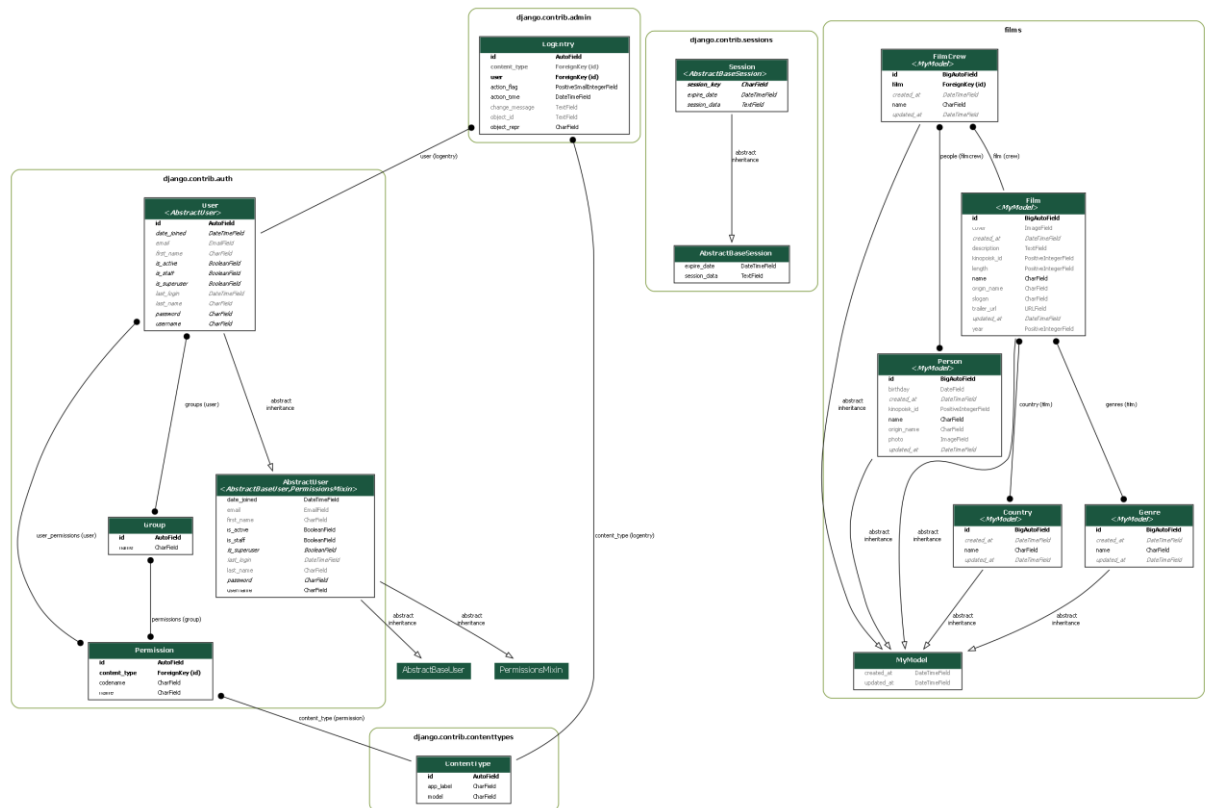


Б23-215 Аллабергенов Максат

Задание 8.2 Реализовать возможность добавления полного списочного состава съемочной группы (режиссеры, продюсеры, сценаристы и др.). Необходимо также реализовать возможность наличия у фильма несколько стран-производителей и нескольких жанров

1) Диаграмма классов:



2) Пояснение изменений:

Была добавлена модель FilmCrew, отвечающая за состав съемочной группы. Т.е. в данной модели хранится информация о профессии, людей данной профессии и информация к какому фильму относятся члены данной группы

Были изменены отношения между фильмом и страной, и фильмом и жанром на многие ко многим для отображения в информации о фильме нескольких стран производителей и жанров.

Были изменены шаблоны:

- 1) `person/detail.html` (представление всех связанных с персоной фильмами, а также профессии в фильме)

```
{% if person.filmcrew_set.all %}
    {% for prof in person.filmcrew_set.all %}
        {% ifchanged %}
<dt class="col-md-3 text-md-end">
            {{ prof.name.title }}
        </dt>
<dd class="col-md-9">
            <ol>
                {% for crew in person.filmcrew_set.all %}
                    {% if crew.name == prof.name %}
```

```

        <li>
            <a href = "{% url 'films:film_detail'
crew.film.id %}">{{crew.film.name}}</a>
        </li>
    {% endif %}
{% endfor %}
</ol>
</dd>
{% endifchanged %}
{% endfor %}

```

- 2) film/detail.com (представление нескольких стран и жанров и представление всей съемочной группы)

```

{% if film.crew.all %}
    {% for prof in film.crew.all %}
        <dt class="col-md-3 text-md-end">
            {{prof.name.title}}
        </dt>
        <dd class="col-md-9">
            {% for person in prof.people.all %}
                <a href="{% url 'films:person_detail' person.id %}">{{
person.name }}</a>{% if not forloop.last %}, {% endif %}
            {% endfor %}
            {% if user.is_superuser %}
                <a href="{% url 'films:crew_update' id=prof.id %}"
title="Редактировать съемочную группу" class="btn btn-primary"><i
class="bi-pencil-square"></i></a>
                <a href="{% url 'films:crew_delete' id=prof.id %}"
title="Удалить съемочную группу" class="btn btn-danger"><i class="bi-
trash"></i></a>
            {% endif %}
        </dd>
    {% endfor %}
{% endif %}
{% if user.is_superuser %}
    <dd class="col-md-3 text-md-end">
        <a href="{% url 'films:crew_create' id=film.id %}"
title="Добавить съемочную группу" class="btn btn-primary"><i class="bi-
plus-lg"></i></a>
    </dd>

```

Добавлена форма создания объекта FilmCrew:

```

class CrewForm(forms.ModelForm):
    class Meta:
        model = FilmCrew
        fields = ['film', 'name', 'people']
        widgets = {
            'people': autocomplete.ModelSelect2Multiple(
                url='films:people_autocomplete'),

```

```
        'film' : forms.HiddenInput(),
    }
```

Добавлены шаблоны:

1) создания FilmCrew

```
{% extends 'films/base.html' %}
{% load django_bootstrap5 %}
{% load films_tags %}

{% block breadcrumb %}
    <nav>
        <ol class="breadcrumb">
            <li class="breadcrumb-item">
                <a href="{% url 'films:film_list' %}">{{
'films:film'|model_verbosename_plural }}</a>
            </li>
            <li class="breadcrumb-item">
                <a href="{% url 'films:film_detail' id=form.instance.id %}">{{
form.instance.id }}</a>
            </li>
            <li class="breadcrumb-item active">Добавление съемочной группы</li>
        </ol>
    </nav>
{% endblock %}

{% block content %}
    <h1>Добавление съемочной группы</h1>
    <form method="POST" enctype="multipart/form-data">
        {% csrf_token %}
        {% bootstrap_form form %}
        {{ form.media }}
        {% bootstrap_button button_type='submit' content='Сохранить' %}
    </form>
{% endblock %}
```

2) редактирования FilmCrew

```
{% extends 'films/base.html' %}
{% load django_bootstrap5 %}
{% load films_tags %}

{% block breadcrumb %}
    <nav>
        <ol class="breadcrumb">
            <li class="breadcrumb-item">
                <a href="{% url 'films:film_list' %}">{{
'films:film'|model_verbosename_plural }}</a>
            </li>
            <li class="breadcrumb-item">
                <a href="{% url 'films:film_detail' form.instance.film.id %}">{{
form.instance.film.id }}</a>
            </li>
        </ol>
    </nav>
{% endblock %}
```

```

        </li>
        <li class="breadcrumb-item active">Редактирование съёмочной
группы</li>
    </ol>
</nav>
{% endblock %}
{% block content %}
    <h1>Редактирование съёмочной группы</h1>
    <form method="POST" enctype="multipart/form-data">
        {% csrf_token %}
        {% bootstrap_form form %}
        {{ form.media }}
        {% bootstrap_button button_type='submit' content='Сохранить' %}
    </form>
{% endblock %}

```

3) удаления FilmCrew

```

{% extends 'films/base.html' %}
{% load films_tags %}
{% load django_bootstrap5 %}

{% block breadcrumb %}
    <nav>
        <ol class="breadcrumb">
            <li class="breadcrumb-item">
                <a href="{% url 'films:film_list' %}">{{
'films:film'|model_verbos_name_plural }}</a>
            </li>
            <li class="breadcrumb-item">
                <a href="{% url 'films:film_detail' crew.film.id %}">{{
crew.film.name }}</a>
            </li>
            <li class="breadcrumb-item active">Удаление</li>
        </ol>
    </nav>
{% endblock %}

{% block content %}
    <div class="alert alert-danger">
        <h4 class="alert-heading">Вы уверены?</h4>
        <p>Вы уверены, что хотите безвозвратно удалить съёмочную группу «{{
crew.name }}»?</p>
        <form method="POST">
            {% csrf_token %}
            {% bootstrap_button 'Да, удалить!' button_type='submit'
button_class='btn-primary' %}
            <a href="{% url 'films:film_detail' crew.film.id %}" class="btn btn-
light ms-4">Нет</a>
        </form>
    </div>

```

```
</div>
{% endblock %}
```

Соответственно, были добавлены представления к соответствующим шаблонам в файл views.py

```
@user_passes_test(check_admin)
def crew_create(request, id):
    film = get_object_or_404(Film, id=id)
    if request.method == 'POST':
        form = CrewForm(request.POST, request.FILES, initial={'film': film})
        if form.is_valid():
            form.save()
            messages.success(request, 'Съемочная группа создана')
            return redirect('films:film_detail', id=film.id)
    else:
        form = CrewForm(instance=film, initial={'film': film, 'name': None})
    return render(request, 'films/film/crew_create.html',
                  {'form': form})

@user_passes_test(check_admin)
def crew_update(request, id):
    crew = get_object_or_404(FilmCrew, id=id)
    if request.method == 'POST':
        form = CrewForm(request.POST, request.FILES, instance=crew)
        if form.is_valid():
            form.save()
            messages.success(request, 'Съемочная группа изменена')
            return redirect('films:film_detail', id=crew.film.id)
    else:
        form = CrewForm(instance=crew)
    return render(request, 'films/film/crew_update.html',
                  {'form': form})

@user_passes_test(check_admin)
def crew_delete(request, id):
    crew = get_object_or_404(FilmCrew, id=id)
    film = get_object_or_404(Film, id=crew.film.id)
    if request.method == 'POST':
        crew.delete()
        messages.success(request, 'Съемочная группа удалена')
        return redirect('films:film_detail', id=film.id)
    return render(request, 'films/film/crew_delete.html',
                  {'crew': crew})
```