Круговое меню

Теперь можно раcсчитывать углы секторов меню и углы поворота пунктов. Используем такие формулы:

[угол сектора] = 360° / [количество пунктов меню]

[поворот пункта n] = (n - 1) \* [угол сектора]

В нашем меню может быть максимум 6 пунктов, поэтому для него получатся такие значения:

[угол сектора] = 360° / 6 = 60°

[поворот 1 пункта] = (1 - 1) \* 60° = 0°

[поворот 2 пункта] = (2 - 1) \* 60° = 60°

Изображение выглядит как текст, канцелярские товары, конверт, визитка

Автоматически созданное описание

Для вращения пунктов, мы используем rotate. Но одного вращения недостаточно, так как элементы списка будут пересекаться, ведь их угол больше 60°.

Чтобы исправить это, мы искривим пункты с помощью skew. А угол наклона посчитаем так:

[угол наклона] = 90° - [угол сектора]

В нашем случае получится 30°.

Итого: первый элемент списка нужно наклонить на 30°, а второй — повернуть на 60° и наклонить на 30°.

Кстати, возможны две записи:

transform: rotate(60deg) skew(30deg); // 1 вариант

transform: skew(30deg) rotate(60deg); // 2 вариант

Внутри пунктов меню расположены ссылки. Для них задан блочный тип, размеры и фон — эти исходные стили нужно просто раскомментировать.

На предыдущем шаге мы повернули и скосили пункты меню, но при этом скосилось и их содержимое, то есть ссылки. Чтобы вернуть им нормальный вид, нужно их «разнаклонить» и развернуть обратно.

«Разнаклонять» будем на противоположный угол, то есть на -30°.

Расчёт угла разворота сложнее, вот формула:

[угол разворота] = -1 \* (90° – ([угол сектора] / 2))

В нашем случае получится:

[угол разворота] = -1 \* (90° - (60° / 2)) = -60°

Обратите внимание на порядок трансформаций для ссылок: сначала skew, потом rotate.

И ещё одна важная деталь. Нам нужно сместить ссылки так, чтобы они располагались во внутренней половине пунктов меню. Для этого используем маргины.

Вот и вся техника для создания круглого меню:

1. Переносим ось вращения пунктов в угол с помощью transform-origin и совмещаем этот угол с центром меню.
2. Наклоняем и поворачиваем пункты меню, используя формулы:
3. [угол сектора] = 360° / [количество пунктов меню]
4. [поворот пункта n] = (n - 1) \* [угол сектора]

[угол наклона] = 90° - [угол сектора]

1. «Разнаклоняем» и разворачиваем содержимое пунктов меню:
2. [угол наклона содержимого] = -1 \* [угол наклона пункта]

[угол разворота] = -1 \* (90° – ([угол сектора] / 2))

1. Смещаем содержимое пунктов к центру круга.
2. Закругляем контейнер меню с помощью border-radius и обрезаем всё лишнее с помощью overflow: hidden.

А после этих шагов можно применять дополнительное оформление. Например, задавать разные фоны пунктов, рамку для контейнера меню и так далее.

Давайте напоследок добавим третий пункт меню, чтобы у нас получился полукруг. Угол поворота третьего пункта будет равен 120°:

Серия заданий сделана по мотивам этой статьи [Building a Circular Navigation with CSS Transforms](http://tympanus.net/codrops/2013/08/09/building-a-circular-navigation-with-css-transforms/). Примеры были адаптированы и упрощены для тренажёра.

<!DOCTYPE html>

<html lang="ru">

<head>

<title>Круговое меню, финал</title>

<meta charset="utf-8">

<link rel="stylesheet" href="style.css">

</head>

<body>

<nav class="circle-menu">

<div class="center"></div>

<ul>

<li><a href="#">monitor</a></li>

<li><a href="#">aperture</a></li>

</ul>

</nav>

</body>

</html>

body {

margin: 0;

padding: 0;

background-color: #f5f5f5;

}

.circle-menu {

position: relative;

width: 400px;

height: 400px;

margin: 50px auto;

background-color: white;

box-shadow: 0 0 3px #cccccc;

}

.circle-menu ul {

position: absolute;

width: 300px;

height: 300px;

margin: 50px;

padding: 0;

list-style: none;

border-radius: 50%;

overflow: hidden;

}

.circle-menu .center {

position: absolute;

top: 195px;

left: 195px;

z-index: 1000;

border: 5px solid #34495e;

border-radius: 50%;

box-shadow: 0 0 3px #cccccc;

}

.circle-menu li {

position: absolute;

top: -10px;

left: -10px;

width: 160px;

height: 160px;

transform-origin: 100% 100%;

overflow: hidden;

}

.circle-menu li:nth-child(1) {

transform: rotate(0deg) skew(30deg);

}

.circle-menu li:nth-child(2) {

transform: rotate(60deg) skew(30deg);

}

.circle-menu li:nth-child(3) {

transform: rotate(120deg) skew(30deg);

}

.circle-menu li a {

display: block;

width: 160px;

height: 160px;

margin-top: 40px;

margin-left: 40px;

font-size: 0;

background: rgba(241, 196, 15, 0.5) url("icons/monitor-4x.png") no-repeat 50% 40%;

transition: background-color 0.5s;

transform: skew(-30deg) rotate(-60deg);

}

.circle-menu li:nth-child(even) a {

background-color: rgba(241, 196, 15, 0.75);

}

.circle-menu li a:hover {

background-color: rgba(241, 196, 15, 1);

}

/\* Иконки \*/

.circle-menu li:nth-child(2) a {

background-image: url("icons/aperture-4x.png");

}

.circle-menu li:nth-child(3) a {

background-image: url("icons/audio-4x.png");

}

.circle-menu li:nth-child(4) a {

background-image: url("icons/battery-empty-4x.png");

}

.circle-menu li:nth-child(5) a {

background-image: url("icons/bluetooth-4x.png");

}

.circle-menu li:nth-child(6) a {

background-image: url("icons/browser-4x.png");

}