

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ



МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

*Факультет Информационных технологий
Кафедра Информатики и информационных технологий*

направление подготовки

09.03.02 «Информационные системы и технологии»

ЛАБОРАТОРНАЯ РАБОТА № 4

Дисциплина: Функциональное программирование

Тема: Применение функционального программирования в TypeScript

Выполнил(а): студент(ка) группы 221-3710

Сычугова П.А.

(Фамилия И.О.)

Дата, подпись _____
(Дата) (Подпись)

Проверил: _____
(Фамилия И.О., степень, звание) (Оценка)

Дата, подпись _____
(Дата) (Подпись)

Замечания: _____

Москва

2025

Цель: Применить принципы функционального программирования для разработки небольшого веб-приложения на TypeScript.

Задание:

Разработайте веб-приложение "Калькулятор", которое позволяет пользователю выполнять следующие операции:

- Сложение, вычитание, умножение и деление.
- Возведение в степень.
- Вычисление квадратного корня.

Ход работы

1. Листинг основного кода app.ts

```
interface CalcData {
  currentText: string;
  lastText: string;
  currentOp: string | null;
  justCalculated: boolean;
}

const defaultCalcData: CalcData = {
  currentText: "0",
  lastText: "",
  currentOp: null,
  justCalculated: false,
};

const calcActions: Record<string, (x: number, y: number) => number> = {
  "+": (x, y) => x + y,
  "-": (x, y) => x - y,
  "*": (x, y) => x * y,
  "/": (x, y) => x / y,
```

```
"^": (x, y) => Math.pow(x, y),
};

function addNumber(data: CalcData, num: string): CalcData {
  if (data.justCalculated) {
    return { ...data, currentText: num, justCalculated: false };
  }
  if (data.currentText === "0") {
    return { ...data, currentText: num };
  }
  return { ...data, currentText: data.currentText + num };
}

function setOperation(data: CalcData, op: string): CalcData {
  if (data.currentOp && data.lastText !== "") {
    const res = executeCalc(data);
    return {
      ...data,
      currentText: res,
      lastText: res,
      currentOp: op,
      justCalculated: true,
    };
  }
  return { ...data, lastText: data.currentText, currentOp: op, justCalculated: true };
}

function executeCalc(data: CalcData): string {
  if (!data.currentOp || !data.lastText) return data.currentText;
```

```

const x = parseFloat(data.lastText);
const y = parseFloat(data.currentText);
const action = calcActions[data.currentOp];

if (data.currentOp === "/" && y === 0) return "Ошибка";

const result = action(x, y);
return String(result);
}

function pressEquals(data: CalcData): CalcData {
  if (!data.currentOp) return data;
  const res = executeCalc(data);
  return { currentText: res, lastText: "", currentOp: null, justCalculated: true };
}

function calcSquareRoot(data: CalcData): CalcData {
  const val = parseFloat(data.currentText);
  if (val < 0) {
    return { ...data, currentText: "Ошибка", justCalculated: true };
  }
  const res = Math.sqrt(val);
  return { ...data, currentText: String(res), justCalculated: true };
}

function toggleSign(data: CalcData): CalcData {
  if (data.currentText.charAt(0) === "-" || data.currentText === "Ошибка") return
data;

  if (data.currentText.charAt(0) === "-") {

```

```

    return { ...data, currentText: data.currentText.slice(1) };
  } else {
    return { ...data, currentText: "-" + data.currentText };
  }
}

function resetCalc(): CalcData {
  return { ...defaultCalcData };
}

function updateDisplay(data: CalcData) {
  const screen = document.getElementById("screen") as HTMLInputElement;
  screen.value = data.currentText;
}

function startCalcApp() {
  let calcData: CalcData = { ...defaultCalcData };

  const numberBtns = document.querySelectorAll<HTMLButtonElement>(".num-
btn");
  numberBtns.forEach((btn) => {
    btn.addEventListener("click", () => {
      const digit = btn.getAttribute("data-num")!;
      calcData = addNumber(calcData, digit);
      updateDisplay(calcData);
    });
  });

  const opBtns = document.querySelectorAll<HTMLButtonElement>(".op-btn");
  opBtns.forEach((btn) => {

```

```
    btn.addEventListener("click", () => {
        const oper = btn.getAttribute("data-op")!;
        calcData = setOperation(calcData, oper);
        updateDisplay(calcData);
    });
});

const eqBtn = document.getElementById("btn-equal")!;
eqBtn.addEventListener("click", () => {
    calcData = pressEquals(calcData);
    updateDisplay(calcData);
});

const sqrtBtn = document.getElementById("btn-root")!;
sqrtBtn.addEventListener("click", () => {
    calcData = calcSquareRoot(calcData);
    updateDisplay(calcData);
});

const clrBtn = document.getElementById("btn-reset")!;
clrBtn.addEventListener("click", () => {
    calcData = resetCalc();
    updateDisplay(calcData);
});

const signBtn = document.getElementById("btn-sign")!;
signBtn.addEventListener("click", () => {
    calcData = toggleSign(calcData);
    updateDisplay(calcData);
});
```

```
updateDisplay(calcData);  
}  
  
window.addEventListener("DOMContentLoaded", startCalcApp);
```

2. Листинг HTML страницы index.html

```
<!DOCTYPE html>  
<html lang="ru">  
  
  <head>  
    <meta charset="UTF-8" />  
    <title>Калькулятор</title>  
    <link rel="stylesheet" href="style.css" />  
  </head>  
  
  <body>  
    <div class="calculator-container">  
      <input type="text" id="screen" disabled value="0" />  
      <div class="row">  
        <button class="num-btn" data-num="7">7</button>  
        <button class="num-btn" data-num="8">8</button>  
        <button class="num-btn" data-num="9">9</button>  
        <button class="op-btn" data-op="/">÷</button>  
      </div>  
      <div class="row">  
        <button class="num-btn" data-num="4">4</button>  
        <button class="num-btn" data-num="5">5</button>  
        <button class="num-btn" data-num="6">6</button>  
        <button class="op-btn" data-op="*">×</button>
```

```

</div>
<div class="row">
  <button class="num-btn" data-num="1">1</button>
  <button class="num-btn" data-num="2">2</button>
  <button class="num-btn" data-num="3">3</button>
  <button class="op-btn" data-op="-">−</button>
</div>
<div class="row">
  <button class="num-btn" data-num="0">0</button>
  <button class="op-btn" data-op="+">+</button>
  <button class="op-btn" data-op="^">x^y</button>
  <button id="btn-root">√</button>
</div>
<div class="row">
  <button id="btn-sign">±</button>
  <button id="btn-equal">=</button>
  <button id="btn-reset">C</button>
</div>
</div>
<script src="dist/app.js"></script>
</body>

</html>

```

3. Стили style.css

```

body {
  font-family: 'Arial', sans-serif;
  background: linear-gradient(135deg, #667eea, #764ba2);
  display: flex;
  justify-content: center;
  align-items: center;

```



```
    height: 100vh;
    margin: 0;
}

.calculator-container {
    background-color: #1e1e2f;
    border-radius: 15px;
    padding: 25px 30px;
    box-shadow: 0 0 15px rgba(118, 75, 162, 0.7);
    width: 260px;
}

#screen {
    width: 100%;
    height: 50px;
    border: none;
    border-radius: 10px;
    margin-bottom: 20px;
    font-size: 2rem;
    text-align: right;
    padding-right: 15px;
    background-color: #2a2a40;
    color: #f5f5f7;
}

.row {
    display: flex;
    margin-bottom: 12px;
}
```

```
button {  
    flex: 1;  
    margin: 0 6px 0 0;  
    padding: 15px 0;  
    font-size: 1.1rem;  
    border: none;  
    border-radius: 8px;  
    cursor: pointer;  
    transition: background-color 0.3s ease;  
    color: #fff;  
    font-weight: 600;  
}
```

```
button:last-child {  
    margin-right: 0;  
}
```

```
.num-btn {  
    background-color: #3b3b5c;  
}
```

```
.num-btn:hover {  
    background-color: #4e4e7a;  
}
```

```
.op-btn {  
    background-color: #805ad5;  
}
```

```
.op-btn:hover {
```

```
    background-color: #9f7aea;
}

#btn-equal {
    background-color: #48bb78;
    flex: 2;
}

#btn-equal:hover {
    background-color: #38a169;
}

#btn-reset {
    background-color: #e53e3e;
    flex: 1;
}

#btn-reset:hover {
    background-color: #c53030;
}

#btn-root {
    background-color: #319795;
}

#btn-root:hover {
    background-color: #2c7a7b;
}

#btn-sign {
```

```
background-color: #6c757d;  
color: white;  
font-weight: bold;  
}
```

Пример работы:

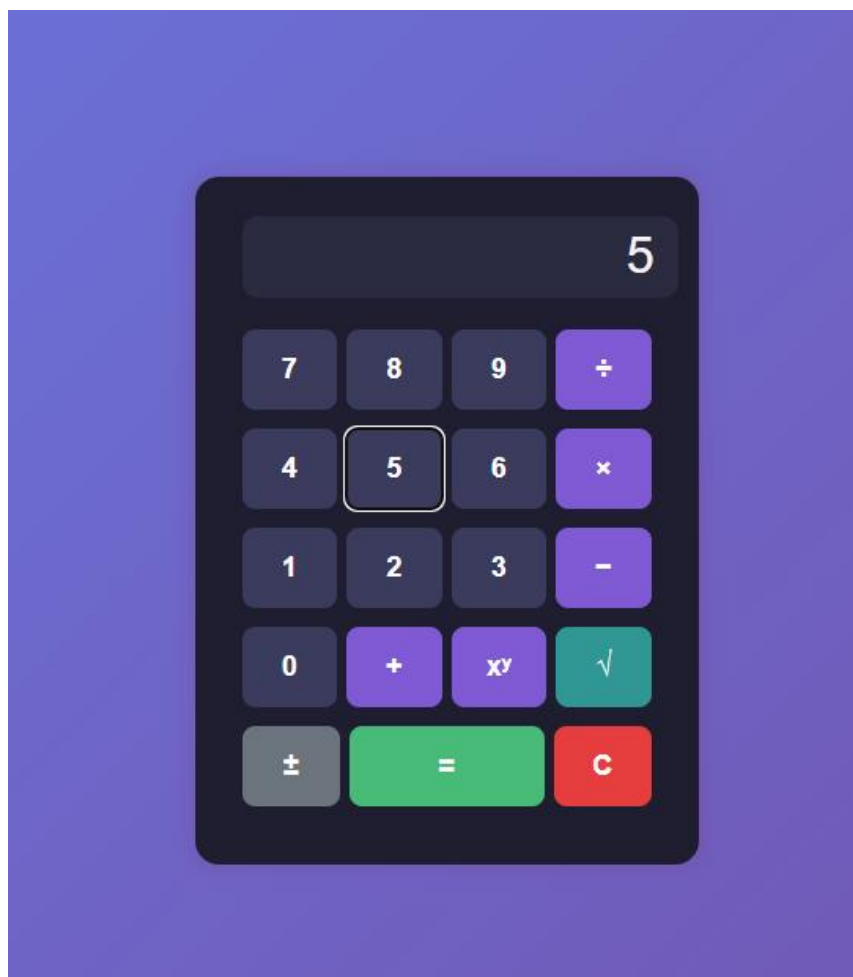


Рисунок 1. Пример калькулятора

Репозиторий: <https://github.com/AnekiChan/fp-lab4>