

(:~ Amir-SHIRALI POUR and Hassina-BOUFATIS ~:)

(: 1) XQuery : Tweets :)

(: 1. Indiquer le nombre de tweets et d'utilisateurs dans la base. :)

```
count(//*[@users/user])
```

(:~ Second solution: ~:)

```
count(//*[@tweets/tweet])
```

(: 2. Donner l'ensemble des hashtags contenus dans la base. :)

(: First solution: :)

```
for $x in doc("document.xml")/tweeter/tweets/tweet/message
return
```

```
  <result>
    {$x/hashtag}
  </result>
```

(: Second solution celle qui marche :)

```
for $x in doc("document.xml")//message
return   $x/hashtag
```

(: 3. Créer une liste de paires tweet-auteur, avec chaque paire contenue dans un element result. :)

```
for $u in //user return for $t in //tweet[@user_id = $u/
@user_id] return <result><user>{$u/data(@user_id)} : {$u}</
user><tweet>{$t//text()}</tweet></result>
```

(:~ 4- Pour chaque utilisateur, lister le nom de l'utilisateur et la date de tous ses tweets, le tout regroupé dans un élément result. ~:)

```
for $u in //user
let $t := //tweets/tweet[@user_id = $u/@user_id] return
let $v := //tweets/tweet[@user_id = $u/@user_id]/author return
<result>
  <user>{($u//text())}</user>
  <tweets>{($v/username), ($t/date) }</tweets>
</result>
```

(:~ 5 - Lister les utilisateurs qui ont publié un tweet qui a été retwitté. ~:)

```
for $u in //user
let $t := //tweets/tweet[@user_id = $u/@user_id]
return
```

```

    if ($t/retweet)
    then $u
    else()

```

(::~ 7 – Lister les utilisateurs de la plateforme en ordre alphabétique. ::)

```

for $u in //tweets/tweet/author
  order by $u/username/text()
    ascending
  return $u/username/text()

```

(::~ 8 – Listez les tweets contenant l'hashtag #I<3XML (nous l'avons fait avec le hashtag banana) ::)

```

for $t in //tweets/tweet/message
  return
    if ($t//hashtag/text() = "Banana")
    then $t
    else()

```

(::~ 10. Pour chaque utilisateur, indiquer l'ensemble des hashtags qu'il a utilisés dans ses Tweets. ::)

```

for $u in //user
  let $t := //tweets/tweet[@user_id = $u/@user_id]/message return
  let $v := //tweets/tweet[@user_id = $u/@user_id]/author return
<result>
  <user>{($u//text())}</user>
  <tweets>{($v/username), ($t/hashtag) }</tweets>
</result>

```

(::~ 11 – Pour chaque tweet ayant des références utilisateur, retournez le tweet avec la liste des références utilisateur. ::)

```

for $t in //tweets/tweet/message
  return $t/usernameReference/text()

```

(::~ 9. Trouvez le tweet le plus ancien ainsi que le plus récent. ::)

```

for $t in //tweets/date
for $u in //tweets/date
  return
    if ($t/seconds < $u/seconds)
    then $t/seconds
    else($u/seconds)

```

(::~ Second solution ::)

```

let $min := min($tweets/tweets/date/text())
let $max := max($tweets/tweets/date/text())
return $max

```

(::~ Ex.2 ::)

```
(::~ Id : identifiant de la station
La : latitude
Lg : longitude
Av : nombre de places occupées
Fr : nombre de places libres
To : nombre total de places
Cb : lecteur de carte bancaire disponible (0 pour non, 1 pour oui)
~:)
```

```
for $u in //tweets/tweet/author order by $u/username/text()
ascending return $u/username/text()
```

```
<html>
<head>
  <title>TP Données du Web</title>
</head>
<body>
  <ol>
    let $name := {($file/@na)}
    let $occupiedPlaces := $file/@av
    let $totalPlaces := $file/@to
    { for $file in doc("velomagg.xml")//si

      return
        <li>
          station name : $name
          occupiedPlaces: $occupiedPlaces
          totalPlaces: $totalPlaces
        </li>
    }
  </ol>
</body>
</html>
```