

# Inverzni Inženjering

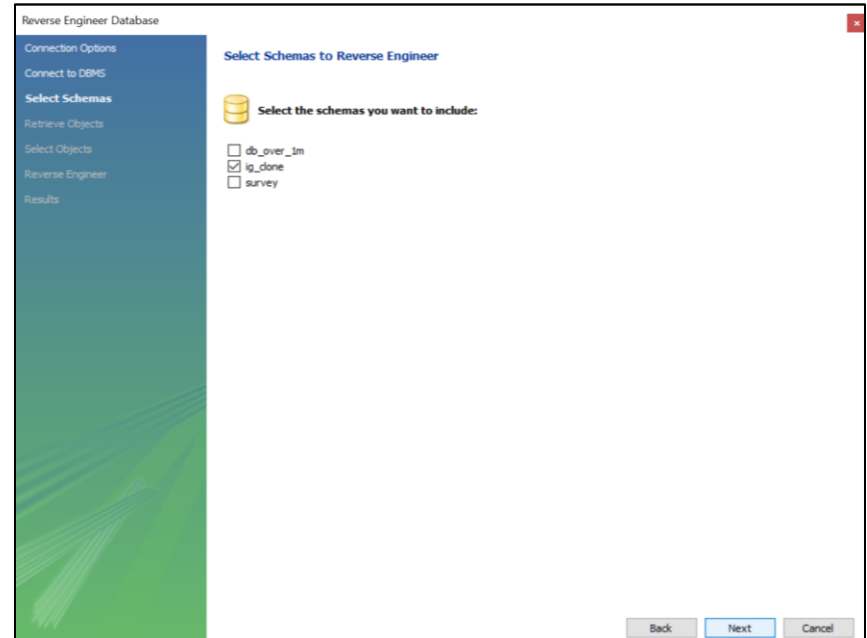
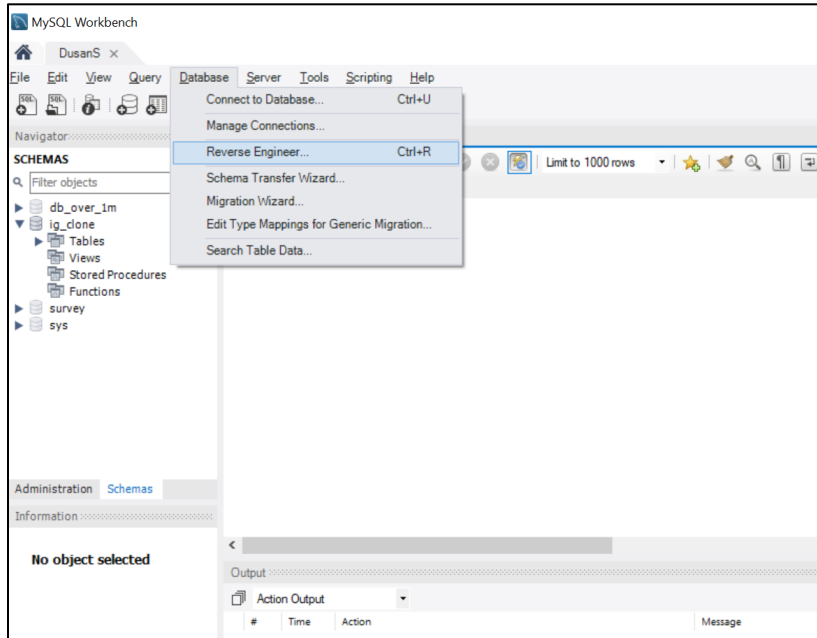
---

# Instagram Clone

## – Reverse Engineering –

---

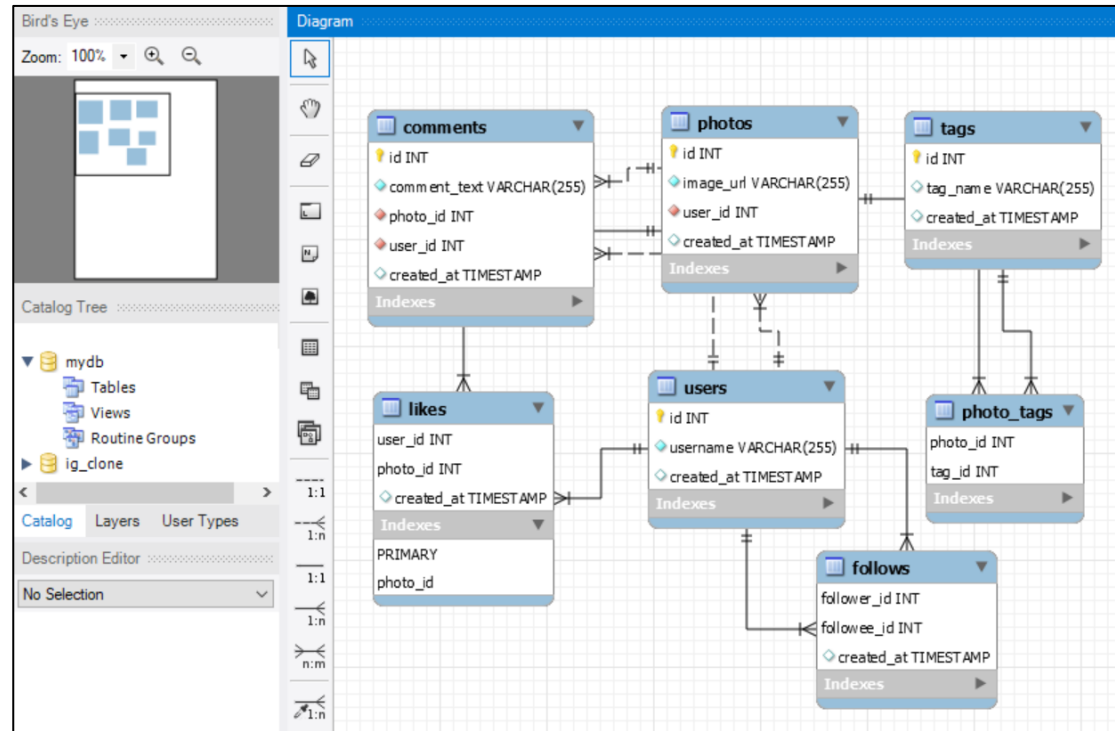
Inverznim inženjeringom želimo da nam Workbench prikaže EER dijagram ukoliko smo definisali šemu svih tabela u bazi.



# Instagram Clone

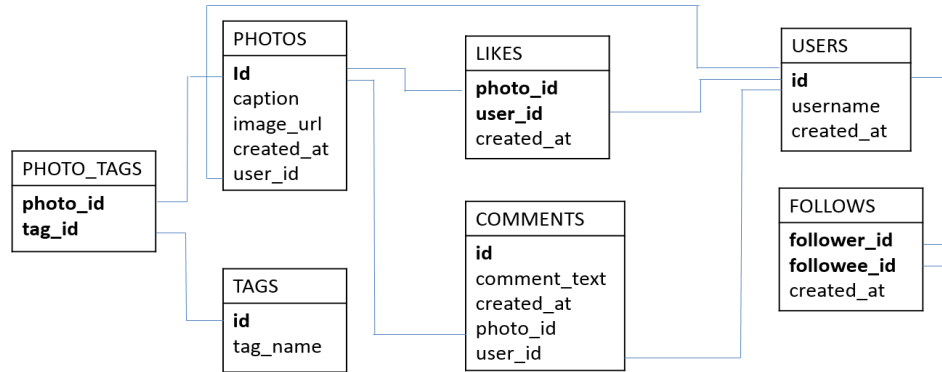
## – Reverse Engineering –

- Inverznim inženjeringom je prikazan EER dijagram.
- EER dijagram je koristan da se lakše razumeju veze između relacija i šema svake relacije



# Zadatak br 1

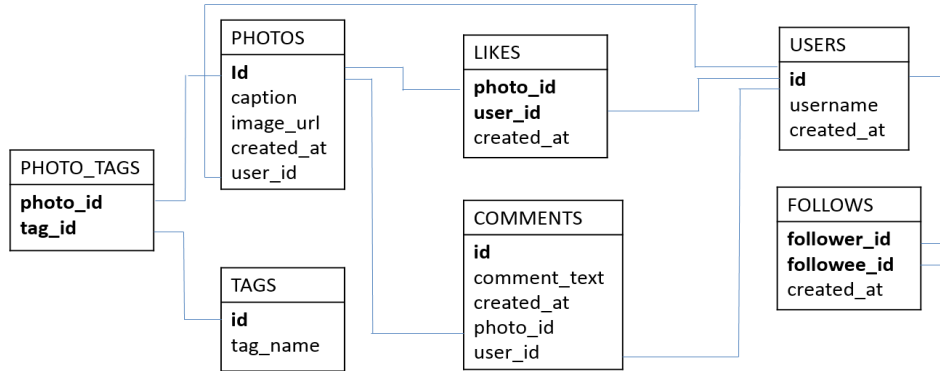
Želimo da nagradimo korisnike koji najduže koriste našu platformu. Pronađite prvih 5 korisnika koji su se registrovali na platformi.



```
SELECT *
FROM users
ORDER BY created_at
LIMIT 5;
```

# Zadatak br 2

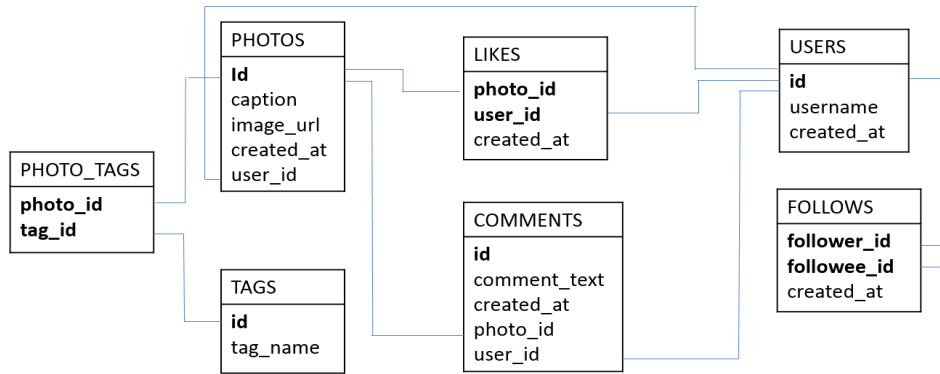
Želimo da saznamo kog dana u nedelji se korisnici najčešće registruju jer želimo da pokrenemo ad kampanju tog dana.



```
SELECT
DAYNAME(created_at) AS day, COUNT(*) AS total
FROM users
GROUP BY day
ORDER BY total DESC
LIMIT 2;
```

# Zadatak br 3

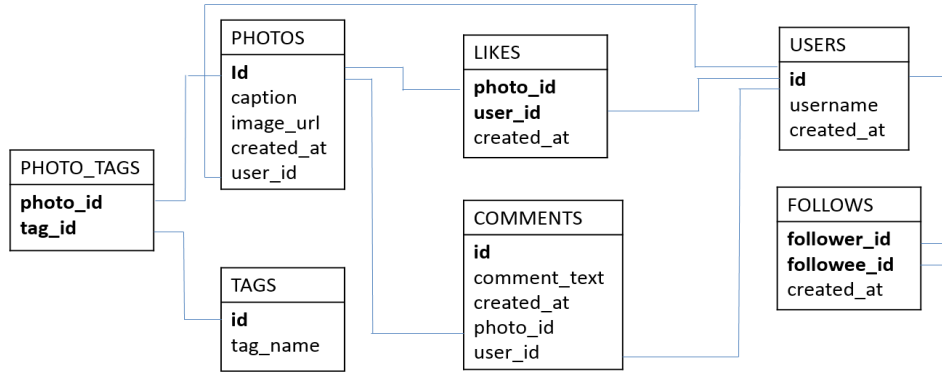
Želimo da motivišemo neaktivne korisnike. Prikaži korisnike koji nisu postovali ni jednu fotografiju.



```
SELECT username
FROM users
LEFT JOIN photos
  ON users.id = photos.user_id
WHERE photos.id IS NULL;
```

# Zadatak br 4

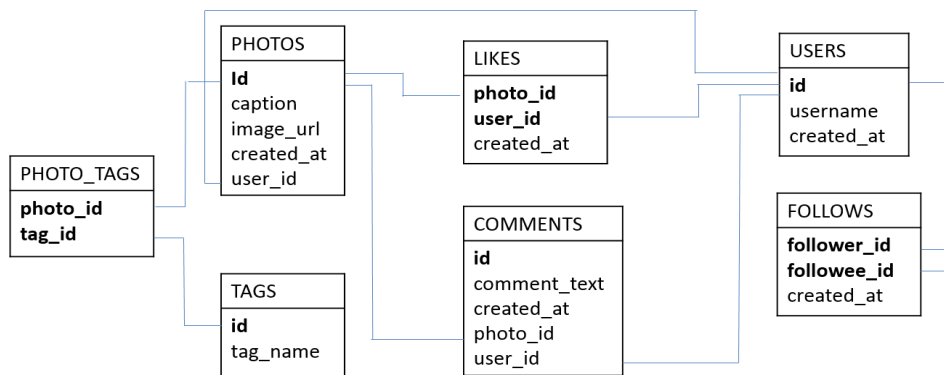
Želimo da pronađemo korisnika koji je postavio fotografiju sa najviše lajkova



```
SELECT username, photos.id, photos.image_url, COUNT(*) AS total
FROM photos INNER JOIN likes ON likes.photo_id = photos.id
        INNER JOIN users ON photos.user_id = users.id
GROUP BY photos.id
ORDER BY total DESC
LIMIT 1;
```

# Zadatak br 5

Investitor želi da zna koliko često prosečan korisnik postuje fotografije. Odrediti broj fotografija po korisniku

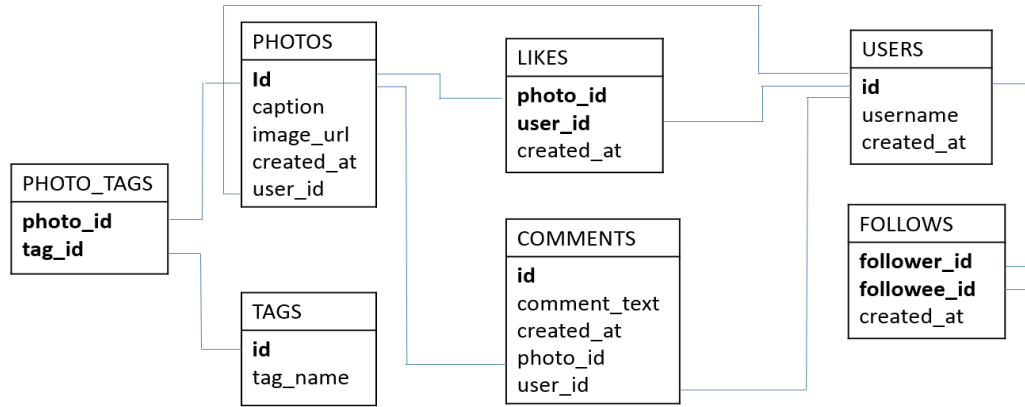


```
SELECT (SELECT Count(*) FROM photos) / (SELECT Count(*) FROM users) AS avg;
```



# Zadatak br 6

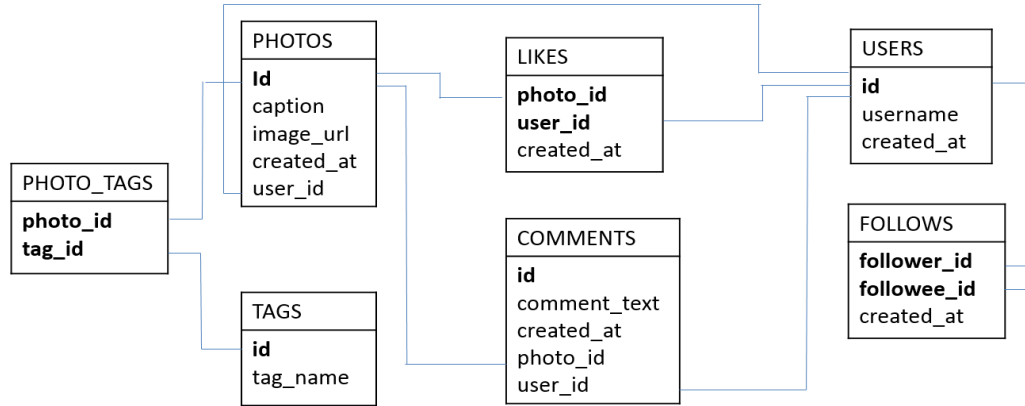
Brand kompanija želi da zna koji hashtag je najbolje da koristi. Potrebno je prikazati 5 najpopularnijih hashtag-ova.



```
SELECT tags.tag_name, Count(*) AS total
FROM photo_tags JOIN tags ON photo_tags.tag_id = tags.id
GROUP BY tags.id
ORDER BY total DESC
LIMIT 5;
```

# Zadatak br 7

Postoji problem sa botovima na sajtu. Prikazati korisnike koji su lajkovali svaku fotografiju na sajtu



```
SELECT username, Count(*) AS num_likes
FROM users INNER JOIN likes ON users.id = likes.user_id
GROUP BY likes.user_id
HAVING num_likes = (SELECT Count(*) FROM photos);
```