

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное автономное образовательное учреждение высшего образования  
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ  
(КАФЕДРА 43)

ОЦЕНКА

ПРЕПОДАВАТЕЛЬ

Старший преподаватель  
должность, уч. степень, звание

подпись, дата

Н.А. Соловьева  
инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №5

«Серверные сценарии. Форма и обработка get и post запросов»

по дисциплине: Web-технологии

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

4236  
номер группы

подпись, дата

Л. Мвале  
инициалы, фамилия

Санкт-Петербург 2025

**Цель работы:** получение опыта написания и применения серверных сценариев для обработки данных, введенных пользователем.

## 1 Варианты задания

Таблица 1 – Тема сайта

№ варианта	Тема
21	Промышленные роботы

Таблица 2 – Вариант задания

№	Метод	Оформление вывода данных	Дополнительный элемент формы
21	Post	Горизонтальная таблица	file

## 2 Скриншоты

Анкета: Промышленные роботы

Промышленные роботы

Анкета

Главная Промышленный робот Используемые источники pr3 анкета таблица из БД

Заполните анкету

Фамилия:

Имя:

Отчество:

Группа:

Какой язык программирования вы предпочитаете? ☐ Python ☐ Java ☐ C++

Какие технологии вы используете? ☐ IoT ☐ AI ☐ Robotics ☐ Automation

Email:

Загрузите файл:

Контакты

Рисунок 1 – Исходная страница формы form.html

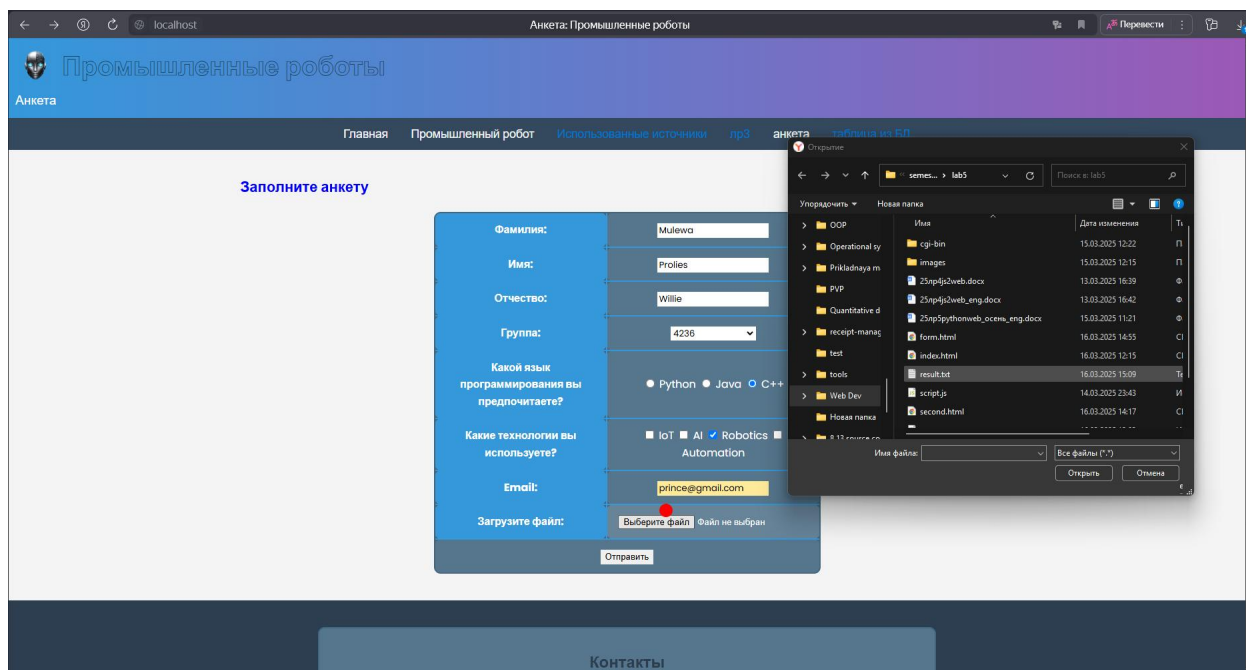


Рисунок 2 – Заполненная форма

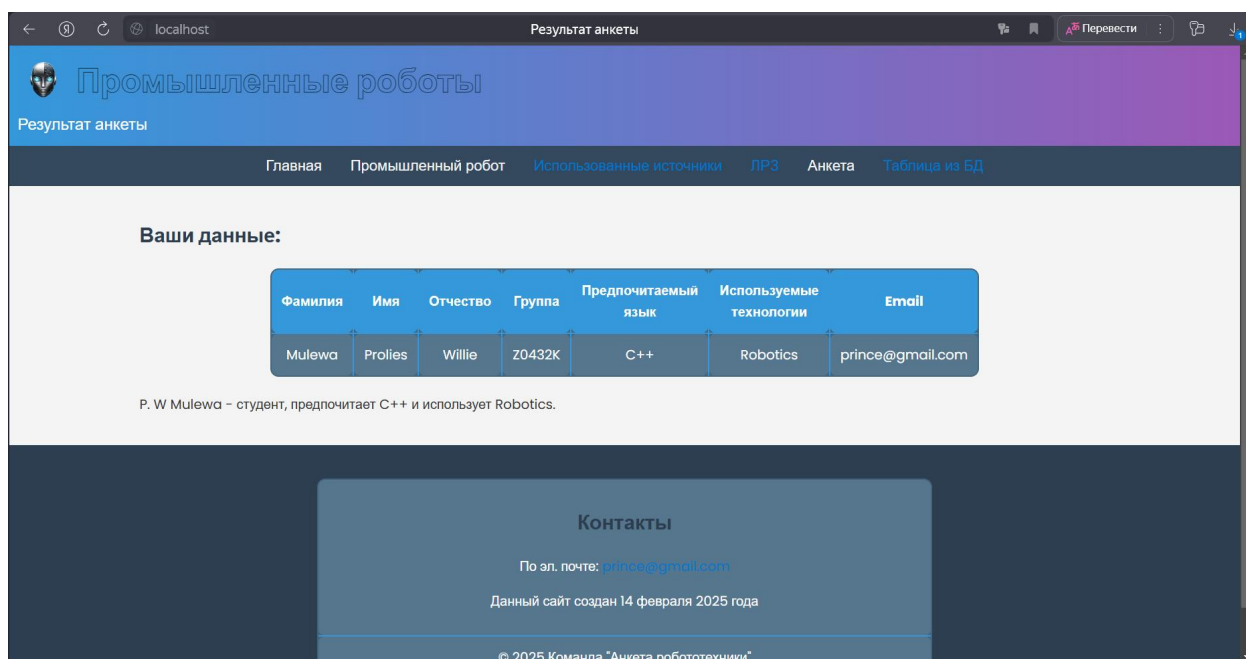


Рисунок 3 – Страница, сформированная веб-сервером

### 3 Листинг

#### Листинг функции проверки валидации

```
function formVal(form) {  
    // Validate surname  
    if (form.surname.value.length < 4 || form.surname.value.length > 30) {  
        alert("Фамилия должна содержать от 4 до 30 символов.");  
        form.surname.focus();  
        return false;  
    }  
    if (!form.surname.value.match(/^[a-zA-Za-яА-Я]+$/i)) {  
        alert("Фамилия должна содержать только буквы.");  
        form.surname.focus();  
        return false;  
    }  
  
    // Validate name  
    if (form.name.value.length < 4 || form.name.value.length > 30) {  
        alert("Имя должно содержать от 4 до 30 символов.");  
        form.name.focus();  
        return false;  
    }  
    if (!form.name.value.match(/^[a-zA-Za-яА-Я]+$/i)) {  
        alert("Имя должно содержать только буквы.");  
        form.name.focus();  
        return false;  
    }  
  
    // Validate patronymic  
    if (form.patron.value.length < 4 || form.patron.value.length > 30) {  
        alert("Отчество должно содержать от 4 до 30 символов.");  
        form.patron.focus();  
        return false;  
    }  
    if (!form.patron.value.match(/^[a-zA-Za-яА-Я]+$/i)) {  
        alert("Отчество должно содержать только буквы.");  
        form.patron.focus();  
        return false;  
    }  
  
    // Validate group selection  
    if (form.group.selectedIndex == 0) {  
        alert("Пожалуйста, выберите вашу группу.");  
        return false;  
    }  
  
    // Validate radio buttons  
    if (!form.question1[0].checked && !form.question1[1].checked  
        && !form.question1[2].checked) {  
        alert("Пожалуйста, выберите вариант ответа.");  
        return false;  
    }  
}
```

```

    }

    // Validate checkboxes
    if (!form.question2[0].checked && !form.question2[1].checked
    && !form.question2[2].checked && !form.question2[3].checked) {
        alert("Пожалуйста, выберите хотя бы один вариант ответа.");
        return false;
    }

    // Validate email
    if (!form.email.value.match(/^[^\s@]+@[^\s@]+\.[^\s@]+$/)) {
        alert("Пожалуйста, введите корректный email.");
        form.email.focus();
        return false;
    }

    return true;
}

```

### Листинг страницы формы form.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta
        charset="UTF-8"
        name="viewport"
        content="width=device-width, initial-scale=1.0" />
    <title>Анкета</title>
    <link rel="stylesheet" href="css/styles.css">
    <link rel="icon" href="images/icon.ico" type="image/x-icon">
    <script type="text/javascript" src="js/events.js"></script>
</head>
<body>
<!-- Шапка сайта с названием и логотипом -->
<header>
    <a href="#" class="logo-container">
        
        <span class="logo-text">Промышленные роботы</span>
    </a>
    <p>Анкета</p>
</header>

<!-- Навигационная панель с ссылками на разделы сайта -->
<nav align="center" >
    <a href="index.html">Главная</a>
    <a href="second.html">Промышленный робот</a>
    <a href="source.html">Использованные источники</a>
    <a href="#">лр3</a>
    <a href="form.html">анкета</a>
    <a href="#">таблица из БД</a>

```

```

</nav>
<main>
  <h2>Заполните анкету</h2>
  <form method="POST" action="http://localhost/cgi-bin/process_form.py"
enctype="multipart/form-data" onsubmit="return formVal(this);">
    <table>
      <tr>
        <th>Фамилия:</th>
        <td><input type="text" name="surname" placeholder="Введите
фамилию"></td>
      </tr>
      <tr>
        <th>Имя:</th>
        <td><input type="text" name="name" placeholder="Введите имя"></td>
      </tr>
      <tr>
        <th>Отчество:</th>
        <td><input type="text" name="patron" placeholder="Введите
отчество"></td>
      </tr>
      <tr>
        <th>Группа:</th>
        <td>
          <select name="group">
            <option value="">Выберите группу</option>
            <option value="Z0432K">4236</option>
            <option value="Z0431">Z0431</option>
            <option value="Z0431K">Z0431K</option>
          </select>
        </td>
      </tr>
      <tr>
        <th>Какой язык программирования вы предпочитаете?</th>
        <td>
          <input type="radio" name="question1" value="Python"> Python
          <input type="radio" name="question1" value="Java"> Java
          <input type="radio" name="question1" value="C++"> C++
        </td>
      </tr>
      <tr>
        <th>Какие технологии вы используете?</th>
        <td>
          <input type="checkbox" name="question2" value="IoT"> IoT
          <input type="checkbox" name="question2" value="AI"> AI
          <input type="checkbox" name="question2" value="Robotics"> Robotics
          <input type="checkbox" name="question2" value="Automation">
Automation
        </td>
      </tr>
      <tr>
        <th>Email:</th>

```

```

email"></td>
        </tr>
        <tr>
            <th>Загрузите файл:</th>
            <td><input type="file" name="file"></td>
        </tr>
        <tr>
            <td colspan="2"><input type="submit" value="Отправить"></td>
        </tr>
    </table>
</form>
</main>
<!-- Подвал сайта с контактной информацией -->
<footer>
    <table width="100%" border="0">
        <tr>
            <td>
                <address>
                    <h2>Контакты</h2>
                    <p>С
                        <button class="author-button" name="author-
info">АВТОРОМ</button> сайта можно связаться:</p>
                    <address>
                        По эл. почте: <a
href="mailto:lieson.mwale22.ru">lieson.mwale22@gmail.com</a><br />
                        По телефону: <a href="tel:+78005553535">+7(800)555-35-35</a>
                    </address>
                    <p>Данный сайт создан 14 февраля 2025 года</p>
                </address>
            </td>
        </tr>
        <tr>
            <td>&copy; 2025 Команда "История робототехники"</td>
        </tr>
    </table>
</footer>
</body>
<script src="script.js" defer></script>
</html>

```

## Листинг серверного сценария

```

#!C:\Users\1\anaconda3\python.exe
# -*- coding: utf-8 -*-

```

```

import cgi
import cgiib
import codecs
import os

```

```

cgitb.enable() # Enable error reporting

# Ensure stdout uses UTF-8
import sys
sys.stdout = codecs.getwriter("utf-8")(sys.stdout.detach())

# Create the 'uploads' directory if it doesn't exist
if not os.path.exists("uploads"):
    os.makedirs("uploads")

# Get form data
form = cgi.FieldStorage()

# Extract values and decode to UTF-8
def get_utf8_value(field_name, default="(unknown)"):
    value = form.getvalue(field_name, default)
    return value.encode("latin1").decode("utf-8") if isinstance(value, str) else default

# Extract values
surname = form.getvalue('surname', '(unknown)').encode("latin1").decode("utf-8")
name = form.getvalue('name', '(unknown)').encode("latin1").decode("utf-8")
patron = form.getvalue('patron', '(unknown)').encode("latin1").decode("utf-8")
group = form.getvalue('group', '(unknown)').encode("latin1").decode("utf-8")
question1 = form.getvalue('question1', '(unknown)').encode("latin1").decode("utf-8")
question2 = form.getlist('question2')
email = form.getvalue('email', '(unknown)').encode("latin1").decode("utf-8")

# Convert checkboxes list to UTF-8
question2 = [q.encode("latin1").decode("utf-8") for q in question2]

# Handle file upload
uploaded_filename = "(no file uploaded)" # Default if no file is uploaded

if "file" in form:
    uploaded_file = form["file"]
    if uploaded_file.filename:
        file_path = f"uploads/{uploaded_file.filename}" # Save to 'uploads/' directory
        try:
            with open(file_path, "wb") as f:
                f.write(uploaded_file.read())
            uploaded_filename = uploaded_file.filename # Store filename for display
        except Exception as e:
            uploaded_filename = f"Error saving file: {e}"

# Save data to a UTF-8 file
try:
    with open("result.txt", "a", encoding="utf-8") as f:
        f.write(f'{surname}|{name}|{patron}|{group}|{question1}|{','.join(question2)}|{email}\n')
except Exception as e:
    print("Content-Type: text/plain\r\n\r\n")
    print(f"Error saving data: {e}")

```



```

exit()

# Generate response
print("Content-Type: text/html; charset=utf-8\r\n\r\n")
print(f"""
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <title>Результат анкеты</title>
  <link rel="stylesheet" href="../styles.css">
</head>
<body>

  <!-- Шапка сайта -->
  <header>
    <a href="#" class="logo-container">
      
      <span class="logo-text">Промышленные роботы</span>
    </a>
    <p>Результат анкеты</p>
  </header>

  <!-- Горизонтальная линия для разделения контента -->

  <!-- Навигация -->
  <nav align="center">
    <a href="../index.html">Главная</a>
    <a href="../second.html">Промышленный робот</a>
    <a href="../source.html">Использованные источники</a>
    <a href="#">ЛР3</a>
    <a href="../form.html">Анкета</a>
    <a href="#">Таблица из БД</a>
  </nav>

  <main>
    <h2>Ваши данные:</h2>
    <table border="1">
      <tr>
        <th>Фамилия</th>
        <th>Имя</th>
        <th>Отчество</th>
        <th>Группа</th>
        <th>Предпочитаемый язык</th>
        <th>Используемые технологии</th>
        <th>Email</th>
        <th>Загруженный файл</th>
      </tr>
      <tr>
        <td>{surname}</td>
        <td>{name}</td>
        <td>{patron}</td>

```

```

        <td>{group}</td>
        <td>{question1}</td>
        <td>{'', '.join(question2)}</td>
        <td>{email}</td>
        <td>{uploaded_filename}
    </tr>
</table>
    <p>{name[0]}. {patron[0]} {surname} - студент, предпочитает {question1} и
использует {'', '.join(question2)}.</p>
</main>

<footer>
    <table width="100%" border="0">
        <tr>
            <td>
                <address>
                    <h2>Контакты</h2>
                    <address>
                        По эл. почте: <a href="mailto:lieson.mwale22.ru">{email}</a><br />
                    </address>
                    <p>Данный сайт создан 14 февраля 2025 года</p>
                </address>
            </td>
        </tr>
        <tr>
            <td>&copy; 2025 Команда "Анкета робототехники"</td>
        </tr>
    </table>
</footer>

</body>
<script src="script.js" defer></script>
</html>
""")

```

## Листинг сервера

```

import os
import sys
from http.server import HTTPServer, CGIHTTPRequestHandler

class MyCGIHandler(CGIHTTPRequestHandler):
    cgi_directories = ["/cgi-bin"] # Ensure this is correctly set

# Set the web directory and port
webdir = '.'
port = 8081

if len(sys.argv) > 1:
    webdir = sys.argv[1]

```

```
if len(sys.argv) > 2:
    port = int(sys.argv[2])

print(f"Web directory: {webdir}, Port: {port}")

os.chdir(webdir) # Change to web directory

server_address = ("", port)
httpd = HTTPServer(server_address, MyCGIHandler) # Use the custom handler
print(f"Server running on port {port}...")
httpd.serve_forever()
```

## **Вывод**

В ходе выполнения лабораторной работы был получен опыт по написанию и применению серверных сценариев для обработки данных, введенных пользователем. А также выполнены дополнительные условия из расширенного задания.