



МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
ИМ. Н. Э. БАУМАНА

ФАКУЛЬТЕТ  
«ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ»

КАФЕДРА  
«АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ»

## **Лабораторная работа №4**

по учебной дисциплине  
**«Разработка интернет приложений»**

на тему  
**«Создание прототипа веб-приложения  
с использованием фреймворка Django»**

Вариант №1

**Группа:** ИУ5Ц-73Б

**Студент:** Каунов А. А.

**Преподаватель:** Гапанюк Ю. Е.

## 1. Описание задания

### Изучить возможности функционального программирования в языке Python:

1. Создайте прототип веб-приложения с использованием фреймворка Django:

- Создайте виртуальное окружение;
- Установите в него Django;
- Создайте проект и приложение Django.

2. Создайте представления и шаблоны (по желанию можно использовать модели), реализующие концепцию master/detail со следующей функциональностью:

- На странице master в виде списка HTML выводится информация о трех объектах (например, о трех сортах мороженого). Каждая строка списка представляет собой гиперссылку, при нажатии на которую происходит переход к странице detail;
- Страница detail содержит детальное описание объекта (сорта мороженого), фотографию, гиперссылку на master-страницу;
- Фотография относится к статическому содержимому сайта;
- Страница detail должна выводить данные с использованием таблицы HTML.;
- Шаблон страницы detail получает от представления данные о детальном объекте с использованием контекста;
- НЕОБЯЗАТЕЛЬНЫЙ ПУНКТ. По желанию можно использовать верстку с применением Bootstrap (или аналогичного фреймворка), а также представления на основе классов (class-based views).

## 2. Текст программы

### Файл Base.html

```
<!DOCTYPE html>
<html lang="en" class="h-100">
  <head>
    <meta charset="utf-8">
    <title>{%block title%}{%endblock%}</title>
  </head>
  <body>
    {%block content%}{%endblock%}
  </body>
</html>
```

### Файл Detail.html:

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

{% block content %}
  <table border="1px">
    <tr align="center">
      <th>Под</th>
      <th>Вид</th>
```

```

        <th>Сорт</th>
        <th>Рег. название</th>
        <th>Страна происхождения</th>
    </tr>
    {% for apple in data.apples %}
        {% if apple.id == data.id %}
            <tr align="center">
                <td>{{ apple.genus }}</td>
                <td>{{ apple.species }}</td>
                <td>{{ apple.sort }}</td>
                <td>{{ apple.name }}</td>
                <td>{{ apple.origin }}</td>
                <td></td>
            </tr>
            {% endif %}
        {% endfor %}
    </table>
    <a href="{% url 'master_url' %}">Главная</a>
{% endblock %}

```

### Файл Detail\_element.html:

```

<tr align="center">
    <td>{{ element.genus }}</td>
    <td>{{ element.species }}</td>
    <td>{{ element.sort }}</td>
    <td>{{ element.name }}</td>
    <td>{{ element.origin }}</td>
    <td></td>
</tr>

```

### Master.html:

```

{% extends 'base.html' %}

{% block title %}Список заказов{% endblock %}

{% block content %}
    Сорта яблок:
    <ul>
        {% for apple in data.apples %}
            <li><a href="{% url 'apple_url' apple.id %}">{{ apple.title }}</a> </li>
            {% empty %}
                <li>Список пуст</li>
            {% endfor %}
        </ul>
    {% endblock %}

```

### Urls.py:

```

from django.contrib import admin
from django.urls import path
from lab4 import views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.GetApples, name='master_url'),
    path('apple/<int:id>/', views.GetApple, name='apple_url'),
]

```

### Views.py:

```

from django.shortcuts import render

```

```

from datetime import date

from django.http import HttpResponse

def hello(request):
    return render(request, 'index.html', {'data': {
        'current_date': date.today(),
        'list': ['cats', 'dogs', 'pythons']
    }})

def GetApples(request):
    return render(request, 'master.html', {'data' : {
        'current_date': date.today(),
        'apples': [
            {'title': 'Голден Делишес', 'id': 1},
            {'title': 'Ред делишес', 'id': 2},
            {'title': 'Фуджи', 'id': 3},
        ]
    }})


def GetApple(request, id):
    return render(request, 'detail.html', {'data': {
        'current_date': date.today(),
        'id': id,
        'apples': [
            {'title': 'Голден Делишес', 'id': 1, 'genus': 'Яблоня', 'species': 'Яблоня
домашняя',
            'sort': 'Голден Делишес', 'name': 'Golden Delicious', 'origin': 'США', 'image':
'images/golden_delishes.png'},
            {'title': 'Ред делишес', 'id': 2, 'genus': 'Яблоня', 'species': 'Яблоня
домашняя',
            'sort': 'Ред делишес', 'name': 'Red Delicious', 'origin': 'США', 'image':
'images/red_delishes.png'},
            {'title': 'Фуджи', 'id': 3, 'genus': 'Яблоня', 'species': 'Яблоня низкая',
            'sort': 'Фуджи', 'name': 'Fuji', 'origin': 'Фудзисаки', 'image':
'images/fuji.png'},
        ]
    }})

```

### 3. Вывод программы

Сорта яблок:

- [Голден Делишес](#)
- [Ред делишес](#)
- [Фуджи](#)

Род	Вид	Сорт	Рег. название	Страна происхождения	
Яблоня	Яблоня домашняя	Голден Делишес	Golden Delicious	США	

[Главная](#)

Род	Вид	Сорт	Рег. название	Страна происхождения	
Яблоня	Яблоня домашняя	Ред делишес	Red Delicious	США	

[Главная](#)