**1.- Crear los nodos**

create (maria: persona{nombre:"María", edad: 27}), (juan: persona{nombre:"Juan", edad:35}),

(javier: persona{nombre:"Javier", edad:40}),

(pedro: persona{nombre:"Pedro", edad:20}),

(ana: persona{nombre:"Ana", edad:18}),

(jorge: persona{nombre:"Jorge", edad:23}),

(antonio: persona{nombre:"Antonio", edad:60})

**2.- Crear las relaciones**

create (maria: persona{nombre:"María", edad: 27}), (juan: persona{nombre:"Juan", edad:35}),

(javier: persona{nombre:"Javier", edad:40}),

(pedro: persona{nombre:"Pedro", edad:20}),

(ana: persona{nombre:"Ana", edad:18}),

(jorge: persona{nombre:"Jorge", edad:23}),

(antonio: persona{nombre:"Antonio", edad:60}),

(javier)-[:conoce]->(juan),

(javier)-[:conoce]->(antonio),

(javier)-[:conoce]->(maria),

(antonio)-[:conoce]->(jorge),

(maria)-[:conoce]->(ana),

(maria)-[:conoce]->(pedro)

**3.- Muestra todos los amigos de Javier.**

**match (javier{nombre:"Javier"})-[:conoce]->(persona)**

**return persona**

**4.- Muestra los conocidos indirectos.**

**match (javier{nombre:"Javier"})-[:conoce\*2]->(persona)**

**return persona**

**5.- Muestra los conocidos directos e indirectos.**

**match (javier{nombre:"Javier"})-[:conoce\*1..2]->(persona)**

**return persona**

**6.- Muestra los conocidos a partir del nivel 1.**

**match (javier{nombre:"Javier"})-[:conoce\*]->(persona)**

**return persona**

**7.- Muestra la media de edad de todas las personas.**

**MATCH (PERSONA)RETURN AVG(PERSONA.edad)**

**8.- Muestra una lista con todas las edades**

**match (PERSONA) return COLLECT (PERSONA.edad)**

**9.- Utilizando el operador especial label, muestra la etiqueta, el nodo raíz y contar la cantidad de personas que conoce Javier. No sabemos el nombre de la relación ni a donde apunta.**

match (javier{nombre:"Javier"})-[r]->()

return labels(javier),count(\*)

**10.- Utilizando el operador especial type, cuenta las relaciones agrupadas por le tipo de las mismas.**

match (javier{nombre:"Javier"})-[r]->()

return type(r),count(\*)

**11.- Crea una relación nueva (con un nombre diferente) entre Javier y otro nodo.**

match (javier{nombre:"Javier"}),(maria{nombre:"María"})

create (javier)-[:Hermanos]->(maria)

**12.- Repite ahora el ejercicio 8.**

**Calcula la edad mínima, la máxima y la suma de las edades.**

match (p:persona)

return min(p.edad) as Edad,max(p.edad),sum(p.edad)