

МГТУ имени Н. Э. Баумана
факультет «Информатика и Системы управления»
кафедра ИУ5 «Системы обработки информации и управления»
дисциплина «Разработка Интернет-Приложений»

Отчет по Рубежному контролю № 2
Вариант 22

Выполнил:

студент группы
ИУ5-51Б
Удодова Ксения

Проверил:

преподаватель каф. ИУ5
Гапанюк Ю.Е.

Москва, 2021 г.

Предметная область: Библиотека – Язык программирования

Задание:

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

Создайте проект Python Django с использованием стандартных средств Django.

Создайте модель Django ORM, содержащую две сущности, связанные отношением один-ко-многим в соответствии с Вашим вариантом из условий рубежного контроля №1.

С использованием стандартного механизма Django сгенерируйте по модели макет веб-приложения, позволяющий добавлять, редактировать и удалять данные.

Создайте представление и шаблон, формирующий отчет, который содержит соединение данных из двух таблиц.

Текст программы:

- **asgi.py**

```
import os

from django.core.asgi import get_asgi_application

os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'rip_rk2.settings')

application = get_asgi_application()
```

- **settings.py**

```
"""
Django settings for mysite project.

Generated by 'django-admin startproject' using Django 4.0.

For more information on this file, see
https://docs.djangoproject.com/en/4.0/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/4.0/ref/settings/
"""

from pathlib import Path
import os

# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/4.0/howto/deployment/checklist/
```

```

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-34p7&pk*1(a0u93kfoa$o8=fu_m-
rm_eo5corw!y8_1lqka6y9'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []

# Application definition

INSTALLED_APPS = [
    'rk2.apps.Rk2Config',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'rip_rk2.urls'
TEMPLATE_DIR = os.path.join(BASE_DIR, "templates")

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [TEMPLATE_DIR,],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'mysite.wsgi.application'

# Database
# https://docs.djangoproject.com/en/4.0/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'rk2',
        'USER': 'dbuser',
        'PASSWORD': '123',
        'HOST': 'localhost',
    }
}

```

```

        'PORT': 3306,
        'OPTIONS': {'charset': 'utf8'},
        'TEST_CHARSET': 'utf8',
    }
}

# Password validation
# https://docs.djangoproject.com/en/4.0/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME':
        'django.contrib.auth.password_validation.UserAttributeSimilarityValidator'
    },
    {
        'NAME':
        'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME':
        'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME':
        'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]

# Internationalization
# https://docs.djangoproject.com/en/4.0/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_TZ = True

# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/4.0/howto/static-files/

STATIC_URL = 'static/'

# Default primary key field type
# https://docs.djangoproject.com/en/4.0/ref/settings/#default-auto-field

DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'

```

- **urls.py**

```

from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),

```

```

    path('', include('rk2.urls'))
]

```

- **wsgi.py**

```

from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('rk2.urls'))
]

```

- **admin.py**

```

from django.contrib import admin

# Register your models here.

```

- **apps.py**

```

from django.apps import AppConfig

class Rk2Config(AppConfig):
    default_auto_field = 'django.db.models.BigAutoField'
    name = 'rk2'

```

- **models.py**

- **from django.db import models**

```

class Language(models.Model):
    id = models.AutoField('id', primary_key=True)
    name = models.CharField(max_length=30)
    extension = models.CharField(max_length=20)
    ide = models.CharField(db_column='IDE', max_length=40) # Field name
    made_lowercase.

    def __str__(self):
        return f'{self.name}'

    class Meta:
        managed = False
        db_table = 'language'

class Library(models.Model):
    id = models.AutoField('id', primary_key=True)
    name = models.CharField(max_length=30)
    lang = models.ForeignKey('Language', models.DO_NOTHING,
    db_column='lang_id')

    class Meta:
        managed = False
        db_table = 'library'

```

- **tests.py**

```

from django.test import TestCase

# Create your tests here.

```

- **views.py**

```
from django.db import models

class Language(models.Model):
    id = models.AutoField('id', primary_key=True)
    name = models.CharField(max_length=30)
    extension = models.CharField(max_length=20)
    ide = models.CharField(db_column='IDE', max_length=40) # Field name
    made_lowercase.

    def __str__(self):
        return f'{self.name}'

    class Meta:
        managed = False
        db_table = 'language'

class Library(models.Model):
    id = models.AutoField('id', primary_key=True)
    name = models.CharField(max_length=30)
    lang = models.ForeignKey('Language', models.DO_NOTHING,
db_column='lang_id')

    class Meta:
        managed = False
        db_table = 'library'
```

- **base.html**

```
• <!DOCTYPE html>
<html>
<head>
    <meta charset='utf-8'>
    <title>
        {% block title %} {% endblock %}
    </title>
    <style>
a, a:visited{
    color: black
}
p {
    font-size: 1.2rem;
    font-family: sans-serif;
}

    </style>
</head>
<body>
<div>
    <h3> <a style='margin: 50px;' href='/'>Главная</a> <br>
    <a style='margin: 50px;' href='/lang'>Языки</a> <br>
    <a style='margin: 50px;' href='/library'>Библиотеки</a> <br>
    <a style='margin: 50px;' href='/report'>Отчет</a> </h3>
</div>
{% block content %} {% endblock %}
</body>
</html>
```

- **index.html**

- ```

{% extends 'base.html' %}

{% block title %}
Начальная страница
{% endblock %}

{% block content %}
<h2> Начальная страница </h2>
{% endblock %}

```
- ### lang\_d.html

```

{% extends "base.html" %}

{% block content %}

<h2>Удаление записи</h2>

<p>Подтвердите удаления {{object.name}}</p>

<form action="" method="POST">
 {% csrf_token %}
 <input type="submit" value="Удалить" />
</form>
{% endblock %}

```
- ### lang.html

```

{% extends "base.html" %}

{% block title %}
Изменение
{% endblock %}

{% block content %}

<h2>Параметры записи</h2>
<form action="" method="post">
 {% csrf_token %}
 <table>
 {{ form.as_table }}
 </table>

 <input type="submit" value="Сохранить" />
</form>

{% endblock %}

```
- ### library\_d.html

```

{% extends "base.html" %}

{% block content %}

<h2>Удаление записи</h2>

<p>Подтвердите удаление <i>{{object.name}}</i></p>

<form action="" method="POST">
 {% csrf_token %}
 <input type="submit" value="Удалить" />
</form>
{% endblock %}

```

- **library.html**

```
{% extends "base.html" %}

{% block title %}
Изменение
{% endblock %}

{% block content %}

<h2>Параметры записи</h2>
<form action="" method="post">
 {% csrf_token %}
 <table>
 {{ form.as_table }}
 </table>

 <input type="submit" value="Сохранить" />
</form>

{% endblock %}
```

- **list.html**

```
{% extends 'base.html' %}

{% block title %}
{{ entity }}
{% endblock %}

{% block content %}
<h2>{{ entity }}</h2>
{% if entity == 'Language' %}
Добавить запись
{% endif %}

{% if entity == 'Library' %}
Добавить запись
{% endif %}

 {% for elem in objects %}

 {% if entity == 'Language' %}
 <p>{{ elem.name }} {{ elem.extension }}, {{ elem.id }}</p>
 Изменить

 Удалить
 {% endif %}

 {% if entity == 'Library' %}
 <p> {{ elem.name }}</p>

 Изменить

 Удалить
 {% endif %}

 {% endfor %}

{% endblock %}
```



- **report.html**

- ```
{% extends 'base.html' %}
```

```
{% block title %}
```

Библиотеки

```
{% endblock %}
```

```
{% block content %}
```

```
<ul>
```

```
    {% for lib in libraries %}
```

```
        <hr>
```

```
        <li>
```

```
            <div>
```

Название {{lib.name}}

Язык {{lib.lang.name}}

```
            </div>
```

```
        </li>
```

```
    {% endfor %}
```

```
<hr>
```

```
</ul>
```

```
{% endblock %}
```

Результат работы:

[Главная](#)

[Языки](#)

[Библиотеки](#)

[Отчет](#)

[Начальная страница](#)

[Главная](#)

[Языки](#)

[Библиотеки](#)

[Отчет](#)

Language

[Добавить запись](#)

Python

1991
Pycharm

[Изменить](#)

[Удалить](#)

C#

2000
VS

[Изменить](#)

[Удалить](#)

Java

1995
JDK

[Изменить](#)

[Удалить](#)

C++

1983
VS code

[Изменить](#)

[Удалить](#)

[Главная](#)

[Языки](#)

[Библиотеки](#)

[Отчет](#)

Параметры записи

name

year

program

Delorean

[Изменить](#)
[Удалить](#)

LinqtoSQL

[Изменить](#)
[Удалить](#)

D3

[Изменить](#)
[Удалить](#)

Guava

[Изменить](#)
[Удалить](#)

Название Delorian

Язык Python

[Изменить](#)
[Удалить](#)

Название LinqtoSQL

Язык C#

[Изменить](#)
[Удалить](#)

Название D3

Язык C++

[Изменить](#)
[Удалить](#)

Название Guava

Язык Java

[Изменить](#)
[Удалить](#)