**SAMPLE INVESTIGATIVE QUERIES**

* Query for returning the number of companies that have been dissolved

MATCH (c:Company)

WHERE c.status="Dissolved"

RETURN COUNT(c)

* Query for returning the cases and number of companies found for each case in the dataset

MATCH (c:Cases)-[]->(a:Company)

RETURN c.name, count(a.name) as CpnyNbr

* Query for returning the total number of executives in the dataset

MATCH (a:Company)-[]->(b:Executive)

RETURN COUNT(b)

* Query for returning the number of executives from a particular country

MATCH (a:Company)-[]->(b:Executive)

RETURN b.nationality,count(b)

* Query for returning the number of executive appointments for UK directors

MATCH (e:Executive)

WHERE e.appointments <>"NA"AND e.nationality="UNITED KINGDOM"

RETURN e.name,e.appointments

* Query for returning the number of executive appointments for non-UK directors

MATCH (e:Executive)

WHERE e.appointments <>"NA"AND e.nationality <> "NA" AND e.nationality <> "UNITED KINGDOM"

RETURN e.name,e.appointments,e.nationality

* Query for returning the period of operation for companies along with their names

MATCH (c:Company)

WHERE c.dissolution <> "NA"

WITH c,apoc.date.format(apoc.date.parse(c.dissolution, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Dislv,

apoc.date.format(apoc.date.parse(c.incorporation, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Incorp

WITH c, [x IN split(Incorp, "-") | toInteger(x)] AS Inc, [x1 IN split(Dislv, "-") | toInteger(x1) ] AS Dis

WITH c,duration.inMonths (date({month: Inc[0], day: Inc[1], year: Inc[2]}), date({month: Dis[0], day: Dis[1], year: Dis[2]})) as Dur

RETURN c.name,toInteger(apoc.text.format("%02d", [Dur.months])) AS value LIMIT 10

* Query for returning the period of operation for companies incorporated before 2007 along with their names

MATCH (c:Company)

WHERE c.dissolution <> "NA"

WITH c,apoc.date.format(apoc.date.parse(c.dissolution, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Dislv,

apoc.date.format(apoc.date.parse(c.incorporation, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Incorp

WITH c, [x IN split(Incorp, "-") | toInteger(x)] AS Inc, [x1 IN split(Dislv, "-") | toInteger(x1) ] AS Dis

WITH c,duration.inMonths (date({month: Inc[0], day: Inc[1], year: Inc[2]}), date({month: Dis[0], day: Dis[1], year: Dis[2]})) as Dur

WHERE c.incorporation<"2007"

RETURN c.name,toInteger(apoc.text.format("%02d", [Dur.months])) AS value

* Query for returning the period of operation for companies incorporated in or after 2007 but in or before 2015

MATCH (c:Company)

WHERE c.dissolution <> "NA"

WITH c,apoc.date.format(apoc.date.parse(c.dissolution, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Dislv,

apoc.date.format(apoc.date.parse(c.incorporation, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Incorp

WITH c, [x IN split(Incorp, "-") | toInteger(x)] AS Inc, [x1 IN split(Dislv, "-") | toInteger(x1) ] AS Dis

WITH c,duration.inMonths (date({month: Inc[0], day: Inc[1], year: Inc[2]}), date({month: Dis[0], day: Dis[1], year: Dis[2]})) as Dur

WHERE c.incorporation>="2007" AND c.incorporation<="2015"

RETURN c.name,c.incorporation,toInteger(apoc.text.format("%02d", [Dur.months])) AS value

ORDER BY value DESC

* Query for returning the period of operation for companies dissolved in or after 2015

MATCH (c:Company)

WHERE c.dissolution <> "NA"

WITH c,apoc.date.format(apoc.date.parse(c.dissolution, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Dislv,

apoc.date.format(apoc.date.parse(c.incorporation, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Incorp

WITH c, [x IN split(Incorp, "-") | toInteger(x)] AS Inc, [x1 IN split(Dislv, "-") | toInteger(x1) ] AS Dis

WITH c,duration.inMonths (date({month: Inc[0], day: Inc[1], year: Inc[2]}), date({month: Dis[0], day: Dis[1], year: Dis[2]})) as Dur

WHERE c.dissolution>="2015"

RETURN c.name,c.dissolution,toInteger(apoc.text.format("%02d", [Dur.months])) AS value

ORDER BY value DESC

* Query for returning the period of operation for companies dissolved in or after 2007 and in or before 2015

MATCH (c:Company)

WHERE c.dissolution <> "NA"

WITH c,apoc.date.format(apoc.date.parse(c.dissolution, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Dislv,

apoc.date.format(apoc.date.parse(c.incorporation, 'ms', 'yyyy-MM-dd'), 'ms', 'MM-dd-yyyy') as Incorp

WITH c, [x IN split(Incorp, "-") | toInteger(x)] AS Inc, [x1 IN split(Dislv, "-") | toInteger(x1) ] AS Dis

WITH c,duration.inMonths (date({month: Inc[0], day: Inc[1], year: Inc[2]}), date({month: Dis[0], day: Dis[1], year: Dis[2]})) as Dur

WHERE c.dissolution>="2007" AND c.dissolution<="2015"

RETURN c.name,c.dissolution,toInteger(apoc.text.format("%02d", [Dur.months])) AS value

ORDER BY value DESC

* Query for returning the name and number of companies with same postal codes

MATCH (d:Company)-[]->(e:CompanyAddress)

RETURN COLLECT(d.name),e.postalcodes,count(\*) AS Number

ORDER BY Number DESC

* Query for returning companies where “ultimate beneficial owner” information is available

MATCH (c:Company)-[:OWNER]->(u:UltimateBeneficialOwner)

RETURN c.name,u.name

* Query for returning the name of companies where there is a previous name but no change in the registered address

MATCH (c:Company)

Where c.changeinregisteredaddress="No" AND c.companypreviousnames="Yes"

Return c.name,c.number

* Query for returning the executives of a particular company associated with previous companies when previous address and previous company names exists

MATCH (c:Company)-[:PREVIOUSADDRESS]->(a:Address)-[]->(p:PreviousCompany)

WHERE c.number = "4232247"

WITH c, p

MATCH (c)-[:EXECUTIVE]->(e:Executive)

WHERE e.tenureend <> "NA" AND

e.tenurestart >= p.StartofTenure AND e.tenureend <= p.EndofTenure

MERGE (e)-[:PREVEXEC]->(p)

RETURN c,e,p

* Query for returning the executives associated with previous companies when previous address and previous company names exists

MATCH (c:Company)-[:PREVIOUSADDRESS]->(a:Address)-[]->(p:PreviousCompany)

WHERE c.companypreviousnames <> "NA"

WITH c, p

MATCH (c)-[:EXECUTIVE]->(e:Executive)

WHERE e.tenureend <> "NA" AND

e.tenurestart >= p.StartofTenure AND e.tenureend <= p.EndofTenure

MERGE (e)-[:PREVEXEC]->(p)

RETURN c, e, p