**МІНЕСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Національний технічний університет**

**«Харківський політехнічний інститут»**

Навчально науковий інститут комп’ютерних наук та інформаційних технологій

Кафедра Інформатики та інтелектуальної власності

**ЗВІТ**

**до лабораторної роботи 5 з дисципліни**

**«Основи WEB-технологій»**

Студент Яковлев І. А.

Викладач Шуба І. В.

м. Харків

Лабораторна робота 5

Завдання 1

Використовуючи псевдокласи link, visited, active та директиви CSS в

другому завданні другої лабораторної роботи у гіперпосилань видалити

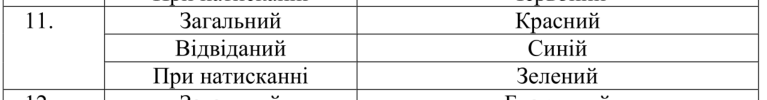
підкреслення та встановити кольори на різні стани гіперпосилання згідно і

параметрами свого варіанту

Опис директив:

Visited - Псевдоклас :visited застосовується до посилань, вже відвіданих користувачем, і задає їм стильове оформлення.

Active - Псевдоклас: Active визначає стиль для активного посилання. Активне посилання стає при натисканні на неї.



<ol type="I" start="9">

<li><a href="#1" name="top">item1</a></li>

<li><a href="#2">item2</a></li>

<li><a href="#3">item3</a></li>

<li><a href="#4">item4</a></li>

<li><a href="#5">item5</a></li>

</ol>

<div>

<a name="1"></a> Lorem

<a href="#top">to top</a>

</div>

<br>

<div>

<a name="2"></a>Lorem

<a href="#top">to top</a>

</div>

<br>

<div>

<a name="3"></a>Lorem

<a href="#top">to top</a>

</div>

<br>

<div>

<a name="4"></a>Lorem

<a href="#top">to top</a>

</div>

<br>

<div>

<a name="5"></a>Lorem

<a href="#top">to top</a>

</div>

a:link{

color: red;

}

a:visited{

color: blue;

}

a:active{

color: green;

}

Завдання 2

Використовуючи псевдоелементи first-letter, first-line, before, after,

псевдокласи nth-child, nth-last-child та директиви CSS в другому завданнi

ДРУГОЇ лабораторної роботи в усiх абзацах кожного розділу(розділ повинен

мати не менше п'яти абзаців) встановити параметри першого символу

(буквиці), першому рядку та виділеного фрагменту згідно із завданням свого

варіанту. Помітити зеленим позаду словом new та попереду червоним словом

old абзаци свого варіанту

Опис директив:

Псевдоелемент :first-letter визначає стиль першого символу в тексті елемента, до якого додається. До цього псевдоелементу можуть застосовуватися лише стильові властивості, пов'язані з властивостями шрифту, полями, відступами, межами, кольором та тлом.

Псевдоклас :nth-child використовується для додавання стилю до елементів на основі нумерації у дереві елементів.



<ol type="I" start="9">

<li><a href="#1" name="top">item1</a></li>

<li><a href="#2">item2</a></li>

<li><a href="#3">item3</a></li>

<li><a href="#4">item4</a></li>

<li><a href="#5">item5</a></li>

</ol>

<div class="parent">

<div>

<a name="1"></a>

<p>Lorem </p>

<p>Lorem </p>

<p>Lorem!</p>

<p>Lorem a!</p>

<p>Lorem a!</p>

<a href="#top">to top</a>

</div>

<br>

<div>

<a name="2"></a>

<p>Lorem a!</p>

<p>Lorem a!</p>

<p> a!</p>

<p> neque, quas perspiciatis reiciendis vero quibusdam. Voluptate eveniet alias odio laborum a!</p>

<p>Lorem ipsum dolor, sit amet consectetur adipisicing elit. <span class="line">Impedit neque, quas perspiciatis reiciendis vero quibusdam. Voluptate eveniet alias odio laborum a!</p>

<a href="#top">to top</a>

</div>

<br>

<div>

<a name="3"></a>

<p>Lorem a!</p>

<p>Lorem ipsum dolor, sit amet consectetur adipisicing elit. <span class="line">Impedit </p>

<p>Lorem a!</p>

<p>Lorem a!</p>

<p>Lorem a!</p>

<a href="#top">to top</a>

</div>

<br>

<div>

<a name="4"></a>

<p>Lorem a!</p>

<p>Lorem a!</p>

<p>Lorem a!</p>

<p>Lorem ipsum dolor, sit amet consectetur adipisicing elit. <span class="line">Impedit aliquam amet nobis,</span> Impedit aliquam amet nobis, illo nisi expedita laborum a!</p>

<p>Lorem a!</p>

<a href="#top">to top</a>

</div>

<br>

<div>

<a name="5"></a>

<p> a!</p>

<p> a!</p>

<p> a!</p>

<p>Lorem ipsum dolor, sit amet consectetur adipisicing elit. <span class="line"> a!</p>

<p> a!</p>

<a href="#top">to top</a>

</div>

</div>

a:link{

color: rgb(255, 108, 108);

}

a:visited{

color: rgb(131, 131, 255);

}

a:active{

color: rgb(121, 255, 121);

}

p::first-letter{

font-size: 70px;

color: rgb(255, 91, 91);

}

p::first-line{

font-size: 1in;

color: rgb(78, 78, 252);

}

.line{

background-color: rgb(77, 255, 115);

font-size: 1pc;

}

.parent div:last-child p:nth-child(even)::after{

content: "NEW";

color: rgb(48, 240, 48);

}

.parent div:nth-child(3) p:nth-child(3)::before{

content: "OLD";

color: red;

}

Завдання 3

Засобами HTMLTа CSS побудувати таблицю розміром 100х100. Розміри

комірок підібрати виходячи з того, щоб горизонтальний, а бажано, і

вертикальний скролінг були відсутніми. Використовуючи псевдокласи nth-

child, nth-last-child та директиви CSS «пофарбувати» в сірий фон комірки

згідно із завданням свого варіанту.



table{

width: 100%;

height: 80vh;

}

tr{

height: 0.999vh;

}

td{

width: 1%;

height: 1%;

background-color: rgb(255, 154, 154);

border: 1px solid;

}

tr:nth-last-child(n+98) td:nth-child(even){

background-color: rgb(100, 99, 99);

}

tr:nth-child(n+98) td:nth-child(even){

background-color: rgb(100, 99, 99);

}

Завдання 4

Використовуючи и over, focus та директиви CSS модифікувати

п'яте завдання четвертої лабораторної роботи таким чином:

для кнопок під час курсору миші фоновий колір та колір

напису мають змінюватись на протилежнi, колiр тiнi має змінитися на

контрастний, для текстових полів під час фокусування фоновий колір та колір напису мають змінюватись на протилежнi, колiр тiнi має змiнитися на

контрастний.

body{

font-size: 200%;

font-style: italic;

font-family: Tahoma;

}

input{

border-radius: 10%;

background-color: rgb(255, 149, 149);

height: 30px;

color: rgb(0, 255, 106);

box-shadow: 5px 5px 5px rgb(128, 68, 68);

font-size: 60%;

}

input:focus{

background-color: rgb(171, 255, 160);

color: rgb(255, 67, 67);

box-shadow: 5px 5px 5px rgb(0, 0, 0);

}

button{

background-color: rgb(183, 255, 142);

color: blue;

height: 50px;

width: 100px;

font-size: 60%;

box-shadow: 5px 5px 5px rgb(95, 95, 121);

}

button:hover{

background-color: rgb(94, 94, 255);

color: rgb(93, 255, 93);

box-shadow: 5px 5px 5px rgb(0, 0, 0);

}

table{

border-color: rgb(255, 0, 0);

background-color: rgb(255, 206, 231);

color: rgb(64, 54, 209);

}

Завдання 5

Модифікувати минуле завдання, перейшовши від простоï заміни кольорів до

організації переходiв для вiдповiдних об'єктiв форм за допомогою директив

transition-property, transition-duration, transition-tirning-function, transition-delay.

Порівняти даний результат з минулим.

Опис директив:

Універсальна властивість, яка дозволяє одночасно задати значення transition-property, transition-duration, transition-timing-function та transition-delay. Встановлює ефект переходу між двома станами елемента, вони можуть бути визначені за допомогою псевдоелемента :hover або :active, динамічно через JavaScript.

body{

font-size: 200%;

font-style: italic;

font-family: Tahoma;

}

input{

border-radius: 10%;

background-color: rgb(255, 149, 149);

height: 30px;

color: rgb(0, 255, 106);

box-shadow: 5px 5px 5px rgb(128, 68, 68);

font-size: 60%;

}

input:focus{

background-color: rgb(171, 255, 160);

color: rgb(255, 67, 67);

box-shadow: 5px 5px 5px rgb(0, 0, 0);

transition: 0.5s;

}

button{

background-color: rgb(183, 255, 142);

color: blue;

height: 50px;

width: 100px;

font-size: 60%;

box-shadow: 5px 5px 5px rgb(95, 95, 121);

}

button:hover{

background-color: rgb(94, 94, 255);

color: rgb(93, 255, 93);

box-shadow: 5px 5px 5px rgb(0, 0, 0);

transition: 0.5s;

}

table{

border-color: rgb(255, 0, 0);

background-color: rgb(255, 206, 231);

color: rgb(64, 54, 209);

}

Завдання 6

Вибір відповідного розділу має відобразити відповідний контент чи за

допомогою директив візуалізації чи за допомогою змiни значень директиви

z-index. Параметри меню налаштувати згiдно до свого варіанту

Опис директив:

Будь-які позиціоновані елементи на веб-сторінці можуть накладатися один на одного в певному порядку, імітуючи цим третій вимір, перпендикулярний екрану. Кожен елемент може перебувати як нижче, так і вище за інші об'єкти веб-сторінки, їх розміщенням по z-осі та управляє z-index. Ця властивість працює тільки для елементів, у яких значення position задано як absolute, fixed чи relative.



body{

position: relative;

}

input{

width: 23vw;

height: 10vh;

}

div{

position: absolute;

width: 50vw;

height: 30vh;

}

.d1{

left: 10vw;

background-color: rgb(200, 255, 168);

top: 70vh;

opacity: 0.3;

}

.d2{

left: 20vw;

background-color: rgb(168, 177, 255);

top: 50vh;

opacity: 0.3;

}

.d3{

left: 30vw;

background-color: rgb(255, 225, 168);

top: 30vh;

opacity: 0.3;

}

.d4{

left: 40vw;

opacity: 0.3;

background-color: rgb(255, 168, 251);

}

.i1:checked + .d1{

z-index: 13;

opacity: 1;

transition: opacity 1s;

}

.i2:checked + .d2{

z-index: 12;

opacity: 1;

transition: opacity 1s;

}

.i3:checked + .d3{

z-index: 11;

opacity: 1;

transition: opacity 1s;

}

.i4:checked + .d4{

z-index: 10;

opacity: 1;

transition: opacity 1s;

}

Завдання 7

Використовуючи псевдоклас checked, селектор -та директиви CSS зробити

селективне (на базi checkbox) меню контролю контенту на прикладі блоків

завдання 4 четвертої лабораторної роботи. Орієнтацію меню вибрати

самостійно.

Опис директив:

Псевдоклас :checked застосовується до елементів інтерфейсу, таких як перемикачі (checkbox) та прапорці (radio), коли вони перебувають у положення "включено". Перемикання елементів у такий стан відбувається за допомогою атрибуту checked тега <input> або користувачем.



input{

position: absolute;

height: 10vh;

width: 20vw;

}

.i1, .i2{

left: -20vw;

}

.i2{

top: 20vh;

}

.i3, .i4, .i5{

right: -20vw;

}

.i4{

top: 20vh;

}

.i5{

top: 40vh;

}

.i1:checked + .block1{

opacity: 1;

z-index: 10;

transition: opacity 1s;

}

.i2:checked + .block2{

opacity: 1;

z-index: 10;

transition: opacity 1s;

}

.i3:checked + .block3{

opacity: 1;

z-index: 10;

transition: opacity 1s;

}

.i4:checked + .block4{

opacity: 1;

z-index: 10;

transition: opacity 1s;

}

.i5:checked + .block5{

opacity: 1;

z-index: 10;

transition: opacity 1s;

}

Завдання 8

Використовуючи псевдоклас hover, вкладені селектори, директиви CSS та

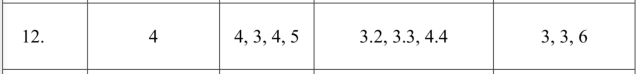
теги списків зробити трирiвневе dropdown-меню з параметрами свого

варіанту. У якості зразку можна застосувати наступний код, в якому

прокоментувати кожну директиву CSS:

Опис директив:

Hover - Определяет стиль элемента при наведении на него курсора мыши, но при этом элемент еще не активирован, иными словами кнопка мыши не нажата.



ul li a, body ul li a:visited{

display: block;

text-align: center;

border: 1px solid;

overflow: hidden;

}

ul{

list-style: none;

}

li{

width: 10vw;

font-size: 50px;

background: rgb(225, 255, 154);

}

ul li{

float: left;

position: relative;

}

ul li ul{

display: none;

}

ul li:hover a{

background: red;

}

ul li:hover ul{

display: block;

position: absolute;

}

ul li:hover ul li a.hide{

background: blue;

}

ul li:hover ul li ul{

display: none;

}

ul li:hover ul li a{

display: block;

background: #ddd;

}

ul li:hover ul li a:hover{

background: green;

}

ul li:hover ul li:hover ul{

display: block;

position: absolute;

z-index: 10;

}

Завдання 9

Використовуючи директиви animation-name,, animation-duration, anim-

timing-function, animation-delay, animation-iteration-count, animation-

direction чи animation створити анімацію руху колового блоку з текстом

Опис директив:

CSS-анімація дає змогу анімувати переходи від однієї конфігурації стилю CSS до іншої. Анімація складається з двох компонентів: стилю, що описує CSS-анімацію, і набору ключових кадрів, які вказують на початковий і кінцевий стани стилю анімації, а також можливі проміжні точки. 

@keyframes example {

0%{left: 10vw; top: 40vh;}

25%{left: 30vw; top: 20vh; width: 33vw;}

50%{left: 50vw; top: 40vh; width: 28vw;}

75%{left: 10vw; top: 40vh; width: 23vw;}

100%{left: 10vw; top: 40vh; width: 40vw;}

}

body{

position: relative;

}

div{

position: absolute;

width: 40vw;

height: 10vh;

left: 10vw;

top: 40vh;

background: rgb(80, 231, 113);

border-radius: 20px;

text-align: center;

animation-name: example;

animation-duration: 3s;

animation-timing-function: linear;

animation-iteration-count: infinite;

animation-iteration-count: 3;

}

p{

position: relative;

top: 3.5vh;

font-size: 3vh;

font-weight: 900;

color: white;

}