis ... | XXX          | Unacceptable principal diagnosis (inpatient da... |
| 542 | NoDX1  | Only used for the default CCSR when the princi... | NoD          | Default   |
| 543 | InvIDX | Used to indicate the diagnosis was invalid and... | Inv          | Invalid   |

544 rows x 4 columns

In [136]:

```
df_output.to_csv("retrieved_data.csv")
```