



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»

Лабораторная работа № 7
по курсу “Разработка интернет-приложений”

“Разработка пользовательского интерфейса для работы с REST API”

Выполнил:
студент группы ИУ5-53Б
Волгина А. Д.
15.11.21

Проверил:
Гапанюк Ю. Е.

2021 г.

Задание

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

Текст программы

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

AppleCardComponent.js:

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

  getHTML(data) {
    return (
      <div class="card" style="width: 300px;">
        <div class="card-body">
          <h5 class="card-title">${data.capacity}</h5>
          <p class="card-text">${data.name}</p>
          <button class="btn btn-primary" id="click-card-${data.id}" data-id="${data.id}">Нажми на
меня</button>
        </div>
      </div>
    )
  }

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

```

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

```

AppleComponent.js:

```

export class StockComponent {
  constructor(parent) {
    this.parent = parent
  }

  getHTML(data) {
    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.capacity}</p>
            </div>
          </div>
        </div>
      </div>
    )
  }
}

```

```

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

```

MainPage.js:

```

import {StockPage} from "../apple/AppPage.js";
import {StockCardComponent} from "../components/apple-card/AppCardComponent.js";
import {ajax} from "../modules/ajax.js";
import {urls} from "../modules/urls.js";

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

  async getData() {

```

```

    return ajax.get(urls.stocks())
  }

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

    const stockPage = new StockPage(this.parent, cardId)
    stockPage.render()
  }

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

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

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

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

```

ApplePage.js:

```

import {MainPage} from "../main/MainPage.js";
import {StockComponent} from "../components/apple/AppleComponent.js";
import {BackButtonComponent} from "../components/button/BackButton.js";
import {ajax} from "../modules/ajax.js";
import {urls} from "../modules/urls.js";

export class StockPage {
  constructor(parent, id) {
    this.parent = parent
    this.id = id
  }

  async getData() {
    return ajax.get(urls.stock(this.id))
  }
}

```

```

    }

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

    getHTML() {
        return (
            `
                <div id="stock-page">
                </div>
            `
        )
    }
    clickBack() {
        const mainPage = new MainPage(this.parent)
        mainPage.render()
    }
    async render() {
        this.parent.innerHTML = ""
        const html = this.getHTML()
        this.parent.insertAdjacentHTML('beforeend', html)

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

        const data = await this.getData()
        const stock = new StockComponent(this.page)
        stock.render(data.data)
    }
}

```

Скриншоты:



Яблоко

Фуджи

Нажми на меня



Яблоко

Ред Делишес

Нажми на меня



Яблоко

Голден Делишес

Нажми на меня

[Назад](#)



Яблоко

Фуджи