***Code***

from nltk import word\_tokenize,pos\_tag

g = Graph("neo4j+s://7826f71c.databases.neo4j.io", auth=("neo4j","dwmcUg2xRr2Y48QnbE30Ffw0sTXXvqHy1GhkBn9D6jc"))

sentence1="Neha teaches sidra"

sentence2="Farukh teaches nadeem"

sentence3="Aneela is student of farukh"

sentence4="Aneela Knows Sidra"

sentence5="Nadeem is favorite\_teacher of neha"

tokens1=word\_tokenize(sentence1)

tokens2=word\_tokenize(sentence2)

tokens3=word\_tokenize(sentence3)

tokens4=word\_tokenize(sentence4)

tokens5=word\_tokenize(sentence5)

posTaggedList1=pos\_tag(tokens1)

posTaggedList2=pos\_tag(tokens2)

posTaggedList3=pos\_tag(tokens3)

posTaggedList4=pos\_tag(tokens4)

posTaggedList5=pos\_tag(tokens5)

print("Sentence:",sentence1)

print("Tokens:",tokens1)

print("Parts of speech:",posTaggedList1)

print("Sentence:",sentence2)

print("Tokens:",tokens2)

print("Parts of speech:",posTaggedList2)

a1=Node("Person",name=tokens1[0])

b1=Node("Person",name=tokens1[2])

Teach1=Relationship.type(tokens1[1])

g.merge(Teach1(a1,b1),"Person","name")

a2=Node("Person",name=tokens2[0])

b2=Node("Person",name=tokens2[2])

Teach2=Relationship.type(tokens2[1])

g.merge(Teach2(a2,b2),"Person","name")

a3=Node("Person",name=tokens3[0])

Student=Relationship.type(tokens3[2])

g.merge(Student(a2,a3),"Person","name")

Knows=Relationship.type(tokens4[1])

g.merge(Knows(a3,b1),"Person","name")

Favourite=Relationship.type(tokens5[2])

g.merge(Favourite(b2,a1),"Person","name")

**Output**

