In [60]:

```
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")
```

In [61]:

```
uri = "bolt:http//127.0.0.1:7687"
driver = GraphDatabase.driver(uri, auth = (c.username, c.password))
session = driver.session()
```

In [62]:

```
#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 lette
r code from the dictionary
children["Chapter Description"] = children['Abbreviation'].apply(lambda x: parent_diction
ary[x[0:3]])

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

Out[62]:

	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	NoDX1	Only used for the default CCSR when the princi...	NoD	Default
543	InvlDX	Used to indicate the diagnosis was invalid and...	Inv	Invalid

544 rows × 4 columns

In [63]:

In [63]:

```
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 [64]:

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

In [65]:

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