Neo4J Analysis:

1. Load the nodes - candidates, contributors and committees

```
LOAD CSV FROM "file:///Users/alfredarsenault/documents/W205_project/candidate.csv" AS
line
CREATE (:Candidate { ID: line [0], Name: line[1], Office: line[2], CommitteeID: line[3] } )

CREATE INDEX ON  :Candidate(ID)

LOAD CSV FROM "file:///Users/alfredarsenault/documents/W205_project/
individual_contributions.csv" AS line
CREATE (:Contributor {Name: line[3], Employer: line[4] })

CREATE INDEX ON :Contributor(Name)

LOAD CSV FROM "file:///Users/alfredarsenault/documents/W205_project/committee.csv" AS
line
CREATE (:Committee {ID: line[0], Name: line[1], Office: line[2], CandidateID: line[3] } )

CREATE INDEX ON :Committee(ID)
```

2. Load the edges - the two contributions files (from committees and individuals)

```
LOAD CSV FROM "file:///Users/alfredarsenault/documents/W205_project/pass_through.csv" AS
line
MATCH (can: Candidate {ID: line[3]})
MATCH(com: Committee {ID: line[0]})
CREATE UNIQUE (u)



LOAD CSV FROM "file:///Users/alfredarsenault/documents/W205_project/pass_through.csv" AS
line MATCH (can: Candidate {ID: line[3]}) MATCH(com: Committee {ID: line[0]}) CREATE
UNIQUE (can) -[:DONATED {w: line[0]}] -> (com)
```

3. Queries:

```
MATCH(hillary:Candidate)
WHERE hillary.Name STARTS WITH "CLINT"
RETURN hillary
```