



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ

Отчет по лабораторной работе №4

Студент Вардумян Арсен Тигранович
фамилия, имя, отчество

Группа ИУ5-51Б

Студент 18.12.2021 Вардумян А.Т.
подпись, дата фамилия, и.о.

Преподаватель 18.12.2021 Гапанюк Ю.Е.
подпись, дата фамилия, и.о.

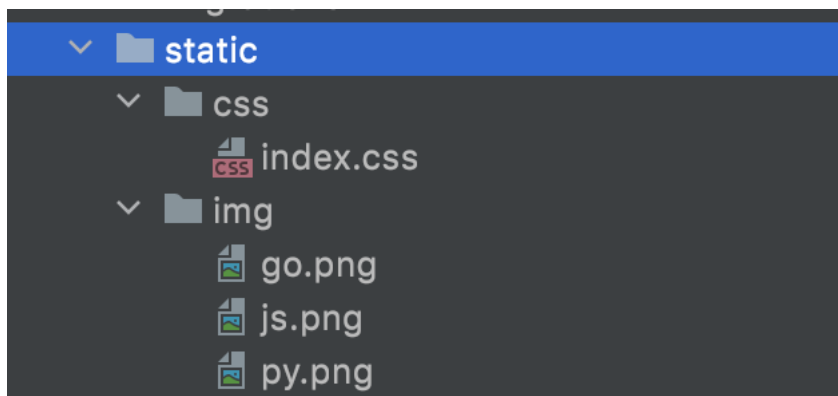
Описание задания:

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).

Текст программы:

Программа использует стандартный шаблон приложения Django. В отчете приведен лишь код, написанный исполнителем самостоятельно.

“/static”:



“urls.py”:

```
from django.contrib import admin
from django.urls import path
from Lab4 import views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.get_langs, name='master_url'),
    path('lang/<str:lang_id>', views.get_lang, name='lang_url'),
]
```

“settings.py”:

```
STATIC_URL = '/static/'
```

“views.py”:

```
from django.shortcuts import render

langs = [
    {
        'title': 'JavaScript',
        'id': 'js',
        'description': 'JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.',
    },
    {
        'title': 'Golang',
        'id': 'go',
        'description': 'Go (also called Golang or Go language) is an open source programming language used for general purpose. Go was developed by Google engineers to create dependable and efficient software. Most similarly modeled after C, Go is statically typed and explicit. The language was designed by taking inspiration for the productivity and relative simplicity of Python, with the ability of C.',
    },
    {
        'title': 'Python',
        'id': 'py',
        'description': "Python is an interpreted, object-oriented, high-level programming language with "
        "dynamic semantics. Its high-level built in data structures, combined with dynamic "
        "typing and dynamic binding, make it very attractive for Rapid Application "
        "Development, as well as for use as a scripting or glue language to connect existing "
```

```

        "components together.",
    },
]

def get_langs(request):
    return render(request, 'langs.html', {
        'langs': langs,
    })

def get_lang(request, lang_id):
    for lang in langs:
        if lang['id'] == lang_id:
            return render(request, 'lang.html', {
                'lang': lang,
            })

```

“base.html”:

```

<!doctype html>
<html lang="en" class="h-100">
<head>
    <meta charset="utf-8">
    {% load static %}
    <link rel="stylesheet" type="text/css" href="{% static 'css/index.css' %}">
    <title>{% block title %}{% endblock %}</title>
</head>
<body>
{% block content %}{% endblock %}
<footer>Made by Vardumyan Arsen IU5-51B</footer>
</body>
</html>

```

“lang.html”:

```

{% extends 'base.html' %}

{% block title %}{{ lang.title }}{% endblock %}

{% block content %}
    {% load static %}
    <div class="container">
        <div class="card">
            <div class="card__header" style="text-align: center">
                {% if lang.id == "py" %}
                    
                {% else %}
                    
                {% endif %}
            </div>
            <div class="card__body">
                <h4>{{ lang.title }}</h4>
                <p>{{ lang.description }}</p>
            </div>
            <div class="card__footer">
                <div class="user">
                    <div class="user__info">

```

```

        <a href="{% url 'master_url' %}" class="link">
            <span class="mask">
                <div class="link-container">
                    <span class="link-title1 title">Back</span>
                    <span class="link-title2 title">Back</span>
                </div>
            </span>
        </a>
    </div>
</div>
</div>
</div>
</div>
{% endblock %}

```

“langs.html”:

```

{% extends 'base.html' %}

{% block title %}Best technologies{% endblock %}

{% block content %}
    <div class="main-menu">
        <ul>
            {% for lang in langs %}
                <li><a href="{% url 'lang_url' lang.id %}">{{ lang.title }}</a> </li>
            {% empty %}
                <li>Список пуст</li>
            {% endfor %}
        </ul>
    </div>
{% endblock %}

```

“index.css”:

```

@import
url("https://fonts.googleapis.com/css2?family=Quicksand:wght@300..700&display=swap");

*,
*::before,
*::after {
    box-sizing: border-box;
    padding: 0;
    margin: 0;
}

body {
    font-family: "Quicksand", sans-serif;
    display: grid;
    place-items: center;
    height: 100vh;
}

.container {
    display: flex;
    flex-wrap: wrap;
    justify-content: center;
    max-width: 1200px;
    margin-block: 2rem;
}

```

```

        gap: 2rem;
    }

    img {
        max-width: 100%;
        display: block;
        object-fit: cover;
    }

    .card {
        display: flex;
        flex-direction: column;
        width: clamp(25rem, calc(20rem + 2vw), 22rem);
        overflow: hidden;
        box-shadow: 0 .1rem 1rem rgba(0, 0, 0, 0.1);
        border-radius: 1em;
        background: #ECE9E6;
        background: linear-gradient(to right, #FFFFFF, #ECE9E6);
    }

    .card__body {
        padding: 1rem;
        display: flex;
        flex-direction: column;
        gap: .5rem;
    }

    .card__body h4 {
        font-size: 1.5rem;
        text-transform: capitalize;
    }

    .card__footer {
        display: flex;
        padding: 1rem;
        margin-top: auto;
    }

    ul li:before {
        ...
        background: linear-gradient(45deg, #f69ec4, #f9dd94);
    }

    .main-menu {
        width: 200px;
    }

    .main-menu h2 {
        font: 400 40px/1.5 Helvetica, Verdana, sans-serif;
        margin: 0;
        padding: 0;
    }

    .main-menu ul {
        list-style-type: none;
        margin: 0;
        padding: 0;
    }

    .main-menu ul li {
        font-size: 20px;
    }

```

```

        border-bottom: 1px solid #ccc;
    }

    .main-menu ul li:last-child {
        border: none;
    }

    .main-menu ul li a {
        text-decoration: none;
        color: #000;
        display: block;
        width: 200px;

        -webkit-transition: font-size 0.3s ease, background-color 0.3s ease;
        -moz-transition: font-size 0.3s ease, background-color 0.3s ease;
        -o-transition: font-size 0.3s ease, background-color 0.3s ease;
        -ms-transition: font-size 0.3s ease, background-color 0.3s ease;
        transition: font-size 0.3s ease, background-color 0.3s ease;
    }

    .main-menu ul li a:hover {
        font-size: 30px;
        background: #f6f6f6;
    }

    .link {
        height: 30px;
        align-items: center;
        color: #000;
        text-decoration: none;
        display: flex;
    }

    /* Hide extra text */
    .mask {
        position: relative;
        padding: 0;
        height: 20px;

        /* Remove overflow to see how it works :) */
        overflow: hidden;
    }

    .link-container {
        transition: transform 0.4s ease;
    }

    .title {
        display: block;

        /* Set same font-size and line height */
        font-size: 20px;
        line-height: 20px;

        transition: transform 0.4s ease;
    }

    .link-title1 {
        transform-origin: right center;
    }

    .link-title2 {

```

```
    transform-origin: left center;
    transform: rotate(20deg);
}

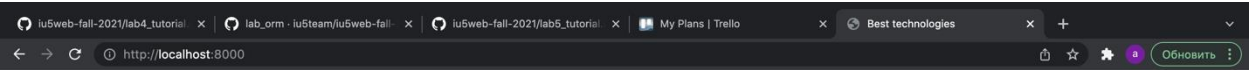
/* Hover Action*/

/* Move up two texts (20px = font-size) */
.link:hover .link-container {
    transform: translateY(-20px);
}

/* Rotate texts a little bit */
.link:hover .link-title1 {
    transform: rotate(20deg);
}

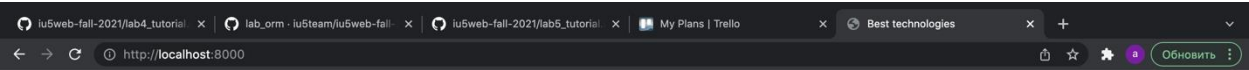
.link:hover .link-title2 {
    transform: rotate(0);
}
```


Экранные формы с примерами выполнения программы:



JavaScript
Golang
Python

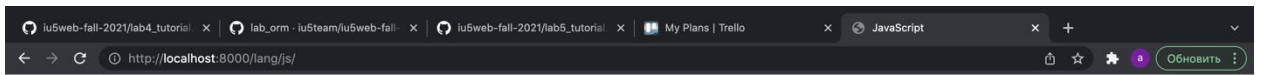
Made by Vardumyan Arsen IU5-51B



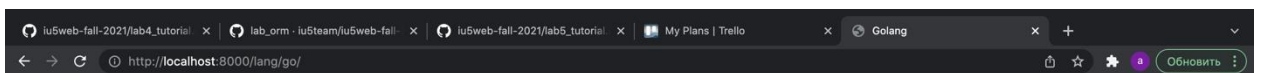
JavaScript
Golang
Python

Made by Vardumyan Arsen IU5-51B

localhost:8000/lang/js/



Made by Vardumyan Arsen IU5-51B



Made by Vardumyan Arsen IU5-51B



Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.

[Back](#)