МГТУ им. Н.Э. Баумана

Факультет «Информатика и системы управления»

ДИСЦИПЛИНА:

«Разработка интернет-приложений»

Отчёт по лабораторной работе №4-5

|  |
| --- |
| Выполнил:  Студент 3 курса  Факультет ИУ  Группа  ИУ5-52  Миронов С.В.  Преподаватель  Гапанюк Ю.Е. |

**Задание.**

* Создать проект
* Реализовать view, в которых генерируются html-страницы
* В шаблонах должны быть использованы рассмотренные конструкции: переменные, вложенные значения, циклы, условия
* Все шаблоны должны расширять базовый шаблон
* Для элементов списка использовать тег include
* По нажатии на элемент списка должна открываться страница информации об элементе

**Код программы**

Mysite/urls.py

**from** django.contrib **import** admin  
**from** django.urls **import** path  
**from** polls.views **import \*  
from** django.views.generic **import** ListView, DetailView  
**from** polls.models **import \*  
from** django.conf **import** settings  
**from** django.conf.urls.static **import** static  
**from** django.conf.urls **import** include  
  
urlpatterns **=** [  
 path('admin', admin.site.urls),  
 path('addform', Add.as\_view()),  
 path('register', RegisterFormView.as\_view(), name**=**'reg'),  
 path('login', LoginFormView.as\_view(), name**=**'login'),  
 path('lul', LogoutView.as\_view(), name**=**'exit'),  
 path('create', create),  
 path('profile', update\_profile, name**=**'profile'),  
 path('', GamesList.as\_view(), name**=**'games\_all'),  
 path('games/<int:id>', GameView.as\_view(), name**=**'game'),  
 path('api-auth/', include('rest\_framework.urls', namespace**=**'rest\_framework')),  
 path('gamesAPI', GamesAPIView.as\_view(), name**=**'gameAPI-list'),  
 path('gamesAPI/<int:pk>', GameAPIView.as\_view(), name**=**'gameAPI'),  
]

polls/admin.py

**from** django.contrib **import** admin  
**from** polls.models **import \****# Register your models here.*admin.site.register(Games)  
admin.site.register(Profile)

polls/models.py

**from** django.db **import** models  
**from** django.contrib.auth.models **import** User  
**from** django.db.models.signals **import** post\_save  
**from** django.dispatch **import** receiver  
**import** os  
**from** django.conf **import** settings

**class Profile**(models.Model)**:** user**=**models.OneToOneField(User, on\_delete**=**models.CASCADE)  
 age**=** models.IntegerField(blank**=**True, null**=**True)  
 image**=**models.ImageField(null**=**True, default **=** os.path.join('media/img/extra.png'), upload\_to**=**"img/users")  
 **def \_\_str\_\_**(self)**:  
 return** self.user.username  
  
 **@receiver**(post\_save, sender**=**User)  
 **def create\_user\_profile**(sender, *instance*, *created*, **\*\***kwargs)**:  
 if** *created***:** Profile.objects.create(user**=***instance*)  
  
 **@receiver**(post\_save, sender**=**User)  
 **def save\_user\_profile**(sender, *instance*, **\*\***kwargs)**:** *instance*.profile.save()  
  
**class Games**(models.Model)**:** name**=**models.CharField(max\_length**=**100)  
 platform**=**models.CharField(max\_length**=**100)  
 date**=**models.DateField()  
 genre**=**models.CharField(max\_length**=**100)  
 rating**=**models.IntegerField()  
 image**=**models.ImageField(upload\_to**=**"img/")  
 info**=**models.TextField(default**=**"")  
  
 **def \_\_str\_\_**(self)**:  
 return** self.name

polls/views.py

**from** django.shortcuts **import** render  
**from** django.views.generic **import** TemplateView,ListView  
**from** polls.models **import \*  
from** django.http **import** HttpResponseRedirect  
**from** django.core.paginator **import** Paginator  
**from** django.contrib.auth.decorators **import** login\_required  
**from** django.db **import** transaction  
**from** polls.forms **import \*  
from** django.views.generic.edit **import** FormView  
**from** django.contrib.auth.forms **import** UserCreationForm  
**from** django.contrib.auth.forms **import** AuthenticationForm  
**from** django.contrib.auth **import** login  
**from** django.views.generic.base **import** View  
**from** django.contrib.auth **import** logout  
**from** django.contrib.auth.models **import** User  
**from** django.contrib **import** messages  
**from** polls.serializers **import \*  
from** rest\_framework **import** generics  
  
**class LogoutView**(View)**:  
 def get**(self, *request*)**:** logout(*request*)  
 **return** HttpResponseRedirect("/")  
  
**class LoginFormView**(FormView)**:** form\_class **=** AuthenticationForm  
 template\_name **=** "login.html"  
 success\_url **=** "/"  
  
 **def form\_valid**(self, *form*)**:** User.profile **=** property(**lambda** *u***:** Profile.objects.get\_or\_create(user**=**u)[0])  
 self.user **=** *form*.get\_user()  
 login(self.request, self.user)  
 **return** super(LoginFormView, self).form\_valid(*form*)  
  
**class RegisterFormView**(FormView)**:** form\_class **=** UserCreationForm  
 success\_url **=** "/profile/"  
 template\_name **=** "register.html"  
 **def form\_valid**(self, *form*)**:** *form*.save()  
 **return** super(RegisterFormView, self).form\_valid(*form*)  
  
  
  
**class GameView**(TemplateView)**:  
 def get**(self, **\***args, *\*\*kwargs*)**:** data **=** {  
 'games'**:** Games.objects.filter(id**=***kwargs*['id']).first()  
 }  
 **return** render(self.request, 'address.html', data)  
  
**class GamesList**(ListView)**:** model**=**Games  
 template\_name**=**'games.html'  
 paginate\_by**=**3  
 queryset**=**Games.objects.all().order\_by("-date")  
  
**class Add**(TemplateView)**:  
 def get**(self, *request*)**:  
 return** render(*request*, 'addform.html')  
  
**class Profile\_View**(TemplateView)**:  
 def get**(self, *request*)**:  
 return** render(*request*, 'profile.html')  
  
**def create**(*request*)**:  
 if** *request*.method **==** "POST"**:** tom **=** Games()  
 tom.name **=** *request*.POST.get("name")  
 tom.platform **=** *request*.POST.get("platform")  
 tom.date **=** *request*.POST.get("date")  
 tom.genre **=** *request*.POST.get("genre")  
 tom.rating **=** *request*.POST.get("rating")  
 tom.image **=** *request*.FILES['image']  
 tom.info **=** *request*.POST.get("info")  
 tom.save()  
 **return** HttpResponseRedirect("/")  
 **return** HttpResponseRedirect("addform")  
  
**@login\_required  
@transaction.atomic  
def update\_profile**(*request*)**:  
 if** *request*.method **==** 'POST'**:** user\_form **=** UserForm(*request*.POST, instance**=***request*.user)  
 profile\_form **=** ProfileForm(*request*.POST,*request*.FILES, instance**=***request*.user.profile)  
 **if** user\_form.is\_valid() **and** profile\_form.is\_valid()**:** user\_form.save()  
 profile\_form.save()  
 '''messages.success(request, ('Your profile was successfully updated!'))'''  
 **return** HttpResponseRedirect('profile')  
 **else:** '''messages.error(request, ('Please correct the error below.'))'''  
 **else:** user\_form **=** UserForm(instance**=***request*.user)  
 profile\_form **=** ProfileForm(instance**=***request*.user.profile)  
 **return** render(*request*, 'profile.html', {  
 'user\_form'**:** user\_form,  
 'profile\_form'**:** profile\_form  
 })

**Шаблоны**

games.html

{%extends "base.html" %}  
{%block head\_name%}Игры{%endblock%}  
  
{%block body%}  
<**div** class=**"row text-center"**>  
 {% for game in object\_list %}  
 <**div** class=**"col-md-3 col-sm-3 col-xs-12 m-1 jumbotron"**>  
 <**a** href=**"/games/{{game.id}}"**><**h3**>{{game.name}}</**h3**></**a**>  
 <**div** class=**"cover-image-list-game"**>  
 <**a** href=**"/games/{{game.id}}"**>  
 <**img** class=**"img-fluid"** src=**"{{game.image.url}}"**, title=**"{{game.name}}"**>  
 </**a**>  
 </**div**>  
 <**div** class=**"game-info"**>  
 <**p** class=**"data-relise"**>  
 <**span** class=**"title-data-relise"**>Дата выхода:</**span**>  
 <**span** class=**"content-data-relise"**>{{game.date|date:"d-m-Y"}}</**span**>  
 </**p**>  
 <**p** class=**"game-genre"**>  
 <**span** class=**"genre-label"**>Жанр:</**span**>  
  
 <**span** class=**"genre-list"**>{{game.genre}}</**span**>  
 </**p**>  
 <**div** class=**"game-platform"**>  
 <**span** class=**"platform-label"**>Платформа:</**span**>  
 <**span** class=**"platform-list"**>{{game.platform}}</**span**>  
 </**div**>  
 </**div**>  
 <**div** class=**"rating-site"**>  
 <**div** class=**"w-rating"**>  
 <**div** class=**"rating-label"**>Рейтинг</**div**>  
 <**div** class=**"rating-search"** style=**"**color: #ffa820**"**>{{game.rating}}</**div**>  
  
 </**div**>  
 <**a** href=**"/games/{{game.id}}"**>Подробнее</**a**>  
 </**div**>  
 </**div**>  
 {% endfor %}  
</**div**>  
  
{% if is\_paginated %}  
</**nav** aria-label=**"games"**>  
 <**ul** class=**"pagination justify-content-center"**>  
 {% if page\_obj.has\_previous %}  
 <**li** class=**"page-item"**><**a** class=**"page-link"** href=**"?page={{ page\_obj.previous\_page\_number }}"**>**&laquo;**</**a**></**li**>  
 {% else %}  
 <**li** class=**"disabled page-item"**><**span** class=**"page-link"**>**&laquo;**</**span**></**li**>  
 {% endif %}  
 {% for i in paginator.page\_range %}  
 {% if page\_obj.number == i %}  
 <**li** class=**"active page-item"**><**span** class=**"page-link"**>{{ i }} <**span** class=**"sr-only"**>(current)</**span**></**span**></**li**>  
 {% else %}  
 <**li** class=**"page-item"**><**a** class=**"page-link"** href=**"?page={{ i }}"**>{{ i }}</**a**></**li**>  
 {% endif %}  
 {% endfor %}  
 {% if page\_obj.has\_next %}  
 <**li** class=**"page-item"**><**a** class=**"page-link"** href=**"?page={{ page\_obj.next\_page\_number }}"**>**&raquo;**</**a**></**li**>  
 {% else %}  
 <**li** class=**"disabled page-item"**><**span** class=**"page-link"**>**&raquo;**</**span**></**li**>  
 {% endif %}  
 </**ul**>  
</**nav**>  
{% endif %}  
{% endblock %}

base.html

<!DOCTYPE html>  
<**html** lang=**"eng"**>  
<**head**>  
 <**meta** charset=**"UTF-8"**>  
 <**title**>{%block head\_name%}{%endblock%}</**title**>  
 {% load staticfiles %}  
 <**link** rel=**"stylesheet"** href=**"{% static 'css/bootstrap.min.css' %}"** type=**"text/css"**>  
 <**link** rel=**"stylesheet"** href=**"https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"** integrity=**"sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO"** crossorigin=**"anonymous"**>  
 <**script** src=**"https://code.jquery.com/jquery-3.3.1.slim.min.js"** integrity=**"sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"** crossorigin=**"anonymous"**></**script**>  
 <**script** src=**"https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js"** integrity=**"sha384-ZMP7rVo3mIykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnjAK/l8WvCWPIPm49"** crossorigin=**"anonymous"**></**script**>  
 <**script** src=**"https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"** integrity=**"sha384-ChfqqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwEULTy"** crossorigin=**"anonymous"**></**script**>  
</**head**>  
<**body**>  
 <**nav** class=**"navbar navbar-dark bg-dark"**>  
 <**a** class=**"navbar-brand"** href=**"{% url 'games\_all' %}"**>  
 QWE  
 </**a**>  
 <**ul** class=**"nav nav-pills"**>  
 <**li** class=**"nav-item"**>  
 <**a** class=**"nav-link"** data-toggle=**"modal"** data-target=**"#addmodal"** href=**"##"**>Add Game</**a**>  
 </**li**>  
 {% if not user.is\_authenticated %}  
 <**li** class=**"nav-item"**>  
 <**a** class=**"nav-link"** href=**"{% url 'reg' %}"**>Регистрация</**a**>  
 </**li**>  
 {% endif %}  
 {% if not user.is\_authenticated %}  
 <**li** class=**"nav-item"**>  
 <**a** class=**"nav-link"** href=**"{% url 'login' %}"**>Войти</**a**>  
 </**li**>  
 {% endif %}  
 {% if user.is\_authenticated %}  
 <**li** class=**"nav-item"**>  
 <**a** class=**"nav-link"** href=**"##"** >Добро пожаловать, {{user.username}} </**a**>  
 </**li**>  
 {% endif %}  
 {% if user.is\_authenticated %}  
 <**li** class=**"nav-item"**>  
 <**a** class=**"nav-link"** href=**"{% url 'profile' %}"**>Личная страница</**a**>  
 </**li**>  
 {% endif %}  
 {% if user.is\_authenticated %}  
 <**li** class=**"nav-item"**>  
 <**a** class=**"nav-link"** href=**"{% url 'exit' %}"**>Выход</**a**>  
 </**li**>  
 {% endif %}  
 </**ul**>  
 </**nav**>  
  
 {%block body %}{%endblock%}  
  
 <**div** class=**"modal fade"** id=**"addmodal"** tabindex=**"-1"** role=**"dialog"** aria-labelledby=**"addmodal"** aria-hidden=**"true"**>  
 <**div** class=**"modal-dialog"** role=**"form"**>  
 <**div** class=**"modal-content"**>  
 <**div** class=**"modal-header"**>  
 <**h5** class=**"modal-title"**>Добавление игры в список</**h5**>  
 <**button** class=**"close"** type=**"button"** data-dismiss=**"modal"** aria-label=**"Close"**><**span** aria-hidden=**"true"**>&times</**span**> </**button**>  
 </**div**>  
 <**div** class=**"modal-body"** id=**"modal-body"**>  
  
 <**form** enctype=**"multipart/form-data"** action=**"create"** method=**"post"** id=**"games\_form"**>  
 <**input** type=**"hidden"** name=**"csrfmiddlewaretoken"** value=**"jyOu3BjOiCWxeBKOqObUBSDWIfVHjMwo9acbjfhYJAXIWd0n95QBgOjqElXrWtZQ"**>  
 <**div** class=**"form-group"**>  
 <**fieldset** class=**"center-block"**>  
 <**div** class=**"form-row field-name"**>  
 <**div**>  
 <**label** class=**"required"** for=**"id\_name"**>Название:</**label**>  
 <**input** type=**"text"** name=**"name"** class=**"vTextField form-control"** maxlength=**"100"** required=**""** id=**"id\_name"**>  
 </**div**>  
 </**div**>  
 <**div** class=**"form-row field-platform"**>  
 <**div**>  
 <**label** class=**"required"** for=**"id\_platform"**>Платформа:</**label**>  
 <**input** type=**"text"** name=**"platform"** class=**"vTextField form-control"** maxlength=**"100"** required=**""** id=**"id\_platform"**>  
 </**div**>  
 </**div**>  
 <**div** class=**"form-row field-date m-2"**>  
 <**div**>  
 <**label** class=**"required"** for=**"id\_date"**>Дата выхода:</**label**>  
 <**input** type=**"date"** class=**"form-control"** name=**"date"** required=**""**>  
 </**div**>  
 </**div**>  
 <**div** class=**"form-row field-genre"**>  
 <**div**>  
 <**label** class=**"required"** for=**"id\_genre"**>Жанры:</**label**>  
 <**input** type=**"text"** name=**"genre"** class=**"vTextField form-control"** maxlength=**"100"** required=**""** id=**"id\_genre"**>  
 </**div**>  
 </**div**>  
 <**div** class=**"form-row field-rating"**>  
 <**div**>  
 <**label** class=**"required"** for=**"id\_rating"**>Рейтинг:</**label**>  
 <**input** type=**"number"** name=**"rating"** class=**"vIntegerField form-control"** required=**""** id=**"id\_rating"**>  
 </**div**>  
 </**div**>  
 <**div** class=**"form-row field-image m-4"**>  
 <**div**>  
 <**label** class=**"required"** for=**"id\_image"**>Изображение:</**label**>  
 <**input** type=**"file"** name=**"image"** accept=**"image/\*"** required=**""** id=**"id\_image"**>  
 </**div**>  
 </**div**>  
 <**div** class=**"form-row field-info"**>  
 <**div**>  
 <**label** class=**"required"** for=**"id\_info"**>Описание:</**label**>  
 <**textarea** name=**"info"** cols=**"40"** rows=**"10"** class=**"vLargeTextField form-control"** required=**""** id=**"id\_info"**></**textarea**>  
 </**div**>  
 </**div**>  
 </**fieldset**>  
  
 <**div** class=**"submit-row"**>  
 <**input** type=**"submit"** value=**"Сохранить"** class=**"default"** name=**"\_save"**>  
 </**div**>  
  
 </**div**>  
 </**form**>  
 </**div**>  
 <**br** class=**"clear"**>  
 </**div**>  
 </**div**>  
 </**div**>  
</**body**>  
</**html**>

address.html

{%extends "base.html" %}  
{%block head\_name%}{{games.name}}{%endblock%}  
{%block title%}{{games.name}}{% endblock %}  
  
{%block body%}  
<**div** class=**"jumbotron"**>  
 <**h2** class=**"text-info"**>{{games.name}}</**h2**>  
 <**img** src=**"{{games.image.url}}"**, title=**"{{game.name}}"**>  
 <**h6** class=**"text-info"**>{{games.date|date:"d-m-Y"}}</**h6**>  
 <**p**>  
 <**span**>Жанр: {{game.genre}}</**span**>  
 </**p**>  
 <**div**>  
 <**div**>Рейтинг:</**div**>  
 <**div** style=**"**color: #ffa820**"**>{{games.rating}}</**div**>  
  
 </**div**>  
 <**div**>  
 <**span**>Платформа:</**span**>  
 <**span**>{{games.platform}}</**span**>  
 </**div**>  
 <**p**>Описание: {{games.info|safe|linebreaks}}</**p**>  
</**div**>  
{% endblock %}

…

Результаты







