**Amir Amemi** **MPSD1**

### **Le compte rendu de TP2**

**3. Premières manipulations**

**3.1. Insertion des données**

-Démarrez un serveur sur la porte 27017 en utilisant le chemin de données « C:/data/» :

Mongo –port 2717

-Utilisez la base « UserDB » :

use UserDB

- Créez la collection « users » dans la base « UserDB »:

db.createCollection("users")

- Insérez les données suivantes :

-5 utilisateurs : toto, titi, tutu, tata, lolo

- titi, tutu, tata suivent toto / tutu, tata suivent titi / toto, titi, tutu suivent tata / tutu et lolo n’ont pas de followers

- toto a posté 4 tweets

- titi a posté 2 tweets

- tutu a posté 2 tweets

-tata et lolo n’ont pas posté de tweet

db.users.insert([{id: 1, username: "toto", followersID: [2, 3, 4], followingID: [2, 3, 4], tweets: [{idtwit: 1, contenu: "Bonjour", date: "2022-04-12"},{idtwit: 2, contenu: "Bonjour", date: "2022-04-12"},{idtwit: 3, contenu: "Bonjour", date: "2022-04-12"}, {idtwit: 4, contenu: "Bonjour", date: "2022-04-12"}]})

db.users.insert({id: 2, username: "titi", followersID: [1], followingID: [3, 4], tweets: [{idtwit: 1, contenu: "Bonjour", date: "2022-04-12"},{idtwit: 2, contenu: "Bonjour", date: "2022-04-12"}]})

db.users.insert({id: 3, username: "tutu", followersID: [1, 4], followingID: [1, 4], tweets: [{idtwit: 1, contenu: "Bonjour", date: "2022-04-12"},{idtwit: 2, contenu: "Bonjour", date: "2022-04-12"}]})

db.users.insert({id: 4, username: "tata", followersID: [1, 2, 3], followingID: [], tweets: []})

db.users.insert( {id: 5, username: "lolo", followersID: [], followingID: [], tweets: []})

-Affichez les bases créées :

Show dbs

- Affichez les collections:

Show collections

**3.2. modification des données**

-Changer le nom de l’utilisateur "lolo" par "lulu":

db.users.update({ username: "lolo" },{ $set: { username: "lulu" } )

- Ajouter le tweet "youpi un nouveau tweet" créé par l’utilisateur "titi" le 13/02/2016 à 17:44:43 :

db.users.update({username: "titi"},{$push: {tweets: {idtwit: 3, contenu: "youpi un nouveau tweet", date: "2016-02-13"}}})

-Ajouter "tata" comme follower de "lulu":

db.users.update({username: "lulu"},{$push: {followersID: 4}})

**3.3. Interrogation**

**-** Lister les utilisateurs (pour vérifier une dernière fois que les informations insérées et mises à jour sont correctes):

db.users.find()

- Obtenir les tweets postés par l’utilisateur "toto" :

db.users.find({username: "toto"}, {tweets:1})

- Obtenir les utilisateurs qui ont utilisé le hashtag « bacasable »:

Db.users.find({"tweets.contenu": /.\*bacasable.\*/},{username: 1, \_id: 0})

- Obtenir les utilisateurs qui sont suivis par les utilisateurs 2 ou 4:

db.users.find({id: {$in: [2, 4]}},{followingID: 1})

- Obtenir les utilisateurs qui sont suivis à la fois par les utlisateurs 2 et 4 :

db.users.find({id: {$in: [2, 4]}, followingID: {$all: [2, 4]}},{username: 1, \_id: 0})

-Obtenir les utilisateurs dont le premier tweet date d’avril 2016:

db.users.find({tweets: {$elemMatch: {date: {$regex: "Avril 2016"}}}},{username: 1, \_id: 0})

- Obtenir les tweets contenant une url (présence de la chaîne de caractères ’http’):

Db.users.find( {tweets: {$elemMatch: {contenu: {$regex: "http"}}}},{tweets: {$elemMatch: {contenu: {$regex: "http"}}}, username: 1, \_id: 0})

- Obtenir le plus grand nombre de followers:

db.users.aggregate([{$project: {username: 1, followersCount: {$size: "$followersID"}}},

{$sort: {followersCount: -1}},{$limit: 1}])

- Obtenir le deuxième tweet de l’utilisateur "titi":

db.users.find({username: "titi"}, {tweets: 2})

-  Obtenir le texte (uniquement) des tweets contenant le mot ’Ceci’:

db.tweets.find({$text: {$search: "Ceci"}}).project({ contenu: 1})