1. Создать двух пользователей (с помощью метода User.objects.create\_user).

#импортируем модель

from NewsPaper.models import User

user1 = User.objects.create\_user('Petr Petrov')

user2 = User.objects.create\_user('Ivan Ivanov')

2. Создать два объекта модели Author, связанные с пользователями.

#импортируем модель

from NewsPaper.models import Author

Author.objects.create(author = user1)

Author.objects.create(author = user2)

3. Добавить 4 категории в модель Category.

импортируем модель

from NewsPaper.models import Category

Category.objects.create(article\_category = 'sport') # (id=1)

Category.objects.create(article\_category = 'politica')

Category.objects.create(article\_category = 'education')

Category.objects.create(article\_category = 'culture')# (id=4)

4. Добавить 2 статьи и 1 новость.

импортируем модель

from NewsPaper.models import Post

author = Author.objects.get(id=1)

# <Post: Post object (1)>

Post.objects.create(

post\_author = author,

category = 'A',

title = 'The record',

content = 'Long – long text'

)

# <Post: Post object (2)>

Post.objects.create(

post\_author = author,

category = 'A',

title = 'The Magic',

content = 'Not very long story'

)

# <Post: Post object (3)>

Post.objects.create(

post\_author = author,

category = 'N',

title = 'Some news',

content = 'We have launched a new project'

)

5. Присвоить им категории (как минимум в одной статье/новости должно быть не меньше 2 категорий).

#статье под id=1 присваиваем категорию 1 - ‘sports’

Post.objects.get(id=1).post\_category.add(Category.objects.get(id=1))

статье под id=1 присваиваем категорию 4 - 'culture'

Post.objects.get(id=1).post\_category.add(Category.objects.get(id=4))

6. Создать как минимум 4 комментария к разным объектам модели Post (в каждом объекте должен быть как минимум один комментарий).

from NewsPaper.models import \*

Comment.objects.create(

comment\_post=Post.objects.get(id=1),

comment\_user=Author.objects.get(id=1).author,

feedback\_text = 'Very interesting article'

)

Comment.objects.create(

comment\_post=Post.objects.get(id=2),

comment\_user = Author.objects.get(id=1).author,

feedback\_text = 'It is interesting'

)

Comment.objects.create(

comment\_post = Post.objects.get(id=3),

comment\_user = Author.objects.get(id=2).author, feedback\_text = "Wow, that's awesome"

)

Comment.objects.create(

comment\_post=Post.objects.get(id=1),

comment\_user = Author.objects.get(id=2).author, feedback\_text = 'It is okay'

)

7. Применяя функции like() и dislike() к статьям/новостям и комментариям, скорректировать рейтинги этих объектов.

Comment.objects.get(id=1).like()

Post.objects.get(id=1).dislike()

Post.objects.get(id=3).like()

проверка рейтинга пользователя

Comment.objects.get(id=1).comment\_rate

проверка рейтинга поста

Post.objects.get(id=1).post\_rate

8. Обновить рейтинги пользователей.

u1 = Author.objects.get(id=1)

u1.update\_rating()

u1.user\_rate

>>> u2 = Author.objects.get(id=2)

#чтобы посмотреть суммарный рейтинг комментариев пользователя, можно обратиться к нему напрямую

>>> u2.author.comment\_set.aggregate(comment\_rating=Sum('comment\_rate'))

>>> u2.update\_rating()

>>> u2.user\_rate

9. Вывести username и рейтинг лучшего пользователя (применяя сортировку и возвращая поля первого объекта).

>>> s = Author.objects.order\_by('user\_rate')

>>> for i in s:

... i.user\_rate

... i.author.username

1

'Ivan Ivanov'

60

'Petr Petrov'

10. Вывести дату добавления, username автора, рейтинг, заголовок и превью лучшей статьи, основываясь на лайках/дислайках к этой статье.

>>> p = Post.objects.order\_by('-post\_rate')

for i in p[:1]:

... i.date\_created

... i.post\_author.author

... i.post\_rate

... i.title

... i.preview()

11. Вывести все комментарии (дата, пользователь, рейтинг, текст) к этой статье.

>>> Post.objects.all().order\_by('-post\_rate')[0].comment\_set.values(

'comment\_date\_created',

'comment\_user',

'comment\_rate', 'feedback\_text'

)

В консоли выведутся комментарии:

<QuerySet [

{

'comment\_date\_created': datetime.date(2021, 5, 11),

'comment\_user': 3,

'comment\_rate': 1,

'feedback\_text': 'Very interesting article'

},

{

'comment\_date\_created': datetime.date(2021, 5, 11),

'comment\_user': 4,

'comment\_rate': 0,

'feedback\_text': 'It is okay'

},

{

'comment\_date\_created': datetime.date(2021, 5, 12),

'comment\_user': 3,

'comment\_rate': 8,

'feedback\_text': 'Very interesting article'

},

{

'comment\_date\_created': datetime.date(2021, 5, 12), 'comment\_user': 4,

'comment\_rate': 8,

'feedback\_text': 'It is okay'

}

]