**Test Queries:**

Below are the Cypher queries which can be used to test the project. Run these queries in the Neo4j browser.

All the authors have a unique ORCID to uniquely identify that author. We can further refine the search using other properties of Author and Publication nodes while querying. Few examples are given below.

1. **Description:** Query to return 400 ‘authored’ relationships

**Query:** *MATCH (a)-[:authored]->(b) RETURN a, b LIMIT 400*

**Screenshot:**

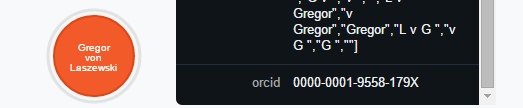


1. **Description:** Query to return the Author node having name ‘Gregor von Laszewski’

**Query:** *MATCH (a:Author) WHERE has(a.othernames) and any(othername in a.othernames WHERE othername = "Gregor von Laszewski") RETURN a*

**Screenshot:** We can see that there is a single node having name ‘Gregor von Laszewski’





1. **Description:** Query to return all the nodes having name ‘Gregor’

**Query:** *MATCH (a:Author) WHERE has(a.othernames) and any(othername in a.othernames WHERE othername = "Gregor") RETURN a*

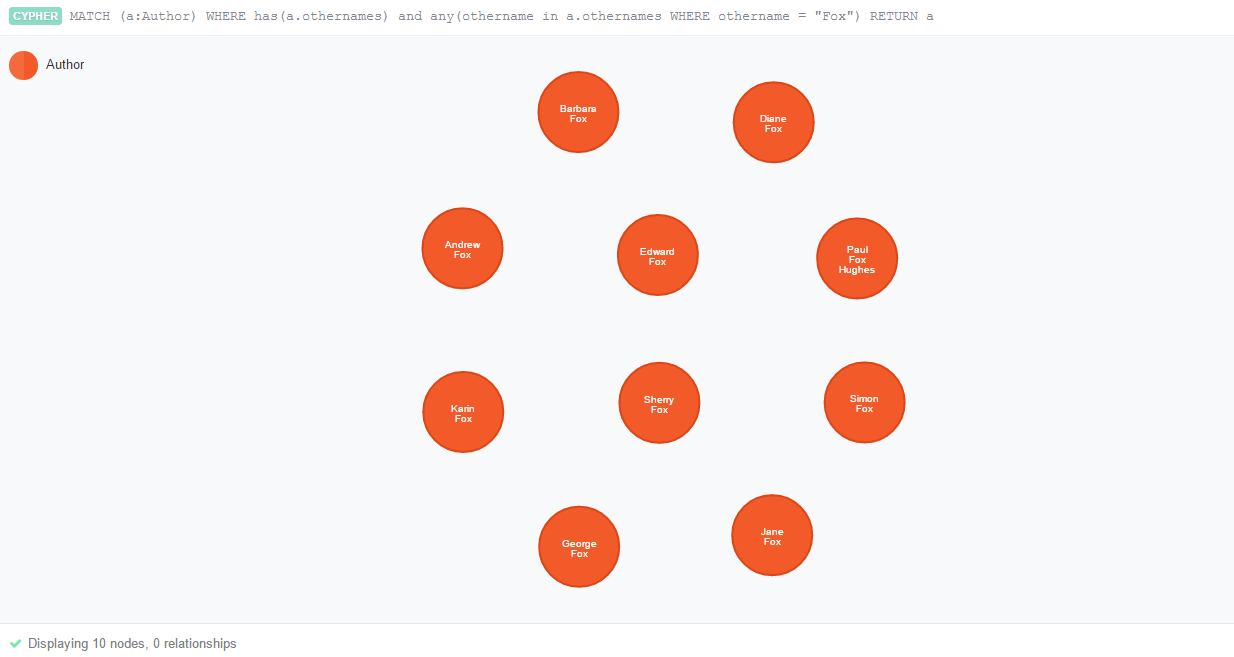
**Screenshot:** We can see that there are 6 nodes having name ‘Gregor’



1. **Description:** Query to return all the nodes having name ‘Fox’

**Query:** *MATCH (a:Author) WHERE has(a.othernames) and any(othername in a.othernames WHERE othername = "Fox") RETURN a*

**Screenshot:** We can see that there are 10 nodes having name ‘Fox’



1. **Description:** Query to return all the nodes having name ‘s fox’, ignoring the case

**Query:** *MATCH (a:Author) WHERE has(a.othernames) and any(othername in a.othernames WHERE othername =~ "(?i)s fox") RETURN a*

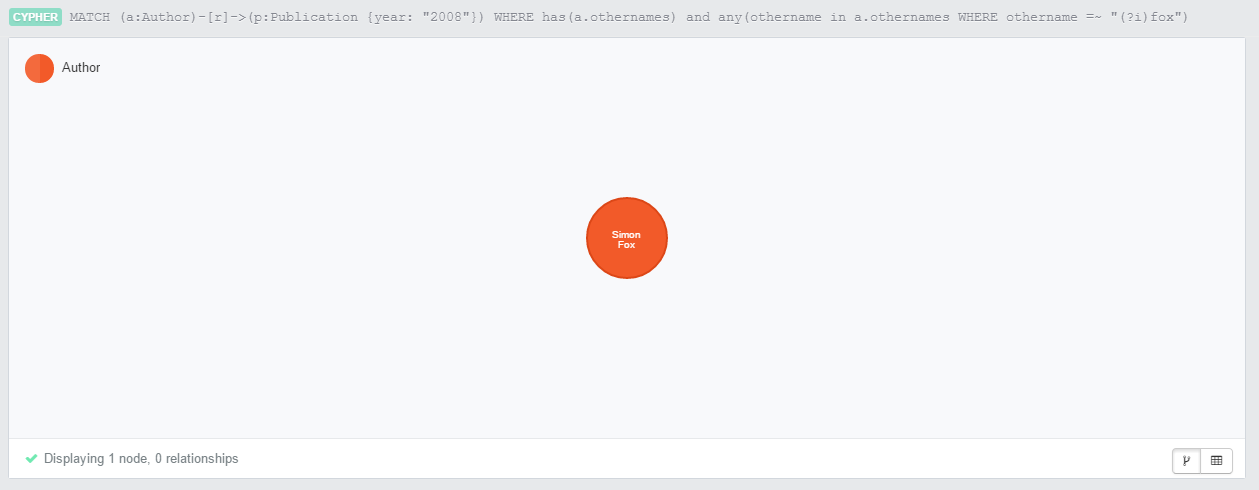
**Screenshot:** We can see that there are 2 nodes having name ‘s fox’



1. **Description:** Query to return all the nodes having name ‘fox’ ignoring case and who authored a publication in the year 2008

**Query:** *MATCH (a:Author)-[r]->(p:Publication {year: "2008"}) WHERE has(a.othernames) and any(othername in a.othernames WHERE othername =~ "(?i)fox") RETURN a*

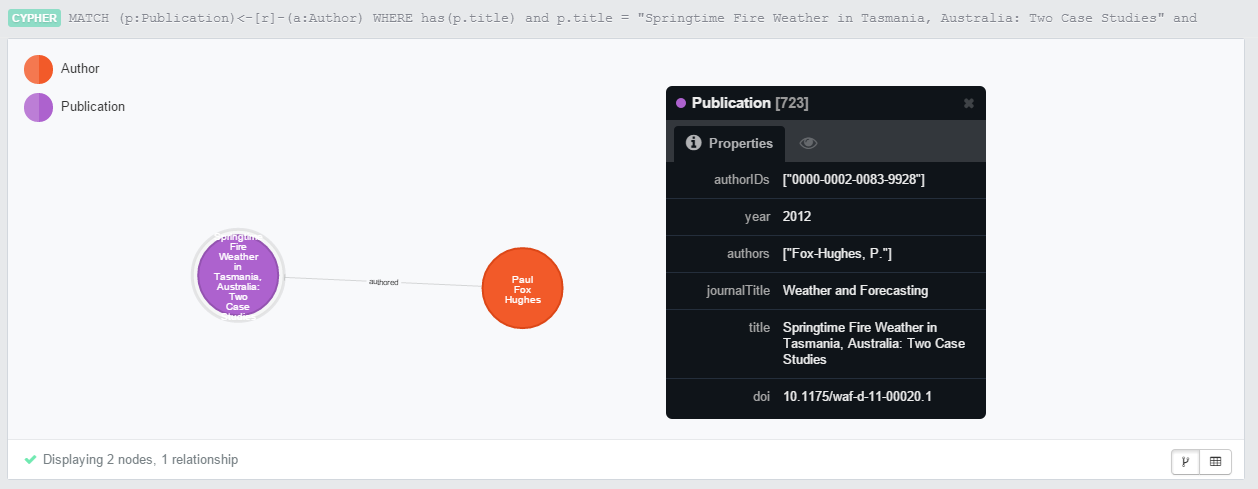
**Screenshot:** We can see that out of 10 nodes having the name ‘fox’, only one author has a publication in the year 2008

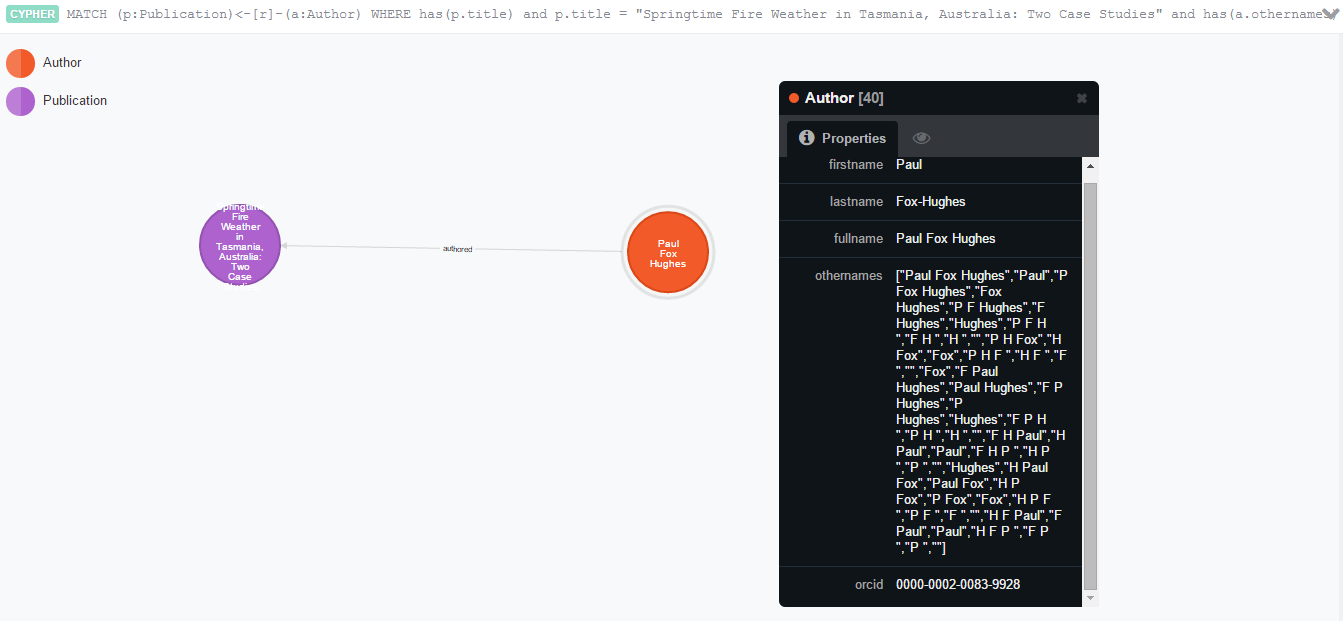


1. **Description:** Query to find the author named ‘fox’ ignoring case who authored the publication ‘Springtime Fire Weather in Tasmania, Australia: Two Case Studies’

**Query:** *MATCH (p:Publication)<-[r]-(a:Author) WHERE has(p.title) and p.title = "Springtime Fire Weather in Tasmania, Australia: Two Case Studies" and has(a.othernames) and any(othername in a.othernames WHERE othername =~ "(?i)fox") RETURN a, p*

**Screenshot:** We can see that out of 10 authors having the name ‘fox’, only one author has authored the publication ‘Springtime Fire Weather in Tasmania, Australia: Two Case Studies’. Thus we are able to identify the author uniquely as every author has unique orcid.





1. **Description:** Query to find the author named ‘Gregor’ who has authored a publication which is also authored by the author named ‘Fox’

**Query:** *MATCH (a:Author)-[r]->(p:Publication)<-[s]-(b:Author) WHERE has(a.othernames) and any(othername in a.othernames WHERE othername = "Gregor") and has(b.othernames) and any(othername in b.othernames WHERE othername = "Fox") RETURN a*

**Screenshot:** We can see that out of the 6 authors named ‘Gregor’, there is only one author with orcid ‘0000-0001-9558-179X’ who has co-authored with an author named ‘Fox’

