Assignment : 1

XQueries :

1.) Retrieve the usernames and the number of posts for each user:

```
for $user in //user
return concat("Username: ", $user/@username, ", Number of Posts: ",
count($user/posts/post), "&#xa;")
```

```
Username: john_doe, Number of Posts: 3

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: jane_smith, Number of Posts: 2

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: ashay_shah, Number of Posts: 2

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: prince_patel, Number of Posts: 2

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: yash_rathore, Number of Posts: 2

Username: , Number of Posts: 0

Username: ravi_v, Number of Posts: 2

Username: , Number of Posts: 0
```

```
Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: vijay_yadav, Number of Posts: 2

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: kuldeep, Number of Posts: 2

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: tarun, Number of Posts: 2

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: , Number of Posts: 0

Username: deepak, Number of Posts: 2

Username: , Number of Posts: 0

Username: , Number of Posts: 0

```

The output is a sequence of strings, where each string contains the username and the number of posts for a user, separated by a comma. The
 character sequence is used to add a new line after each string.

2.) Get the usernames of all users who have at least one friend:

for $user in //user[friends/user]
return $user/@username

```
john_doe
jane_smith
ashay_shah
prince_patel
yash_rathore
ravi_v
vijay_yadav
kuldeep
tarun
deepak

```

3.) Return the first name, last name, and email of the user with id "u002"

```
let $user := //user[@id = "u002"]
return <user>
        <first_name>{$user/@first_name}</first_name>
        <last_name>{$user/@last_name}</last_name>
        <email>{$user/@email}</email>
       </user>
```

```
<user>
          <first_name first_name="Jane"></first_name>
          <last_name last_name="Smith"></last_name>
          <email email="janesmith@example.com"></email>
</user>
```

The output is in xml format.
$user := //user[@id = "u002"] : Assigns the userid = “u002”.
return <user>
<first_name>{$user/@first_name}</first_name>
<last_name>{$user/@last_name}</last_name>
<email>{$user/@email}</email>
</user>

We are asking the user information in xml format.

XPath :

1.) Select the id of all users who have the occupation "Therapist"

/social_networking_website/user[profile/@occupation="Therapist"]/@id

```
u002
u005
```

2.) Select the captions of all images and videos in the third user's posts

/social_networking_website/user[3]/posts/post/*[self::image or self::video]/@caption

```
At the beach
My dog doing tricks

```

3.) Select all the posts posted by user id "u002".

//user[@id="u002"]/posts

```
<posts>
        <post id="p004" timestamp="2022-02-14T10:00:00Z">
          <image caption="Happy Valentine&apos;s Day!">https://example.com/valentinesday.
        </post>
        <post id="p005" timestamp="2023-03-05T16:45:00Z">
          <video caption="My cat doing tricks">https://example.com/catvideo.mp4</video>
        </post>
      </posts>
```