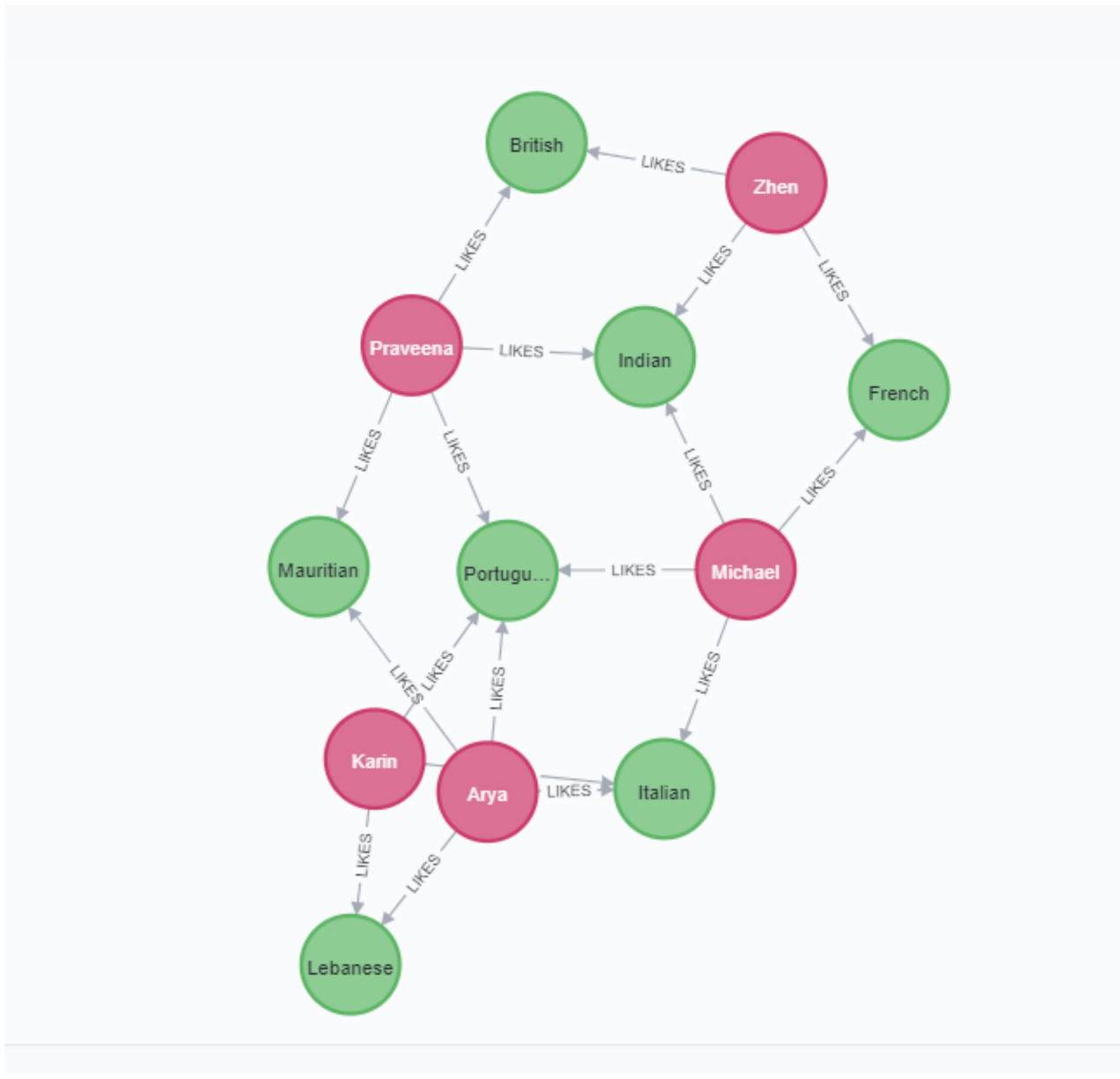
SIMILITUD DE COSENO

NOMBRE: DAVID ISRAEL LEON

Podemos usar el algoritmo Cosine Similarity para calcular la similitud entre dos cosas. Luego, podríamos usar la similitud calculada como parte de una consulta de recomendación. Por ejemplo, para obtener recomendaciones de películas basadas en las preferencias de los usuarios que han otorgado calificaciones similares a otras películas que ha visto.

In [ ]:
```
#CREAMOS LOS NODOS
```

In [ ]:
```
MERGE (french:Cuisine {name:'French'})
MERGE (italian:Cuisine {name:'Italian'})
MERGE (indian:Cuisine {name:'Indian'})
MERGE (lebanese:Cuisine {name:'Lebanese'})
MERGE (portuguese:Cuisine {name:'Portuguese'})
MERGE (british:Cuisine {name:'British'})
MERGE (mauritian:Cuisine {name:'Mauritian'})

MERGE (zhen:Person {name: "Zhen"})
MERGE (praveena:Person {name: "Praveena"})
MERGE (michael:Person {name: "Michael"})
MERGE (arya:Person {name: "Arya"})
MERGE (karin:Person {name: "Karin"})

MERGE (praveena)-[:LIKES {score: 9}]->(indian)
MERGE (praveena)-[:LIKES {score: 7}]->(portuguese)
MERGE (praveena)-[:LIKES {score: 8}]->(british)
MERGE (praveena)-[:LIKES {score: 1}]->(mauritian)

MERGE (zhen)-[:LIKES {score: 10}]->(french)
MERGE (zhen)-[:LIKES {score: 6}]->(indian)
MERGE (zhen)-[:LIKES {score: 2}]->(british)

MERGE (michael)-[:LIKES {score: 8}]->(french)
MERGE (michael)-[:LIKES {score: 7}]->(italian)
MERGE (michael)-[:LIKES {score: 9}]->(indian)
MERGE (michael)-[:LIKES {score: 3}]->(portuguese)

MERGE (arya)-[:LIKES {score: 10}]->(lebanese)
MERGE (arya)-[:LIKES {score: 10}]->(italian)
MERGE (arya)-[:LIKES {score: 7}]->(portuguese)
MERGE (arya)-[:LIKES {score: 9}]->(mauritian)

MERGE (karin)-[:LIKES {score: 9}]->(lebanese)
MERGE (karin)-[:LIKES {score: 7}]->(italian)
MERGE (karin)-[:LIKES {score: 10}]->(portuguese)
```



Se devuleve la similitud de michael y arya

In [ ]:
```
MATCH (p1:Person {name: 'Michael'})-[likes1:LIKES]->(cuisine)
MATCH (p2:Person {name: "Arya"})-[likes2:LIKES]->(cuisine)
RETURN p1.name AS from,
       p2.name AS to,
       algo.similarity.cosine(collect(likes1.score), collect(likes2.score)) AS similarity
```

"from"	"to"	"similarity"
"Michael"	"Arya"	0.9788908326303921

In [ ]:
```

```