

Instrucciones: Resuelva los ejercicios, detallando los pasos más importantes, genere con esta información un archivo pdf. Este archivo llámelo "NOSQL_(nro padrón)", es decir por ejemplo: NOSQL_77463. En la cabecera de este archivo escriba su nombre y apellido y número de padrón. Suba este archivo al campus antes de las 11:59 PM del sábado 25/05/2024 para que podamos entregarle la corrección antes del parcial, o antes de las 11:59 PM del martes 28/05/2024 en caso de que no tenga apuro por el resultado.

PARCIALITO VI: NOSQL

CONSULTAS DE MONGODB

Trabajando con el set de datos utilizado en el *Taller VII: MongoDB* con la siguiente URI de conexión:

`mongodb+srv://estudiante:bddfiuba@tweets.cbxukni.mongodb.net/tweets`

Resuelva los siguientes enunciados:

1. Obtener los ids, cantidad de hashtags y likes para tweets que tengan 50 o más likes ('`favorite_count`') y hayan sido a las 3 de la tarde. Ordenar la salida de forma descendente por cantidad de likes.
Utilice una única consulta básica con `find(<query>, <proyeccion>).sort({}).limit({})`
2. Para cada hashtag obtener los usuarios que lo utilizaron además del máximo, mínimo y promedio de retweets, sólo teniendo en cuenta aquellos tweets que utilicen más de 3 hashtags (primero se deben filtrar los tweets y luego hallar los valores por cada hashtag).
Se debe utilizar el pipeline de agregación.
3. Dada la consulta: Anexo: Consulta ejercicio 3 o disponible en github
Explicar qué sucede en cada paso del pipeline y en forma resumida qué resuelve la query completa.

CONSULTAS DE NEO4J

Resuelva las siguientes consultas de Neo4j utilizando la base "Crime Investigation" vista en clase (*Taller VIII: Neo4j*).

4. Investigue los crímenes cometidos en 165 Laurel Street, muestre las personas que participaron de algún crimen y si tienen relación entre ellas muéstrela.
5. Muestre la (o las) persona(s) que ha(n) realizado mas de 7 comunicaciones telefónicas.

ANEXO: CONSULTA EJERCICIO 3

```
[
  {
    $match: {
      lang: /es|pt/,
      'place.country': 'Brasil',
    },
  },
  {
    $group: {
      _id: { $ifNull: [ // coalesce
        '$in_reply_to_status_id_str',
        '$_id',
      ] },
      tweets: { $push: {
        tweet_id: '$_id',
        text: '$full_text',
        user: '$user',
        created_at: '$created_at.date',
      } },
      avg_retweets: { $avg: '$retweet_count' },
    },
  },
  {
    $project: {
      tweet: { $arrayElemAt: [{
        $filter: {
          input: '$tweets',
          as: 'reply',
          cond: {
            $eq: ["$$reply.tweet_id", '$_id'],
          },
        }, 0 ] },
      replies: { $sortBy: { input: {
        $filter: {
          input: '$tweets',
          as: 'reply',
          cond: { // eliminamos el mismo tweet
            $ne: ["$$reply.tweet_id", '$_id'],
          },
        },
      }, sortBy: {created_at: 1} } } },
      avg_retweets: 1,
    },
  },
]
```