**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №4

«Проект на Flask»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-33Б |  | преподаватель каф. ИУ5 |
| Нагапетян Валерий |  | Гапанюк Ю. Е. |
|  |  |  |

Москва, 2023 г.

**Задание**

Написать Таск-Менеджера с использованием веб-фреймворка Flask, в котором можно будет добавлять дела, редактировать их статус и удалять выполненные.

* Использовать фреймворк Semantic UI для создания стильного пользовательского интерфейса.
* Использовать SQLAlchemy для управления базой данных с задачами.

**Файл main.py**

**Текст программы**

*from* flask *import* Flask, redirect, render\_template, request, url\_for  
*from* flask\_sqlalchemy *import* SQLAlchemy  
*from* sqlalchemy.exc *import* IntegrityError  
  
app = Flask(\_\_name\_\_)  
app.config["SQLALCHEMY\_DATABASE\_URI"] = "sqlite:///Users.sqlite3"  
app.config["SQLALCHEMY\_TRACK\_MODIFICATIONS"] = *False*db = SQLAlchemy(app)  
  
  
*class* ToDo(db.Model):  
 id = db.Column("id", db.Integer, primary\_key=*True*)  
 title = db.Column(db.String(150), unique=*True*, nullable=*False*)  
 is\_complete = db.Column(db.Boolean)  
 current\_id = db.Column(db.Integer)  
  
  
@app.get("/")  
@app.get("/home")  
*def* home():  
 todo\_list = ToDo.query.all()  
 *return* render\_template("index.html", todo\_list=todo\_list, title=f"Главная страница")  
  
  
@app.post("/add")  
*def* add():  
 title = request.form.get("title")  
 new\_todo = ToDo(title=title, is\_complete=*False*)  
  
 *try*:  
 db.session.add(new\_todo)  
 db.session.commit()  
 new\_todo.current\_id = new\_todo.id  
 db.session.commit()  
 *except* IntegrityError:  
 print("exc")  
  
 *return* redirect(url\_for("home"))  
  
  
@app.get("/update/<int:todo\_id>")  
*def* update(todo\_id):  
 todo = ToDo.query.filter\_by(id=todo\_id).first()  
 todo.is\_complete = *not* todo.is\_complete  
  
 db.session.commit()  
 *return* redirect(url\_for("home"))  
  
  
@app.get("/delete/<int:todo\_id>")  
*def* delete(todo\_id):  
 todo = ToDo.query.filter\_by(id=todo\_id).first()  
  
 todo\_list = ToDo.query.all()  
 *for* i *in* range(todo.current\_id, len(todo\_list)):  
 todo\_list[i].current\_id -= 1  
  
 db.session.delete(todo)  
 db.session.commit()  
 *return* redirect(url\_for("home"))  
  
  
*if* \_\_name\_\_ == "\_\_main\_\_":  
 *with* app.app\_context():  
 db.create\_all()  
 app.run(debug=*True*)

**Файл layout.html**

**Текст программы**

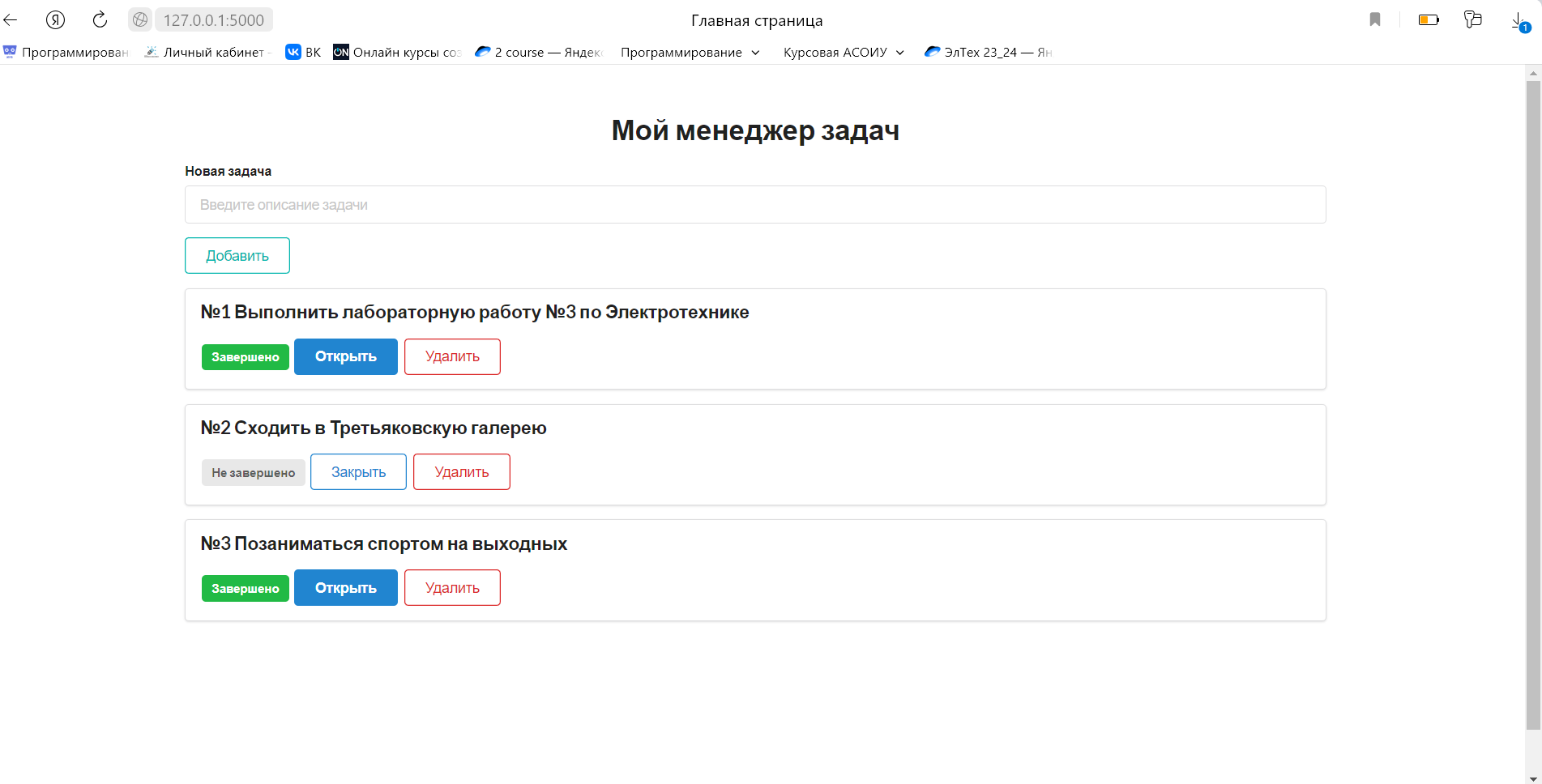
<!doctype *html*>  
<html *lang*="ru">  
<head>  
 <meta *charset*="UTF-8">  
 <meta *name*="viewport"  
 *content*="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">  
 <meta *http-equiv*="X-UA-Compatible" *content*="ie=edge">  
 <title>{{ title if title else 'Базовый шаблон' }} </title>  
 <link *rel*="stylesheet" *href*="{{ url\_for('static', filename='css/style.css') }}">  
 <link *rel*="stylesheet" *href*="https://cdn.jsdelivr.net/npm/semantic-ui@2.4.2/dist/semantic.min.css">  
 <script *src*="https://cdn.jsdelivr.net/npm/semantic-ui@2.4.2/dist/semantic.min.js"></script>  
</head>  
<body>  
{% block content %}{% endblock content %}  
</body>  
</html>

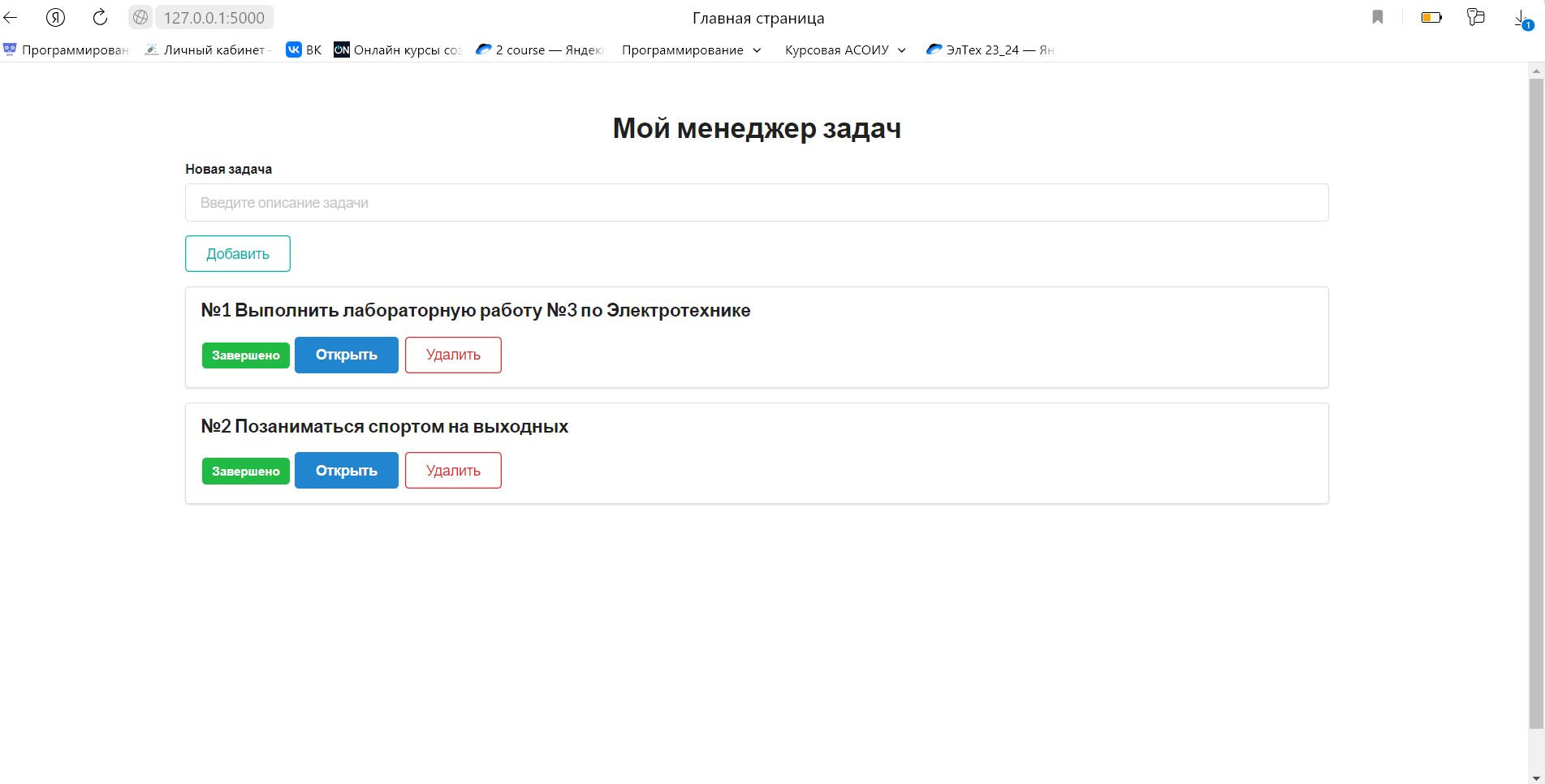
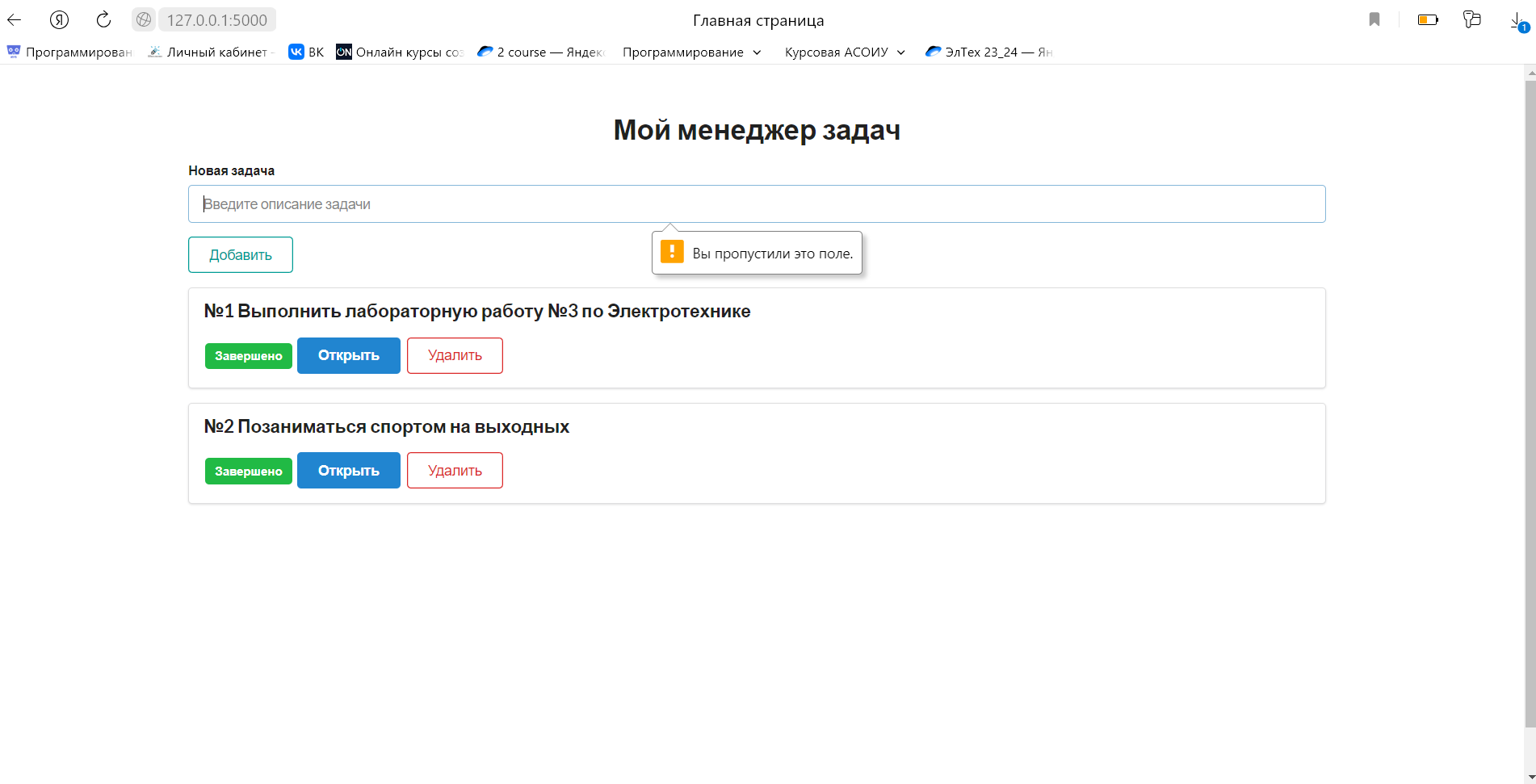
**Файл index.html**

**Текст программы**

{% extends 'layout.html' %}  
  
{% block content %}  
<div *style*="margin-top: 50px;" *class*="ui container">  
 <h1 *class*="ui center aligned header">Мой менеджер задач</h1>  
  
 <form *class*="ui form" *action*="/add" *method*="post">  
 <div *class*="field">  
 <label>Новая задача</label>  
 <input *type*="text" *name*="title" *required placeholder*="Введите описание задачи"><br>  
 </div>  
 <button *class*="ui teal basic button" *type*="submit">Добавить</button>  
 </form>  
  
{% for todo in todo\_list %}  
<div *class*="ui segment">  
 <p *class*="ui big header">№{{ todo.current\_id }} {{ todo.title }}</p>  
  
 {% if todo.is\_complete == False %}  
 <span *class*="ui gray label">Не завершено</span>  
 <a *class*="ui primary basic button" *href*="/update/{{ todo.id }}">Закрыть</a>  
 {% else %}  
 <span *class*="ui green label">Завершено</span>  
 <a *class*="ui blue button" *href*="/update/{{ todo.id }}">Открыть</a>  
 {% endif %}  
 <a *class*="ui negative basic button" *href*="/delete/{{ todo.id }}">Удалить</a>  
</div>  
{% endfor %}  
</div>  
{% endblock content %}

**Результат**

****

**** ****