Ejercicio evaluable node4j

create (pedro: persona{nombre:"Pedro", edad: 45}), (marta: persona{nombre:"Marta", edad:27, ciudad:"Cáceres"}),

(pepe: persona{nombre:"Pepe", edad:25}),

(eva: persona{nombre:"Eva", edad:33, ciudad:"Badajoz"}),

(elisa: persona{nombre:"Elisa", edad:34, ciudad:"Badajoz"}),

(sergio: persona{nombre:"Sergio", edad:26}),

(celia: persona{nombre:"Celia", edad:20}),

(pedro)-[:Es\_amigo{desde:"31/12/2021"}]->(marta),

(pedro)-[:Es\_amigo]->(eva),

(pedro)-[:Es\_amigo]->(pepe),

(pepe)-[:Es\_amigo]->(sergio),

(eva)-[:Es\_amigo]->(marta),

(marta)-[:Es\_amigo]->(eva),

(marta)-[:Es\_amigo]->(elisa),

(sergio)-[:Es\_amigo]->(celia)

1 y 2.-

create (carmen:persona{nombre:"Carmen", edad: 62, ciudad:"Badajoz"}),

(carlos:persona{nombre:"Carlos", edad: 63, ciudad: "Badajoz"}),

(jose:persona{nombre:"José", edad: 42, ciudad: "Badajoz"}),

(jesus:persona{nombre:"Jesús", edad: 35, ciudad: "Badajoz"}),

(jose)-[:Es\_hermano]->(jesus),

(jesus)-[:Es\_hermano]->(jose),

(carmen)-[:Es\_madre]->(jesus),

(carmen)-[:Es\_madre]->(jose),

(carlos)-[:Es\_padre]->(jesus),

(carlos)-[:Es\_padre]->(jose)

3.-

merge (pedro{nombre:"Pedro"}) set pedro.ciudad="Cáceres"

merge (pepe{nombre:"Pepe"}) set pepe.ciudad="Cáceres"

merge (sergio{nombre:"Sergio"}) set sergio.ciudad="Cáceres"

merge (celia{nombre:"Celia"}) set celia.ciudad="Badajoz"

4.-

match (persona{ciudad:"Badajoz"}) return persona

match (persona{ciudad:"Cáceres"}) return persona

5.-

match (n) detach delete n

create (pedro: persona{nombre:"Pedro", edad: 45}), (marta: persona{nombre:"Marta", edad:27, ciudad:"Cáceres", provincia:"Extremadura"}),

(pepe: persona{nombre:"Pepe", edad:25}),

(eva: persona{nombre:"Eva", edad:33, ciudad:"Badajoz", provincia:"Extremadura"}),

(elisa: persona{nombre:"Elisa", edad:34, ciudad:"Badajoz", provincia:"Extremadura"}),

(sergio: persona{nombre:"Sergio", edad:26}),

(celia: persona{nombre:"Celia", edad:20}),

(pedro)-[:Es\_amigo{desde:"31/12/2021"}]->(marta),

(pedro)-[:Es\_amigo]->(eva),

(pedro)-[:Es\_amigo]->(pepe),

(pepe)-[:Es\_amigo]->(sergio),

(eva)-[:Es\_amigo]->(marta),

(marta)-[:Es\_amigo]->(eva),

(marta)-[:Es\_amigo]->(elisa),

(sergio)-[:Es\_amigo]->(celia)

create (carmen:persona{nombre:"Carmen", edad: 62, ciudad:"Badajoz", provincia:"Extremadura"}),

(carlos:persona{nombre:"Carlos", edad: 63, ciudad: "Badajoz", provincia:"Extremadura"}),

(jose:persona{nombre:"José", edad: 42, ciudad: "Badajoz", provincia:"Extremadura"}),

(jesus:persona{nombre:"Jesús", edad: 35, ciudad: "Badajoz", provincia:"Extremadura"}),

(jose)-[:Es\_hermano]->(jesus),

(jesus)-[:Es\_hermano]->(jose),

(carmen)-[:Es\_madre]->(jesus),

(carmen)-[:Es\_madre]->(jose),

(carlos)-[:Es\_padre]->(jesus),

(carlos)-[:Es\_padre]->(jose)

6.-

match (eva{nombre:"Eva"})-[:Es\_amigo\*1..2]-(persona) return persona

7.-

match (pedro{nombre:"Pedro"})-[:Es\_amigo\*1..2]-(persona) return persona.nombre, persona.edad as nombre, persona