

Отчет по РК № 2 по курсу
"Разработка Интернет-Приложений"

Выполнил:

Студент группы

ИУ5-55Б

Смыслов Максим

Москва, МГТУ – 2021

Задание:

Рубежный контроль представляет собой разработку веб-приложения с использованием фреймворка Django. Веб-приложение должно выполнять следующие функции:

1. Создайте проект Python Django с использованием стандартных средств Django.
2. Создайте модель Django ORM, содержащую две сущности, связанные отношением один-ко-многим в соответствии с Вашим вариантом из условий рубежного контроля №1.
3. С использованием стандартного механизма Django сгенерируйте по модели макет веб-приложения, позволяющий добавлять, редактировать и удалять данные.
4. Создайте представление и шаблон, формирующий отчет, который содержит соединение данных из двух таблиц.

№ варианта	Класс 1	Класс 2
13	Книга	Библиотека

Текст программы:

rk2/urls.py

```
urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.index, name="main"),
    path('book/', include([
        path('', views.read_book, name='read_book'),
        path('create/', views.create_book, name="create_book"),
        path('update/<int:book_id>/', views.update_book, name="update_book"),
        path('delete/<int:book_id>/', views.delete_book, name="delete_book"),
    ])),
    path('library/', include([
        path('', views.read_library, name='read_library'),
        path('create/', views.create_library, name="create_library"),
        path('update/<int:library_id>/',
            views.update_library, name="update_library"),
        path('delete/<int:library_id>/',
            views.delete_library, name="delete_library"),
    ])),
    path('report/', views.report, name="report"),
]
```

books/models.py

```

from django.db import models

✓ class Library(models.Model):
    name = models.CharField(max_length=256, verbose_name="Library name")

✓    def __str__(self):
        return self.name

✓ class Book (models.Model):
    name = models.CharField(max_length=256, verbose_name="Book name")
    cost = models.PositiveIntegerField(verbose_name="Book cost")
✓    library = models.ForeignKey(
        Library,
        on_delete=models.SET_DEFAULT,
        null=True,
        default=None,
        related_name="books"
    )

✓    def __str__(self):
        return self.name

```

books/views.py

```

from django.db.models.aggregates import Avg
from django.http.response import HttpResponse
from django.shortcuts import get_list_or_404, get_object_or_404, redirect,
render
from .models import Library, Book
from django.db import models

def index(request):
    return render(request, 'index.html')

# BOOKS

def read_book(request):
    books = Book.objects.all()
    return render(request, 'book/book_list.html', {'books': books})

def create_book(request):
    if request.method == 'GET':
        libraries = Library.objects.all()
        return render(request, 'book/create_book.html', {"libraries":
libraries})
    else:
        dto = {}
        for key in request.POST:
            if key in Book.__dict__:

```

```
        dto[key] = request.POST[key]
        dto['library'] = get_object_or_404(Library,
pk=request.POST['library'])
        new_book = Book(**dto)
        new_book.save()
        return redirect('read_book')

def update_book(request, book_id):
    if request.method == 'GET':
        libraries = Library.objects.all()
        book = get_object_or_404(Book, pk=book_id)
        return render(request, 'book/update_book.html', {"book": book,
"libraries": libraries})
    else:
        book = get_object_or_404(Book, pk=book_id)
        for key in request.POST:
            if key in book.__dict__ and key != 'library':
                setattr(book, key, request.POST[key])
        if 'library' in request.POST:
            setattr(book, 'library', get_object_or_404(
                Library, pk=request.POST['library']))
        book.save()
        return redirect('read_book')

def delete_book(request, book_id):
    book = get_object_or_404(Book, pk=book_id)
    book.delete()
    return redirect(request.META.get('HTTP_REFERER'))

# libraries

def read_library(request):
    libraries = Library.objects.all()
    return render(request, 'library/library_list.html', {'libraries':
libraries})

def create_library(request):
    if request.method == 'GET':
        return render(request, 'library/create_library.html')
    else:
        new_library = Library(name=request.POST['name'])
        new_library.save()
        return redirect('read_library')

def update_library(request, library_id):
    if request.method == 'GET':
```

```

        library = get_object_or_404(Library, pk=library_id)
        return render(request, 'library/update_library.html', {"library":
library})
    else:
        library = get_object_or_404(Library, pk=library_id)
        for key in request.POST:
            if key in library.__dict__:
                setattr(library, key, request.POST[key])
        library.save()
        return redirect('read_library')

```

```

def delete_library(request, library_id):
    library = get_object_or_404(Library, pk=library_id)
    library.delete()
    return redirect(request.META.get('HTTP_REFERER'))

```

```

# REPORT

```

```

def report(request):
    expensive_books = Book.objects.filter(cost__gt=1000)
    avg_prices = []
    for library in Library.objects.all():
        avg_prices.append({"library": library, "price": Book.objects.filter(
            library=library.pk).aggregate(Avg('cost'))['cost__avg']})
    return render(request, 'report.html', {"expensive_books": expensive_books,
"avg_prices": avg_prices})

```

templates/inc/base.html

```

{% load static %}

```

```

<!doctype html>
<html lang="en" class="h-100">

<head>
    <!-- ===== CHARSET AND VIEWPORT ===== -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-
to-fit=no">
    <!-- ===== CACHE TURN OFF ===== -->
    <meta http-equiv="cache-control" content="no-cache">
    <meta http-equiv="expires" content="0">
    <!-- ===== COPY RIGHT ===== -->
    <meta name="author" content="ИУ5-55 Смыслов Максим">
    <title>
        ИУ5-55 Смыслов Максим РК-2
    </title>

```

```

<!-- ===== BOOTSTRAP CORE CSS ===== -->
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
1BmE4kWbQ78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous">
<!-- ===== SHARED STYLES ===== -->
<link rel="stylesheet" href="{% static 'css/navbar.css' %}">
</head>

<body class="d-flex flex-column h-100">
  <header>
    <!-- Navbar -->
    {% include 'inc/navbar.html' %}
  </header>

  <!-- Begin page content -->
  <main role="main" class="flex-shrink-0">
    <div class="container">
      <div class="row">
        <div class="col-md-12">
          <!-- Content -->
          {% block content %}
          {% endblock content %}
        </div>
      </div>
    </div>
  </main>

  <footer class="footer mt-auto py-3">
    <!-- Footer -->
  </footer>

  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
  integrity="sha384-
ka7Sk0GlN4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
crossorigin="anonymous"></script>
</html>

```

templates/inc/navbar.html

```

{% load static %}
<nav class="navbar navbar-expand-sm navbar-dark bg-dark p-3">
  <a class="navbar-brand ml-5" href="{% url 'main' %}">Library</a>

  <ul class="navbar-nav ml-auto">
    <li class="nav-item">
      <a href="{% url 'main' %}" class="nav-link">Main Page</a>
    </li>
  </ul>

```

```

<li class="nav-item">
  <a href="{% url 'read_book' %}" class="nav-link">All books</a>
</li>
<li class="nav-item">
  <a href="{% url 'create_book' %}" class="nav-link">Create book</a>
</li>
<li class="nav-item">
  <a href="{% url 'read_library' %}" class="nav-link">All librarys</a>
</li>
<li class="nav-item">
  <a href="{% url 'create_library' %}" class="nav-link">Create
library</a>
</li>
<li class="nav-item">
  <a href="{% url 'report' %}" class="nav-link">Report</a>
</li>
</ul>
</nav>

```

templates/book/book_card.html

```

{% load static %}

<div class="card mb-3">
  <div class="row g-0">
    <div class="col-md-4">
      
    </div>
    <div class="col-md-8">
      <div class="card-body">
        <h5 class="card-title">{{ book.name }}</h5>
        <p class="card-text">{{ book.library.name }}</p>
        <p class="card-text">
          This is a wider card with supporting text below as a
natural lead-in to additional content. This
          content is a little bit longer.
        </p>
        <p class="card-text"><small class="text-muted">{{ book.cost
}}</small></p>
        <a href="{% url 'delete_book' book.pk %}" class="btn btn-
danger"> Delete </a>
        <a href="{% url 'update_book' book.pk %}" class="btn btn-
primary"> Update </a>
      </div>
    </div>
  </div>
</div>

```

templates/book/book_form.html

```

<div class="mb-3">
    <label for="bookName" class="form-label">Book Name</label>
    <input type="text" name="name" value="{{ book.name }}"
required="required" class="form-control" id="bookName" />
</div>

<div class="mb-3">
    <label for="bookCost" class="form-label">Book Cost</label>
    <input
        type="number"
        min="0"
        name="cost"
        value="{{ book.cost }}"
        required="required"
        class="form-control"
        id="bookCost"
    />
</div>

<div class="mb-3">
    <label for="bookLibrary" class="form-label">Select library</label>
    <select name="library" class="form-select" aria-label="Default select
example" id="bookLibrary">
        {% for library in libraries %}
        <option value="{{ library.pk }}">{{ library.name }}</option>
        {% endfor %}
    </select>
</div>

```

templates/book/book_list.html

```

{% extends 'inc/base.html' %} {% load static %} {% block content %}
<div class="container">
    <h1 class="text-center mt-4">All Books</h1>
</div>
<!-- Page Content -->
<div class="d-flex flex-column">
    {% for book in books %} {% include 'book/book_card.html' with book=book
%} {% endfor %}
</div>
{% endblock content %}

```

templates/book/create_book.html

```

{% extends 'inc/base.html' %} {% load static %} {% block content %}
<div class="container">
    <h1 class="text-center mt-4">New Book</h1>
</div>

```



```
<!-- Page Content -->
<div>
    <form action="{% url 'create_book' %}" method="POST" class="form">
        {% csrf_token %} {% include 'book/book_form.html' with book=book %}
        <button type="submit" class="btn btn-primary">Create</button>
    </form>
</div>
{% endblock content %}
```

templates/book/update_book.html

```
{% extends 'inc/base.html' %} {% load static %} {% block content %}
<div class="container">
    <h1 class="text-center mt-4">Update Book</h1>
</div>
<!-- Page Content -->
<div>
    <form action="{% url 'update_book' book.pk %}" method="POST"
class="form">
        {% csrf_token %} {% include 'book/book_form.html' with book=book %}
        <button type="submit" class="btn btn-primary">Save</button>
    </form>
</div>
{% endblock content %}
```

Результат работы:

Main Page



New Book

Book Name

Book Cost

Select library

 ▼

Create

All Books



Старик и море

Ленина

This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

333

Delete

Update



Метро 2033

Ленина

This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

1480

Delete

Update

All libraries



Ленина

Some quick example text to build on the card title and make up the bulk of the card's content.

Delete

Update



Пушкина

Some quick example text to build on the card title and make up the bulk of the card's content.

Delete

Update

Report Page

Expensive books



Метро 2033

Ленина

This is a wider card with supporting text below as a natural lead-in to additional content. This content is a little bit longer.

1480

Delete

Update

Average costs

Average cost 906.5



Average cost 401.5



New Library

Library Name

Create