

Отчет по лабораторной работе №7 по курсу
" Разработка пользовательского интерфейса для работы с
REST API."

Выполнил:

Студент группы

ИУ5-55Б

Бахман А.А.

Задание:

На основе методических указаний разработайте пользовательский интерфейс для работы с REST API. Используйте REST API, разработанный Вами в предыдущей лабораторной работе.

Текст программы:

server/index.js

```
'use strict';

const express = require('express');
const path = require('path');
const app = express();
const htmlPath = path.resolve(__dirname, '..', 'src/index.html')
const staticPath = path.resolve(__dirname, '..', 'src')
const port = 3000;

app.use('/', express.static(staticPath));

app.all('*', (req, res) => {
  res.sendFile(htmlPath);
});

app.listen(port, () => {
  console.log(`Server listening http://localhost:${port}`);
});
```

src/components/BackButtonComponent.js

```
export class BackButtonComponent {
  constructor(parent) {
    this.parent = parent;
  }

  addListeners(listener) {
    document
      .getElementById("back-button")
      .addEventListener("click", listener)
  }

  getHTML() {
    return (
      `
      <button id="back-button" class="btn btn-primary"
type="button">Назад</button>
      `
    )
  }

  render(listener) {
    const html = this.getHTML()
    this.parent.insertAdjacentHTML('beforeend', html)
    this.addListeners(listener)
  }
}
```

```
}  
}
```

src/components/ButtonComponent.js

```
export class ButtonComponent {  
  constructor(parent) {  
    this.parent = parent;  
  }  
  
  render() {  
    this.parent.insertAdjacentHTML('beforeend', '<button type="button"  
class="btn btn-primary">Hello world 4!</button>')  
  }  
}
```

src/components/PersonComponent.js

```
export class PersonComponent {  
  constructor(parent) {  
    this.parent = parent  
  }  
  
  getHTML(data) {  
    var str = data.sex==true?"Мужчина":"Женщина";  
    return (  
      `<div class="card mb-3" style="width: 540px;">  
        <div class="row g-0">  
          <div class="col-md-4">  
            <!---->  
          </div>  
          <div class="col-md-8">  
            <div class="card-body">  
              <h5 class="card-title">${data.name}</h5>  
              <p class="card-text">${data.age}</p>  
              <p class="card-text">${str}</p>  
            </div>  
          </div>  
        </div>  
      </div>  
    `)  
  }  
  
  render(data) {  
    const html = this.getHTML(data)  
    this.parent.insertAdjacentHTML('beforeend', html)  
  }  
}
```

src/components/PersonCardComponent.js

```
export class PersonCardComponent {  
  constructor(parent) {
```

```

        this.parent = parent;
    }

    getHTML(data) {
        var str = data.sex===true?"Мужчина":"Женщина";
        return (
            `
                <div class="card" style="width: 300px;">
                    <!---->
                    <div class="card-body">
                        <h5 class="card-title">${data.name}</h5>
                        <p class="card-text">${data.age}</p>
                        <p class="card-text">${str}</p>
                        <button class="btn btn-primary" id="click-card-
${data.pk}" data-id="${data.pk}">Подробнее</button>
                    </div>
                </div>
            `
        )
    }

    addListeners(data, listener) {
        document
            .getElementById(`click-card-${data.pk}`)
            .addEventListener("click", listener)
    }

    render(data, listener) {
        const html = this.getHTML(data)
        this.parent.insertAdjacentHTML('beforeend', html)
        this.addListeners(data, listener)
    }

    // render(data) {
    //     const html = this.getHTML(data)
    //     this.parent.insertAdjacentHTML('beforeend', html)
    // }
}

```

src/modules/ajax.js

```

class Ajax {
    async get(url) {
        const response = await fetch(url, {
            //mode: 'no-cors',
            method: "GET",
            headers: {
                "Content-Type": "text/plain"
            },
        });
        const responseData = await response.json();
    }
}

```

```

        return {
            status: response.status,
            data: responseData
        };
    }
}

```

```
export const ajax = new Ajax();
```

src/modules/urls.js

```

class Urls {
    constructor() {
        this.url = 'http://localhost:8000/';
    }

    people() {
        return `${this.url}app/`
    }

    person(id) {
        return `${this.url}app/${id}/`
    }
}

```

```
export const urls = new Urls()
```

src/pages/main/MainPage.js

```

//import {ButtonComponent} from "../../components/button/ButtonComponent.js";
//import {ButtonComponent} from "C:/Users/Xiaomi/Desktop/Study/5
sem/Python/лa67/src/components/button/ButtonComponent.js";

```

```

import {PersonCardComponent} from "../../components/person-
card/PersonCardComponent.js";
import {PersonPage} from "../../person/PersonPage.js";
import {ajax} from "../../modules/ajax.js";
import {urls} from "../../modules/urls.js";

```

```

export class MainPage {
    constructor(parent) {
        this.parent = parent;
    }

    get page() {
        return document.getElementById('main-page')
    }

    getHTML() {
        return (
            `
            <div id="main-page" class="d-flex flex-wrap"><div/>
            `
        )
    }
}

```

```

    )
  }

  // getData() {
  //   return [
  //     {
  //       id: 1,
  //       src:
"https://i.pinimg.com/originals/c9/ea/65/c9ea654eb3a7398b1f702c758c1c4206.jpg",
  //       title: "Акция",
  //       text: "Такой акции вы еще не видели 1"
  //     },
  //     {
  //       id: 2,
  //       src:
"https://i.pinimg.com/originals/c9/ea/65/c9ea654eb3a7398b1f702c758c1c4206.jpg",
  //       title: "Акция",
  //       text: "Такой акции вы еще не видели 2"
  //     },
  //     {
  //       id: 3,
  //       src:
"https://i.pinimg.com/originals/c9/ea/65/c9ea654eb3a7398b1f702c758c1c4206.jpg",
  //       title: "Акция",
  //       text: "Такой акции вы еще не видели 3"
  //     },
  //   ]
  // }

  async getData() {
    return ajax.get(urls.people())
  }

  clickCard(e) {
    const cardId = e.target.dataset.id

    const personPage = new PersonPage(this.parent, cardId)
    personPage.render()
  }

  async render() {
    this.parent.innerHTML = ''
    const html = this.getHTML()
    this.parent.insertAdjacentHTML('beforeend', html)

    const data = await this.getData()
    data.data.forEach((item) => {
      const personCard = new PersonCardComponent(this.page)
      personCard.render(item, this.clickCard.bind(this))
    })
  }
}

```

```
}
```

src/pages/main/PersonPage.js

```
import {PersonComponent} from "../../components/person/PersonComponent.js";
import {MainPage} from "../../main/MainPage.js";
import {BackButtonComponent} from "../../components/back-button/BackButtonComponent.js";
import {ajax} from "../../modules/ajax.js";
import {urls} from "../../modules/urls.js";
export class PersonPage {
  constructor(parent, id) {
    this.parent = parent
    this.id = id
  }

  // getData() {
  //   return {
  //     id: 1,
  //     src:
"https://i.pinimg.com/originals/c9/ea/65/c9ea654eb3a7398b1f702c758c1c4206.jpg",
  //     title: "Акция",
  //     text: "Такой акции вы еще не видели"
  //   }
  // }

  async getData() {
    return ajax.get(urls.person(this.id))
  }
  get page() {
    return document.getElementById('person-page')
  }

  getHTML() {
    return (
      `
      <div id="person-page">
      </div>
    `
  )
  }

  async render() {
    this.parent.innerHTML = ''
    const html = this.getHTML()
    this.parent.insertAdjacentHTML('beforeend', html)

    const data = await this.getData()
    const person = new PersonComponent(this.page)
    person.render(data.data)
  }
}
```

```

        const backButton = new BackButtonComponent(this.page)
        backButton.render(this.clickBack.bind(this))
    }

    clickBack() {
        const mainPage = new MainPage(this.parent)
        mainPage.render()
    }
}

```

src/index.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Lab7</title>
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css">
</head>
<body>
    <div id="root"></div>
    <script src="main.js" type="module"></script>

    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
"></script>
</body>
</html>

```

src/main.js

```

import {MainPage} from "../pages/main/MainPage.js";

const root = document.getElementById('root');


const mainPage = new MainPage(root)
mainPage.render()

```

Результат работы:

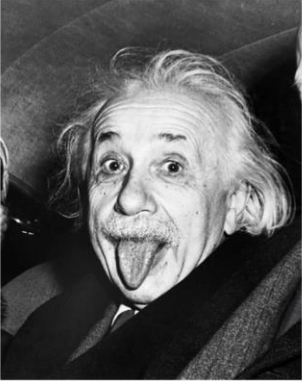
← → ↺ 🏠 localhost:3000 🔍 ☆ ⚙️ 👤

🌐 Сервисы 🌐 Google 🌐 Яндекс 🌐 YouTube 🌐 GISMETEO: погода... 🌐 Википедия — своб... 🌐 WoordHunt (Борд... 🌐 Visio 🌐 Microsoft – офици...




Bauman N.E.
34.00
Мужчина

[Подробнее](#)




EINSTEIN
70.00
Мужчина

[Подробнее](#)



MOZART
35.00
Мужчина

[Подробнее](#)




SELENA GOMEZ
25.00
Женщина

[Подробнее](#)


← → ↺ 🏠 localhost:3000 🔍 ☆ ⚙️ 👤

🌐 Сервисы 🌐 Google 🌐 Яндекс 🌐 YouTube 🌐 GISMETEO: погода... 🌐 Википедия — своб... 🌐 WoordHunt (Борд... 🌐 Visio 🌐 Microsoft – офици...




ANGELINA JOLIE
40.00
Женщина

[Подробнее](#)




BEYONCE
40.00
Женщина

[Подробнее](#)



KIRA KNITLY
30.00
Женщина

[Подробнее](#)



SHAKIRA
40.00
Женщина

[Подробнее](#)



localhost:3000



Сервисы



Google



Яндекс



YouTube



GISMETEO: погода...



Википе



SHAKIRA

40.00

Женщина

Назад