МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ государственное БЮДЖЕТНОЕ

образовательное учреждение

высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет автоматики и вычислительной техники

Кафедра автоматики



**ОТЧЁТ**

**по практике №1**

**«***Получение практических навыков работы с JS и выполнение простейших примитивов в рамках знакомства с языком***»**

**по дисциплине: «***Технологии и методы программирования***»**

Вариант 3

|  |  |
| --- | --- |
| Выполнил:  Студент гр. «АБ-121», «АВТФ»  *Кремзер Алёна Дмитриевна*  *Кремзер Алёна Александровна*  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2023г. | Проверил:  ассистент кафедры ЗИ  Медведев М.А.  «\_\_\_» \_\_\_\_\_\_\_\_\_ 2023г.  (оценка, подпись) |

Новосибирск

2023

**1.Цели и задачи работы**: Получение практических навыков работы с JS и выполнение простейших примитивов в рамках знакомства с языком

**2.Задание к работе**: выполнить указанные ниже задания согласно варианту. Вариант выбирается согласно номеру в списке группы.

**Часть 1**

Функция находит в строке пары одинаковых фрагментов и возвращает

номер строки с первым фрагментом. С помощью функции найти все пары одинаковых фрагментов.

**Часть 2**

Напишите мини-игру (игра из слов) связав с событием на клик или отправки из формы текста.

**Текст программы**

**Script.js**

function findDuplicates(str) {  
 let duplicates = [];  
 for (let i = 0; i < str.length; i++) {  
 for (let j = i + 1; j <= str.length; j++) {  
 let subStr = str.slice(i, j);  
 let index = str.indexOf(subStr, i + 1);  
 if (index !== -1) {  
 if (!(index - i < subStr.length)) {  
 duplicates.push({substring: subStr, indexes: [i+1, index+1]});  
 }  
 }  
 }  
 }  
 return duplicates;  
}  
  
$(document).ready(function () {  
 $('.btn-task1').click( function() {  
 const input = $("input").val();  
 const result = findDuplicates(input);  
 const myElement = $('.result');  
 myElement.removeClass("d-none");  
  
 $('.list-group').empty();  
 // Создаем новый элемент списка  
 for(let i =0;i<result.length;i++){  
 var listItem = $("<li>", {class: "list-group-item", text: `${result[i].substring} - [${result[i].indexes[0]}, ${result[i].indexes[1]}]`});  
  
 // Добавляем новый элемент в конец списка  
 $(".list-group").append(listItem);} });});

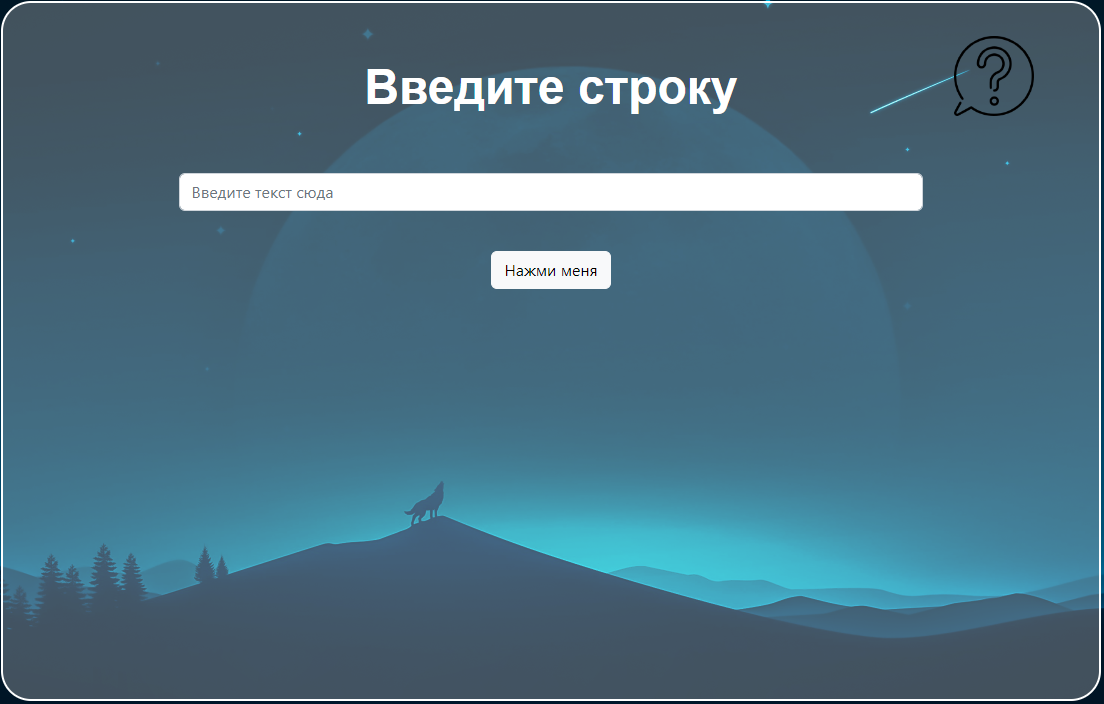
**main.html**

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta http-equiv="X-UA-Compatible" content="IE=edge">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <title>Лабораторная</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi" crossorigin="anonymous">  
 <link rel="stylesheet" href="/style.css">  
</head>  
<body>  
  
<section>  
 <nav class="navbar sticky-top navbar-expand-lg navbar-dark bg-dark">  
 <div class="container-fluid">  
 <a class="navbar-brand">Практика 1</a>  
 <div class="collapse navbar-collapse" id="navbarScroll">  
 <ul class="navbar-nav me-auto my-2 my-lg-0 navbar-nav-scroll" style="--bs-scroll-height: 100px;">  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="http://localhost:63342/practic-1/main.html">Практика 1</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="http://localhost:63342/practic-1/main2.html">Практика 2</a>  
 </li>  
 </ul>  
 </div>  
 </div>  
 </nav>  
  
 <section class="section">  
 <div class="card">  
 <div class="card-body">  
 <div class="container text-center">  
 <div class="row align-items-center">  
 <div class="col-2">  
 <img class="restart d-none" src="http://localhost:63342/practic-1/free-icon-reloading-4659424.png" alt="рестарт">  
 </div>  
 <div class="col-8">  
 <p class="title">  
 Введите строку  
 </p>  
 </div>  
 <div class="col-2">  
 <span class="d-inline-block" tabindex="0" data-bs-toggle="tooltip" data-bs-placement="right" data-bs-title="Функция находит в строке пары одинаковых фрагментов и возвращает номер строки с первым фрагментом. С помощью функции найти все пары одинаковых фрагментов.">  
 <img class="restart" src="http://localhost:63342/practic-1/free-icon-question-1055522.png" alt="рестарт">  
 </span>  
 </div>  
 </div>  
 </div>  
 <div class="mb-3 input-field">  
 <input type="text" class="form-control" id="exampleFormControlInput1" placeholder="Введите текст сюда">  
 </div>  
 <div class="button-div text-center">  
 <button class="btn-task1 btn btn-light mx-auto">Нажми меня</button>  
 </div>  
 <div class="result d-none">  
 <ul class="list-group list-group-flush">  
 </ul>  
 </div>  
 </div>  
 </div>  
 </section>  
</section>  
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.1/jquery.min.js" integrity="sha512-aVKKRRi/Q/YV+4mjoKBsE4x3H+BkegoM/em46NNlCqNTmUYADjBbeNefNxYV7giUp0VxICtqdrbqU7iVaeZNXA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Qbsw3" crossorigin="anonymous"></script>  
<script src="/script.js"></script>  
</body>  
</html>

**Style.css**

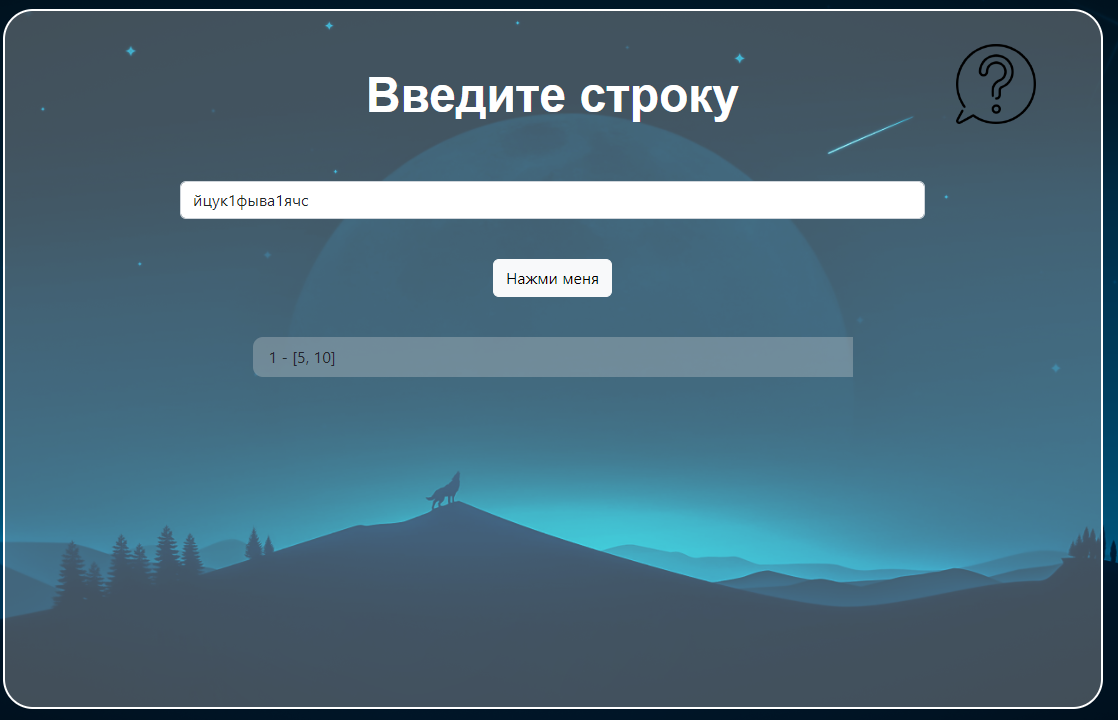
section {  
 display: block;  
 min-height: 100vh;  
 background-image: url("http://localhost:63342/practic-1/Ffcj66m.png");  
 background-size: cover;  
 background-position: center;  
 background-repeat: no-repeat;  
}  
  
.section {  
 display: flex;  
 justify-content: center;  
 align-items: center;  
 width: 100%;  
}  
  
.card {  
 display: flex;  
 background: rgba(217, 217, 217, 0.3);  
 width: 1100px;  
 height: 700px;  
 border: 2px solid #FFFFFF;  
 border-radius: 30px;  
}  
  
  
.title {  
 margin-top: 40px;  
 text-align: center;  
 font-family: 'Inter', sans-serif;  
 font-style: normal;  
 font-weight: 700;  
 font-size: 48px;  
 line-height: 58px;  
 color: #FFFFFF;  
}  
  
.input-field {  
 margin-top: 40px;  
 margin-left: auto;  
 margin-right: auto;  
 width: 70%;  
}  
  
.button-div {  
 margin-top: 40px;  
}  
  
.button-div-2 {  
 margin-top: 15px;  
}  
  
.button-div-3 {  
 margin-top: 50px;  
}  
  
.navbar {  
 position: fixed;  
 min-width: 100vw;  
 border: 2px solid white;  
 border-radius: 0px 0px 15px 15px;  
}  
  
.restart {  
 display: inline-block;  
 width: 80px;  
 height: 80px;  
}  
  
.result {  
 margin-top: 40px;  
 margin-left: auto;  
 margin-right: auto;  
 width: 600px;  
 max-height: 340px;  
 border-radius: 10px 0px 0px 10px;  
 overflow: auto;  
}  
  
.fields-container {  
 display: flex;  
 justify-content: center;  
  
}  
  
.gamer-1 .list-group-item, .gamer-2 .list-group-item {  
 background: rgba(217, 217, 217, 0);  
}  
  
  
  
.gamer-container-2, .gamer-container-1 {  
  
 margin: 0 20px;  
 width: 500px;  
 left: 440px;  
 top: 446px;  
  
  
}  
  
.gamer-2, .gamer-1{  
 border-radius: 5px 5px 5px 5px;  
 min-height: 450px;  
 max-height: 450px;  
 background: rgba(217, 217, 217, 0.3);  
 overflow: auto;  
}  
  
.btn-confirm {  
 display: inline-block;  
}  
  
.results {  
 margin-top: 100px;  
 display: flex;  
 justify-content: center;  
 align-items: center;  
 text-align: center;  
 font-family: 'Inter', sans-serif;  
 font-style: normal;  
 font-weight: 700;  
 font-size: 48px;  
 line-height: 58px;  
 color: #FFFFFF;  
}  
  
.result-1, .result-2 {  
 margin: 0 200px;  
  
}  
  
.div-win {  
 margin-top: 50px;  
 font-family: 'Inter', sans-serif;  
 font-style: normal;  
 font-weight: 700;  
 font-size: 48px;  
 line-height: 58px;  
 color: #FFFFFF;  
}  
  
  
  
  
::-webkit-scrollbar {  
 width: 10px;   
 height: 10px;   
}  
  
  
::-webkit-scrollbar-thumb {  
 background-color: #666;   
  
}  
  
  
::-webkit-scrollbar-track {  
 background-color: #eee;   
}  
  
.list-group-item {  
 background: rgba(217, 217, 217, 0.3);  
}

**Результат работы программы**



Интерфейс программы

После ввода строки, и нажатие кнопки с надписью “Нажми меня”, программа выдаст результат в виде таблицы



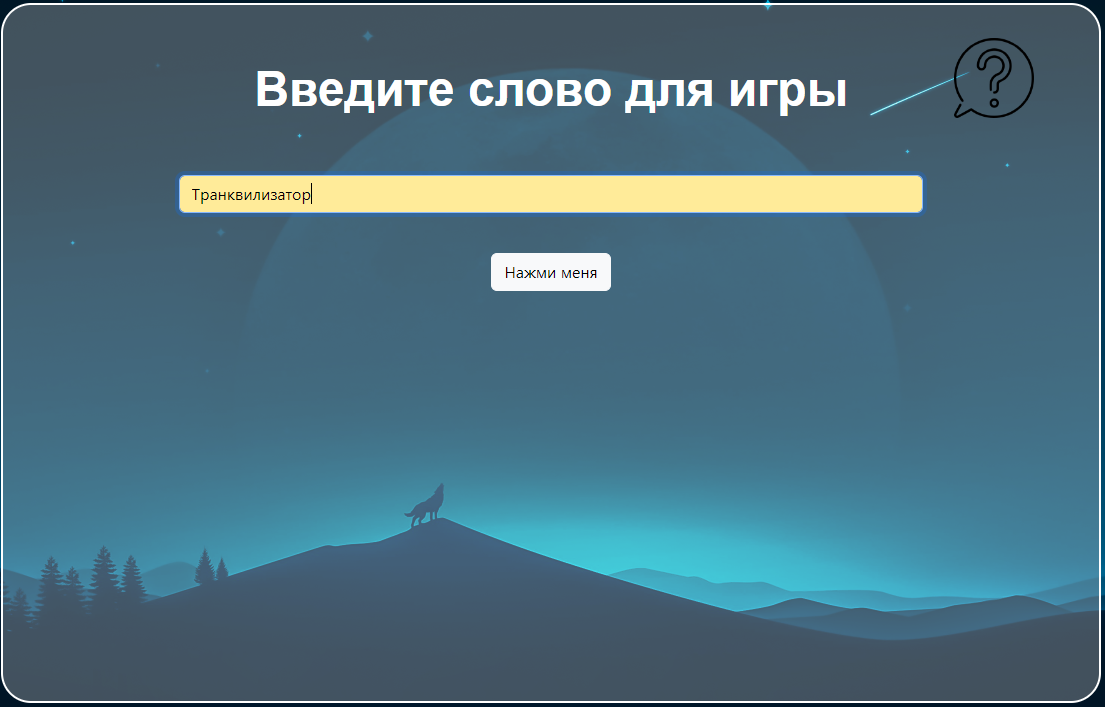
**Script2.js**

const ***tooltipTriggerList*** = ***document***.querySelectorAll('[data-bs-toggle="tooltip"]')  
const ***tooltipList*** = [...***tooltipTriggerList***].map(tooltipTriggerEl => new bootstrap.Tooltip(tooltipTriggerEl))  
  
const ***input*** = ***document***.querySelector('.input-word');  
let ***word*** = '';  
let ***arr1*** = [];  
let ***arr2*** = [];  
let ***check*** = 0;  
  
$(***document***).ready(function () {  
 $('.btn-task1').click( function() {  
  
 $('.gamer-1-input').keypress(function(event) {  
 if (event.keyCode === 13) { // 13 - код клавиши Enter  
 const input = ***document***.querySelector('.gamer-1-input');  
 check\_word(input.value).then((data) => {  
 if (data === undefined){  
 ***console***.log('нет слолва');  
 word\_not\_exist(input);  
 }else {  
 ***console***.log(data);  
 if(word\_in\_word(***word***,data)) {  
 if(***arr1***.includes(data))  
 {  
 word\_not\_exist(input);  
 } else {  
 ***arr1***.push(data);  
 let listItem = $("<li>", {class: "list-group-item", text: data});  
 // Добавляем новый элемент в конец списка  
 $(".list-group-1").append(listItem);  
 input.value = "";  
 }  
 } else {  
 word\_not\_exist(input);  
 }  
 } });  
 }  
 });  
  
 $('.gamer-2-input').keypress(function(event) {  
 if (event.keyCode === 13) { // 13 - код клавиши Enter  
 const input = ***document***.querySelector('.gamer-2-input');  
 check\_word(input.value).then((data) => {  
 if (data === undefined){  
 ***console***.log('нет слолва');  
 word\_not\_exist(input);  
 }else {  
 ***console***.log(data);  
 if(word\_in\_word(***word***,data)) {  
 if(***arr2***.includes(data))  
 {  
 word\_not\_exist(input);  
 } else {  
 ***arr2***.push(data);  
 let listItem = $("<li>", {class: "list-group-item", text: data});  
 // Добавляем новый элемент в конец списка  
 $(".list-group-2").append(listItem);  
 input.value = "";  
 }  
 } else {  
 word\_not\_exist(input);  
 }  
 } });  
 }  
 });  
  
 $('.btn-confirm').click( function() {  
  
 var myModal = new bootstrap.Modal(***document***.getElementById('exampleModal'));  
 myModal.show();  
  
 $('.btn-primary').click(function() {  
 myModal.hide();  
 if (***check*** === 1) {  
 $('.fields-container').addClass('d-none'); //button-div-2  
 $('.button-div-2').addClass('d-none');  
 $('.title').text("Результаты");  
 $('.results').removeClass('d-none');  
 let player1 = ***arr1***.length;  
 let player2 = ***arr2***.length;  
 $('.result-1').text(player1.toString());  
 $('.result-2').text(player2.toString());  
 if (player1 > player2) {  
 $('.who-win').text("Player 1 win");  
 } else if (player2 > player1) {  
 $('.who-win').text("Player 2 win");  
 } else {  
 $('.who-win').text("Draw");  
 }  
 $('.div-win').removeClass('d-none');  
 $('.button-div-3').removeClass('d-none');  
  
 } else {  
 $('.gamer-container-1').css({  
 '-webkit-filter': 'blur(5px)',  
 'filter': 'blur(5px)'  
 });  
 $('.gamer-1-input').prop('disabled', true);  
 $('.gamer-1-input').attr('placeholder', '');  
 $('.gamer-2-input').prop("disabled", false);  
 $('.gamer-2-input').attr('placeholder', 'Введите слово');  
 ***check*** = 1;  
 }  
 });  
 });  
  
 $('.button-div-3').click( function () {  
 ***window***.location.reload();  
 });  
  
 if (***input***.value.length < 5){  
 alert("Слово должно быть длиннее 5 букв");  
 return;  
 }  
  
 check\_word(***input***.value).then((data) => {  
 if (data === undefined){  
 ***console***.log('нет слолва');  
 word\_not\_exist(***input***);  
 }else {  
 ***console***.log(data);  
 ***word*** = data;  
 $('.input-field').addClass('d-none');  
 $('.button-div').addClass('d-none');  
 $('.fields-container').removeClass('d-none'); //button-div-2  
 $('.button-div-2').removeClass('d-none');  
 $('.title').text(***word***);  
 $('.restart').addClass('d-none');  
  
  
 } });  
 });  
});  
  
  
function check\_word(s) {  
 const apiKey = 'dict.1.1.20230224T095439Z.9184bff1006b1511.31b84c39e763f3ba41bbf23fbe6fa0493221bc98';  
 return fetch(`https://dictionary.yandex.net/api/v1/dicservice.json/lookup?key=${apiKey}&lang=ru-ru&text=${s}`).then(response => {  
 if (response.status === 401) {  
 ***console***.log("error");  
 return;  
 }  
 return response.json();}).then(json => {  
  
 if (json.def.length !== 0) {  
 return json.def[0].text;  
  
 }});  
}  
function word\_not\_exist(input){ /\*добавить к каждому полю\*/  
 input.style.transition = 'background-color 1s ease';  
 input.style.backgroundColor = '#f25c5c';  
 setTimeout(() => {  
 input.style.backgroundColor = "";  
 }, 1000);  
}  
  
function word\_in\_word(bigWord,smallWord){  
 const charCount = new ***Map***();  
 for (const char of bigWord.toLowerCase()) {  
 charCount.set(char, (charCount.get(char) || 0) + 1);  
 }  
  
  
 let canMakeWord = true;  
 for (const char of smallWord.toLowerCase()) {  
 const count = charCount.get(char);  
 if (!count || count < 1) {  
 canMakeWord = false;  
 break;  
 }  
 charCount.set(char, count - 1);  
 }  
  
 if (canMakeWord) {  
 ***console***.log(`Можно составить слово ${smallWord} из букв ${bigWord}.`);  
 return true;  
 } else {  
 ***console***.log(`Нельзя составить слово ${smallWord} из букв ${bigWord}.`);  
 return false;  
 }  
}

**main2.html**

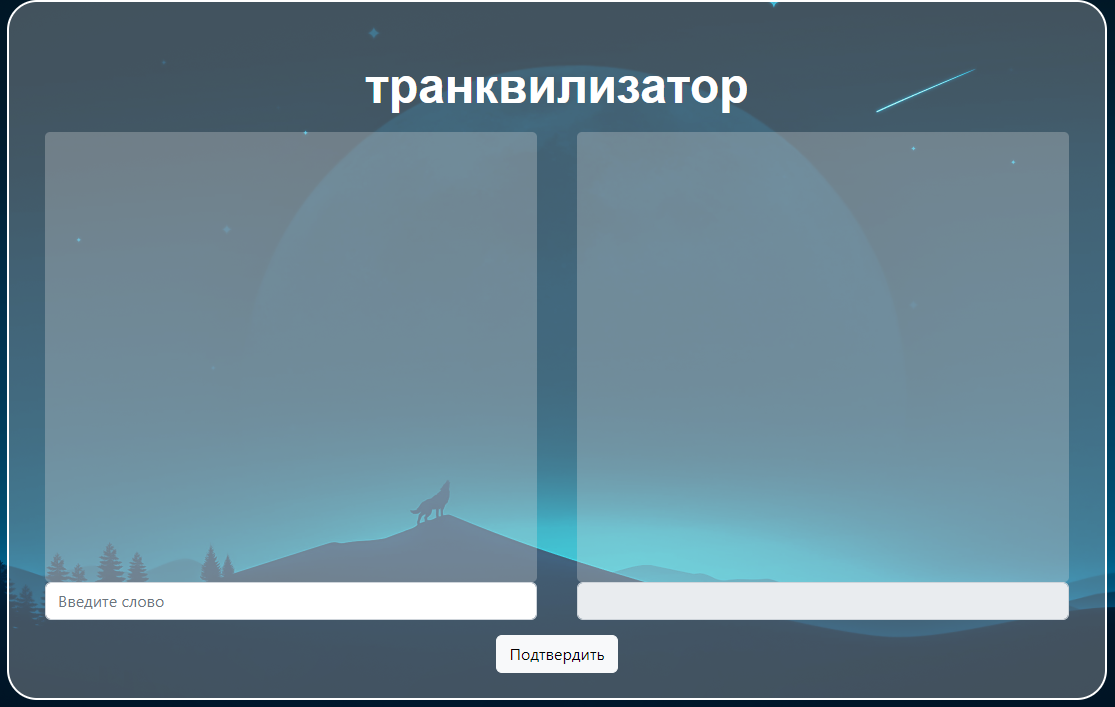
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta http-equiv="X-UA-Compatible" content="IE=edge">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <title>Лабораторная</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi" crossorigin="anonymous">  
 <link rel="stylesheet" href="/style.css">  
</head>  
<body>  
  
<section>  
 <nav class="navbar sticky-top navbar-expand-lg navbar-dark bg-dark">  
 <div class="container-fluid">  
 <a class="navbar-brand">Практика 2</a>  
 <div class="collapse navbar-collapse" id="navbarScroll">  
 <ul class="navbar-nav me-auto my-2 my-lg-0 navbar-nav-scroll" style="--bs-scroll-height: 100px;">  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="http://localhost:63342/practic-1/main.html">Практика 1</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="http://localhost:63342/practic-1/main2.html">Практика 2</a>  
 </li>  
 </ul>  
 </div>  
 </div>  
 </nav>  
  
 <section class="section">  
 <div class="card">  
 <div class="card-body">  
 <div class="container text-center">  
 <div class="row align-items-center">  
 <div class="col-2">  
 <img class="restart d-none" src="http://localhost:63342/practic-1/dice.png" alt="random">  
 </div>  
 <div class="col-8">  
 <p class="title">  
 Введите строку  
 </p>  
 </div>  
 <div class="col-2">  
 <span class="d-inline-block" tabindex="0" data-bs-toggle="tooltip" data-bs-placement="right" data-bs-title="Пишем длинное слово. Каждый игрок записывает как можно больше слов, которые получится составить из букв исходного слова.">  
 <img class="restart" src="http://localhost:63342/practic-1/free-icon-question-1055522.png" alt="рестарт">  
 </span>  
 </div>  
 </div>  
 </div>  
 <div class="mb-3 input-field">  
 <input type="text" class="form-control input-word" id="exampleFormControlInput1" placeholder="Введите текст сюда">  
 </div>  
 <div class="button-div text-center">  
 <button class="btn-task1 btn btn-light mx-auto">Нажми меня</button>  
 </div>  
 <div class="fields-container d-none">  
 <div class="gamer-container-1">  
 <div class="gamer-1">  
 <ul class="list-group list-group-1 list-group-flush">  
  
 </ul>  
 </div>  
 <input type="text" class="form-control gamer-1-input" placeholder="Введите слово">  
 </div>  
 <div class="gamer-container-2">  
 <div class="gamer-2">  
 <ul class="list-group list-group-2 list-group-flush">  
  
 </ul>  
 </div>  
 <input type="text" class="form-control gamer-2-input" placeholder="" disabled>  
 </div>  
 </div>  
 <div class="button-div-2 text-center d-none">  
 <button class="btn-confirm btn btn-light mx-auto">Подтвердить</button>  
 </div>  
  
 <!--result-->  
 <div class="results d-none">  
 <span class="result-1"></span>  
 <span class="result-2"></span>  
  
 </div>  
 <div class="div-win text-center d-none">  
 <span class="who-win"></span>  
 </div>  
 <div class="button-div-3 text-center d-none">  
 <button class="btn-confirm btn btn-light mx-auto">Играть заново</button>  
 </div>  
  
  
 </div>  
 </div>  
 </section>  
  
 <!-- Modal -->  
 <div class="modal fade" id="exampleModal" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">  
 <div class="modal-dialog">  
 <div class="modal-content">  
 <div class="modal-header">  
 <h1 class="modal-title fs-5" id="exampleModalLabel">Предупреждение</h1>  
 <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>  
 </div>  
 <div class="modal-body">  
 Вы хотите закончить ход ?  
 </div>  
 <div class="modal-footer">  
 <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Отмена</button>  
 <button type="button" class="btn btn-primary">Закончить ход</button>  
 </div>  
 </div>  
 </div>  
 </div>  
  
  
</section>  
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.1/jquery.min.js" integrity="sha512-aVKKRRi/Q/YV+4mjoKBsE4x3H+BkegoM/em46NNlCqNTmUYADjBbeNefNxYV7giUp0VxICtqdrbqU7iVaeZNXA==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Qbsw3" crossorigin="anonymous"></script>  
<script src="/script2.js"></script>  
</body>  
</html>

**Результат работы программы**



Страница ввода слова для игры

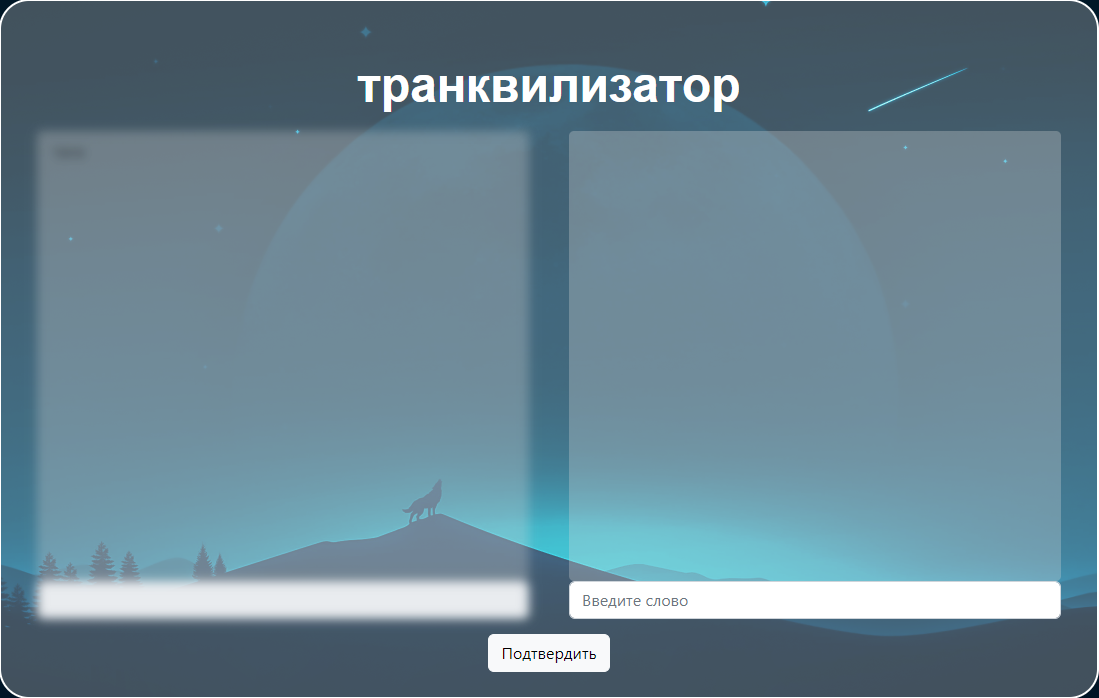
После ввода слова для игры, и нажатие кнопки с надписью “Нажми меня”, программа откроет поле для игры.



Страница ввода слов

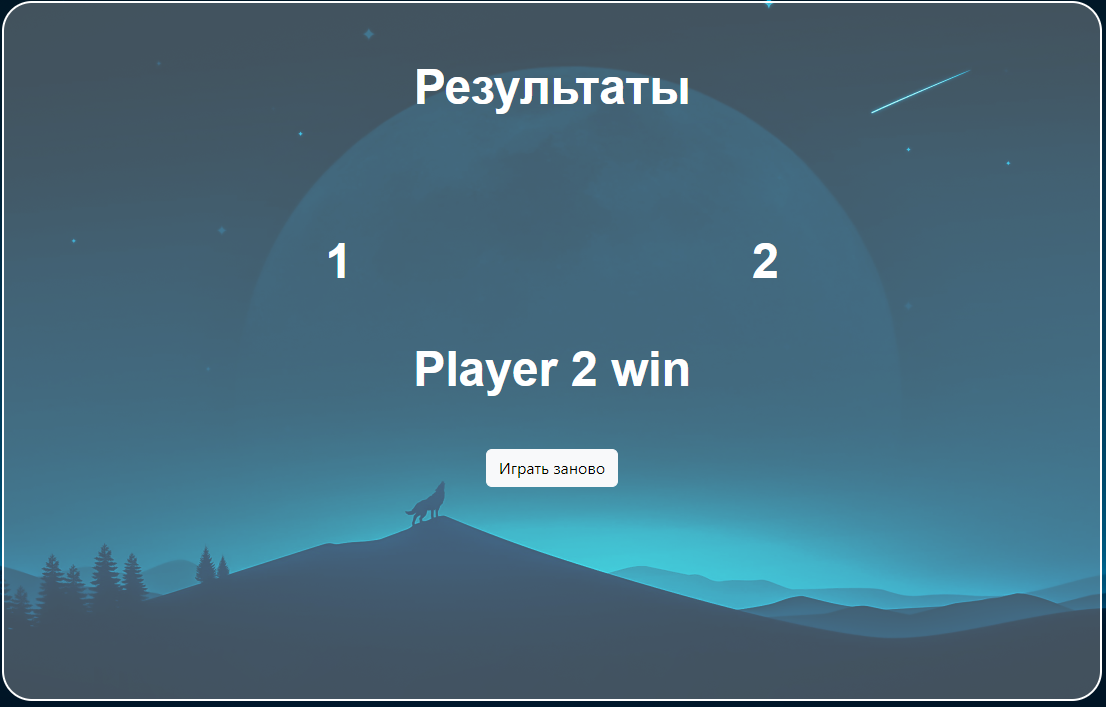
После ввода слова, и нажатия кнопки enter на клавиатуре, слово добавится в таблицу над полем ввода

После окончания хода нужно нажать кнопку с надписью “Подтвердить”, после чего ход передастся второму игроку



После того как первый игрок закончил свой ход, открывается поле для ввода слов второго игрока

После окончания хода нужно нажать кнопку с надписью “Подтвердить”, после чего игра закончится и подведутся итоги игры.



Страница с результатами

**Вывод**:   
Для решения этой задачи была написана функция, которая принимает на вход строку str и находит в ней все пары одинаковых фрагментов. Функция возвращает список кортежей, где каждый кортеж содержит номер строки и позиции первого и второго фрагментов. Если в строке не находится ни одной пары одинаковых фрагментов, то функция возвращает пустой список.

Также для решения второй задачи была написана мини-игра из слов, связанная с событием на клик или отправки из формы текста. Игра предлагает составить слова из заданного слова. Если пользователь вводит правильное слово, оно отображается в таблице над полем ввода. Игра заканчивается, когда второй пользователь заканчивает свой ход, после чего подводятся результаты игры.

В результате выполнения лабораторной работы мной были изучены основы языка гипертекстовой разметки HTML, CSS, основы языка программирования JavaScript, а именно создание переменных, циклов, работа с промисами и функцией fetch для работы с api яндекс словаря.