**Что вы должны сделать в консоли *Django*?**

from news.models import \*

* 1. Создать двух пользователей (с помощью метода User.objects.create\_user(‘username’)).

u1 = User.objects.create\_user(username='Alex')

u2 = User.objects.create\_user(username='Ivan')

* 1. Создать два объекта модели *Author*, связанные с пользователями.

Author.objects.create(authorUser = u1)

Author.objects.create(authorUser = u2)

* 1. Добавить 4 категории в модель *Category.*

Category.objects.create(name\_category='Human')

Category.objects.create(name\_category ='Planet')

Category.objects.create(name\_category ='Computers')

Category.objects.create(name\_category ='Different')

* 1. Добавить 2 статьи и 1 новость.

author1=Author.objects.get(id=1)

Post.objects.create(author=author1, categoryType='NW', title='Human is a good person', text='I think, human is a good. But not all of them. But may be all of them')

Post.objects.create(author=author1, categoryType='AR', title='All about Mars', text='There are many planets in our world. And now a tell you about Mars. Not Snickers. About Mars.')

author2=Author.objects.get(id=2)

Post.objects.create(author=author2, categoryType='AR', title='PC or not PC', text='Personal computer is very nice thing. Any person in our country must have him.')

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

Post.objects.get(id=1).postCategory.add(Category.objects.get(id=1))

Post.objects.get(id=2).postCategory.add(Category.objects.get(id=2))

Post.objects.get(id=3).postCategory.add(Category.objects.get(id=3), Category.objects.get(id=2))

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

Comment.objects.create(commentPost=Post.objects.get(id=1), commentUser=Author.objects.get(id=1).authorUser, text='This is true')

Comment.objects.create(commentPost=Post.objects.get(id=2), commentUser=Author.objects.get(id=1).authorUser, text='This is not true')

Comment.objects.create(commentPost=Post.objects.get(id=3), commentUser=Author.objects.get(id=2).authorUser, text='Oh! I can not believe in this!')

Comment.objects.create(commentPost=Post.objects.get(id=1), commentUser=Author.objects.get(id=2).authorUser, text='Author is great pearson!')

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

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

Comment.objects.get(id=2).dislike()

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

Comment.objects.get(id=4).like()

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

Post.objects.get(id=2).like()

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

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

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

* 1. Обновить рейтинги пользователей.

Author.objects.get(id=1).update\_rating()

Author.objects.get(id=2).update\_rating()

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

a = Author.objects.order\_by('-ratingAuthor')[:1]

a[0].ratingAuthor

a[0].authorUser.username

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

p = Post.objects.order\_by('-rating')[:1]

p[0].dateCreation.date()

p[0].author.authorUser.username

p[0].rating

p[0].title

p[0].preview()

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

c = Comment.objects.all()

for i in c:

i.dateCreation.data()

i.commentUser.username

i.rating

i.text