|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | | | |
| Институт информационных технологий (ИТ) | |
| Кафедра инструментального и прикладного программного обеспечения (ИиППО) | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ОТЧЕТ ПО ПРАКТИЧЕСКИМ РАБОТАМ** | | | |
| **по дисциплине** | | | |
| **«Разработка клиентских частей интернет-ресурсов»** | | | |
| Выполнил студент группы ИКБО-01-19 | | Иноземцев Н.С. | |
|  | |  | |
| Принял  *Ассистент* | | Коваленко М.А. | |
| Практические работы выполнены | «\_\_\_»\_\_\_\_\_\_\_2020 г. | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись студента) | |
| «Зачтено» | «\_\_\_»\_\_\_\_\_\_\_2020 г. | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись руководителя) | |
|  |  | |  | |

Москва 2020

**Оглавление**

[Практическая работа №4: «Работа с внешними таблицами стилей (CSS)» 3](#_Toc55490252)

[**Задача 1.** 3](#_Toc55490253)

[**Задача 2.** 4](#_Toc55490254)

[**Задача 3.** 7](#_Toc55490255)

[**Задача 4.** 11](#_Toc55490256)

[**Задача 5.** 13](#_Toc55490257)

[**Задача 6.** 14](#_Toc55490258)

[**Задача 7.** 17](#_Toc55490259)

[**Задача 8.** 23](#_Toc55490260)

[**Задача 9.** 24](#_Toc55490261)

[**Задача 10.** 27](#_Toc55490262)

[**Задача 11: «CSS3-селекторы»** 31](#_Toc55490263)

# Практическая работа №4: «Работа с внешними таблицами стилей (CSS)»

**Цель работы:**

**Задача 1.**

Используя CSS-коды спецсимволов вывести:

1. ‘Левая одиночная и правая одиночная кавычки’
2. ‚Нижняя одиночная кавычка‚
3. “Левая и правая двойные кавычки”
4. „Нижняя двойная кавычка„
5. «Левая и правая двойные угловые кавычки»

Добавить комментарии к кодам с описанием самих символов, разные стили для каждой из вышеперечисленных строк.

**Описание выполнения задания:**

Листинг 1 – используя псевдокласс after и css-коды спецсимволов выводим данные

<div></div>

div:after {

white-space: pre;

content: '\275B' 'Левая одиночная и правая одиночная кавычки' '\275C' '\0a' '\201A' 'Нижняя одиночная кавычка' '\201A' '\0a' '\275D' 'Левая и правая двойные кавычки' '\275E' '\0a' '\201E' 'Нижняя двойная кавычка' '\201E' '\0a' '\AB' 'Левая и правая двойные угловые кавычки' '\BB';

}

Результат выполнения кода на странице (Рисунок 1):

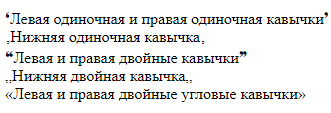


Рисунок 1 - результат выполнения кода

**Задача 2.**

Создать список ключевых слов с оформлением (использовать свойства *text-decoration* со значениями underline, line-through, overline (к одному из ключевых слов одновременно применить два эффекта), *text-decoration-style* с разными значениями, *text-decoration-color* с разными цветами, задать шрифты и выделение тексту).

Пример списка (нужно создать свой):

1. **А**

* Авангард

1. **Б**

* Большие данные

1. *В*

* *Винчестер*

1. ***Г***

* *Гигабайт*

**Описание выполнения задания:**

Листинг 2 – создаем список и применяем к нему разные свойства

<p id="p1">I. A</p>

<ul id="ul1">

<li>Авангард</li>

<li>Авиатор</li>

</ul>

<p id="p2">II. Б</p>

<ul id="ul2">

<li>Бегемот</li>

<li>Большие данные</li>

</ul>

<p id="p3">III. В</p>

<ul id="ul3">

<li>Велосипед</li>

<li>Винчестер</li>

</ul>

<p id="p4">IV. Г</p>

<ul id="ul4">

<li>Гигабайт</li>

<li>Город</li>

</ul>

#p1 {

color: blue;

font-style: oblique;

}

#ul1 {

color: blue;

text-decoration: overline underline;

text-decoration-style: wavy;

text-decoration-color: red;

}

#p2 {

color: purple;

}

#ul2 {

color: purple;

text-decoration: line-through;

text-decoration-style: dashed;

text-decoration-color: black;

}

#p3 {

color: green;

font-style: italic;

}

#ul3 {

color: green;

text-decoration: underline;

text-decoration-style: double;

text-decoration-color: blue;

}

#p4 {

color: gray;

font-size: 20pt;

}

#ul4 {

color: gray;

text-decoration: line-through;

text-decoration-style: double;

text-decoration-color: white;

}

Результат выполнения кода на странице (Рисунок 2):

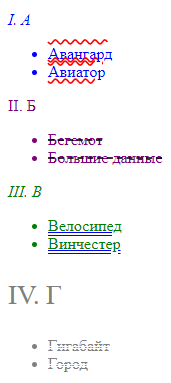


Рисунок 2 – список с разными стилями

**Задача 3.**

Выбрать два шаблона таблиц (один из Примеров 1.1. и 1.2, второй – из Примеров 2.1 и 2.2) и создать по ним свои таблицы.

Пример 1.1 (Таблица имеет фоновое изображение, и строка выделяется при наведении курсором мыши на нее):

**Описание выполнения задания:**

Листинг 3 – выбираем шаблон 1.2 и шаблон 2.1 и создаем таблицы

**Таблица 1.2:**

<table id="tb1">

<tr style="color :#5e38a5">

<td>Company</td>

<td>Q1</td>

<td>Q2</td>

<td>Q3</td>

<td>Q4</td>

</tr>

<tr class="under">

<td>Microsoft</td>

<td>20.3</td>

<td>30.5</td>

<td>23.5</td>

<td>40.3</td>

</tr>

<tr class="under">

<td>Google</td>

<td>50.2</td>

<td>40.63</td>

<td>45.23</td>

<td>39.3</td>

</tr>

<tr class="under">

<td>Apple</td>

<td>25.4</td>

<td>30.2</td>

<td>33.3</td>

<td>36.7</td>

</tr>

<tr class="under">

<td>IBM</td>

<td>20.4</td>

<td>15.6</td>

<td>22.3</td>

<td>39.3</td>

</tr>

</table>

#tb1 {

height: 40%;

width: 50%;

border: solid 2px #89bdff;

border-spacing: 0;

border-collapse: collapse;

}

.under {

background-color: #c6deff;

border-top: dashed 2px white;

color: #6a6ea3;

}

.under:hover {

background-color: #89bdff;

}

.up {

color: #0833b6;

border-bottom: solid 2px #0833b6;

}

Результат выполнения кода на странице (Рисунок 3):

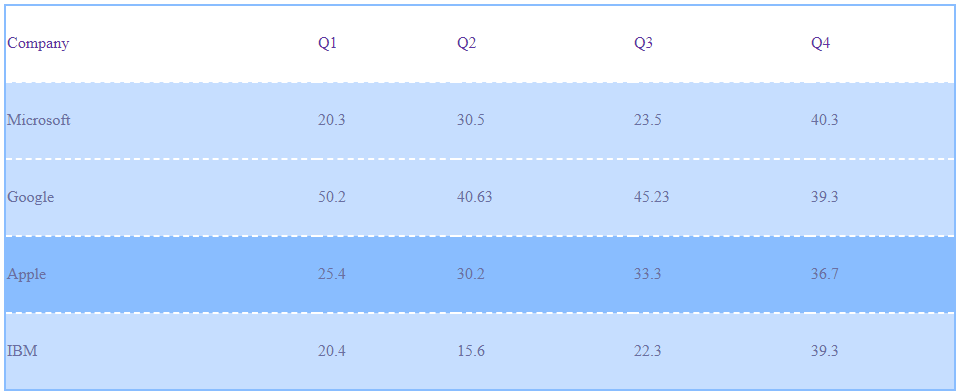


Рисунок 3 – таблица 1.2

**Таблица 2.1:**

<table id="tb2">

<tr>

<td class="up">Comedy</td>

<td class="up">Adventure</td>

<td class="up">Action</td>

<td class="up">Children</td>

</tr>

<tr class="under2">

<td>Scary movie</td>

<td>Indiana Jones</td>

<td>The Punisher</td>

<td>Wall-E</td>

</tr>

<tr class="under2">

<td>Epic movie</td>

<td>Star Wars</td>

<td>Bad Boys</td>

<td>Madagascar</td>

</tr>

<tr class="under2">

<td>Spartan</td>

<td>LOTR</td>

<td>Die Hard</td>

<td>Finding Nemo</td>

</tr>

<tr class="under2">

<td>Dr. Dolittle</td>

<td>The Mummy</td>

<td>300</td>

<td>A Bug's Life</td>

</tr>

</table>

#tb2 {

height: 40%;

width: 50%;

border-spacing: 30px;

}

.under2 {

color: #7577ab;

}

Результат выполнения кода на странице (Рисунок 4):

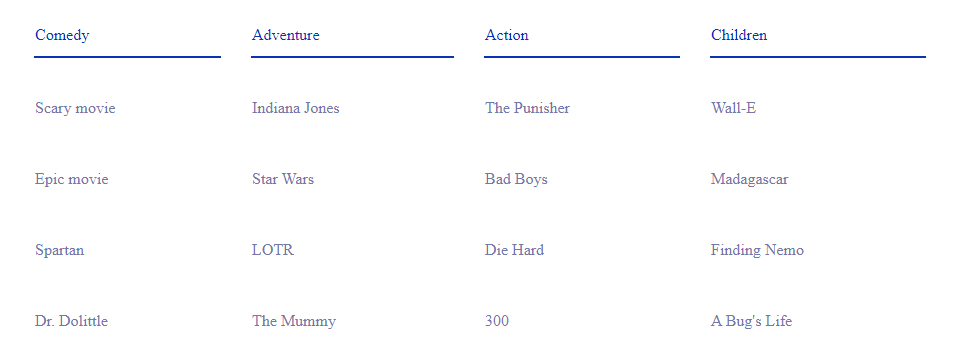


Рисунок 4 – таблица 2.1

**Задача 4.**

Создать батарейки как представлено на рисунке. Размеры заданы в пикселах и не масштабируются. Использовать псевдоэлемент **::after**.

**Описание выполнения задания:**

Листинг 4 – создаем 3 блока <div> и задаем им стиль

<div class="battery battery1"></div>

<div class="battery battery2"></div>

<div class="battery battery3"></div>

.battery {

width: 170px;

height: 50px;

border: 5px solid black;

margin: 20px 20px;

position: relative;

box-shadow: 25px 0px 0px -20px black;

}

.battery1:after {

content: '';

position: absolute;

width: 33%;

height: 100%;

background-color: red;

}

.battery2:after {

content: '';

position: absolute;

width: 66%;

height: 100%;

background-color: yellow;

}

.battery3:after {

content: '';

position: absolute;

width: 100%;

height: 100%;

background-color: green;

}

Результат выполнения кода на странице (Рисунок 5):

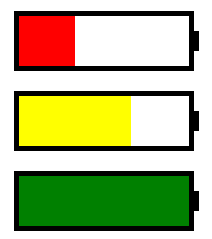
****

Рисунок 5 – батарейки

**Задача 5.**

Создать адаптивный квадратный корень на CSS (знак квадратного корня должен отображаться корректно независимо от используемого числа).

**Описание выполнения задания:**

Листинг 5 – создаем адаптивный корень с помощью text-decoration

<div>

<span>&#8730

<span id="num">144</span>

</span>

</div>

span {

font-size: 20pt;

}

#num {

text-decoration: overline;

text-decoration-style: solid;

text-decoration-color: black;

}

#num:before {

content: '';

position: absolute;

width: 7px;

border-top: 2px solid black;

transform: translate(-7px, -2px);

}

Результат выполнения кода на странице (Рисунки 6-7):



Рисунок 6



Рисунок 7

**Задача 6.**

Создать столбчатую диаграмму.

**Описание выполнения задания:**

Листинг 6 – создаем несколько блоков <div> указываем стиль и объединяем их

<div id="diagram">

<div id="col1"></div>

<div id="col2"></div>

<div id="col3"></div>

<div id="col4"></div>

<div id="col5"></div>

</div>

#diagram {

width: 220px;

height: 220px;

display: flex;

flex-direction: row;

margin-top: 20px;

}

#col1 {

width: 40px;

margin-top: 90%;

background-color: red;

border-right: solid 5px white;

}

#col1:before {

content: '10%';

position: absolute;

transform: translate(0px, -20px);

}

#col2 {

width: 40px;

margin-top: 75%;

background-color: orange;

border-right: solid 5px white;

}

#col2:before {

content: '25%';

position: absolute;

transform: translate(0px, -20px);

}

#col3 {

width: 40px;

margin-top: 50%;

background-color: yellow;

border-right: solid 5px white;

}

#col3:before {

content: '50%';

position: absolute;

transform: translate(0px, -20px);

}

#col4 {

width: 40px;

margin-top: 25%;

background-color: greenyellow;

border-right: solid 5px white;

}

#col4:before {

content: '75%';

position: absolute;

transform: translate(0px, -20px);

}

#col5 {

width: 40px;

margin-top: 0%;

background-color: green;

}

#col5:before {

content: '100%';

position: absolute;

transform: translate(0px, -20px);

}

Результат выполнения кода на странице (Рисунок 8):

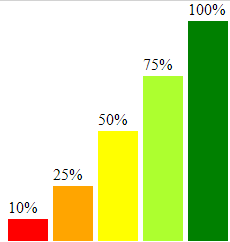


Рисунок 8 – столбчатая диаграмма

**Задача 7.**

Создать страницу со следующим содержимым:

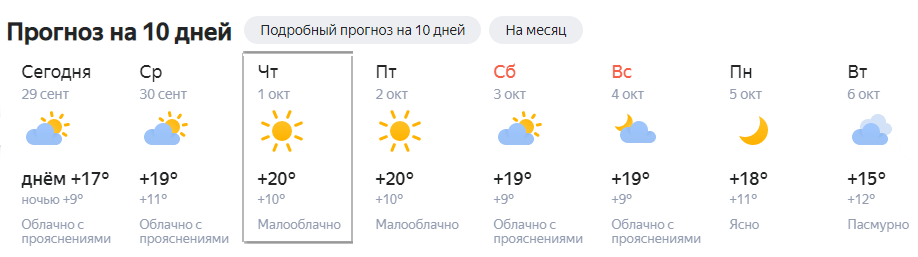


Рисунок 9 – условие задачи

Добавить отображение рамки вокруг каждой из колонок погоды при наведении на них курсором мыши (как показано на рисунке выше).

**Описание выполнения задания:**

Листинг 7

<div id="widget">

<div id="upper">

<h1>Прогноз на 10 дней </h1>

<a href="https://yandex.ru/pogoda/details?lat=55.682533&lon=37.623493&via=ms">Подробный прогноз на 10 дней</a>

<a href="https://yandex.ru/pogoda/month?lat=55.682533&lon=37.623493&via=ms">На месяц</a>

</div>

<div id="lower">

<div class="item">

<h2 class="weekend">Сегодня</h2>

<span>4 нояб</span>

<img src="4nov.svg" width="48px" height="48px">

<h2>днем +8&deg;</h2>

<span>ночью +7&deg;</span><br>

<span>Облачно с прояснениями</span>

</div>

<div class="item">

<h2>Чт</h2>

<span>5 нояб</span>

<img src="5nov.svg" width="48px" height="48px">

<h2>+8&deg;</h2>

<span>+6&deg;</span><br>

<span>Небольшой дождь</span>

</div>

<div class="item">

<h2>Пт</h2>

<span>6 нояб</span>

<img src="6nov.svg" width="48px" height="48px">

<h2>+9&deg;</h2>

<span>+6&deg;</span><br>

<span>Небольшой дождь</span>

</div>

<div class="item">

<h2 class="weekend">Сб</h2>

<span>7 нояб</span>

<img src="6nov.svg" width="48px" height="48px">

<h2>+12&deg;</h2>

<span>+7&deg;</span><br>

<span>Небольшой дождь</span>

</div>

<div class="item">

<h2 class="weekend">Вс</h2>

<span>8 нояб</span>

<img src="5nov.svg" width="48px" height="48px">

<h2>+9&deg;</h2>

<span>+3&deg;</span><br>

<span>Небольшой дождь</span>

</div>

<div class="item">

<h2>Пн</h2>

<span>9 нояб</span>

<img src="4nov.svg" width="48px" height="48px">

<h2>+5&deg;</h2>

<span>0&deg;</span><br>

<span>Облачно с прояснениями</span>

</div>

<div class="item">

<h2>Вт</h2>

<span>10 нояб</span>

<img src="4nov.svg" width="48px" height="48px">

<h2>+3&deg;</h2>

<span>-1&deg;</span><br>

<span>Облачно с прояснениями</span>

</div>

<div class="item">

<h2>Ср</h2>

<span>11 нояб</span>

<img src="4nov.svg" width="48px" height="48px">

<h2>+1&deg;</h2>

<span>-1&deg;</span><br>

<span>Облачно с прояснениями</span>

</div>

<div class="item">

<h2>Чт</h2>

<span>12 нояб</span>

<img src="4nov.svg" width="48px" height="48px">

<h2>-1&deg;</h2>

<span>-1&deg;</span><br>

<span>Облачно с прояснениями</span>

</div>

<div class="item">

<h2>Пт</h2>

<span>13 нояб</span>

<img src="13nov.svg" width="48px" height="48px">

<h2>+1&deg;</h2>

<span>-1&deg;</span><br>

<span>Пасмурно</span>

</div>

</div>

</div>

#widget {

width: 50%;

height: 30%;

display: flex;

flex-direction: column;

}

#upper {

display: flex;

flex-direction: row;

width: 100%;

}

#lower {

display: flex;

flex-direction: row;

justify-content: center;

}

a {

background-color: #eceef2;

color : #222426;

border-radius: 20px;

padding: 6px 16px;

font-size: 14px;

text-decoration: none;

outline: 0;

touch-action: manipulation;

cursor: pointer;

align-self: center;

margin-left: 8px;

}

a:hover {

background-color: #dfe1e7;

}

.item {

display: flex;

flex-direction: column;

width: 10%;

height: 100%;

padding: 2px;

}

.item:hover {

border: solid 2px gray;

padding: 0;

}

h2 {

font-size: 18px;

margin: 0;

padding: 0;

font-family: 'YS Text Web',Helvetica Neue,Arial;

}

.weekend {

color:red;

}

span {

font-size: 14px;

color: #939cb0;

font-family: 'YS Text Web',Helvetica Neue,Arial;

}

Результат выполнения кода на странице (Рисунок 10):

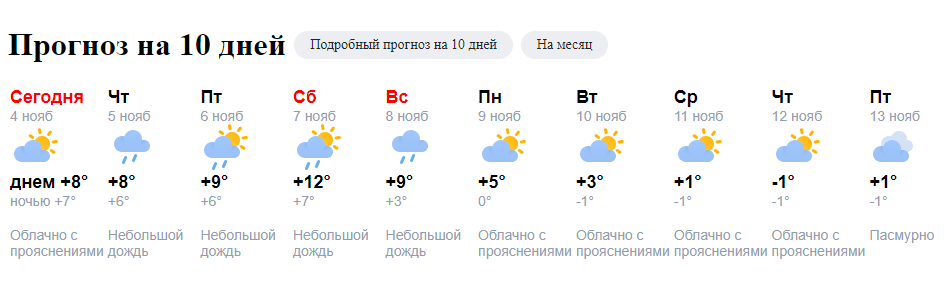


Рисунок 10 – погода на 10 дней

**Задача 8.**

Создать пять блоков *div* и задать им параметры поворота, масштабируемости, наклона и сдвига при помощи CSS-свойства *transform*.

**Описание выполнения задания:**

Листинг 8 – создаем 5 блоков <div>

<div id="container">

<div style="background-color: black; transform: translate(100px, 100px);"></div>

<div style="background-color: red; transform: rotate(35deg);"></div>

<div style="background-color: orange; transform: scale(1.5, 0.5);"> </div>

<div style="background-color: purple; transform: skew(40deg, 20deg);"></div>

<div style="background-color: pink;"></div>

</div>

#container {

display: flex;

flex-direction: row;

width: 100%;

height: 100%;

}

div {

width: 20%;

height: 50%;

}

Результат выполнения кода на странице (Рисунок 11):

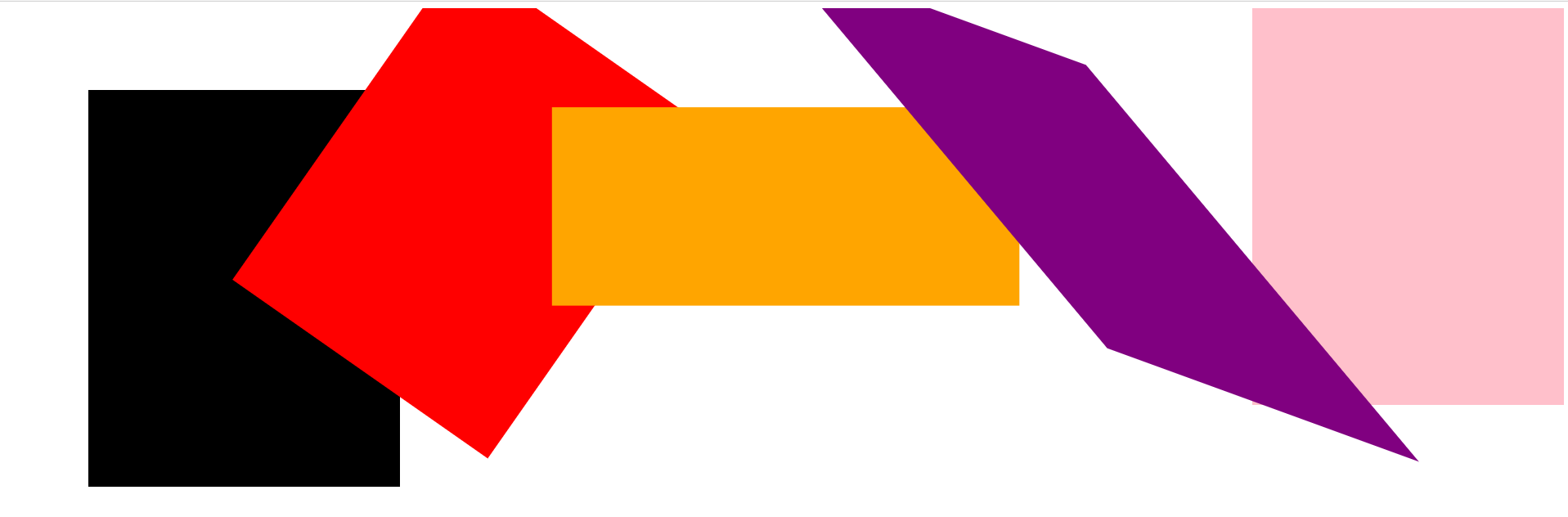


Рисунок 11 – блоки <div> с разными свойствами transform

**Задача 9.**

Создать 3 анимационных кнопки: с поворотом кнопки на 90 градусов (использовать CSS-свойство *transform* с функцией *rotateX),* две кнопки с разными плавными изменениями свойств кнопки (использовать CSS-свойство перехода *transition*).

Задать названия всем кнопкам, тени (использовать ключевое слово *inset*), цветовое градиентное оформление.

**Описание выполнения задания:**

Листинг 9 – пришлось поворачивать кнопки не на 90 а на 80 градусов, иначе кнопка исчезала и ее нельзя было вернуть

<div class="button btn1">

Button1

</div><br>

<div class="button btn2">

Button2

</div><br>

<div class="button btn3">

Button3

</div>

<script type="text/javascript">

let btn1 = document.getElementsByClassName("button")[0];

let btn2 = document.getElementsByClassName("button")[1];

let btn3 = document.getElementsByClassName("button")[2];

let ygol = 80;

btn1.addEventListener('click', function() {

btn1.style.cssText = `transform: rotateX(${ygol}deg)`;

if(ygol != 0) {

ygol = 0

}

else {

ygol = 80;

}

});

btn2.addEventListener('click', function() {

btn2.style.cssText = `transform: rotateX(${ygol}deg)`;

if(ygol != 0) {

ygol = 0

}

else {

ygol = 80;

}

});

btn3.addEventListener('click', function() {

btn3.style.cssText = `transform: rotateX(${ygol}deg)`;

if(ygol != 0) {

ygol = 0

}

else {

ygol = 80;

}

});

</script>

.button {

width: 200px;

height: 50px;

border: solid 1px black;

border-radius: 30px;

display: flex;

justify-content: center;

align-items: center;

}

.button.btn1 {

box-shadow: inset 0px 0px 50px red;

}

.button.btn1:hover {

box-shadow: inset 0px 0px 50px #cc0000;

}

.button.btn2 {

background: linear-gradient(to right, red, purple);

transition: 1s;

color: #FFFFFF;

}

.button.btn2:hover {

background: linear-gradient(to right, #e60000, #660066);

}

.button.btn3 {

color: #FFFFFF;

background: linear-gradient(45deg, rgba(35,36,0,1) 0%, rgba(50,207,21,1) 31%, rgba(102,0,255,1) 61%, rgba(255,0,245,1) 88%);

transition: 2s;

}

.button.btn3:hover {

background: linear-gradient(45deg, rgba(20,21,0,1) 0%, rgba(35,192,6,1) 31%, rgba(87,0,250,1) 61%, rgba(240,0,230,1) 88%);

}

Результат выполнения кода на странице (Рисунок 12):



Рисунок 12 – кнопки с поворотом на 80 градусов и с разным оформлением

**Задача 10.**

Создать выпадающее меню (с собственным оформлением и содержанием) на примере, представленном на рисунке ниже, с изменением вида курсора при наведении на пункты меню.

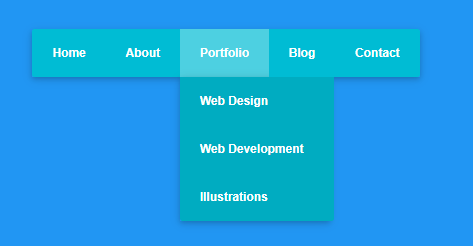


Рисунок 13 – пример меню

**Описание выполнения задания:**

Листинг 10 – создаем меню с помощью тега <ul> и задаем свой стиль

<nav>

<ul class="top">

<li>

<a href="#">Home</a>

</li>

<li>

<a href="#">About</a>

</li>

<li>

<a href="#">Portfolio</a>

<ul class="sub">

<li>

<a href="#">Web Design</a>

</li>

<li>

<a href="#">Web Development</a>

</li>

<li>

<a href="#">Illustrations</a>

</li>

</ul>

</li>

<li>

<a href="#">Blog</a>

</li>

<li>

<a href="#">Contact</a>

</li>

</ul>

</nav>

body {

margin: 0;

}

.top {

width: 50%;

margin: 0;

padding: 0;

background-color: #00bcd4;

display: flex;

flex-direction: row;

}

ul {

list-style-type: none;

}

.top > li {

width: 20%;

height: 70px;

background-color: #00bcd4;

display:table;

}

li {

margin: 0;

padding: 0;

}

li:hover {

background-color: #4dd0e1;

}

.sub {

background-color: #00bcd4;

top: 70px;

left: 20%;

width: 20%;

visibility: hidden;

position: absolute;

display: flex;

flex-direction: column;

margin: 0;

padding: 0;

transform: scaleY(0);

transform-origin: 0 0;

transition: .5s;

}

.sub > li {

display:table;

text-align-last: left;

height: 70px;

}

ul li:hover .sub {

transform: scaleY(1);

visibility: visible;

}

a {

width: 100%;

margin: 0;

padding: 0;

display: block;

border: none;

text-align:center;

vertical-align: middle;

display:table-cell!important;

color:white;

text-decoration: none;

}

Результат выполнения кода на странице (Рисунок 14):

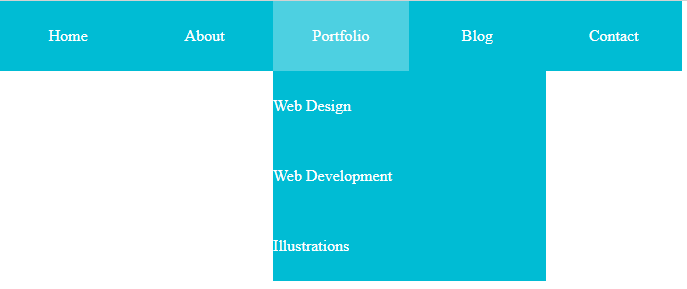


Рисунок 14 – выпадающее меню

**Задача 11: «CSS3-селекторы»**

**Задания:**

1. Выбрать *input* типа *checkbox*.

input[type="checkbox"]

1. Выбрать *input* типа *checkbox*, НЕ отмеченный.

input[type="checkbox"]:not(:checked)

1. Найти все элементы с *id=message* или *message-\**.

[id|="message"]

1. Найти все элементы с *id=message-\**.

[id^="message-"]

1. Найти все ссылки с расширением *href="...zip"*.

a[href$=".zip"]

1. Найти все элементы с атрибутом *data-action*, содержащим *delete* в списке (через пробел).

[data-action~="delete"]

1. Найти все элементы, у которых ЕСТЬ атрибут *data-action*, но он НЕ содержит *delete* в списке (через пробел).

[data-action]:not([data-action~="delete"])

1. Выбрать все чётные элементы списка *#messages*.

#messages li:nth-child(2n)

1. Выбрать один элемент сразу за заголовком *h3#widget-title* на том же уровне вложенности.

h3#widget-title + \*

1. Выбрать все ссылки, следующие за заголовком *h3#widget-title* на том же уровне вложенности.

h3#widget-title ~ a

1. Выбрать ссылку внутри последнего элемента списка *#messages*.

#messages li:last-child a