Abrar Hasnat Project 1 Part 3

3.1 A)

```
SELECT photoID FROM photo AS p WHERE photoPoster = 'TestUser' OR photoID IN
(SELECT photoID FROM Photo WHERE photoPoster != %s AND allFollowers = 1 AND
photoPoster IN
(SELECT username_followed FROM follow WHERE username_follower = 'TestUser'
AND username_followed = photoPoster AND followstatus = 1)) OR photoID IN
(SELECT photoID FROM share NATURAL JOIN belongto NATURAL JOIN photo WHERE
username = %s AND photoPoster != 'TestUser') ORDER BY postingdate DESC
```

3.1 B)

https://github.com/AbrarHasnat/Finstagram/blob/master/init.py

3.2 A) Below is a youtube link to my screen record on my personal NYU email. (Wasnt sure how else to do this, I hope it was okay)

https://www.youtube.com/watch?v=fg0efmNunVU

3.2 B)

The additional features I will be working towards are as follows:

7. Optional (this can be one of your extra features:) Add comments: Users can add comments about photos that are visible to them. Decide what data about the comments should be stored and displayed and how you want to restrict visibility of comments.
    - Using the comments feature would require an additional attribute to photos and would require updating. I will most likely require a field to check if someone is allowed to comment. Potentially someone will be able to like the comment as well. We would display the comment under the photo.
8. Optional (this can be one of your extra features:) Like Photo (restricted to liking Photos that are visible to the user)
    - After permission check if a user has the ability to like someone's photo. It would be a boolean in terms of the like relationship between the picture owner and the liker. The database would keep track of how many likes per photo and display it with comments.
10. Optional (this can be one of your extra features:) Search by tag:Search for photos that are visible to the user and that tag some particular person (the user or someone else)

- We would have to develop a search query to locate the user via tag that would connect the query to the database of users. The query would return matching tags with their basic information (fname, lname, etc..)

1. View visible photos:: Finstagram shows the user the photoID and photoPoster of each photo that is visible to her, arranged in reverse chronological order.

- Can be done by displaying the photos via their photo IDs

2. View further photo info:
Display the following for visible photos (You may include this with the results of "view visible photos" or you may supply a different way for users to see this additional info, such as clicking on additional links).
a) Display the photo or include a link to display it.
b) The firstName and lastName of the photoPoster
c) The timestamp,
d) the usernames, first names and last names of people who have been tagged in the photo (taggees), provided that they have accepted the tags
(Tag.acceptedTag == true)
e) The usernames of people who have liked the photo and the rating they gave it

- This would require to grab more information from the database and render it to the upload page. More attributes would have to be present in the display.