

**КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ
Ордена трудового Красного Знамени
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский технический университет связи и информатики»**

Кафедра «Электроника»



**Задание по дисциплине
«Прикладные информационные системы»**

Лабораторная работа № 1

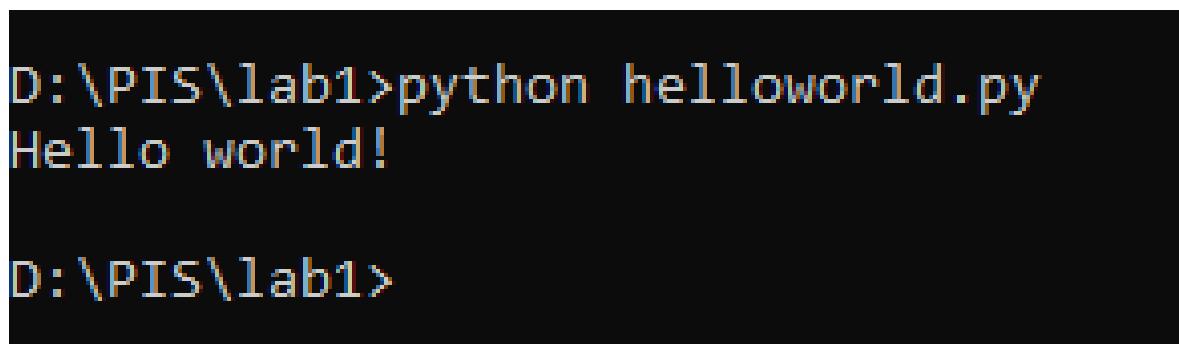
Группа: УБСТ2201
ФИО: Лысов Вячеслав Константинович

Москва 2025

Лабораторная работа №1

Начало работы с Python.

Ознакомился с вводной информацией по работе с Python. Установил Python 3.12. Создал директорию для лабораторных работ на диске D. С помощью редактора кода Visual Studio Code добавил в созданный файл код: print("Hello world!"). Запустил скрипт с помощью команды: python helloworld.py. С результатом можно ознакомиться на рисунке 1.



```
D:\PIS\lab1>python helloworld.py
Hello world!

D:\PIS\lab1>
```

Рисунок 1 — Результат команды python helloworld.py.

Задание. Необходимо написать функцию фильтрации студентов по средней оценке, так, чтобы на экран выводился список студентов, средний балл которых выше заданного. Средний балл, по которому будет проводиться фильтрация, вводится пользователем с клавиатуры. Со следующим списком студентов:

```
groupmates = [
    {
        "name": "Александр",
        "surname": "Иванов",
        "exams": ["Информатика", "ЭЭиС", "Web"],
        "marks": [4, 3, 5]
    },
    {
        "name": "Иван",
        "surname": "Петров",
        "exams": ["История", "АиГ", "КТП"],
        "marks": [4, 4, 4]
    },
    {
        "name": "Кирилл",
        "surname": "Смирнов",
        "exams": ["Философия", "ИС", "КТП"],
        "marks": [5, 5, 5]
    }
]
```

Рисунок 2— Исходные данные для выполнения задания.

```

def print_students(students):
    print("Имя".ljust(15), "Фамилия".ljust(10), "Экзамены".ljust(30), "Оценки".ljust(20))
    for student in students:
        print(student["name"].ljust(15),
              student["surname"].ljust(10),
              str(student["exams"]).ljust(30),
              str(student["marks"]).ljust(20))

```

Рисунок 3 — Код вывода студентов.

```

min_avg = float(input("Введите минимальный средний балл: "))
filtered_students = []
for student in groupmates:
    marks = student["marks"]
    avg_mark = sum(marks) / len(marks)
    if avg_mark >= min_avg:
        filtered_students.append(student)
print("\nСтуденты со средним баллом выше", min_avg)
if filtered_students:
    print_students(filtered_students)
else:
    print("Нет студентов с таким средним баллом")

```

Рисунок 4 — Код фильтрации и вычисления среднего балла.

```

D:\PIS\lab1>python mygroup.py
Введите минимальный средний балл: 3

Студенты со средним баллом выше 3.0
Имя          Фамилия      Экзамены          Оценки
Александр    Иванов       ['Информатика', 'ЭЭиС', 'Web'] [4, 3, 5]
Иван         Петров       ['История', 'АиГ', 'КТП']   [4, 4, 4]
Кирилл       Смирнов     ['Философия', 'ИС', 'КТП'] [5, 5, 5]

D:\PIS\lab1>python mygroup.py
Введите минимальный средний балл: 4.1

Студенты со средним баллом выше 4.1
Имя          Фамилия      Экзамены          Оценки
Кирилл       Смирнов     ['Философия', 'ИС', 'КТП'] [5, 5, 5]

D:\PIS\lab1>python mygroup.py
Введите минимальный средний балл: 5

Студенты со средним баллом выше 5.0
Имя          Фамилия      Экзамены          Оценки
Кирилл       Смирнов     ['Философия', 'ИС', 'КТП'] [5, 5, 5]

```

Рисунок 5 — Результат работы кода.

Начало работы с Django.

Перешёл на официальный сайт фреймворка Django:

<https://www.djangoproject.com/download/>

С помощью командной строки установил Django. Версия 5.2.8.

Команда: `py -m pip install Django==5.2.8`

```
C:\Windows\system32>py -m pip install Django==5.2.8
Collecting Django==5.2.8
  Using cached django-5.2.8-py3-none-any.whl.metadata (4.1 kB)
Requirement already satisfied: asgiref>=3.8.1 in c:\python312\lib\site-packages (from Django==5.2.8) (3.10.0)
Requirement already satisfied: sqlparse>=0.3.1 in c:\python312\lib\site-packages (from Django==5.2.8) (0.5.3)
Requirement already satisfied: tzdata in c:\python312\lib\site-packages (from Django==5.2.8) (2024.2)
Using cached django-5.2.8-py3-none-any.whl (8.3 MB)
Installing collected packages: Django
  Attempting uninstall: Django
    Found existing installation: Django 5.2.7
    Uninstalling Django-5.2.7:
      Successfully uninstalled Django-5.2.7
Successfully installed Django-5.2.8
```

Рисунок 6 — Процесс установки Django.

После установки фреймворка приступим к созданию проекта. Для этого с помощью командной строки необходимо перейти в директорию, в которой находятся ваши лабораторные работы, и выполнить команду:

`django-admin.py startproject project_name`

Для того, чтобы проверить, правильно ли был установлен фреймворк Django, необходимо запустить локальный web-сервер разработки. Для запуска сервера перейдите в директорию `project_name`, в которой находится файл `manage.py` и выполните команду, которая запускает локальный сервер на порту 8000. Команда: `python manage.py runserver`

```
D:\PIS\lab1\lab1>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly
auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
November 17, 2025 - 20:51:26
Django version 5.2.8, using settings 'lab1.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.

WARNING: This is a development server. Do not use it in a production s
ad.
For more information on production servers see: https://docs.djangoproject.com/en/5.2/howto/deployment/checklist/
```

Рисунок 7 — Запуск локального сервера.



The install worked successfully! Congratulations!

[View release notes for Django 5.2](#)

You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

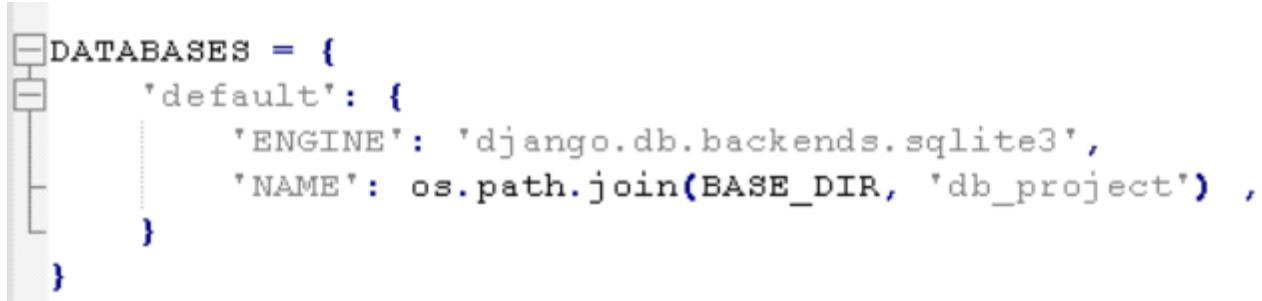
django

Рисунок 8 — Проверка локального сервера.

Задание.

В файле settings.py измените параметр базы данных. Переменной DATABASES есть внутренний словарь с ключами ENGINE и NAME. Для параметра NAME установите следующее значение:

```
'NAME': os.path.join(BASE_DIR, 'db_project_name')
```

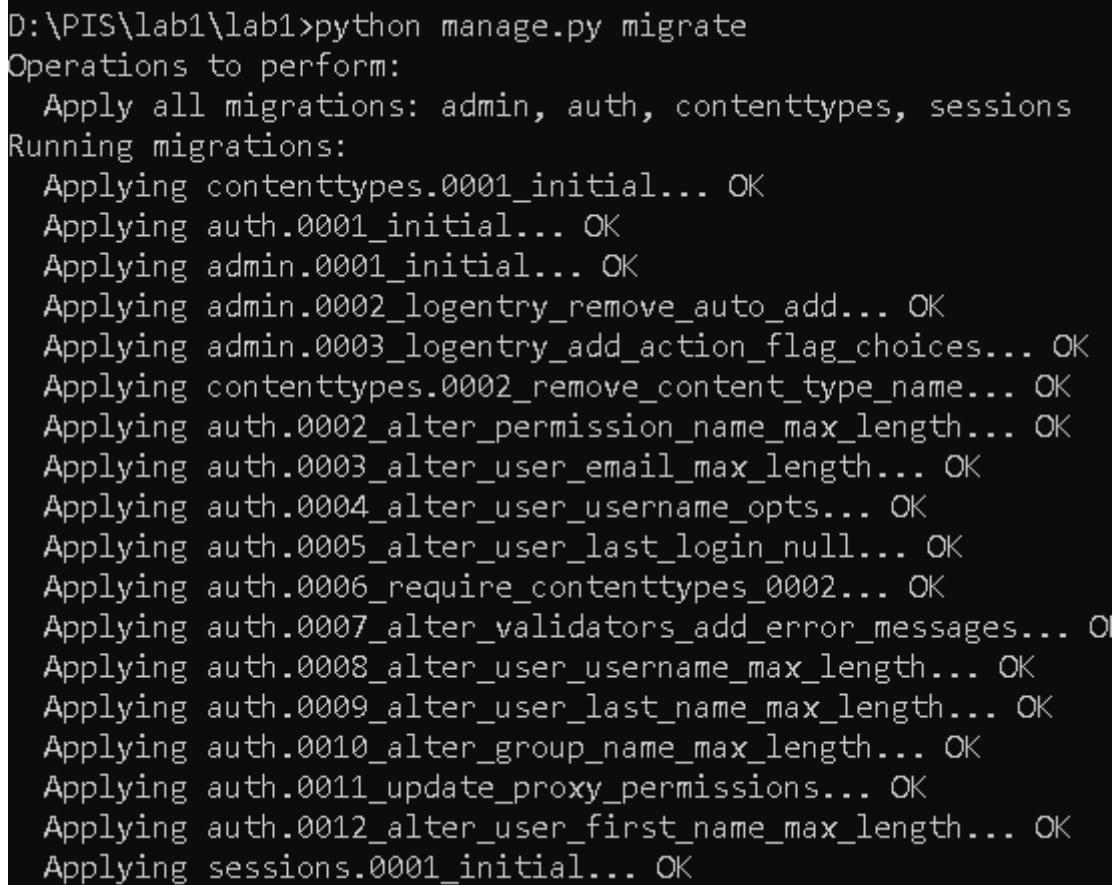


```
DATA BASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.sqlite3',  
        'NAME': os.path.join(BASE_DIR, 'db_project') ,  
    }  
}
```

Рисунок 9 — Внесение изменения в файл settings.py.

Чтобы создать таблицы базы данных, необходимо перейти в директорию project_name и через командную строку выполнить команду:

```
python3 manage.py migrate
```



```
D:\PIS\lab1\lab1>python manage.py migrate  
Operations to perform:  
  Apply all migrations: admin, auth, contenttypes, sessions  
Running migrations:  
  Applying contenttypes.0001_initial... OK  
  Applying auth.0001_initial... OK  
  Applying admin.0001_initial... OK  
  Applying admin.0002_logentry_remove_auto_add... OK  
  Applying admin.0003_logentry_add_action_flag_choices... OK  
  Applying contenttypes.0002_remove_content_type_name... OK  
  Applying auth.0002_alter_permission_name_max_length... OK  
  Applying auth.0003_alter_user_email_max_length... OK  
  Applying auth.0004_alter_user_username_opts... OK  
  Applying auth.0005_alter_user_last_login_null... OK  
  Applying auth.0006_require_contenttypes_0002... OK  
  Applying auth.0007_alter_validators_add_error_messages... O  
  Applying auth.0008_alter_user_username_max_length... OK  
  Applying auth.0009_alter_user_last_name_max_length... OK  
  Applying auth.0010_alter_group_name_max_length... OK  
  Applying auth.0011_update_proxy_permissions... OK  
  Applying auth.0012_alter_user_first_name_max_length... OK  
  Applying sessions.0001_initial... OK
```

Рисунок 10 — Создание таблицы базы данных.

Затем создаем суперпользователя (при создании необходимо ввести логин, email и пароль пользователя):

```
python3 manage.py createsuperuser
```

```
D:\PIS\lab1\lab1>python manage.py createsuperuser
Username (leave blank to use 'lysov.vk'): lysov.vk
Email address: v.k.lysov@edu.mtuci.ru
Password:
Password (again):
This password is too short. It must contain at least 8 characters.
This password is too common.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.
```

Рисунок 11 — Создание суперпользователя.

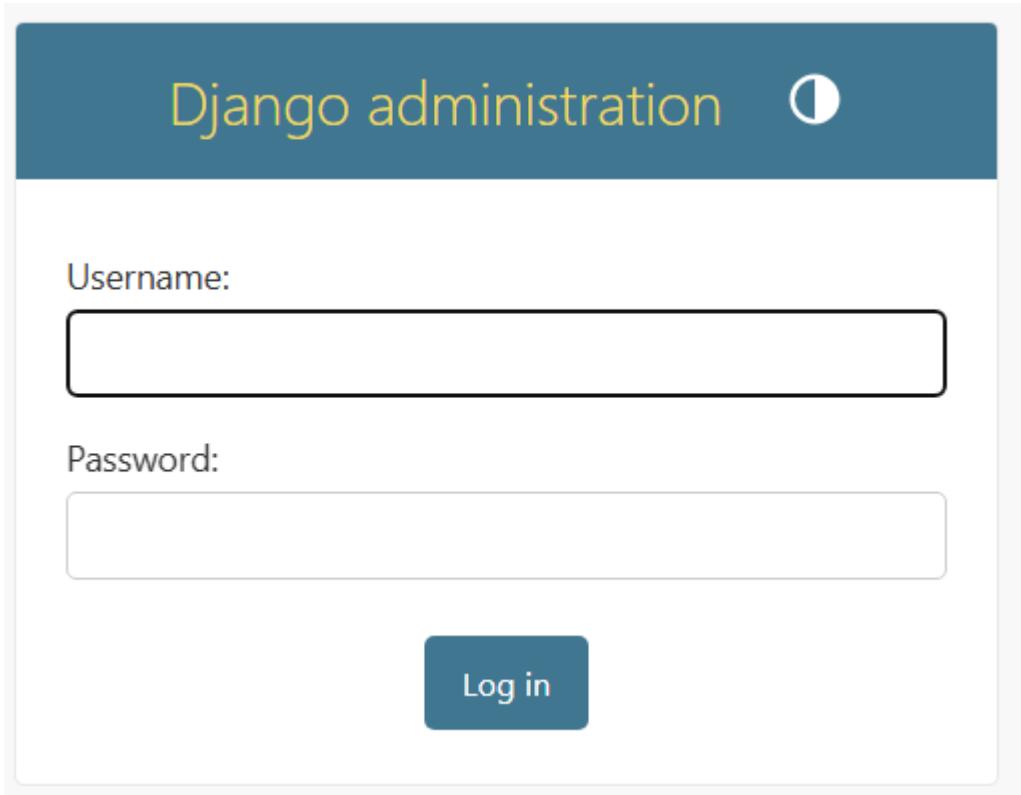


Рисунок 12 — Проверка работы и авторизация.

После успешного авторизации, необходимо выполнить следующее задание:

- Изучите интерфейс административного приложения django;
- Через интерфейс административного приложения создайте нового пользователя с правами суперпользователя;
- Через интерфейс административного приложения создайте нового пользователя без прав суперпользователя;
- Через интерфейс административного приложения «забаньте» одного из пользователей (сделайте пользователя «неактивным»);
- Загрузите ваш проект на любой гит-репозиторий (GitHub, GitLab, Google Code, Bitbucket и т.п.).

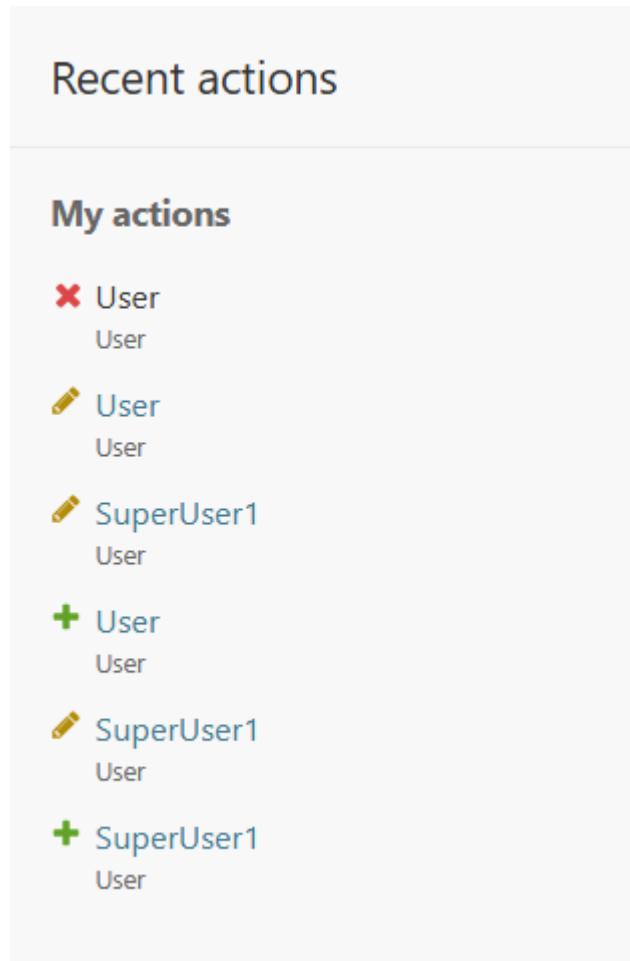


Рисунок 13 — Выполнение всех пунктов задания.

После выполнения всех действий, так будет выглядеть «Recent actions».