Skrypt 13

Korzystając z szablonu:

OCTYPE html>

<html>

<head>

<style type="text/css">

#right{

background-color:lightblue;

border:1px solid black;

}

#ala{

background-color:lightblue;

}

#geo{

background-color:lightblue;

}

.menu-context {

position: absolute;

left:-9999px;

top:-9999px;

background: #222;

color:#fff;

list-style:none;

padding:0;

margin:0;

width:300px;

padding:10px;

border-radius: 4px;

box-shadow:0 2px 4px rgba(0,0,0,0.3);

font-size:13px;

}

.menu-context li {

border-bottom:1px solid #444;

cursor: pointer;

}

.menu-context li a {

color:#fff;

text-decoration: none;

padding:5px 10px;

display: block;

transition:0.5s all;

}

.menu-context li a:hover {

background: #F15C5C;

color:#fff;

}

.menu-context li:last-child {

border:0;

}

</style>

</head>

<body>

<ul class="menu-context">

<li><a href="a.html" id="ala">Element menu 1</a></li>

<li><a href="geo.html" id="geo">Element menu 2</a></li>

<li><a href="">Element menu 3</a></li>

<li><a href="">Element menu 4</a></li>

</ul>

<div id="right">

Klinknij tu

</div>

<script type="text/javascript">

const menu = document.querySelector('.menu-context');

function removeMenu() {

menu.style.left = '-9999px';

menu.style.top = '-9999px';

}

function rightButton(e) {

e.preventDefault();

if (e.button === 2) {

menu.style.left = e.pageX + 'px';

menu.style.top = e.pageY + 'px';

}

}

// Usuwanie menu kontekstowego przeglądarki

window.addEventListener('contextmenu', (e) => {

e.preventDefault();

});

const block = document.querySelector('#right');

block.addEventListener('mousedown', rightButton);

// Ukrycie menu po kliknięciu gdziekolwiek indziej

window.addEventListener('click', removeMenu);

</script>

</body>

</html>

Przygotuj 2 strony dla menu kontekstoweg:

1. Pobierz z geolokalizacji współrzędne i wstaw na mapkę (ma się wyświetlać mapka z markerem (dane pobierane z geolokalizacji).

<!DOCTYPE html>

<html>

<body>

<button onclick="getLocation()">Pobierz</button>

<p id="demo"></p>

<script>

var x = document.getElementById("demo");

function getLocation() {

if (navigator.geolocation) {

navigator.geolocation.getCurrentPosition(showPosition);

} else {

x.innerHTML = "Geolocation is not supported by this browser.";

}

}

function showPosition(position) {

x.innerHTML = "Latitude: " + position.coords.latitude +

"<br>Longitude: " + position.coords.longitude;

}

</script>

</body>

</html>

1. Druga strona niech pobiera z formularz 10 liczb całkowitych i niech sortuje te liczby algorytmem couting sort (sortowanie przez zliczanie) i wyświetla posortowaną tablicę pod formularzem.

Przykład

Jeśli mamy tablicę: [4, 2, 2, 8, 3, 3, 1], to algorytm sortowania przez zliczanie działa w następujących krokach:

1. Tworzy tablicę count z wartościami początkowymi [0, 0, 0, 0, 0, 0, 0, 0, 0].
2. Po zliczeniu wystąpień: [0, 1, 2, 2, 1, 0, 0, 0, 1].
3. Po sumowaniu zliczeń: [0, 1, 3, 5, 6, 6, 6, 6, 7].
4. Na końcu tworzy posortowaną tablicę wyjściową: [1, 2, 2, 3, 3, 4, 8].

Sortowanie przez zliczanie działa bardzo dobrze dla małych liczb całkowitych z ograniczonym zakresem, jednak ze względu na potrzebę tworzenia dużych tablic count, nie jest zalecany do sortowania liczb o dużym zakresie wartości lub liczb zmiennoprzecinkowych.