**Load Company names with official names**

|  |
| --- |
| load csv with headers from "file:///allcompanies\_neo.csv" as row  MERGE (c:Company{Name:row.itemLabel,Official\_Name : row.Official\_Name}) |

**Load Products Names and Relationships**

|  |
| --- |
| load csv with headers from "file:///products\_neo.csv" as row  MERGE (i1:Company {Name: row.itemLabel})  MERGE (i2:Product {Name: row.productLabel})  MERGE (i1)-[r:PRODUCT\_OF]->(i2) |

**Load Subsidiary names and relationships**

|  |
| --- |
| load csv with headers from "file:///sub\_neo.csv" as row  MERGE (i1:Company {Name: row.itemLabel})  MERGE (i2:Company {Name: row.subsidiaryLabel})  MERGE (i2)-[r:SUBSIDIARY\_OF{startDate:row.startDate,endDate:row.endDate}]->(i1) |

**Load Parent Companies and relationships**

|  |
| --- |
| load csv with headers from "file:///parent\_neo.csv" as row  MERGE (i1:Company {Name: row.itemLabel})  MERGE (i2:Company {Name: row.parentLabel})  MERGE (i2)-[r:PARENT\_OF{startDate:row.startDate,endDate:row.endDate}]->(i1) |

**Load Replaced companies and relationships**

|  |
| --- |
| load csv with headers from "file:///replacedby\_neo.csv" as row  MERGE (i1:Company {Name: row.itemLabel})  MERGE (i2:Company {Name: row.replacedByLabel})  MERGE (i1)-[r:REPLACED\_BY]->(i2) |

**Load Company Description**

|  |
| --- |
| load csv with headers from "file:///description\_neo.csv" as row  MATCH (c:Company)  WHERE c.Name=row.itemLabel  SET c.Description = row.Description |

**Load Industry and relationships**

|  |
| --- |
| load csv with headers from "file:///ind\_connection.csv" as row  MERGE (i1:Industry {name: row.industry1})  MERGE (i2:Industry {name: row.ind2})  MERGE (i2)-[r:PART\_OF]->(i1) |

**Load Company-Industry Relations**

|  |
| --- |
| load csv with headers from "file:///ind\_neo\_comp.csv" as row  MATCH (c:Company)  WHERE c.Name=row.itemLabel  MERGE (i:Industry {name: row.industry})  MERGE (c)-[r:INDUSTRY]->(i) |

**Query to get all subsidiaries of company whose name starts with “Wal”**

|  |
| --- |
| MATCH (C1:Company)-[r:SUBSIDIARY\_OF]->(C2:Company)  WHERE C2.Name starts with "Wal"  RETURN C1,C2 |

**Query to check whether a subsidiary relationship exist between “Waymo” and “Google**

|  |
| --- |
| MATCH p= (C1:Company)-[r:SUBSIDIARY\_OF]->(C2:Company)  WHERE C1.Name = "Waymo" and C2.Name="Google"  RETURN p |

**Query to get top 10 Fortune 500 Companies**

|  |
| --- |
| MATCH (n:Company)  where n.Fortune500\_Rank <10  RETURN n.Official\_Name as Official\_Name, n.Fortune500\_Rank  ORDER BY n.Fortune500\_Rank |

**Query to get Parent of the company “AOL”**

|  |
| --- |
| MATCH ((c:Company)-[r:PARENT\_OF]->(c2:Company{Name:'AOL'}))  RETURN c2.Name as Child\_Name,c.Name as Parent\_Name,r.startDate as Start\_Date,r.endDate as End\_Date |

**Query to get all companies in the industry “Retail”**

|  |
| --- |
| MATCH (c:Company)-[r:INDUSTRY]->(i:Industry{Name:'Retail'})  RETURN c,i |

**Query to get all companies replaced by other companies**

|  |
| --- |
| MATCH (c:Company)-[r:REPLACED\_BY]->(c2:Company)  RETURN c.Name as Company\_Name ,c2.Name as Replaced\_By |

**Query to check whether a relationship exists between "Johnson & Johnson" and  “Aveeno"**

|  |
| --- |
| MATCH p= (C1:Company)-[]->(C2:Company)  WHERE C1.Name = "Johnson & Johnson" and C2.Name="Aveeno"  RETURN p |