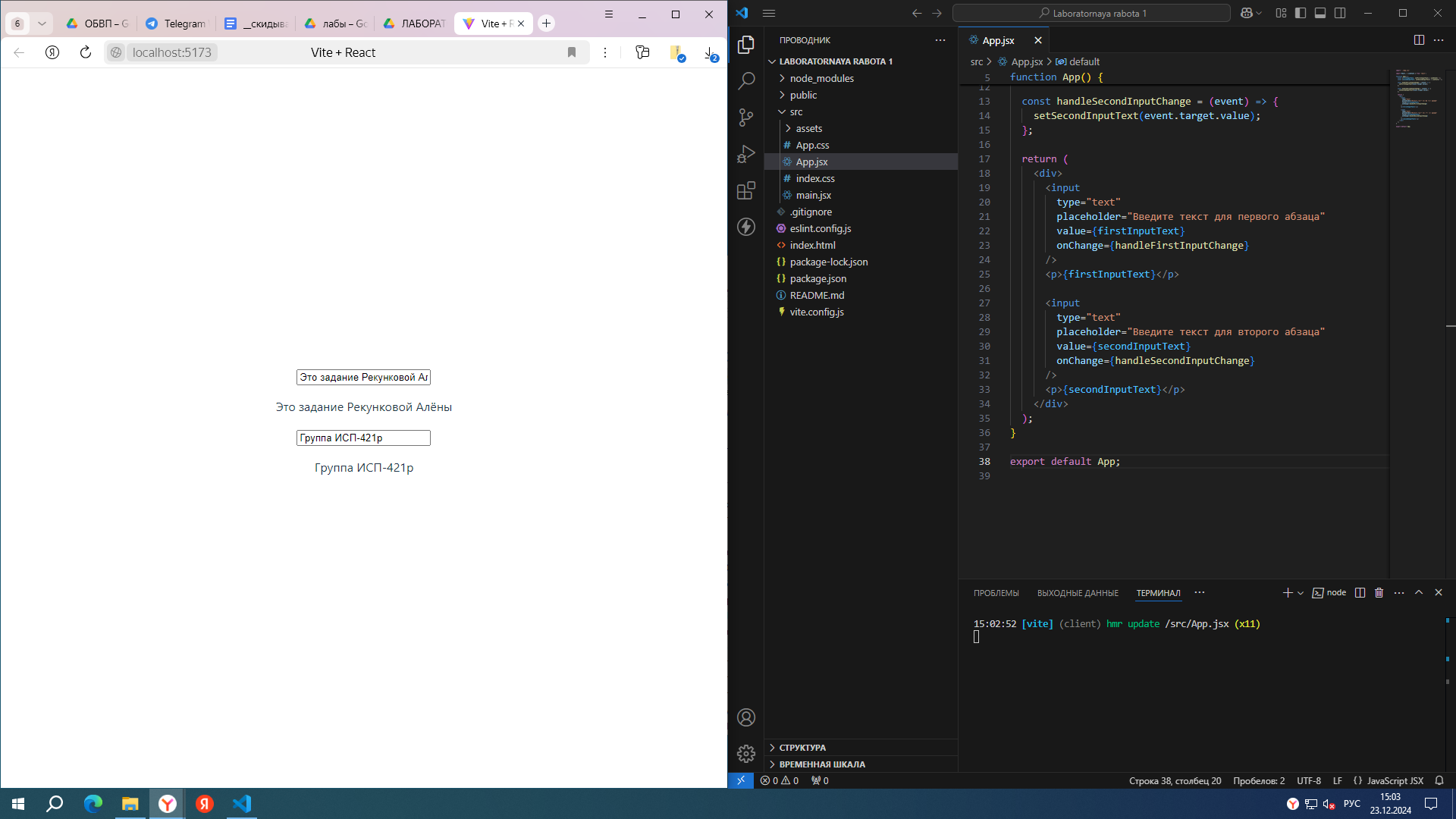
Лабораторная работа №2

Рекункова Алёна ИСП-421р

Задание №1

Сделайте два инпута. Пусть текст первого инпута выводится в первый абзац, а текст второго инпута - во второй абзац.



import './App.css'

import React, { useState } from 'react';

function App() {

const [firstInputText, setFirstInputText] = useState('');

const [secondInputText, setSecondInputText] = useState('');

const handleFirstInputChange = (event) => {

setFirstInputText(event.target.value);

};

const handleSecondInputChange = (event) => {

setSecondInputText(event.target.value);

};

return (

<div>

<input

type="text"

placeholder="Введите текст для первого абзаца"

value={firstInputText}

onChange={handleFirstInputChange}

/>

<p>{firstInputText}</p>

<input

type="text"

placeholder="Введите текст для второго абзаца"

value={secondInputText}

onChange={handleSecondInputChange}

/>

<p>{secondInputText}</p>

</div>

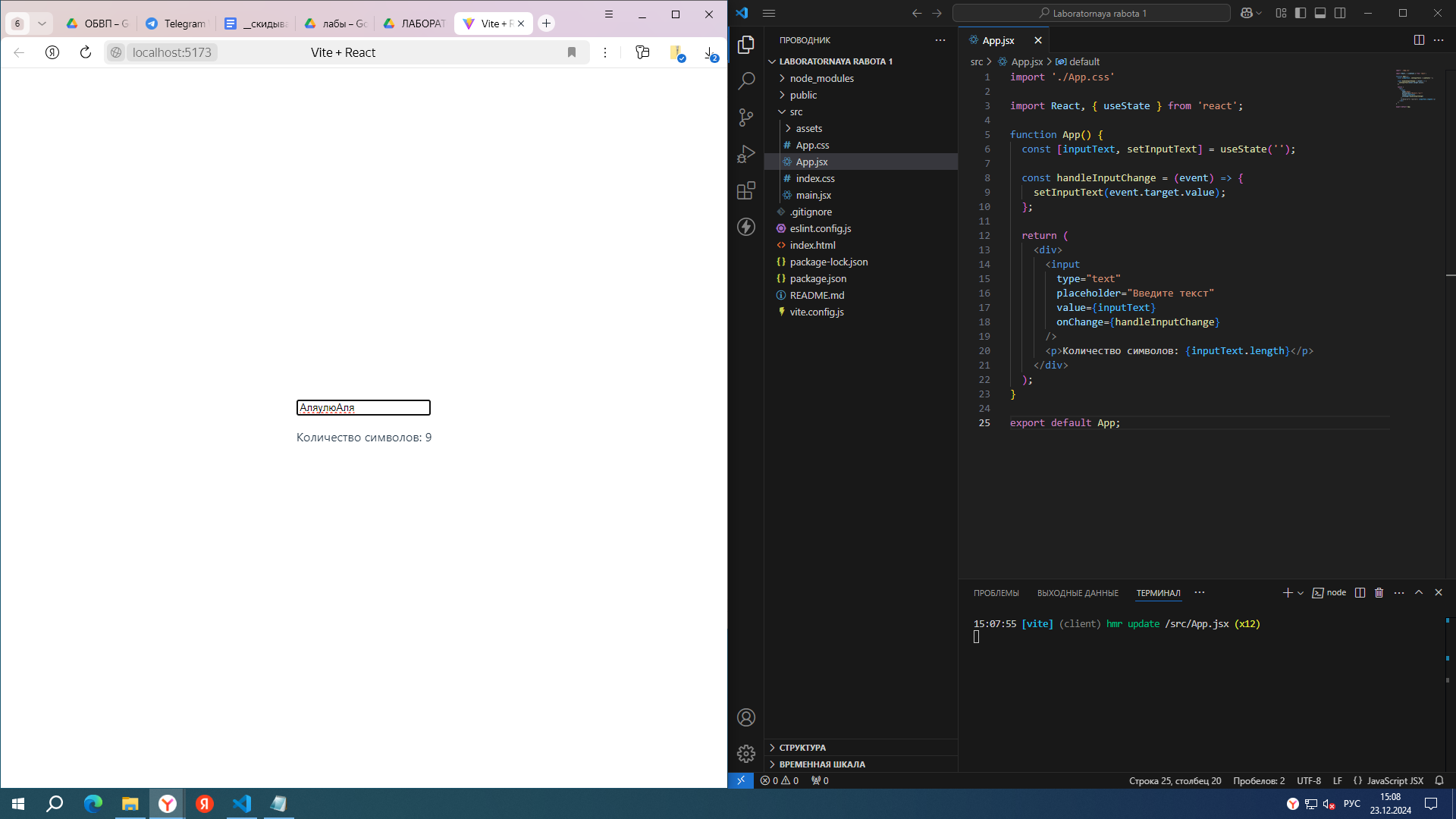
);

}

export default App;

Задание №1

Дан инпут. Дан абзац. Сделайте так, чтобы при вводе текста в инпут, в абзаце выводилось количество введенных в инпут символов.



import './App.css'

import React, { useState } from 'react';

function App() {

const [inputText, setInputText] = useState('');

const handleInputChange = (event) => {

setInputText(event.target.value);

};

return (

<div>

<input

type="text"

placeholder="Введите текст"

value={inputText}

onChange={handleInputChange}

/>

<p>Количество символов: {inputText.length}</p>

</div>

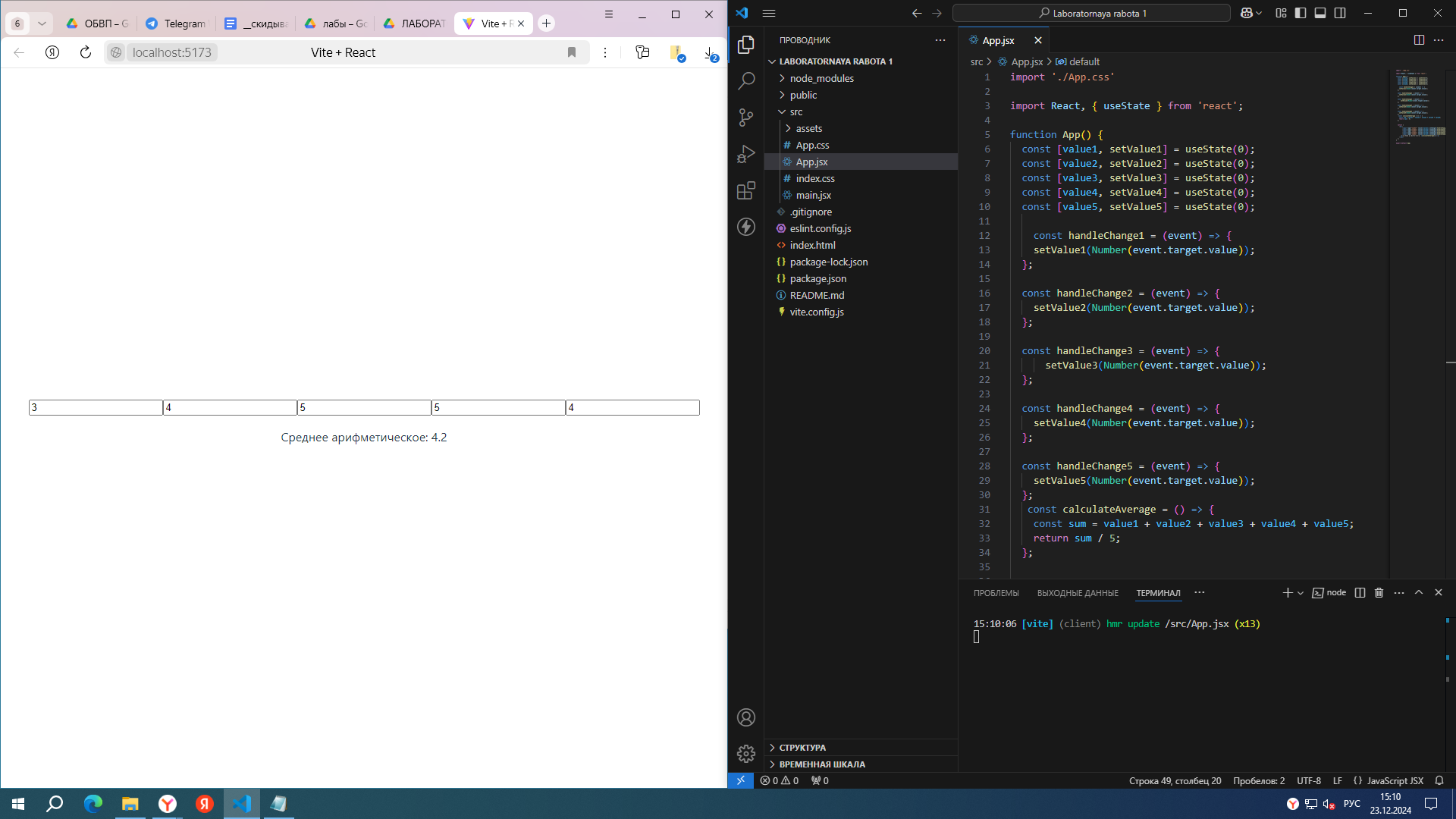
);

}

export default App;

Задание №1

Даны 5 инпутов. Сделайте так, чтобы при вводе чисел в наши инпуты в абзац выводилось среднее арифметическое введенных чисел.



import './App.css'

import React, { useState } from 'react';

function App() {

const [value1, setValue1] = useState(0);

const [value2, setValue2] = useState(0);

const [value3, setValue3] = useState(0);

const [value4, setValue4] = useState(0);

const [value5, setValue5] = useState(0);

const handleChange1 = (event) => {

setValue1(Number(event.target.value));

};

const handleChange2 = (event) => {

setValue2(Number(event.target.value));

};

const handleChange3 = (event) => {

setValue3(Number(event.target.value));

};

const handleChange4 = (event) => {

setValue4(Number(event.target.value));

};

const handleChange5 = (event) => {

setValue5(Number(event.target.value));

};

const calculateAverage = () => {

const sum = value1 + value2 + value3 + value4 + value5;

return sum / 5;

};

return (

<div>

<input type="number" value={value1} onChange={handleChange1} />

<input type="number" value={value2} onChange={handleChange2} />

<input type="number" value={value3} onChange={handleChange3} />

<input type="number" value={value4} onChange={handleChange4} />

<input type="number" value={value5} onChange={handleChange5} />

<p>Среднее арифметическое: {calculateAverage()}</p>

</div>

);

}

export default App;

Задание №1

Дан чекбокс. С помощью чекбокса спросите у пользователя, если ли ему уже 18 лет. Если чекбокс отмечен, покажите пользователю следующий блок кода:

<div>

<h2>Ура, вам уже есть 18</h2>

<p> здесь расположен контент только для взрослых </p>

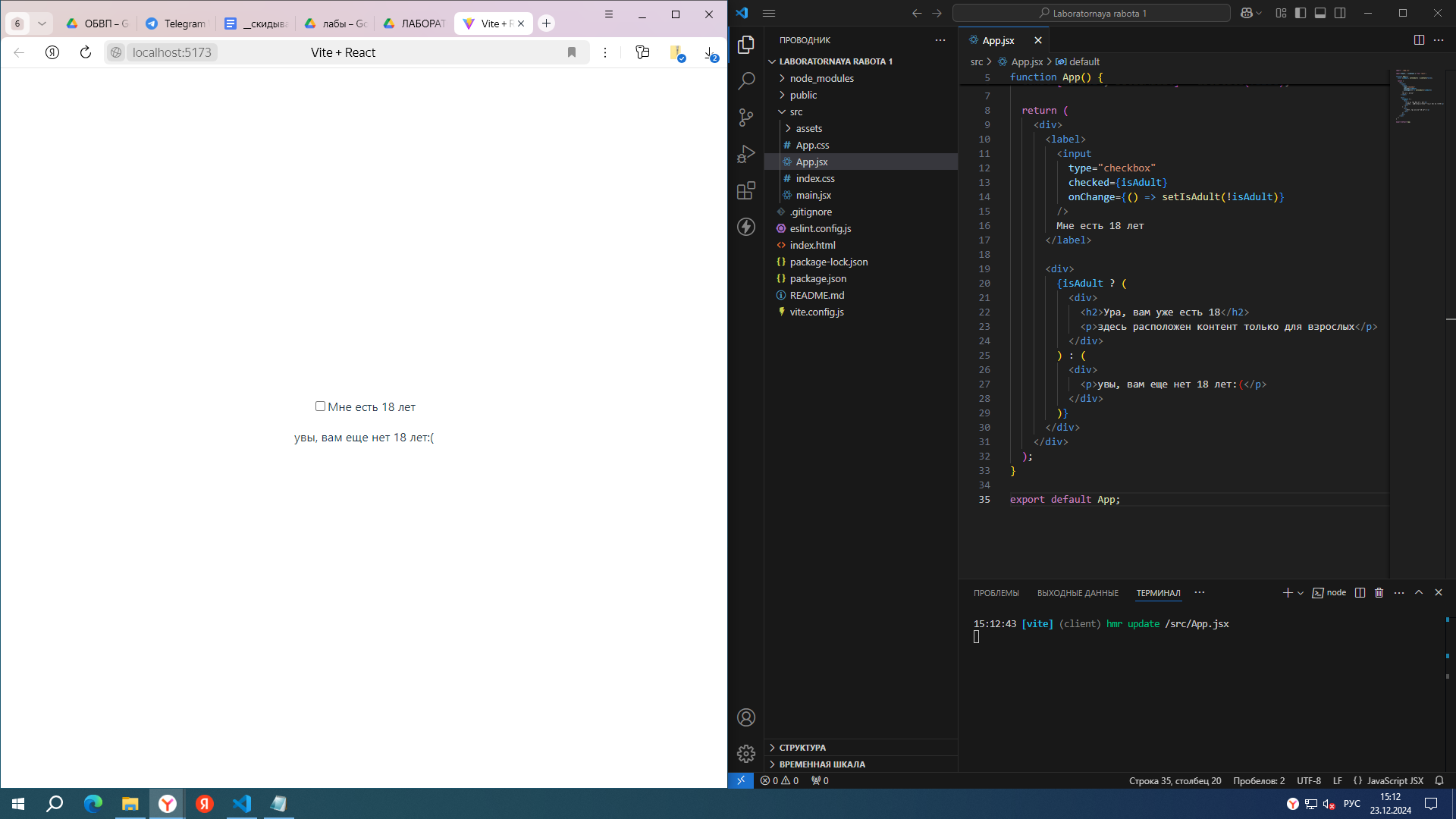
</div>;

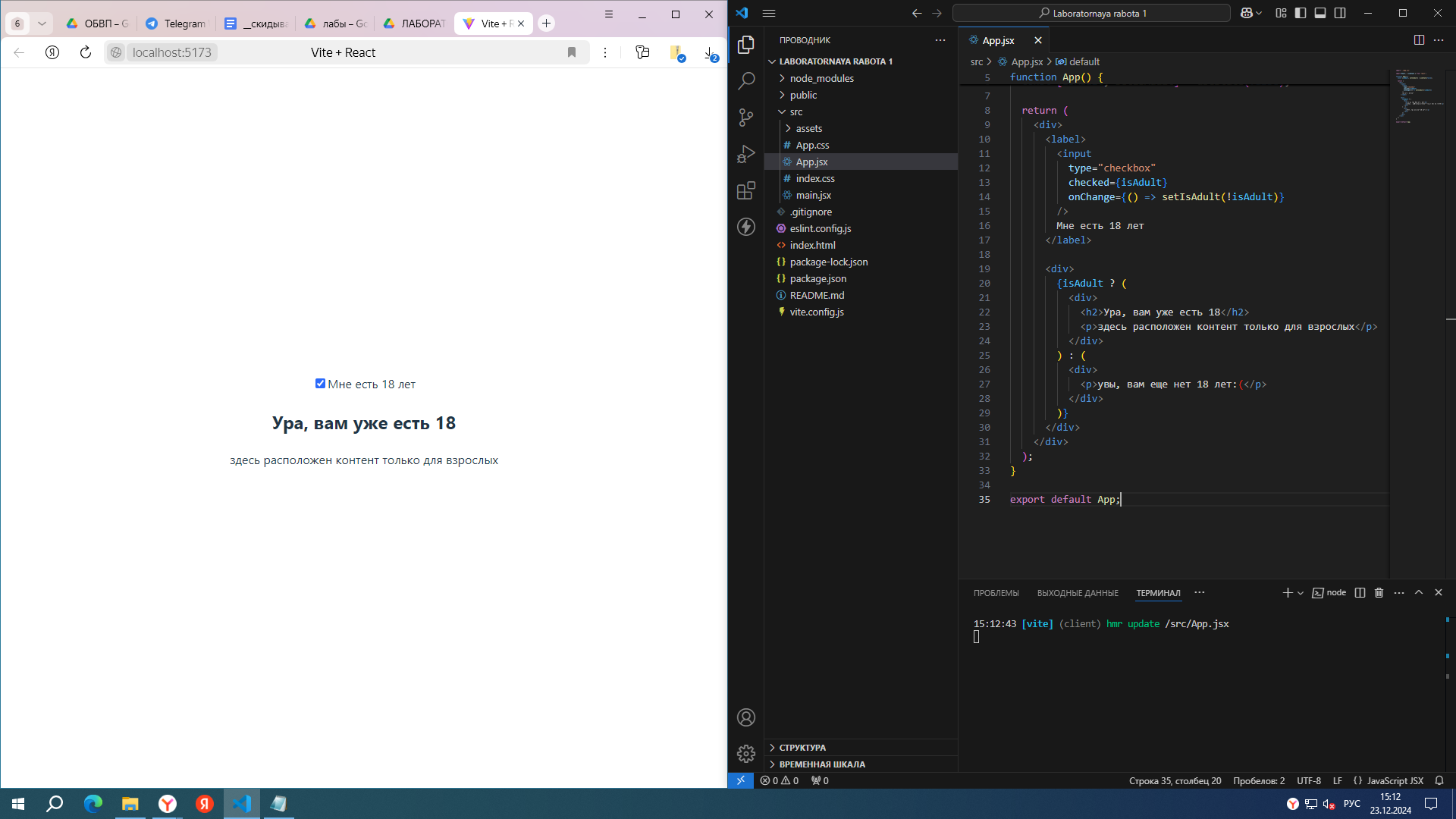
А если чекбокс не отмечен - то следующий:

<div>

<p> увы, вам еще нет 18 лет:( </p>

</div>;





import './App.css'

import React, { useState } from 'react';

function App() {

const [isAdult, setIsAdult] = useState(false);

return (

<div>

<label>

<input

type="checkbox"

checked={isAdult}

onChange={() => setIsAdult(!isAdult)}

/>

Мне есть 18 лет

</label>

<div>

{isAdult ? (

<div>

<h2>Ура, вам уже есть 18</h2>

<p>здесь расположен контент только для взрослых</p>

</div>

) : (

<div>

<p>увы, вам еще нет 18 лет:(</p>

</div>

)}

</div>

</div>

);

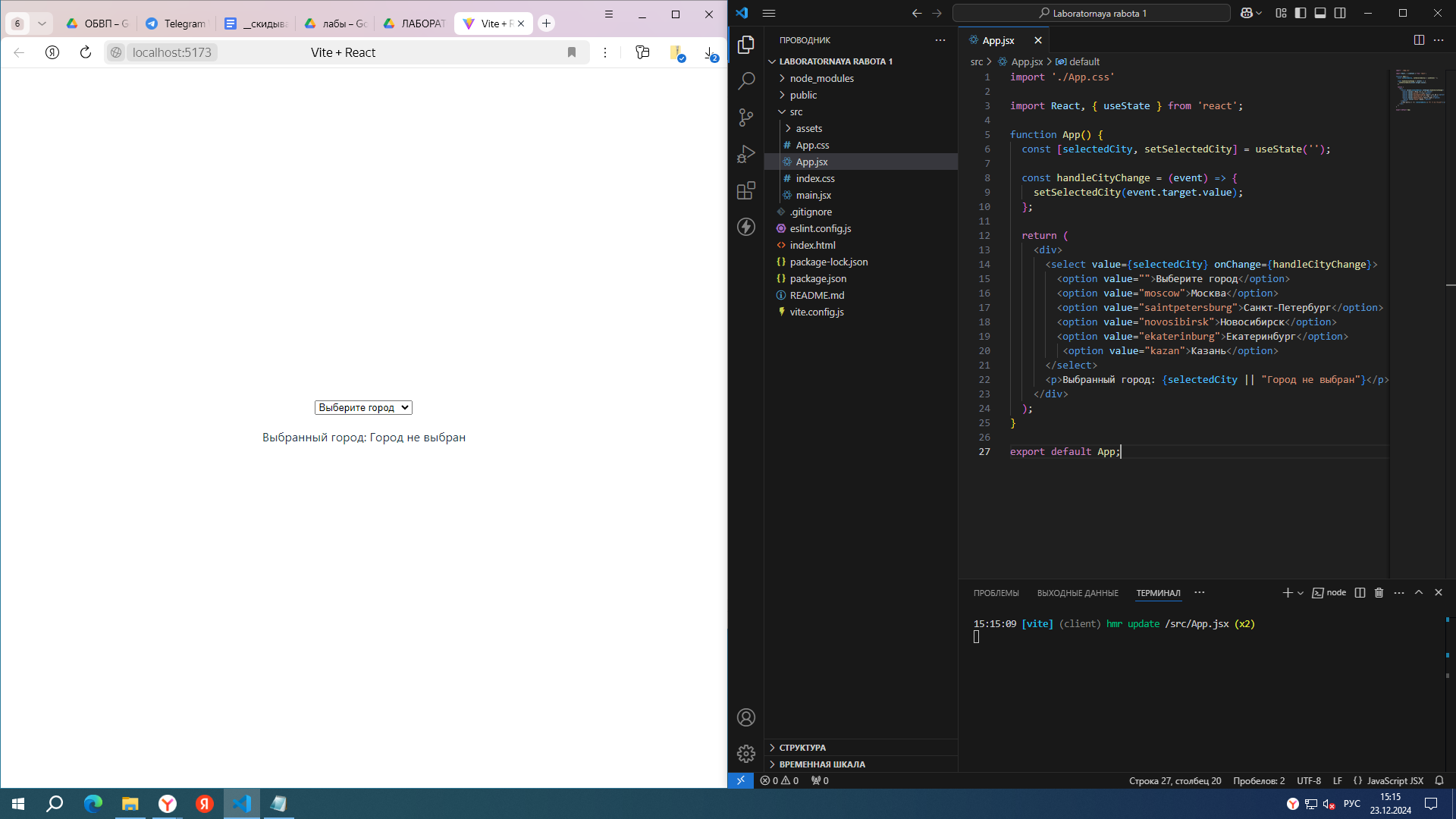
}

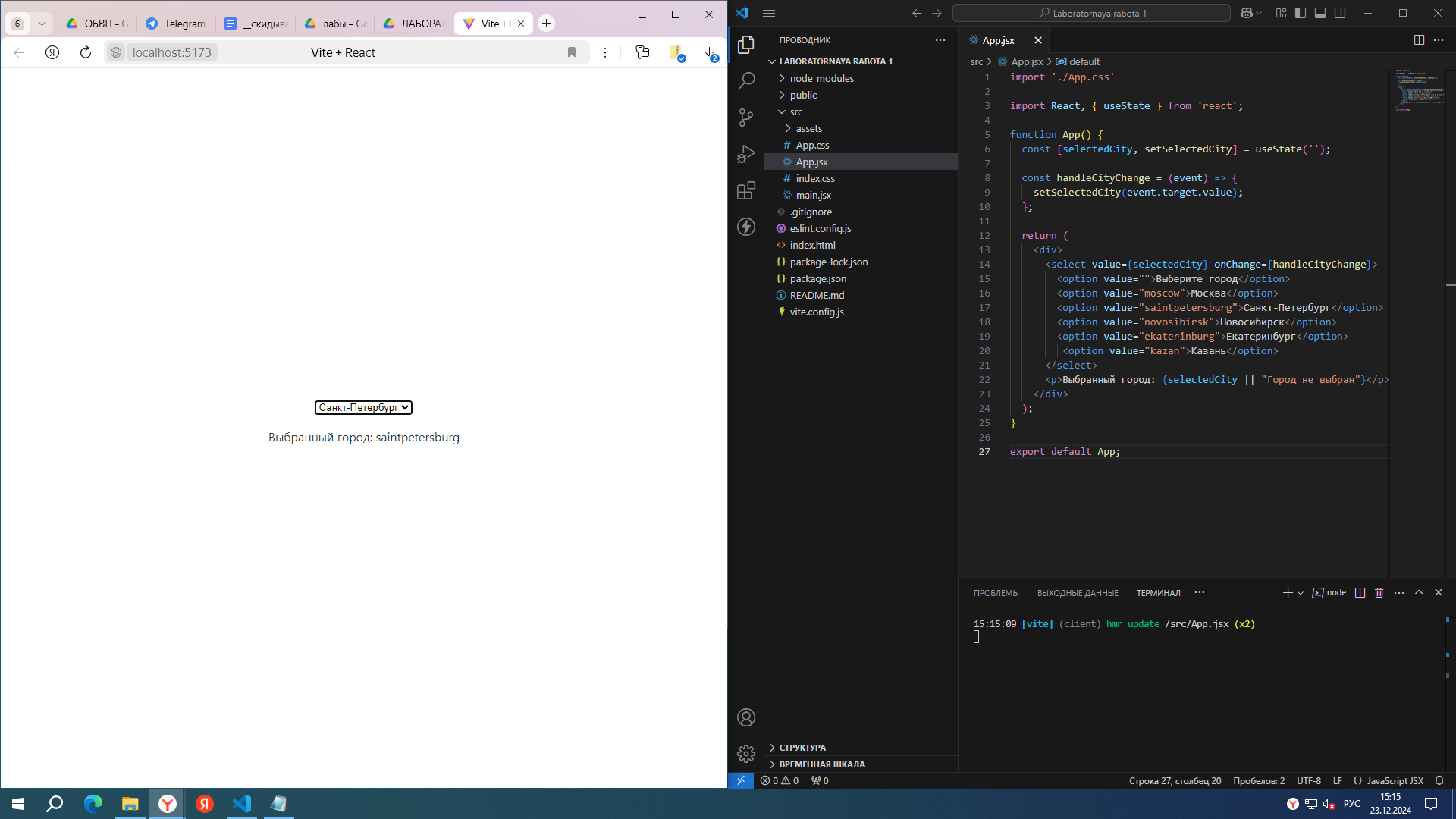
export default App;

Задание №1

Сделайте выпадающий список городов. Сделайте также абзац, в который будет

выводиться выбор пользователя.





import './App.css'

import React, { useState } from 'react';

function App() {

const [selectedCity, setSelectedCity] = useState('');

const handleCityChange = (event) => {

setSelectedCity(event.target.value);

};

return (

<div>

<select value={selectedCity} onChange={handleCityChange}>

<option value="">Выберите город</option>

<option value="moscow">Москва</option>

<option value="saintpetersburg">Санкт-Петербург</option>

<option value="novosibirsk">Новосибирск</option>

<option value="ekaterinburg">Екатеринбург</option>

<option value="kazan">Казань</option>

</select>

<p>Выбранный город: {selectedCity || "Город не выбран"}</p>

</div>

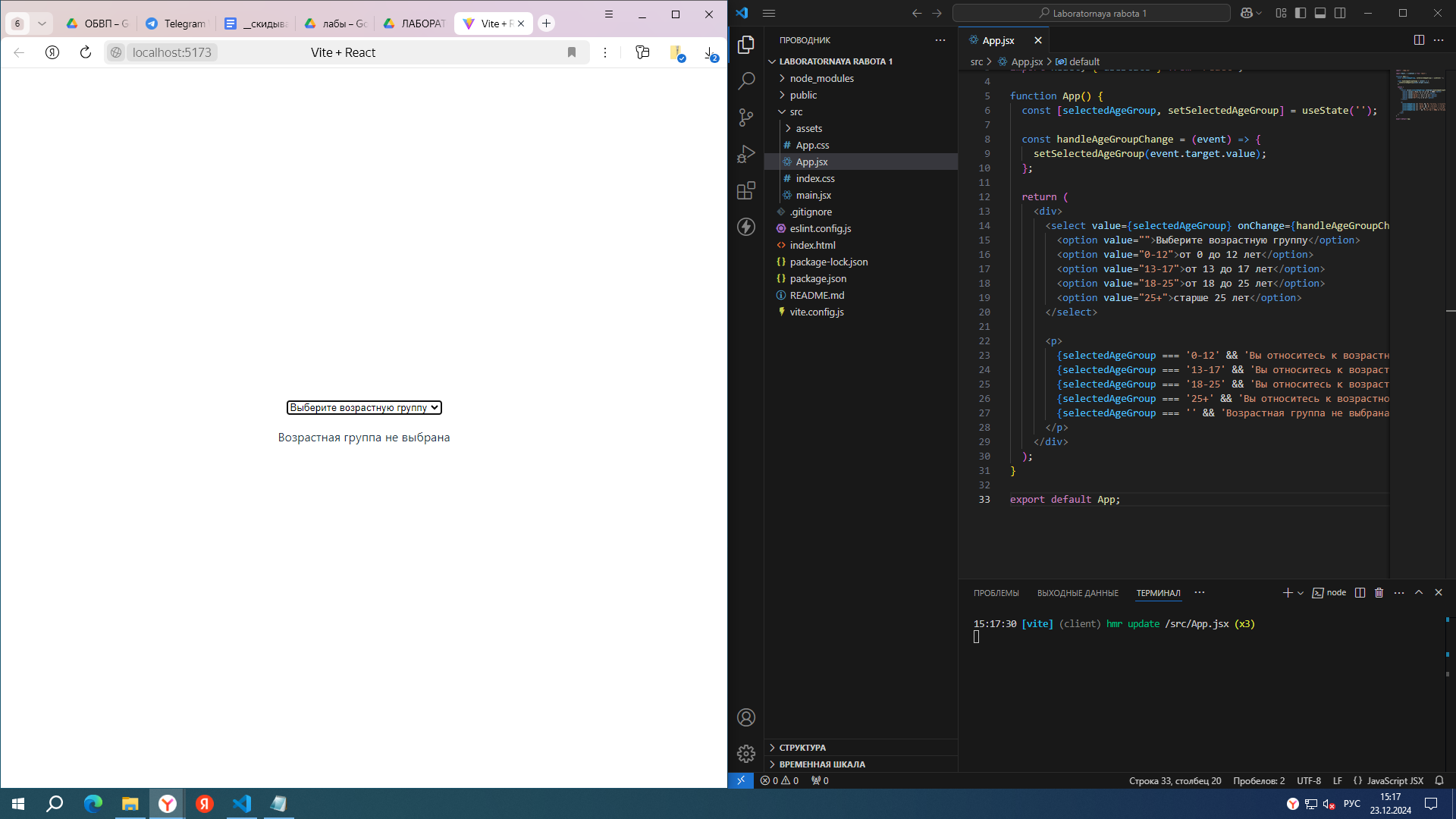
);

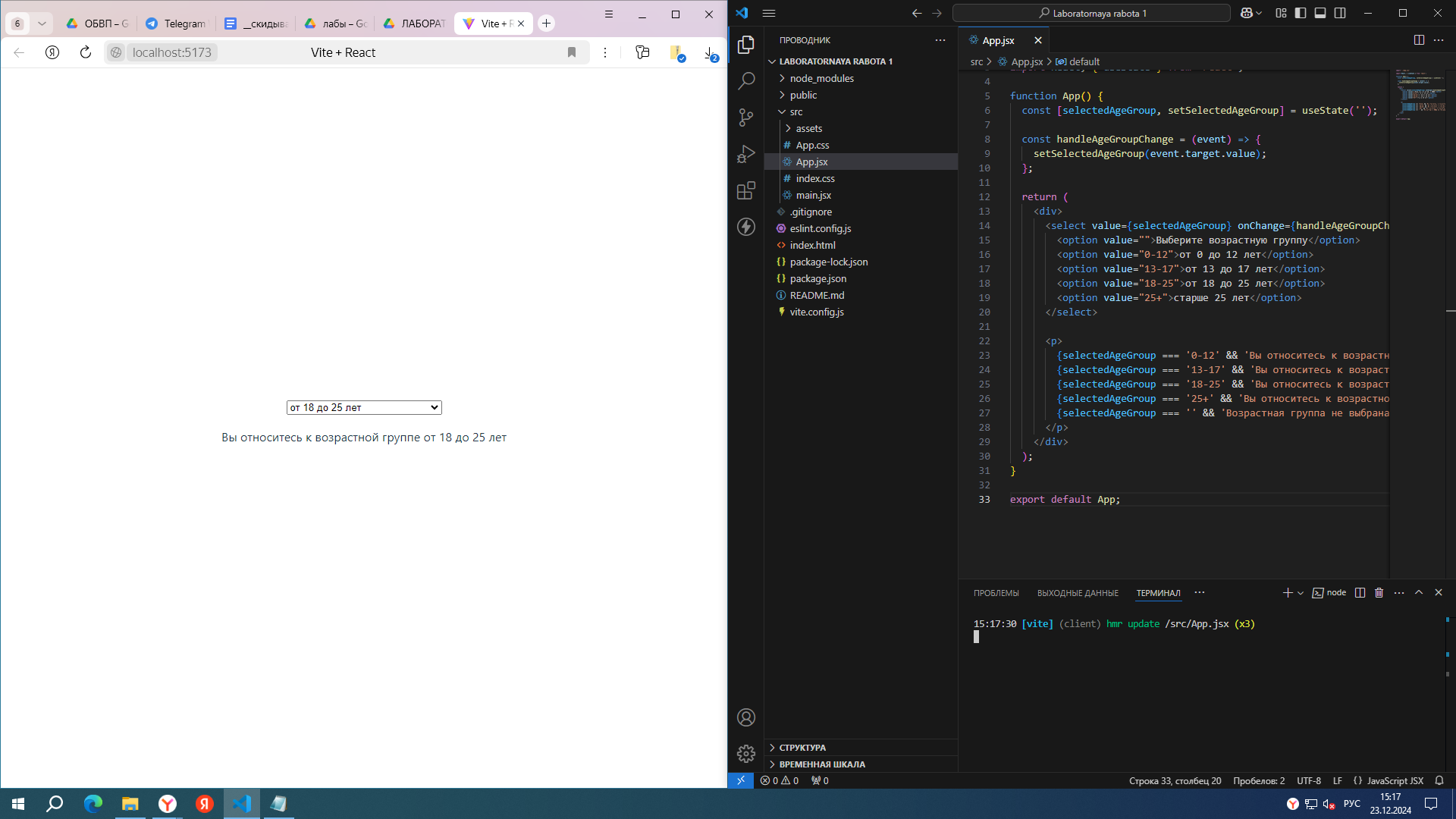
}

export default App;

Задание №1

С помощью выпадающего списка предложите пользователю выбрать к какой возрастной группе он относится: от 0 до 12 лет, от 13 до 17, от 18 до 25, либо старше 25 лет.





import './App.css'

import React, { useState } from 'react';

function App() {

const [selectedAgeGroup, setSelectedAgeGroup] = useState('');

const handleAgeGroupChange = (event) => {

setSelectedAgeGroup(event.target.value);

};

return (

<div>

<select value={selectedAgeGroup} onChange={handleAgeGroupChange}>

<option value="">Выберите возрастную группу</option>

<option value="0-12">от 0 до 12 лет</option>

<option value="13-17">от 13 до 17 лет</option>

<option value="18-25">от 18 до 25 лет</option>

<option value="25+">старше 25 лет</option>

</select>

<p>

{selectedAgeGroup === '0-12' && 'Вы относитесь к возрастной группе от 0 до 12 лет'}

{selectedAgeGroup === '13-17' && 'Вы относитесь к возрастной группе от 13 до 17 лет'}

{selectedAgeGroup === '18-25' && 'Вы относитесь к возрастной группе от 18 до 25 лет'}

{selectedAgeGroup === '25+' && 'Вы относитесь к возрастной группе старше 25 лет'}

{selectedAgeGroup === '' && 'Возрастная группа не выбрана'}

</p>

</div>

);

}

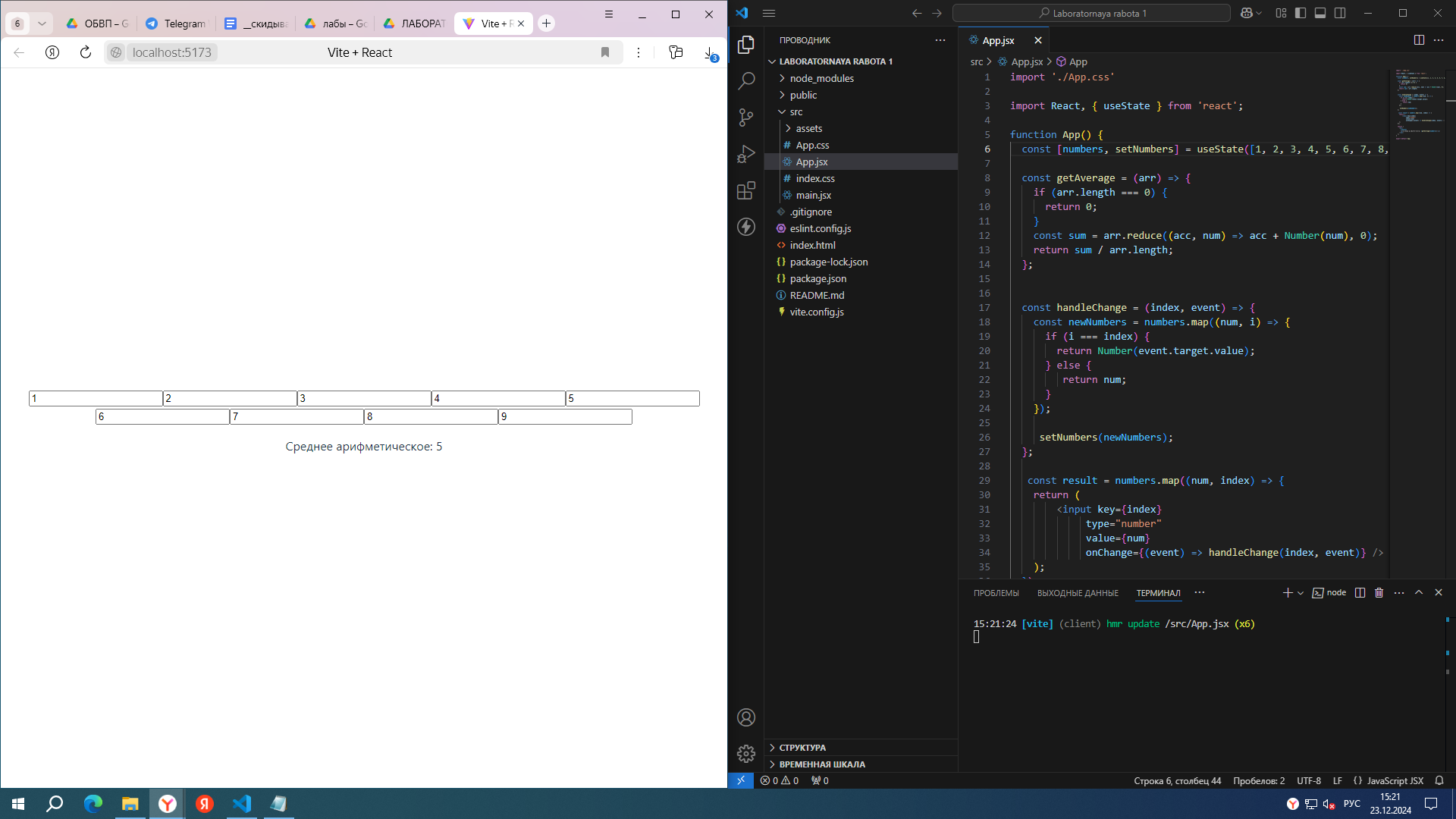
export default App;

Задание №1

Дан следующий массив:

[1, 2, 3, 4, 5, 6, 7, 8, 9];

Выведите на экран среднее арифметическое элементов этого массива. В цикле сделайте инпуты для редактирования элементов.



import './App.css'

import React, { useState } from 'react';

function App() {

const [numbers, setNumbers] = useState([1, 2, 3, 4, 5, 6, 7, 8, 9]);

const getAverage = (arr) => {

if (arr.length === 0) {

return 0;

}

const sum = arr.reduce((acc, num) => acc + Number(num), 0);

return sum / arr.length;

};

const handleChange = (index, event) => {

const newNumbers = numbers.map((num, i) => {

if (i === index) {

return Number(event.target.value);

} else {

return num;

}

});

setNumbers(newNumbers);

};

const result = numbers.map((num, index) => {

return (

<input key={index}

type="number"

value={num}

onChange={(event) => handleChange(index, event)} />

);

});

return (

<div>

{result}

<p>Среднее арифметическое: {getAverage(numbers)}</p>

</div>

);

}

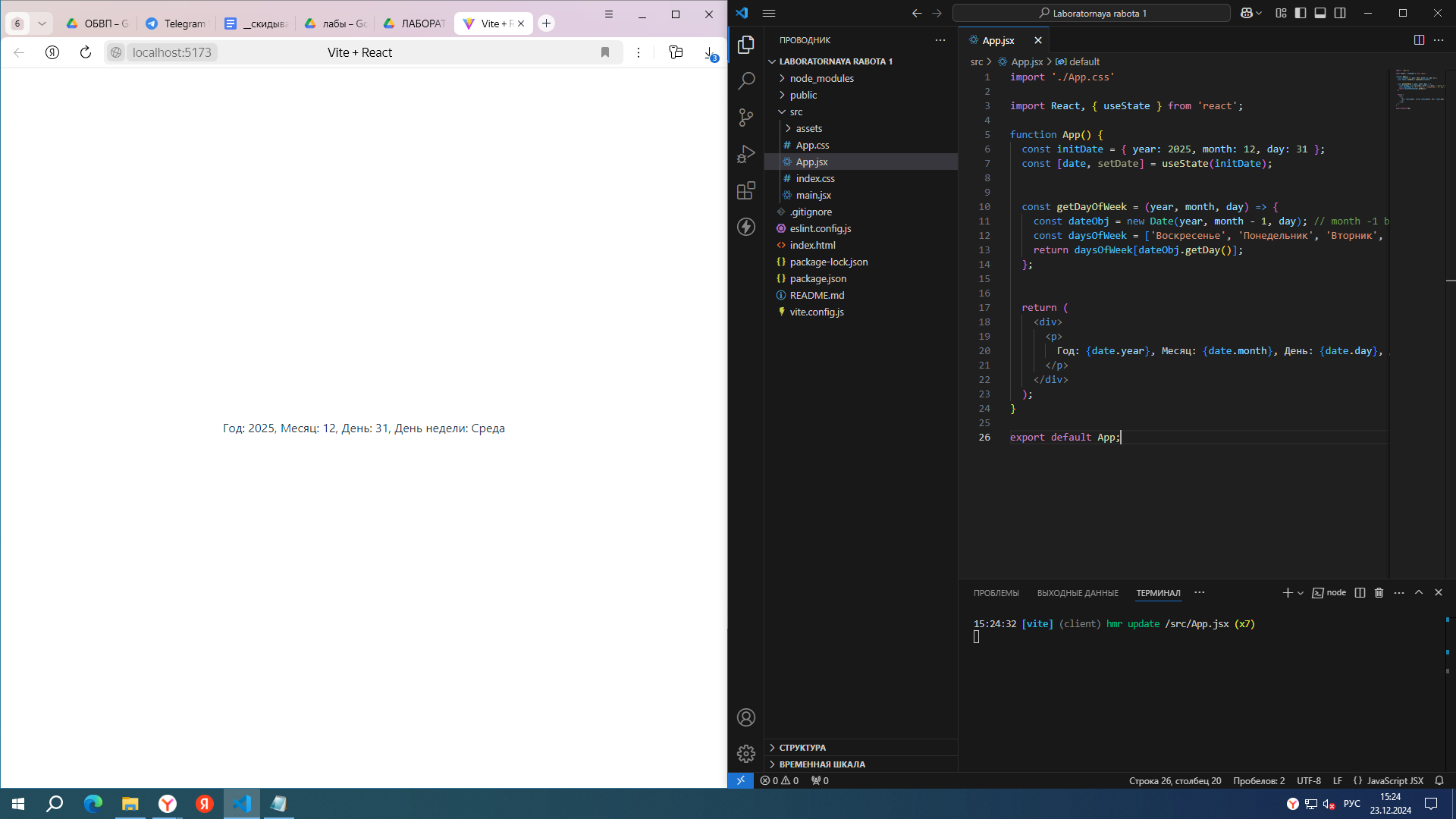
export default App;

Задание №1

Пусть в стейте хранится объект с датой:

const initDate = { year: 2025, month: 12, day: 31 };

Выведите в абзаце год, месяц и день из даты, хранящейся в стейте, а также день недели, соответствующий ей.



import './App.css'

import React, { useState } from 'react';

function App() {

const initDate = { year: 2025, month: 12, day: 31 };

const [date, setDate] = useState(initDate);

const getDayOfWeek = (year, month, day) => {

const dateObj = new Date(year, month - 1, day); // month -1 because months are 0-indexed

const daysOfWeek = ['Воскресенье', 'Понедельник', 'Вторник', 'Среда', 'Четверг', 'Пятница', 'Суббота'];

return daysOfWeek[dateObj.getDay()];

};

return (

<div>

<p>

Год: {date.year}, Месяц: {date.month}, День: {date.day}, День недели: {getDayOfWeek(date.year, date.month, date.day)}

</p>

</div>

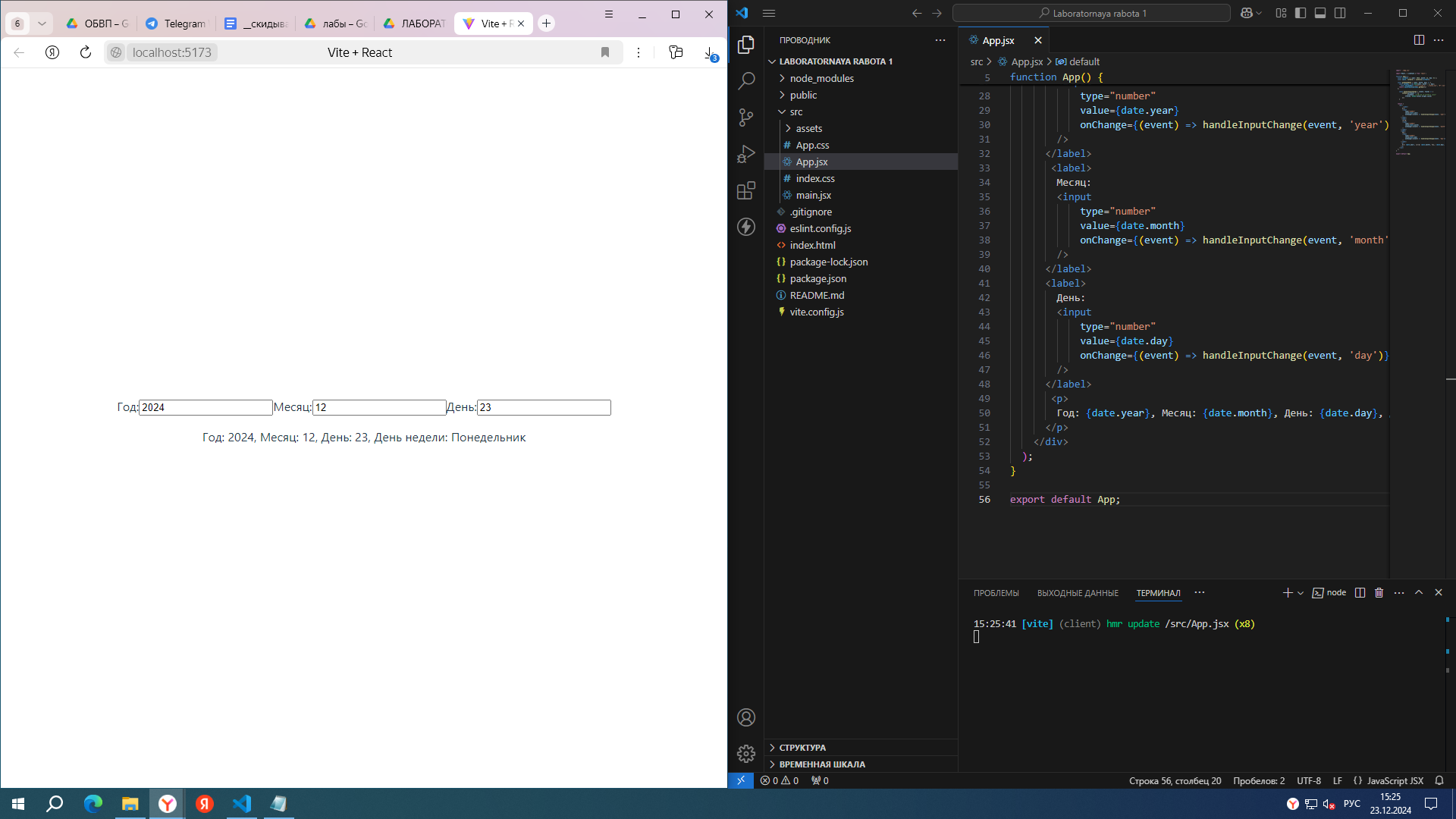
);

}

export default App;

Задание №2

Модифицируйте предыдущую задачу, добавив три инпута для редактирования даты.



import './App.css'

import React, { useState } from 'react';

function App() {

const initDate = { year: 2025, month: 12, day: 31 };

const [date, setDate] = useState(initDate);

const getDayOfWeek = (year, month, day) => {

const dateObj = new Date(year, month - 1, day);

const daysOfWeek = ['Воскресенье', 'Понедельник', 'Вторник', 'Среда', 'Четверг', 'Пятница', 'Суббота'];

return daysOfWeek[dateObj.getDay()];

};

const handleInputChange = (event, field) => {

setDate(prevDate => ({

...prevDate, //Копируем предыдущий объект

[field]: Number(event.target.value)

}))

}

return (

<div>

<label>

Год:

<input

type="number"

value={date.year}

onChange={(event) => handleInputChange(event, 'year')}

/>

</label>

<label>

Месяц:

<input

type="number"

value={date.month}

onChange={(event) => handleInputChange(event, 'month')}

/>

</label>

<label>

День:

<input

type="number"

value={date.day}

onChange={(event) => handleInputChange(event, 'day')}

/>

</label>

<p>

Год: {date.year}, Месяц: {date.month}, День: {date.day}, День недели: {getDayOfWeek(date.year, date.month, date.day)}

</p>

</div>

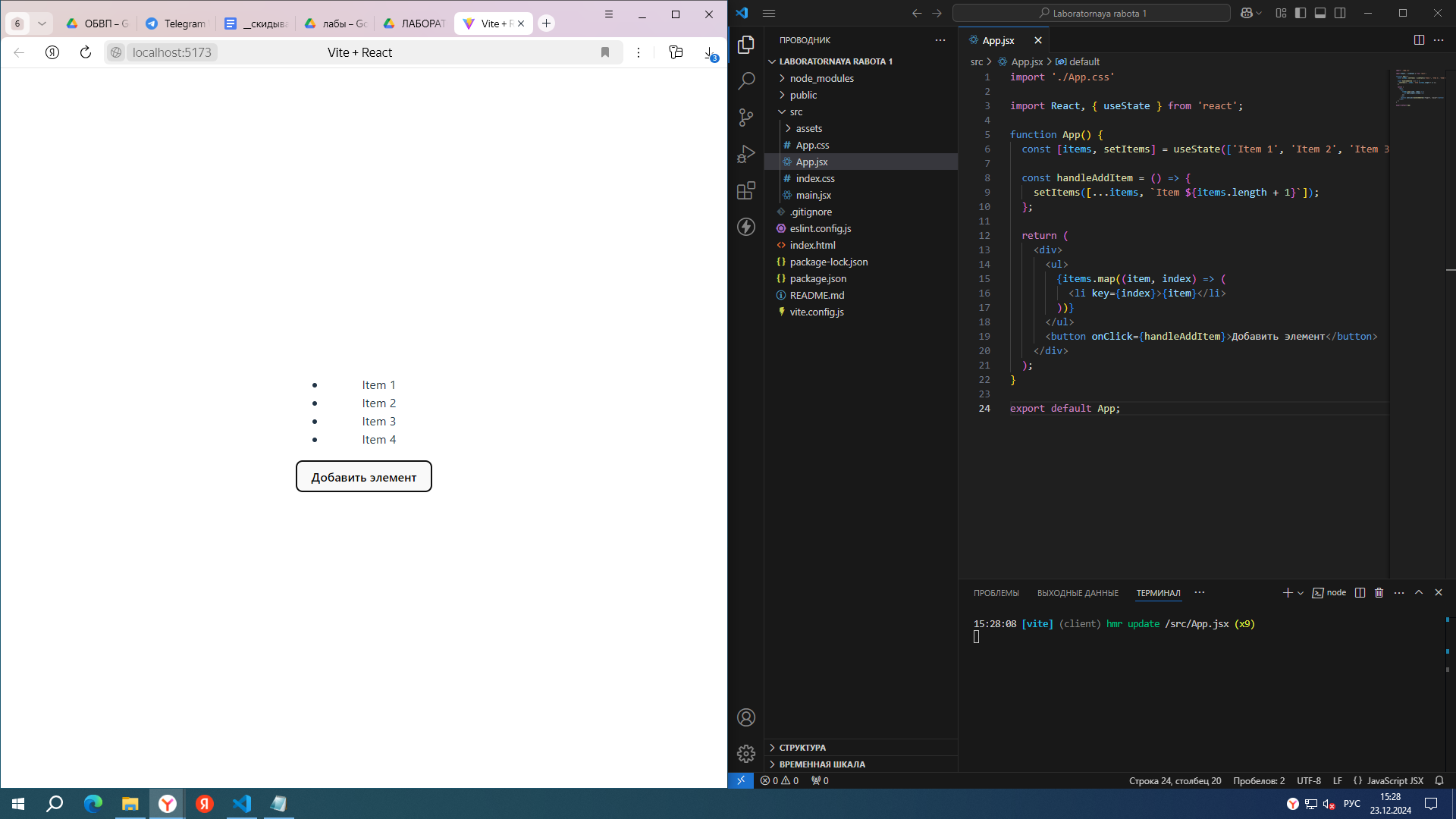
);

}

export default App;

Задание №1

Сделайте кнопку, по нажатию на которую будет добавляться новый элемент в конец массива, тем самым добавляя новый li в конец тега ul.



import './App.css'

import React, { useState } from 'react';

function App() {

const [items, setItems] = useState(['Item 1', 'Item 2', 'Item 3']);

const handleAddItem = () => {

setItems([...items, `Item ${items.length + 1}`]);

};

return (

<div>

<ul>

{items.map((item, index) => (

<li key={index}>{item}</li>

))}

</ul>

<button onClick={handleAddItem}>Добавить элемент</button>

</div>

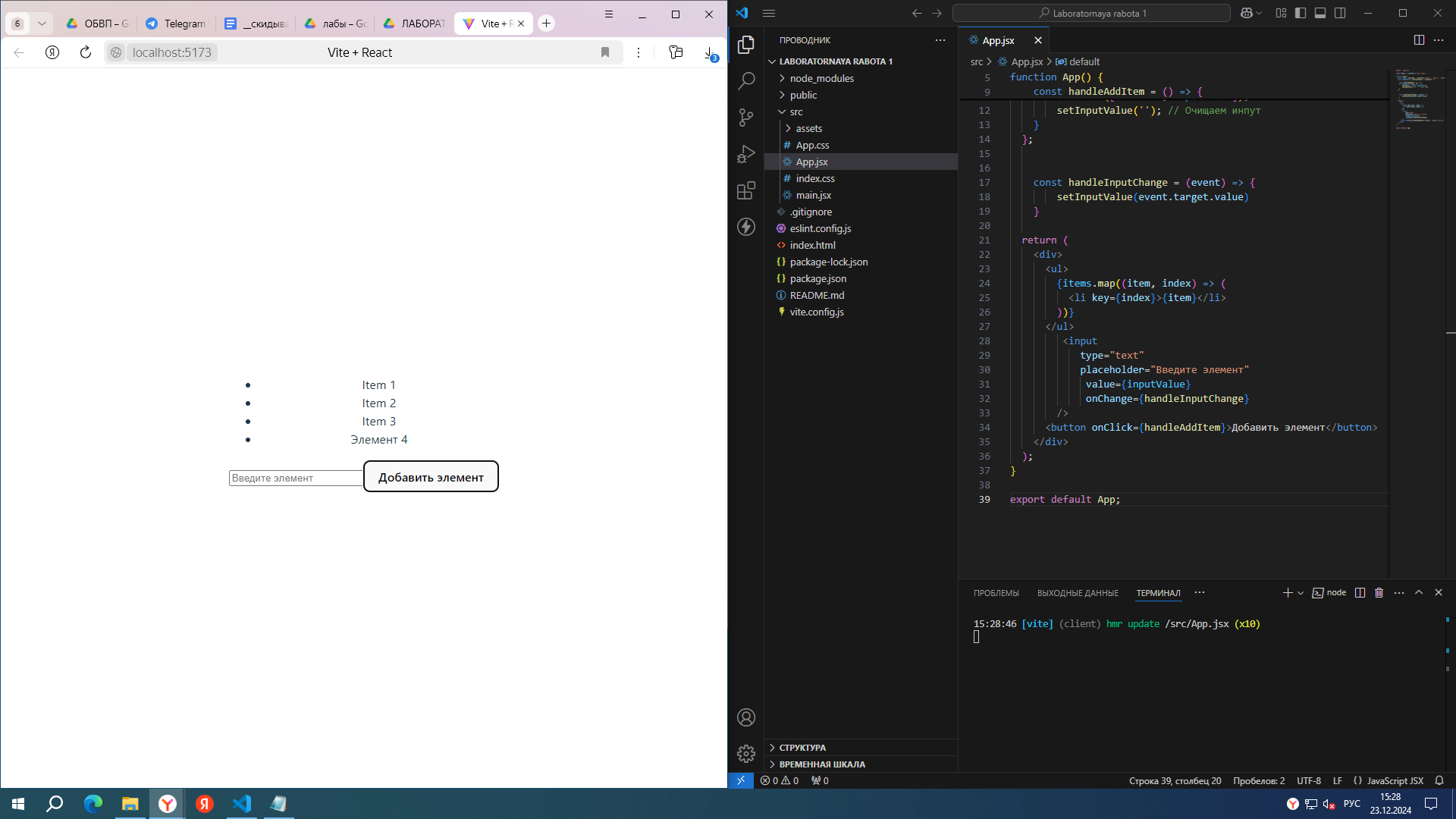
);

}

export default App;

Задание №2

Сделайте инпут и кнопку. По нажатию на кнопку пусть текст инпута станет новым тегом li в конец тега ul.



import './App.css'

import React, { useState } from 'react';

function App() {

const [items, setItems] = useState(['Item 1', 'Item 2', 'Item 3']);

const [inputValue, setInputValue] = useState('');

const handleAddItem = () => {

if (inputValue.trim() !== '') {

setItems([...items, inputValue]);

setInputValue(''); // Очищаем инпут

}

};

const handleInputChange = (event) => {

setInputValue(event.target.value)

}

return (

<div>

<ul>

{items.map((item, index) => (

<li key={index}>{item}</li>

))}

</ul>

<input

type="text"

placeholder="Введите элемент"

value={inputValue}

onChange={handleInputChange}

/>

<button onClick={handleAddItem}>Добавить элемент</button>

</div>

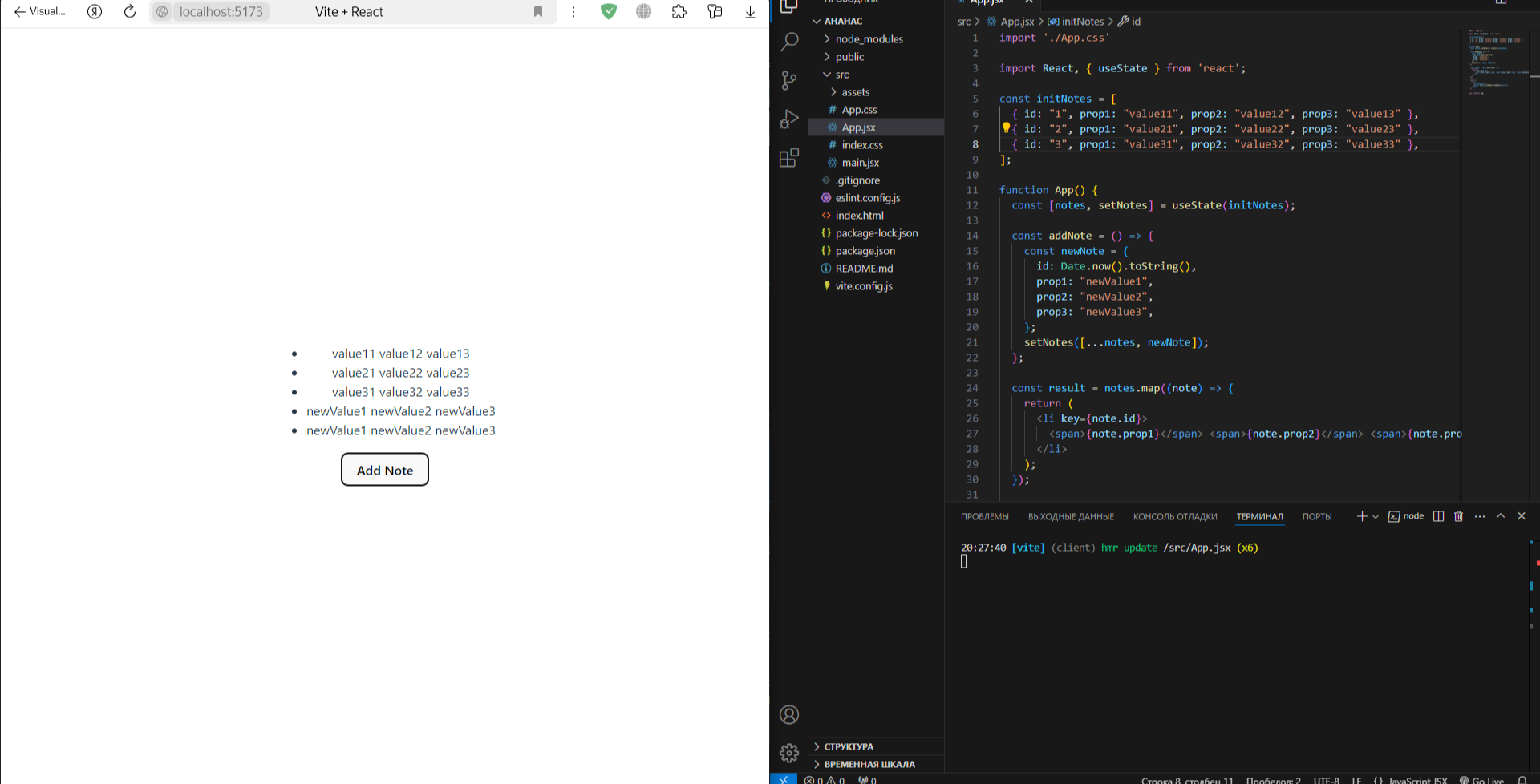
);

}

export default App;

Задание №1

Сделайте кнопку, по нажатию на которую будет добавляться новый элемент в конец массива, тем самым добавляя новый li в конец тега ul.



import './App.css'

import React, { useState } from 'react';

const initNotes = [

{ id: "1", prop1: "value11", prop2: "value12", prop3: "value13" },

{ id: "2", prop1: "value21", prop2: "value22", prop3: "value23" },

{ id: "3", prop1: "value31", prop2: "value32", prop3: "value33" },

];

function App() {

const [notes, setNotes] = useState(initNotes);

const addNote = () => {

const newNote = {

id: Date.now().toString(),

prop1: "newValue1",

prop2: "newValue2",

prop3: "newValue3",

};

setNotes([...notes, newNote]);

};

const result = notes.map((note) => {

return (

<li key={note.id}>

<span>{note.prop1}</span> <span>{note.prop2}</span> <span>{note.prop3}</span>

</li>

);

});

return (

<div>

<ul>{result}</ul>

<button onClick={addNote}>Add Note</button>

</div>

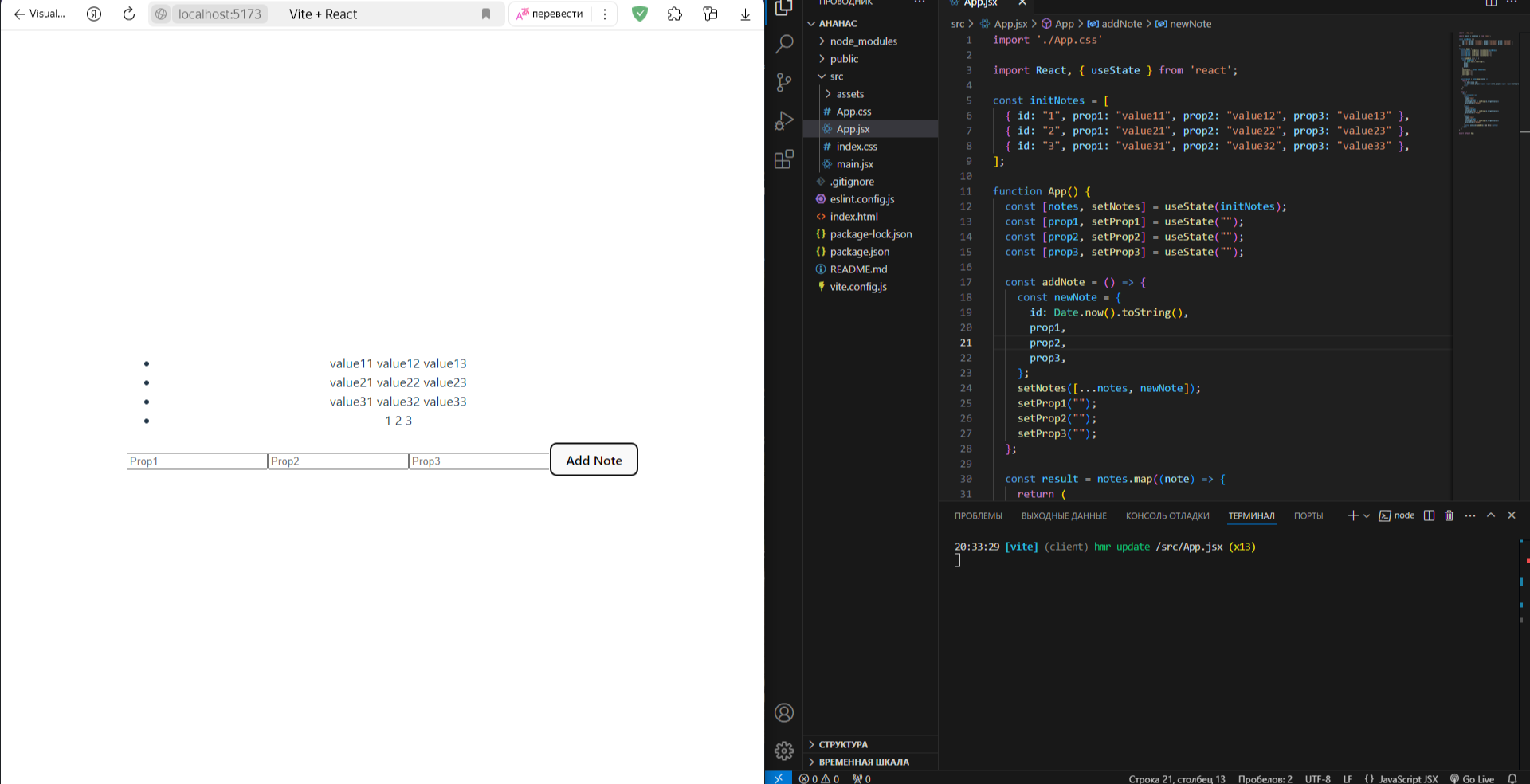
);

}

export default App;

Задание № 2

Сделайте три инпута и кнопку. По нажатию на кнопку из данных инпута создайте новую li в конце тега ul.



import './App.css'

import React, { useState } from 'react';

const initNotes = [

{ id: "1", prop1: "value11", prop2: "value12", prop3: "value13" },

{ id: "2", prop1: "value21", prop2: "value22", prop3: "value23" },

{ id: "3", prop1: "value31", prop2: "value32", prop3: "value33" },

];

function App() {

const [notes, setNotes] = useState(initNotes);

const [prop1, setProp1] = useState("");

const [prop2, setProp2] = useState("");

const [prop3, setProp3] = useState("");

const addNote = () => {

const newNote = {

id: Date.now().toString(),

prop1,

prop2,

prop3,

};

setNotes([...notes, newNote]);

setProp1("");

setProp2("");

setProp3("");

};

const result = notes.map((note) => {

return (

<li key={note.id}>

<span>{note.prop1}</span> <span>{note.prop2}</span> <span>{note.prop3}</span>

</li>

);

});

return (

<div>

<ul>{result}</ul>

<input

type="text"

value={prop1}

onChange={(e) => setProp1(e.target.value)}

placeholder="Prop1"

/>

<input

type="text"

value={prop2}

onChange={(e) => setProp2(e.target.value)}

placeholder="Prop2"

/>

<input

type="text"

value={prop3}

onChange={(e) => setProp3(e.target.value)}

placeholder="Prop3"

/>

<button onClick={addNote}>Add Note</button>

</div>

);

}

export default App;