

MongoDB

1.

```
db.getCollection('tweets')
  .find(
    {
      $and: [
        {
          $expr: {
            $gte: [{ $strLenCP: '$text' }, 50]
          }
        },
        {
          text: { $regex: RegExp('https:', 'i') }
        }
      ]
    },
    { text: 1 }
  )
  .sort({ favorite_count: -1 });
```

2.

```
[
  {
    $project: {
      "user.screen_name": 1,
      "quoted_status_permalink.url": 1,
      "place.name": 1,
    },
  },
  {
    $group: {
      _id: "$_id",
      screen_name: {
        $first: "$user.screen_name",
      },
      quoted: {
        $addToSet: "$quoted_status_permalink.url",
      },
      cities: {
        $addToSet: "$place.name",
      },
    },
  },
]
```

3.

a. \$match

Filtra los tweets **provenientes de iPhone o Android** (distintos tipos de sistemas operativos) **y de Argentina**.

b. \$group

Agrupar por:

- **id del tweet** → en caso de que no sea un tweet que responde a otro, reemplaza el valor de "in_reply_to_status_id_str" (que es null) por su mismo id; si es un tweet que responde a otro, el valor de "in_reply_to_status_id_str" es el id del otro tweet y nada cambia.
- **tweets** → arreglo que contiene el id del tweet, el text del tweet, el usuario (objeto que contiene la información correspondiente al usuario) y la fecha de creación del tweet.
- **avg_retweets** → cantidad de retweets que obtuvo el tweet.

c. \$project

Muestra:

- **text** → objeto que resulta del filtrado del objeto *tweets*; contiene un conjunto de elementos (id del tweet, texto del tweet, usuario que generó el tweet y fecha de creación del mismo) correspondientes a la información de un tweet que no es una respuesta a otro; si el tweet es una respuesta a otro, el objeto almacena el valor nulo.
- **replies** → arreglo ordenado por fecha de creación ascendente que resulta del filtrado del objeto *tweets*; cada elemento es un objeto que contiene un conjunto de elementos (id del tweet, texto del tweet, usuario que generó el tweet y fecha de creación del mismo) correspondientes a la información de un tweet que es una respuesta a otro; será un arreglo de únicamente un elemento pues un tweet sólo puede responder a otro tweet y no a un conjunto, de esta forma el ordenamiento dado es intrascendente pues no existe otro elemento en el arreglo contra el cual se realizará la operación; si el tweet no es una respuesta a otro, el arreglo se encuentra vacío.
- **avg_retweets** → cantidad de retweets que obtuvo el tweet.

Finalmente podemos decir que los documentos exportados por la query representarán o bien un tweet (arreglo de *replies* vacío) o bien una respuesta a un tweet (objeto *text* en nulo) si estos fueron realizados desde un dispositivo iPhone o Android y desde Argentina. Adicionalmente se podrá visualizar la cantidad de retweets de cada uno.

4.

```
MATCH (a:Person {name: "Billy", surname: "Moore"})-[b:FAMILY_REL]-(c:Person)
WHERE NOT (c)-[:PARTY_TO]->(:Crime)
RETURN c
```



5.

```
MATCH (a:Person)-[:PARTY_TO]->(b:Crime)
WITH a, COUNT(b) AS crimes
MATCH (a:Person)-[:PARTY_TO]->(b:Crime)
WHERE crimes > 4
RETURN DISTINCT a.name, a.surname, b.type
```

Para ver si se conocen:

