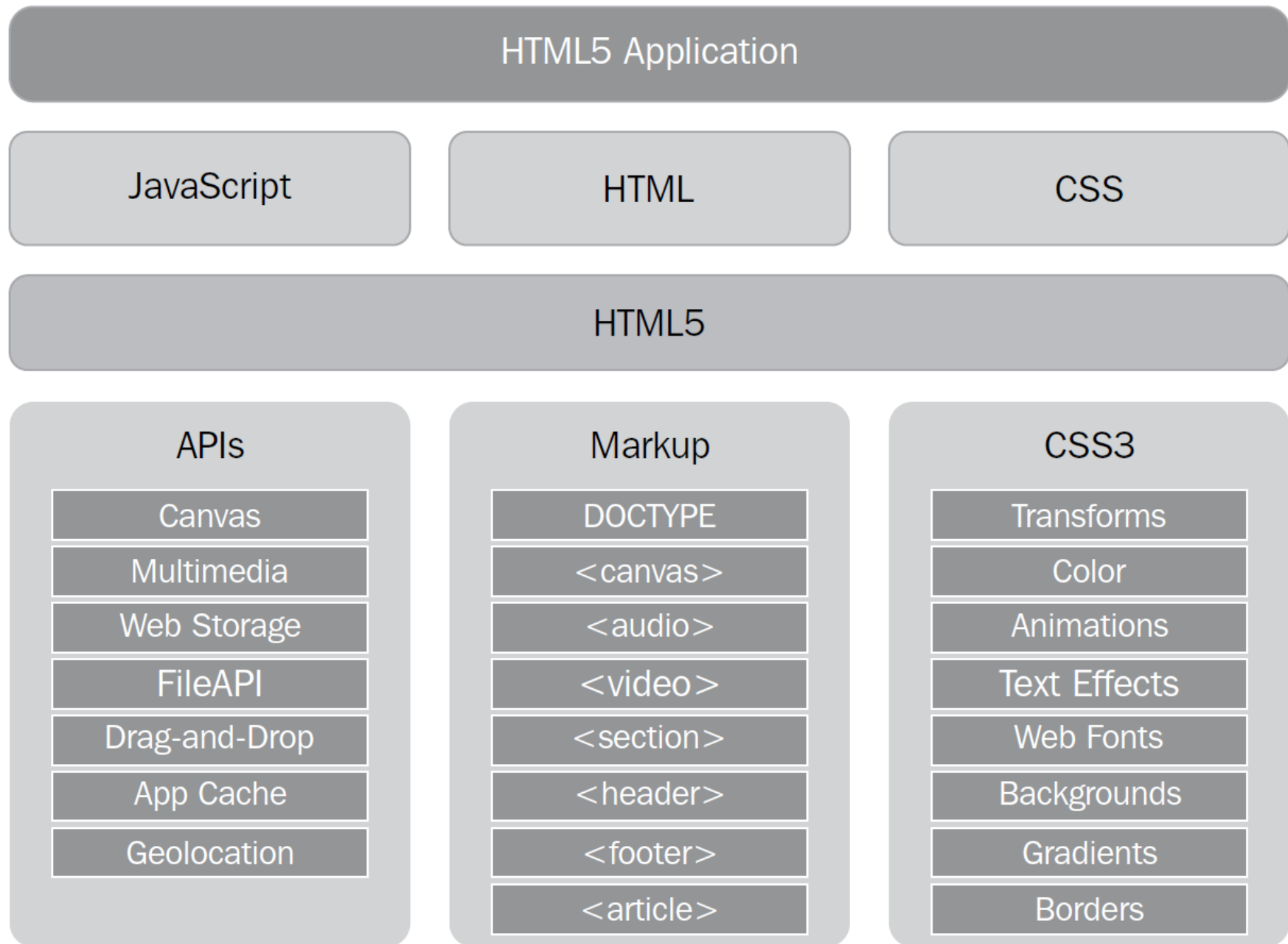# HTML & CSS

Web Programming

Suriya Natsupakpong, PhD

Institute of Field Robotics (FIBO)

King Mongkut's University of Technology Thonburi (KMUTT)

HTML5 Application		
JavaScript	HTML	CSS

HTML5

APIs	Markup	CSS3
Canvas	DOCTYPE	Transforms
Multimedia	<canvas>	Color
Web Storage	<audio>	Animations
FileAPI	<video>	Text Effects
Drag-and-Drop	<section>	Web Fonts
App Cache	<header>	Backgrounds
Geolocation	<footer>	Gradients
	<article>	Borders

Ref: J.M. Gustafson, HTML5 Web Application Development By Example Learn how to build rich, interactive web applications from the ground up

HTML

- HTML, or Hypertext Markup Language, is the standard markup language used to create web pages.

- It serves as the backbone of all web content, defining the structure and layout of a web page by using a series of elements or tags.
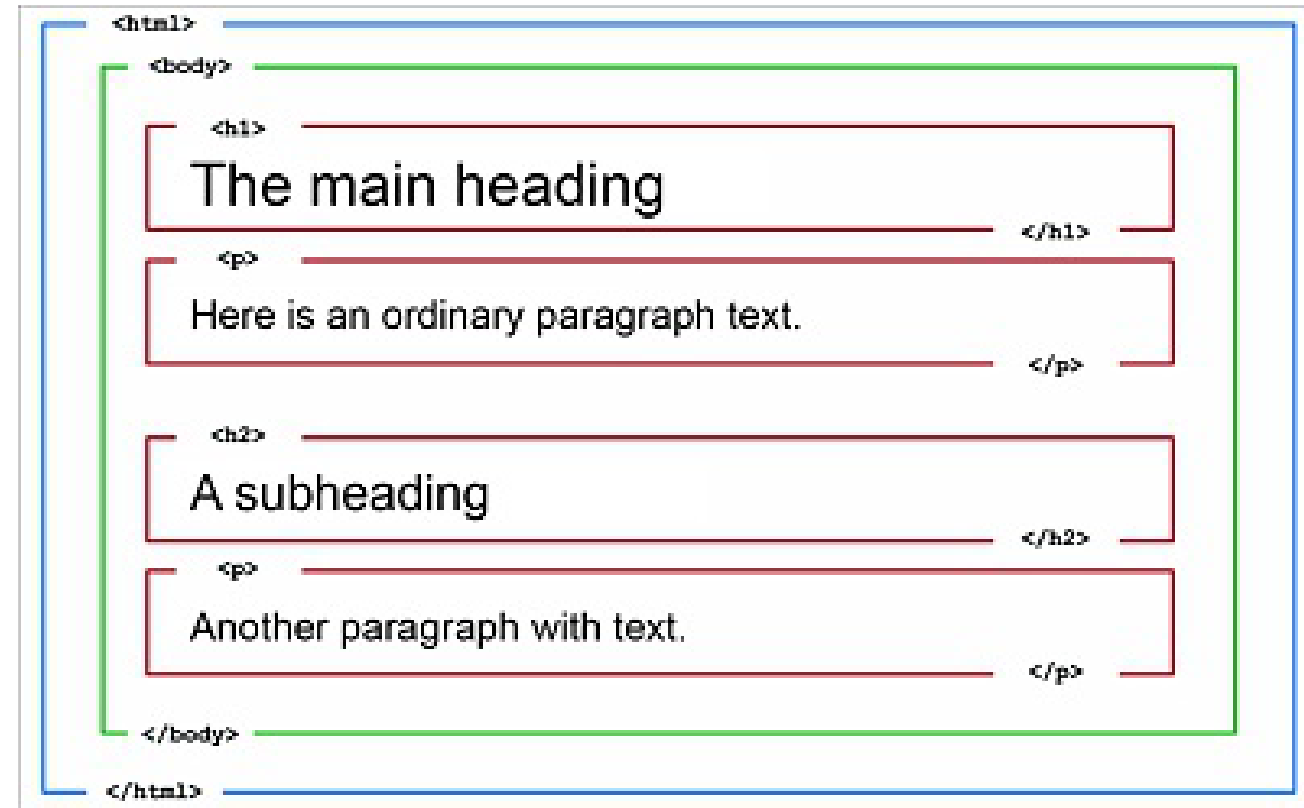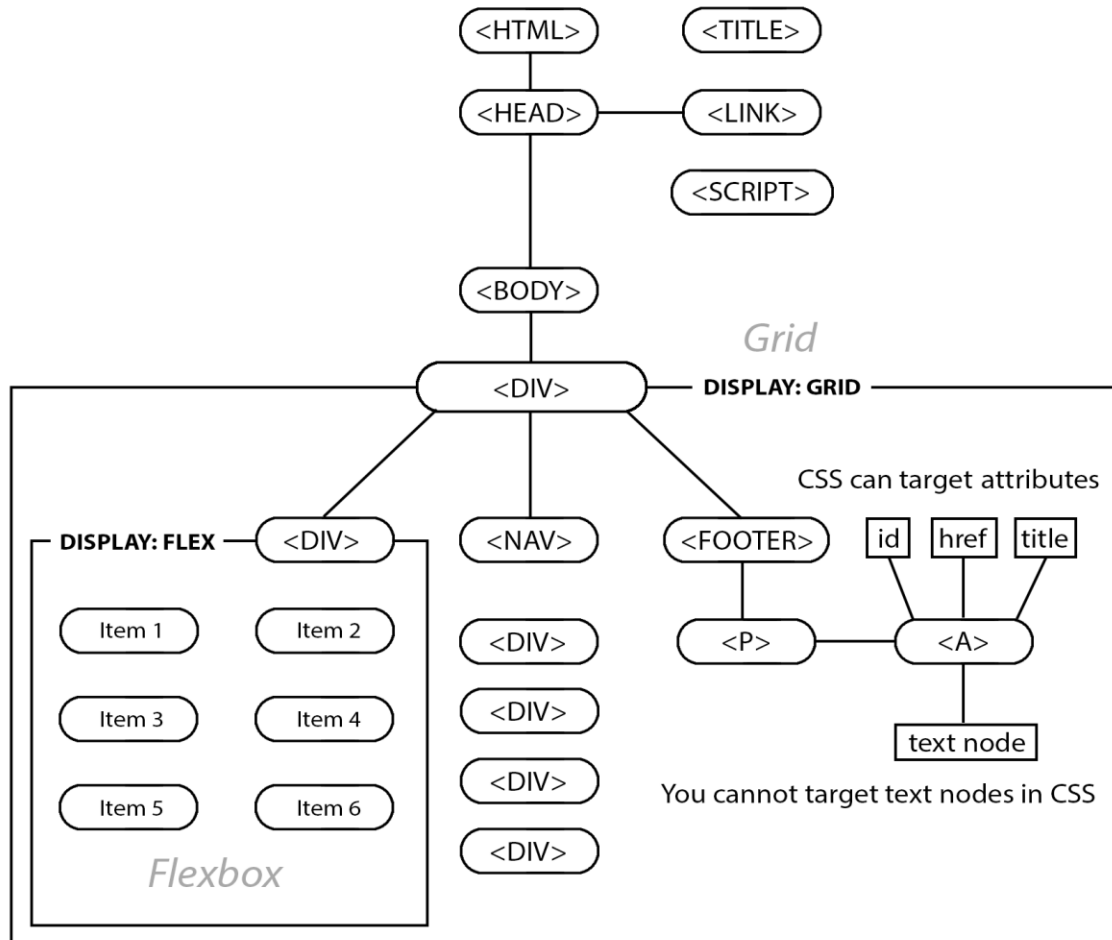
This tag is an HTML5 document. ⟶

This is the responsive meta tag. ⟶

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>My Portfolio</title>
</head>
<body>
    <h1>Hello World!</h1>
</body>
</html>
```

Each HTML tag is opened and closed in the right order.

Basic Page Structure of an HTML Document

Grid

DISPLAY: GRID

CSS can target attributes

You cannot target text nodes in CSS

DISPLAY: FLEX

Flexbox

Ref: Greg Sidelnikov, CSS – Visual Dictionary

Ref: Jürgen Wolf, HTML and CSS : the comprehensive guide

Example of HTML Tags

<div>: Defines a division or a section in an HTML document that is styled with CSS or manipulated with JavaScript

<article>: Defines an article in the document

<aside>: Defines content aside from the other page content

<footer>: Defines the footer for a section in the document

<header>: Defines the header for a section in the document

<nav>: Contains page navigation links

<section>: Defines a section in a document

CSS

Inline:	`<p style="color: blue;">This text is blue.</p>`
Internal	`<style> p { color: red; } </style>`
External	`<link rel="stylesheet" href="styles.css"/>`

- Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML.

- Basic CSS Properties:

  Colors, Fonts, and Layout Techniques

  - Colors can be applied using names, HEX codes, RGB, or HSL values. For example, color: blue; or backgroundcolor: #ffffff;.

  - Fonts are controlled via the font-family, font-size, font-weight, and other font-related properties.

  - Layout Techniques involve using properties like display, position, flexbox, and grid to control how elements are positioned and laid out on the page.

  The Box Model (margin, border, padding, width, height)

  - Margin - Controls the space outside the border.

  - Border - A line that goes around the padding and content.

  - Padding - The space between the border and the content.

  - Width and Height - Control the size of the content area.
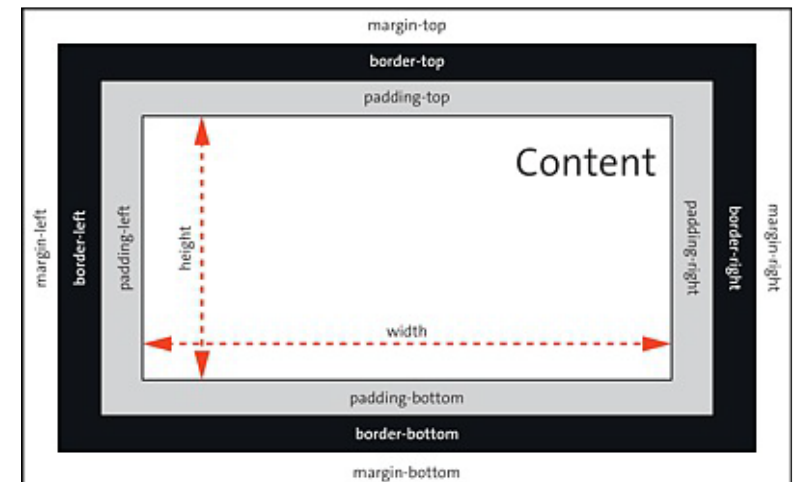
CSS

- Start with a clean canvas. Since each browser comes with its own default styles, you want to remove them first, otherwise these styles will clash with yours. This process is the purpose of a CSS reset by using minireset.css (https://jgthms.com/minireset.css/)

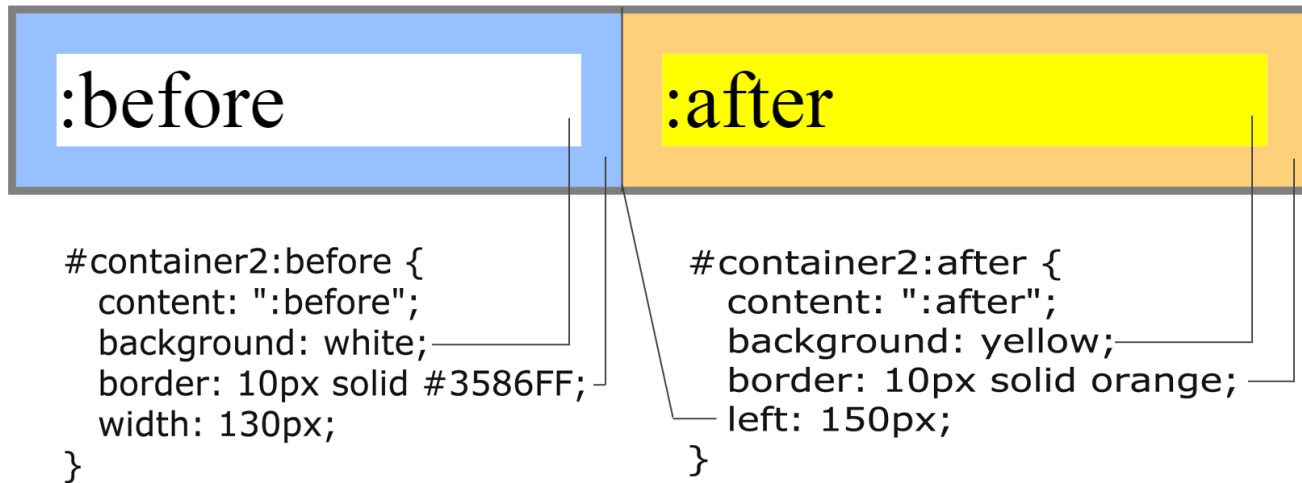- Add main.css

```css
html {
    background-color: blue;
}
```
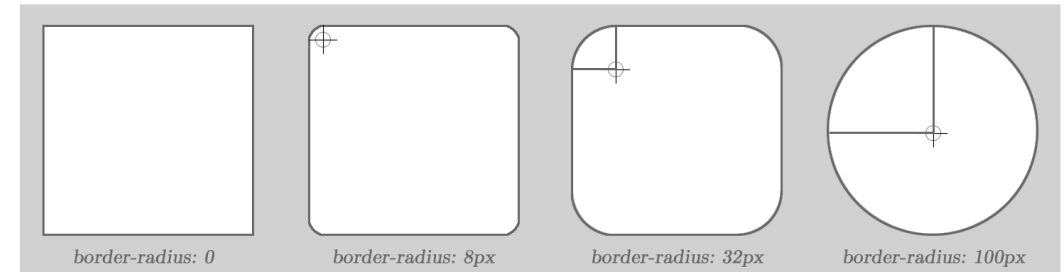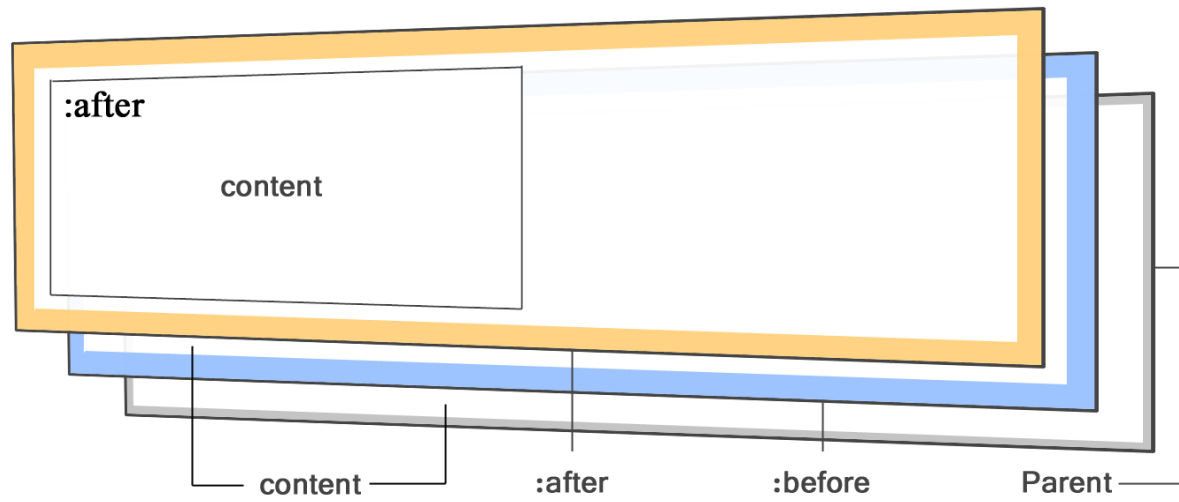
- In index.html file, adding the Google Fonts, minireset.min.css, and main.css
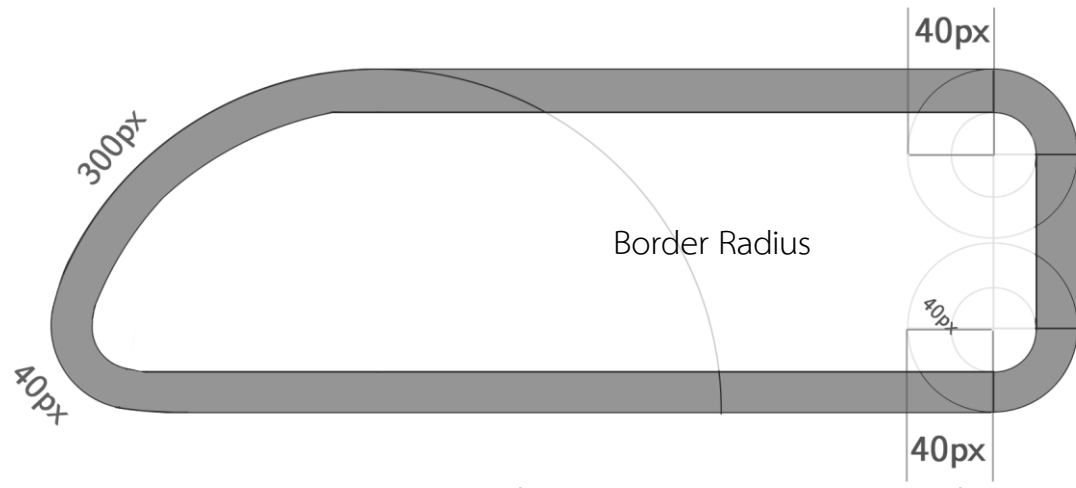
```html
<link href="https://fonts.googleapis.com/css?family=Montserrat|Quicksand" rel="stylesheet">
<link rel="stylesheet" href="css/minireset.min.css" />
<link rel="stylesheet" href="css/main.css" />
```

CSS Grammar

```
HTML_Tag_name {
/* CSS properties go here */
    property: value;
    property: value;
           ...
}
```

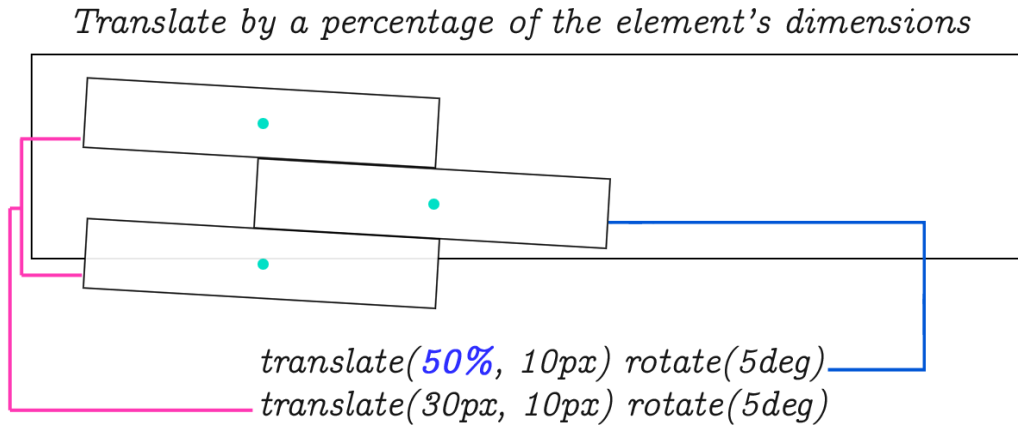|	
<div id = "box">content</div>	#box { property: value; }
<li class = "item">content	.item { line-height: 1.80; }
All HTML elements	:root { font-family: Arial, sans-serif; } * { font-family: Arial, sans-serif; }
All CSS variable names must begin with double dash --	:root { --red-color: red; } div { color: var(--red-color); }
CSS comments	/* comment */

:after

content

content | :after | :before | Parent

border-radius: 0 | border-radius: 8px | border-radius: 32px | border-radius: 100px

:before

:after

```
#container2:before {
    content: ":before";
    background: white;
    border: 10px solid #3586FF;
    width: 130px;
}
```

```
#container2:after {
    content: ":after";
    background: yellow;
    border: 10px solid orange;
    left: 150px;
}
```
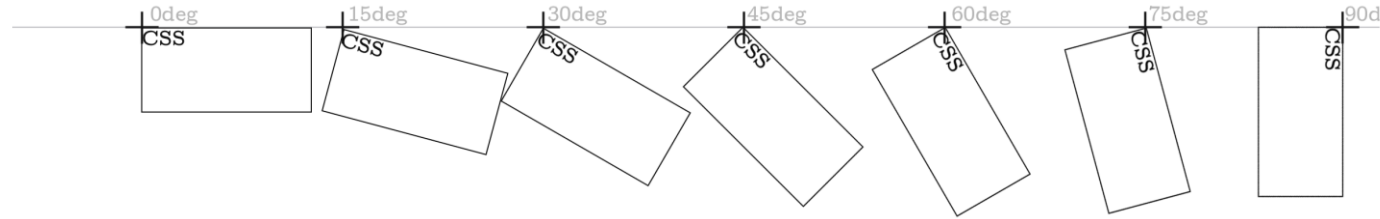
Border Radius

40px

300px

40px

40px

40px

border-top-left-radius:300px | border-top-left-radius:40px |
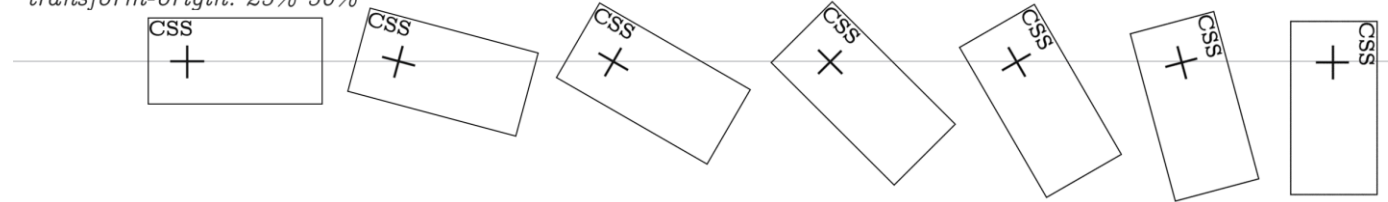border-bottom-left-radius:40px | border-bottom-right-radius:40px

200px

100px

CSS Is Awesome.

200px

500px
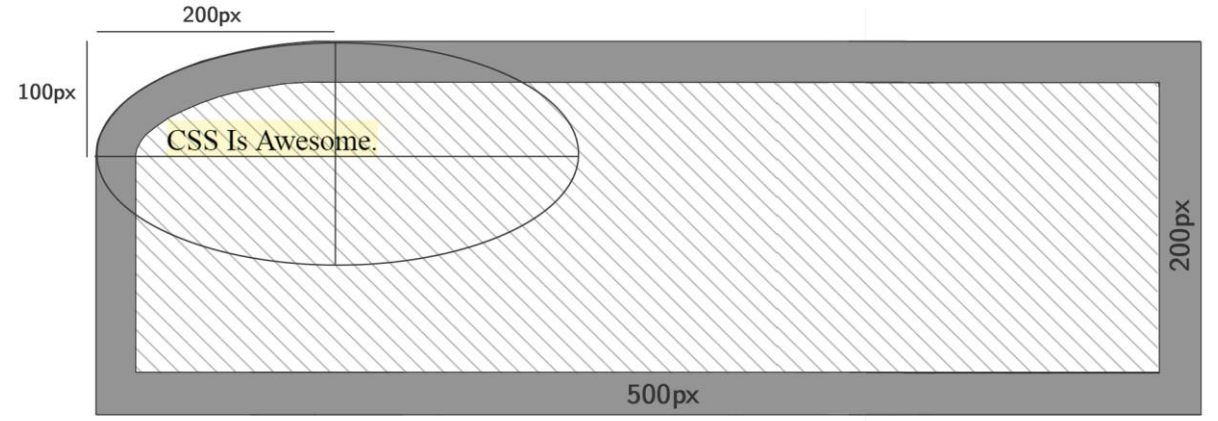
Elliptical border is created by using two parameters border-top-left-radius: 200px 100px

transform-origin: 0 0

0deg   15deg   30deg   45deg   60deg   75deg   90d

CSS   CSS   CSS   CSS   CSS   CSS   CSS

transform-origin: 25% 50%

CSS   CSS   CSS   CSS   CSS   CSS   CSS

transform-origin: 100% 0

CSS   CSS   CSS   CSS   CSS   CSS   CSS

Translate by a percentage of the element's dimensions

translate(50%, 10px) rotate(5deg)
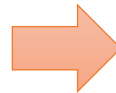*translate(30px, 10px) rotate(5deg)*
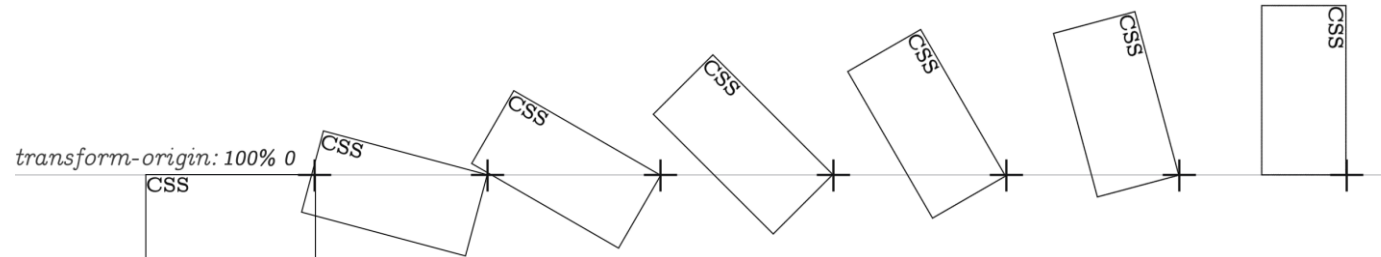
Syntactically Awesome Stylesheets: SASS

- A CSS preprocessor that adds new features not currently available in standard CSS specification.

- SCSS is recommended for advanced CSS specialists.

- Install:
```
npm install -g sass
```

- Compile .sass and .scss files to .css files:

```
sass source/stylesheets/index.scss build/stylesheets/index.css
```
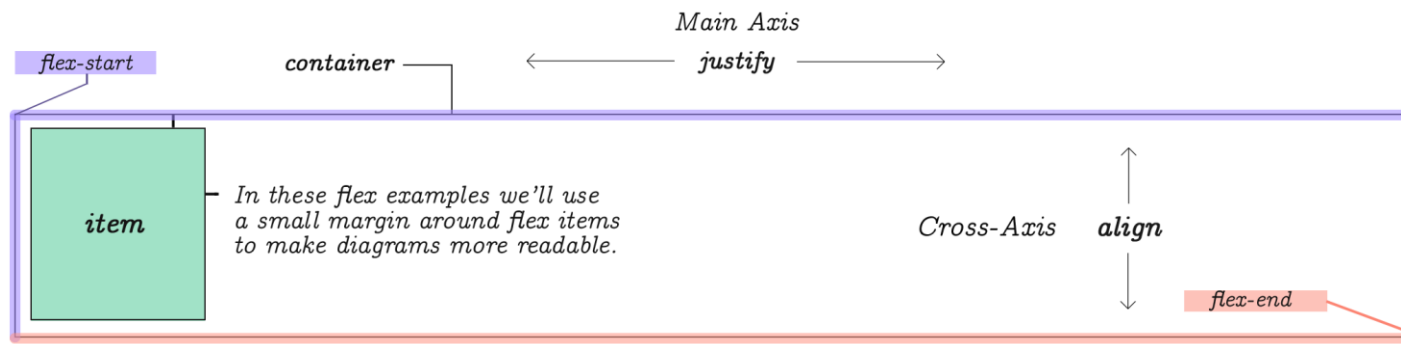
SCSS
```scss
$font-stack: Helvetica, sans-serif;
$primary-color: #333;

body {
  font: 100% $font-stack;
  color: $primary-color;
}
```
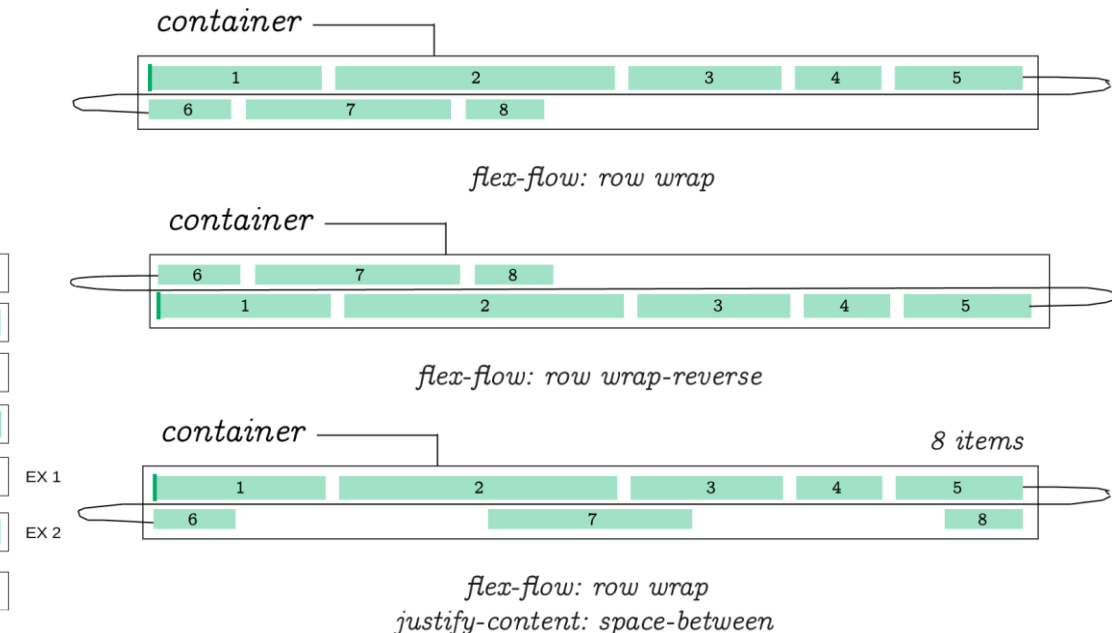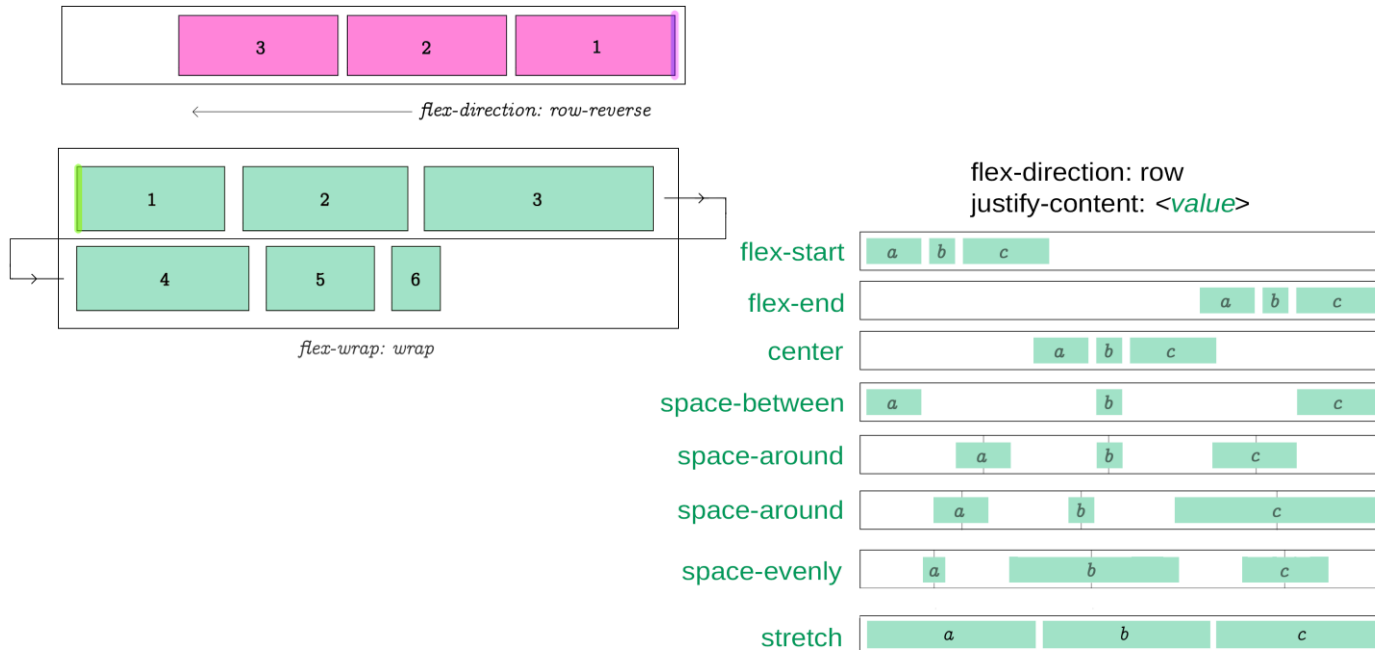
CSS
```css
body {
    font: 100% Helvetica, sans-serif;
    color: #333;
}
```

Flex

flex-start — *container* — Main Axis — *justify*

item — In these flex examples we'll use a small margin around flex items to make diagrams more readable.
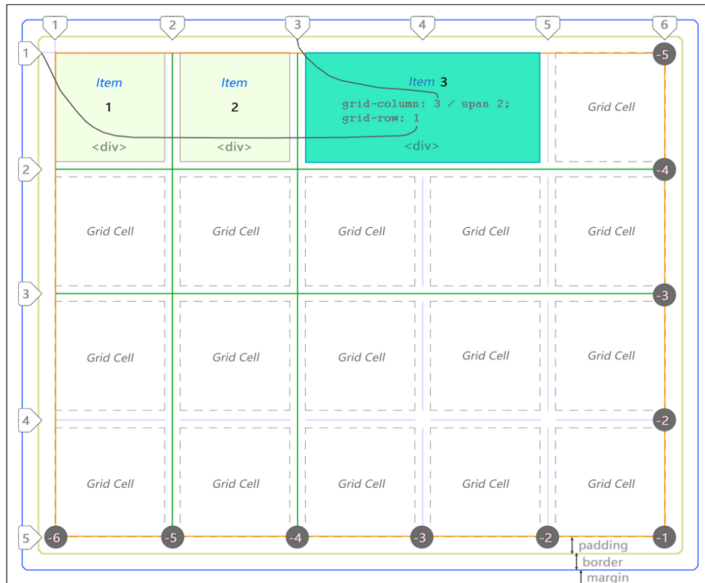
Cross-Axis — *align*

flex-end

- Flex is a set of rules for automatically stretching multiple columns and rows of content across parent container.

- Flex element as a parent container with display:flex. Elements placed inside this container are called items.
  Each container has a flex-start and flex-end points as shown on this diagram.

- Flex provides layout elements that dynamically respond to the dimensions of parent container.

flex-direction: row-reverse

flex-wrap: wrap

flex-direction: row
justify-content: *<value>*

- flex-start
- flex-end
- center
- space-between
- space-around   EX 1
- space-around   EX 2
- space-evenly
- stretch

container

flex-flow: row wrap

container

flex-flow: row wrap-reverse

container

8 items

flex-flow: row wrap
justify-content: space-between

Ref: Greg Sidelnikov, CSS – Visual Dictionary

Grid

Track Column

Track Row
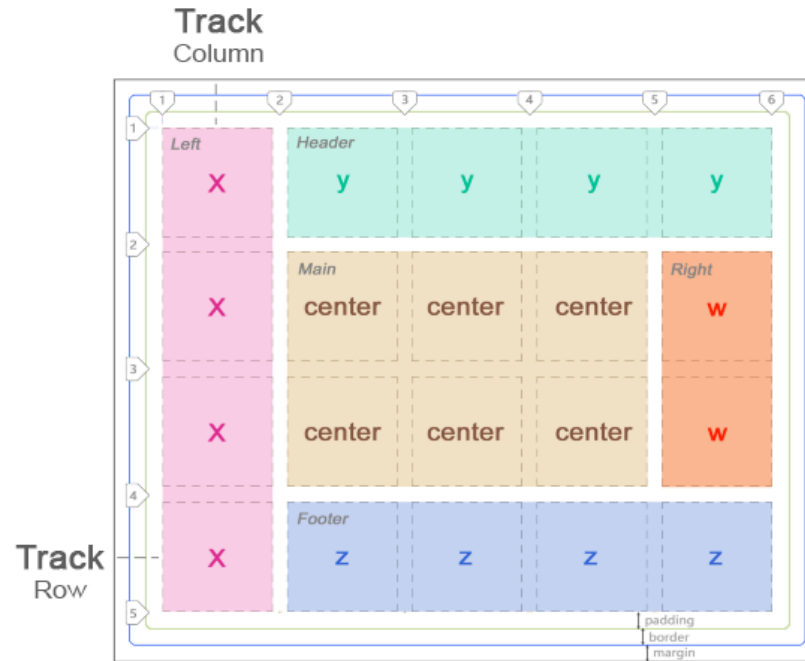
CSS

```
div#grid {
    display: grid;
    grid-template-areas:
        'x y y y y'
        'x center center center w'
        'x center center center w'
        'x z z z z';
}
```

HTML

```
<div id="grid">
        <div style="grid-area: x">Left</div>
        <div style="grid-area: y">Header</div>
        <div style="grid-area: z">Footer</div>
        <div style="grid-area: w">Right</div>
        <div style="grid-area: center">Main</div>
    </div>
```

Animation in CSS

- CSS animation keyframes are specified using the @keyframes directive.

HTML

```
<div class="animationClass"></div>
```

CSS

```css
.animationClass {
  width: 100px;
  height: 100px;
  position: relative;
  animation-name: animationName;
  animation-fill-mode: forward;
  animation-duration: 4s;
  animation-timing-function: linear;
  animation-delay: 2s;
  animation-iteration-count: infinite;
  animation-direction: alternate;
}

@keyframes animationName {
  0% { border-radius: 0; background: yellow; left:0px; top:0px;}
  25% { border-radius: 10px; background: green; left:500px; top:0px;}
  50% { border-radius: 30px; background: orange; left:400px; top:200px;}
  75% { border-radius: 40px; background: blueviolet; left:150px; top:200px;}
  100% { border-radius: 50px; background: teal; left:0px; top:0px;}
}
```

CSS Example

HTML

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="test.css" />
    <title>Document</title>
</head>
<body>
    <h1>Website Header</h1>
    <article>
        <p>Main Content</p>
    </article>
    <footer>
        <p>Privacy Policy. <span>&copy; 2025 Copyright</span></p>
    </footer>
</body>
</html>
```
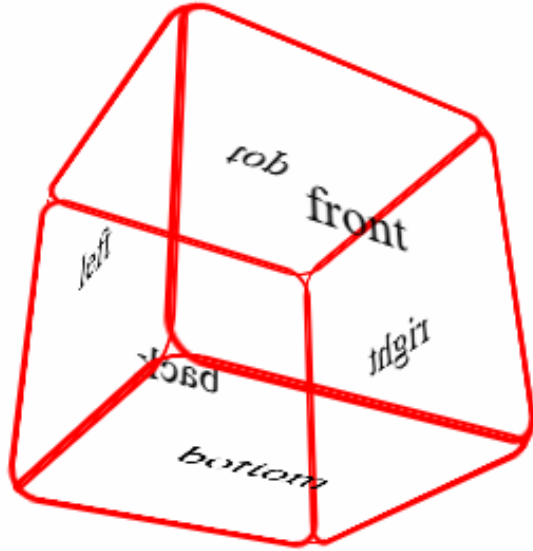
Website Header

Main Content

Privacy Policy. © 2025 Copyright

CSS

```css
body {
    background: black;
    color: white;
}

footer {
    color: red;
}

footer span {
    color: green;
}
```

3D Cube by CSS

HTML

```html
<div class="view">
    <div class="cube">
        <div class="face front">front</div>
        <div class="face back">back</div>
        <div class="face right">right</div>
        <div class="face left">left</div>
        <div class="face top">top</div>
        <div class="face bottom">bottom</div>
    </div>
</div>
```

CSS

```css
.view {
  width: 200px;
  height: 200px;
  perspective: 300px;
  position: fixed;
  bottom: 0;
}
.cube {
  width: 100%;
  height: 100%;
  position: relative;
  transform-style: preserve-3d;
  transform: rotateX(30deg) rotateY(30deg) translateX(50px) translateY(50px);
}
.face {
  position: absolute;
  width: 100px;
  height: 100px;
  background: transparent;
  border: 2px solid red;
  border-radius: 10px;
  backface-visibility: visible;
  text-align: center;
  line-height: 100px;
}
.front { transform: rotateY(0deg) translateZ(50px); }
.right { transform: rotateY(90deg) translateZ(50px); }
.back { transform: rotateY(180deg) translateZ(50px); }
.left { transform: rotateY(-90deg) translateZ(50px); }
.top { transform: rotateX(90deg) translateZ(50px); }
.bottom { transform: rotateX(-90deg) translateZ(50px); }
```
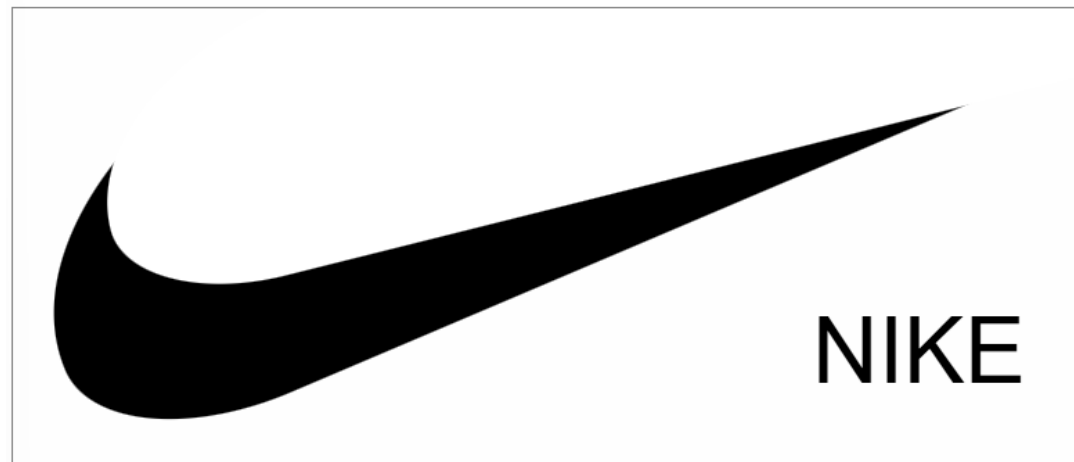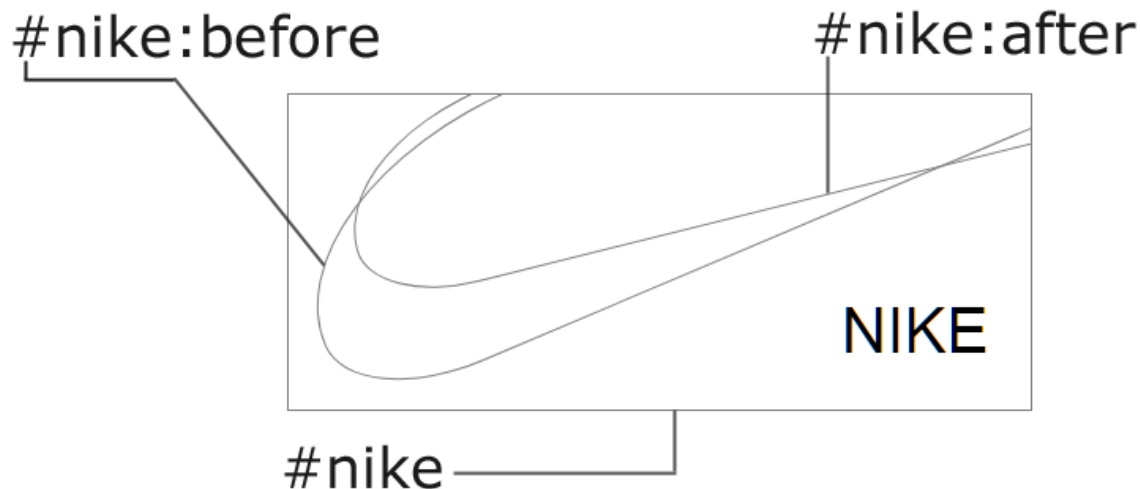
Nike Logo by CSS

```html
<div id="nike">NIKE</div>
```


#nike:before
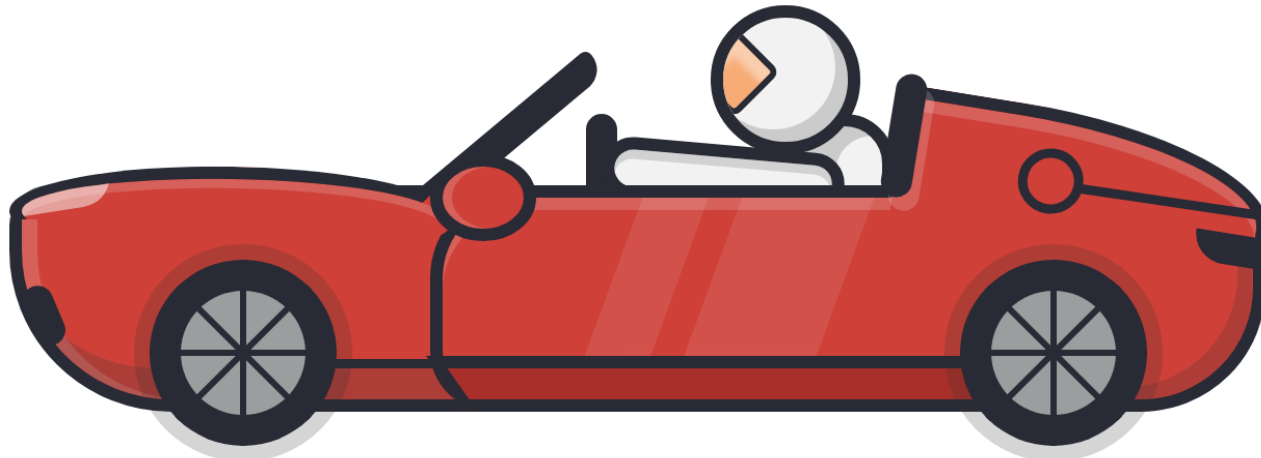
#nike:after

#nike

NIKE

```css
#nike {
    position: absolute;
    top: 300px; left: 300px;
    width: 470px; height: 200px;
    border: 1px solid gray;
    overflow: hidden;
    font-family: Arial, sans-serif;
    font-size: 40px;
    line-height: 300px;
    text-indent: 350px;
    z-index: 3;
}
```

```css
#nike:before {
    content: "";
    position: absolute;
    top: -250px; left: 190px;
    width: 150px; height: 550px;
    background: black;
    border-top-left-radius: 60px 110px;
    border-top-right-radius: 130px 220px;
    transform: rotate(-113deg);
    z-index: 1;
}
```
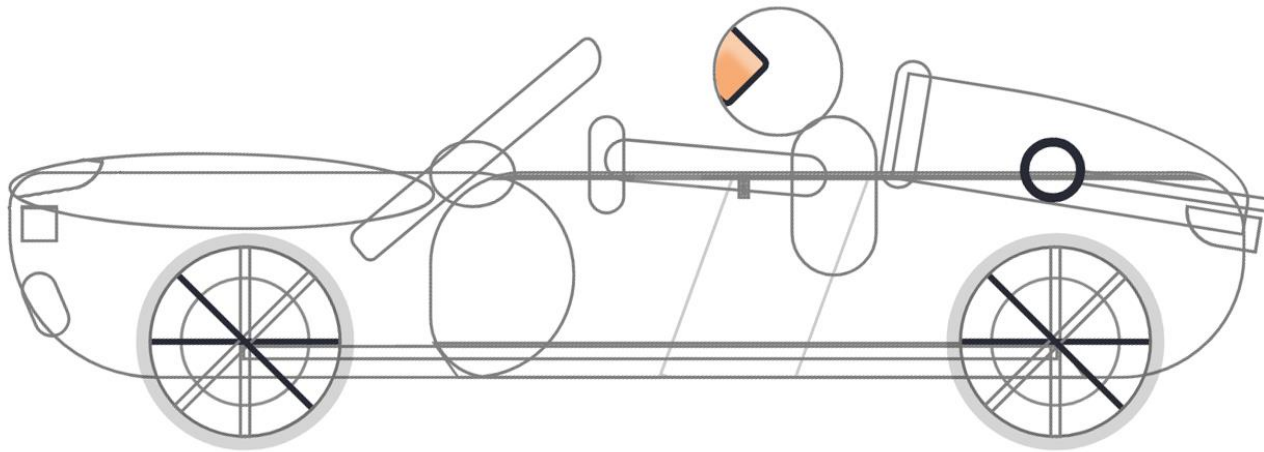
```css
#nike:after {
    content: "";
    position: absolute;
    top: -235px; left: 220px;
    width: 120px; height: 500px;
    background: black;
    border-top-left-radius: 60px 110px;
    border-top-right-radius: 130px 220px;
    background: white;
    transform: rotate(-104deg);
    z-index: 2;
}
```
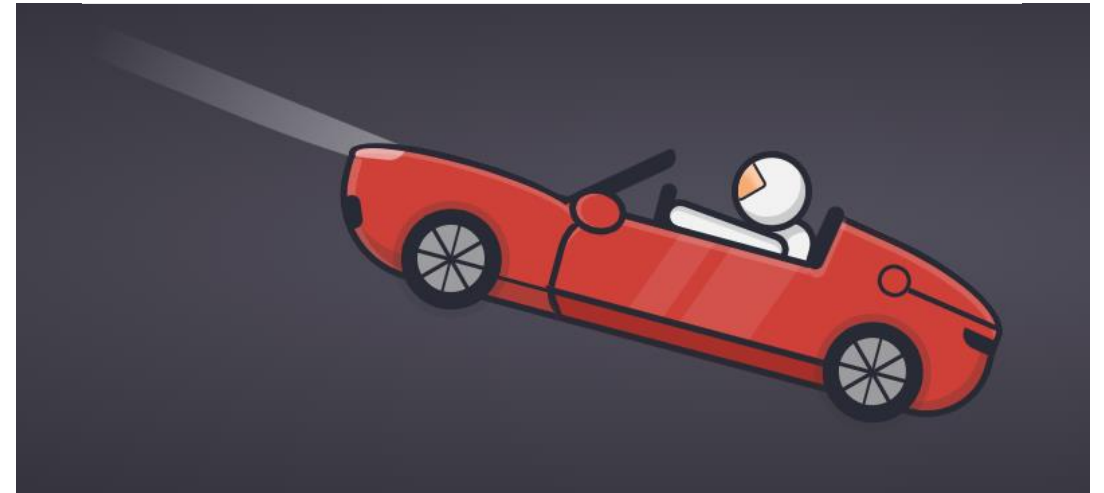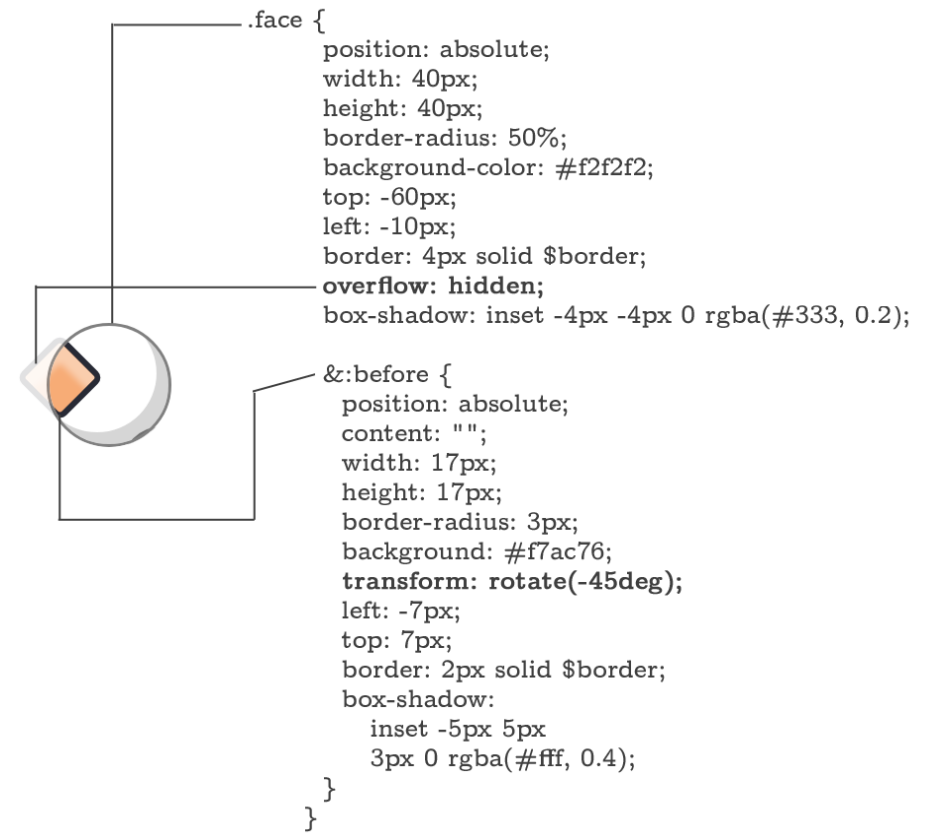
CSS Art

.front { left: -190px; }

.rear { right: -130px; }

```
.face {
    position: absolute;
    width: 40px;
    height: 40px;
    border-radius: 50%;
    background-color: #f2f2f2;
    top: -60px;
    left: -10px;
    border: 4px solid $border;
    overflow: hidden;
    box-shadow: inset -4px -4px 0 rgba(#333, 0.2);

    &:before {
        position: absolute;
        content: "";
        width: 17px;
        height: 17px;
        border-radius: 3px;
        background: #f7ac76;
        transform: rotate(-45deg);
        left: -7px;
        top: 7px;
        border: 2px solid $border;
        box-shadow:
            inset -5px 5px
            3px 0 rgba(#fff, 0.4);
    }
}
```

Ref: Greg Sidelnikov, CSS – Visual Dictionary

CSS Painting by Diana A Smith

https://diana-adrianne.com/purecss-vignes/

https://diana-adrianne.com/purecss-gaze/

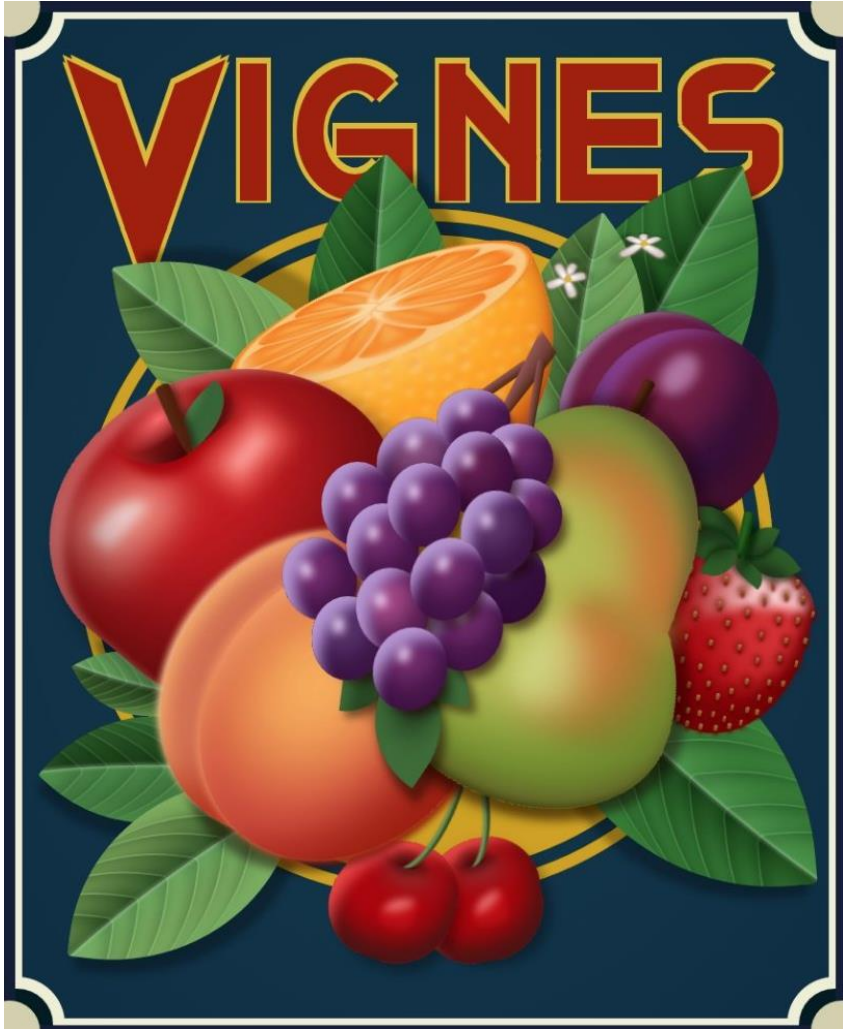# index.html

```html
<body>
  <div class="wallpaper"></div>
  <div class="content">
    <aside class="side">
      <figure class="picture">
        <div class="picture-shadow"></div>
        <img
          id="pictureImage"
          class="picture-image"
          src="images/suriya.jpg"
          srcset="
            images/suriya.jpg    1x,
            images/suriya@2x.jpg 2x,
            images/suriya@3x.jpg 3x
          "
          alt="Portrait of Suriya Natsupakpong"
          width="320"
          height="320"
        />
      </figure>
    </aside>
    <main class="about">
      <h1 class="name">Hi, I'm Suriya Natsupakpong</h1>
      <p class="job">Lecturer </p>
      <hr class="hr" />
```
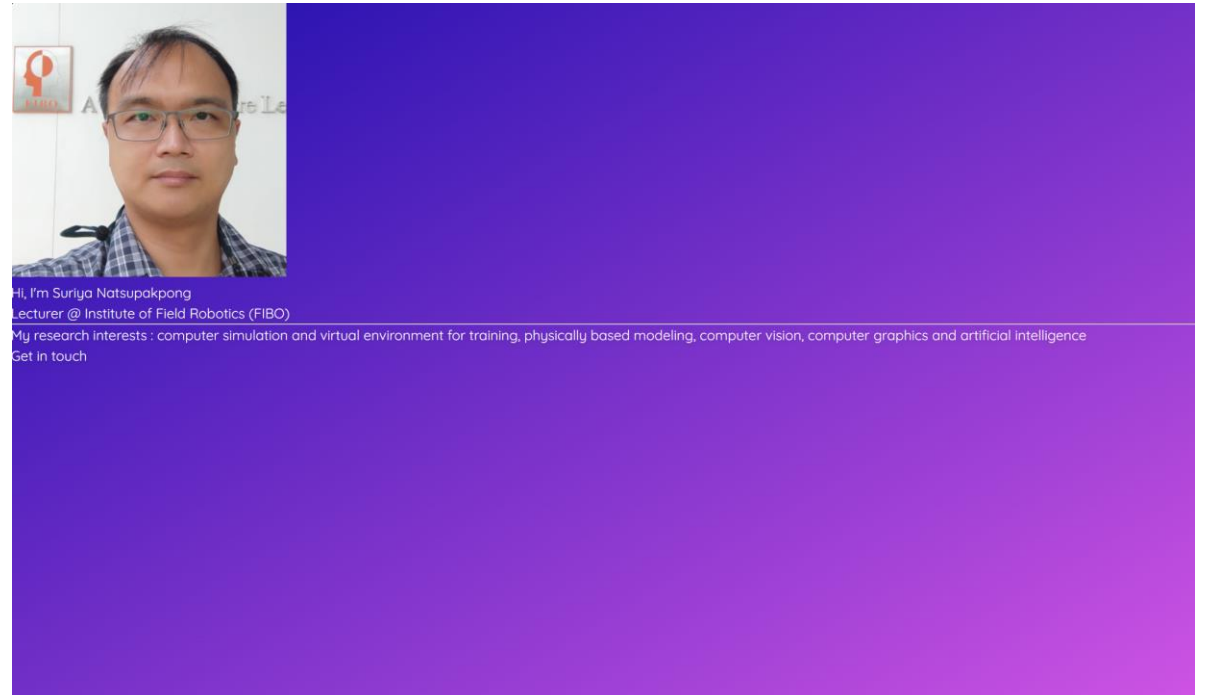
```html
      <div class="description">
        <p>
          My research interests : computer simulation and
          virtual environment for training, physically based
          modeling, computer vision, computer graphics and
          artificial intelligence
        </p>
      </div>
      <div class="contact">
        <ul class="social">
          <li>
            <a>
              <i class="fab fa-twitter"></i>
            </a>
          </li>
          <li>
            <a>
              <i class="fab fa-github"></i>
            </a>
          </li>
          <li>
            <a>
              <i class="fab fa-linkedin"></i>
            </a>
          </li>
        </ul>
        <a class="button" href="mailto:youremail@example.com">
          Get in touch
        </a>
      </div>
    </main>
  </div>
</body>
```

main.css

Hi, I'm Suriya Natsupakpong
Lecturer @ Institute of Field Robotics (FIBO)
My research interests : computer simulation and virtual environment for training, physically based modeling, computer vision, computer graphics and artificial intelligence
Get in touch

```css
html {
    background-color: #5f45bb;
    background-image: linear-gradient(to bottom right, #180cac, #d054e4);
    color: #fff;
    font-family: "Quicksand", sans-serif;
    font-size: 16px;
    -moz-osx-font-smoothing: grayscale;
    -webkit-font-smoothing: antialiased;
    line-height: 1.5;
    min-height: 100vh;
    min-width: 300px;
    overflow-x: hidden;
    text-shadow: 0 3px 5px rgba(0, 0, 0, 0.1);
}
```

```css
a {
    color: currentColor;
    cursor: pointer;
    text-decoration: none;
}
```

```css
/* 2. Shared */
.wallpaper {
  display: block;
  height: 100%;
  left: 0;
  top: 0;
  width: 100%;
}
```

```css
/* 3. Specific */
.wallpaper {
  background-image: url("../images/cloud@fibo.jpg");
  background-position: center;
  background-size: cover;
  opacity: 0.2;
  position: fixed;
}
```

```css
.content {
  display: flex;
  position: relative;
  min-height: 100vh;
}
.side {
  max-height: 20rem;
  max-width: 20rem;
}
.about {
  max-width: 26rem;
}
```

Responsiveness

- any CSS written inside that block will only be active if all the items in the media query list (screen and (max-width: 799px)) are true.

```css
@media screen and (max-width: 799px) {
    .content {
        flex-direction: column;
        justify-content: center;
        align-items: center;
        padding: 5rem 3rem;
    }
    .side {
        margin-bottom: 3rem;
        width: 100%;
    }
}
```



```css
@media screen and (min-width: 800px) {
    .content {
        align-items: center;
        justify-content: space-around;
        justify-content: space-evenly;
        padding: 4rem;
    }
    .side {
        flex-grow: 0;
        flex-shrink: 0;
        height: 20rem;
        margin-left: 4rem;
        order: 2;
        width: 20rem;
    }
    .about {
        flex-grow: 1;
        flex-shrink: 1;
    }
}
```

```css
.name {
  font-size: 2.25rem;
  line-height: 1.125;
  margin-bottom: 0.5rem;
}
```

```css
.job,
.button {
  font-family: "Montserrat", "Quicksand", sans-serif;
  letter-spacing: 0.3em;
  text-transform: uppercase;
}
```

```css
.job {
  color: #ffe479;
  font-size: 0.75rem;
}
```

```css
.hr {
  background-color: #ff470f;
  border: none;
  content: "";
  height: 1px;
  margin-bottom: 1.5rem;
  margin-top: 1.5rem;
  transform-origin: center left;
  width: 4rem;
}
```

```css
.wallpaper,
.picture-shadow,
.picture-image {
  display: block;
  height: 100%;
  left: 0;
  top: 0;
  width: 100%;
}
```

```css
.picture {
  padding-top: 100%;
  position: relative;
  width: 100%;
}
.picture-shadow {
  border-radius: 290486px;
  background-image: radial-gradient(#000 0%, rgba(0, 0, 0, 0) 70%);
  position: absolute;
  top: 10%;
}
.picture-image {
  border-radius: 290486px;
  position: absolute;
}
```

My research interests : computer simulation and virtual environment for training, physically based modeling, computer vision, computer graphics and artificial intelligence

GET IN TOUCH

```css
.description {
  font-size: 1.5rem;
}

.contact {
  display: inline-block;
  margin-top: 1.5rem;
  vertical-align: top;
}
```

```css
.button,
.social a {
  transform-origin: center;
  transition-duration: 100ms;
}
```

```css
.button {
  background-color: #fff;
  border-radius: 290486px;
  box-shadow: 0 1rem 2rem rgba(0, 0, 0, 0.2);
  color: #9013fe;
  display: inline-block;
  font-size: 0.875rem;
  line-height: 1;
  padding: 1.25em 2em;
  text-shadow: none;
  transition-property: box-shadow, transform;
  user-select: none;
  vertical-align: top;
  white-space: nowrap;
  will-change: box-shadow, transform;
}
```
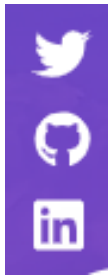
```css
.button:hover {
  box-shadow: 0 1.5rem 3rem rgba(0, 0, 0, 0.2);
  transform: scale(1.02) translateY(-4px);
}

.button:active {
  box-shadow: 0 0.5rem 1rem rgba(0, 0, 0, 0.3);
  transform: scale(0.98) translateY(-2px);
}
```

Font Awesome Library

- Before the closing </head> tag in index.html , include the following script:

```
<script defer src="https://use.fontawesome.com/releases/v5.0.0/js/all.js"></script>
```

- Right after the <div class="contact"> , add these social icons:

```html
        <ul class="social">
          <li>
            <a>
              <i class="fab fa-twitter"></i>
            </a>
          </li>
          <li>
            <a>
              <i class="fab fa-github"></i>
            </a>
          </li>
          <li>
            <a>
              <i class="fab fa-linkedin"></i>
            </a>
          </li>
        </ul>
```

```css
.social {
  display: flex;
  margin-top: 1.5rem;
}
.social li {
  height: 2.5rem;
  margin-right: 0.5rem;
  text-align: center;
  width: 2rem;
}
```

```css
.social a {
  align-items: center;
  display: flex;
  font-size: 1.5rem;
  height: 2rem;
  justify-content: center;
  opacity: 0.5;
  transition-property: opacity, transform;
  width: 2rem;
  will-change: opacity, transform;
}
.social a:hover {
  opacity: 1;
  transform: scale(1.25);
}
.social a:active {
  opacity: 1;
  transform: scale(1.1);
}
```

CSS Animations

- Create a new file animations.css in the /css folder and add this link tag into index.html.

```
<link rel="stylesheet" href="css/animations.css">
```

```css
@keyframes bounceIn {
  from {
    opacity: 0;
    transform: scale(0.5);
  }
  to {
    opacity: 1;
    transform: scale(1);
  }
}
@keyframes fillUp {
  from {
    transform: scaleX(0);
  }
  to {
    transform: scaleX(1);
  }
}
@keyframes picImage {
  from {
    opacity: 0;
    transform: scale(1.2) translateY(-1rem);
  }
  to {
    opacity: 1;
    transform: scale(1) translateY(0);
  }
}
```

```css
@keyframes picShadow {
  from {
    opacity: 0;
    transform: scale(1.2) translateY(4rem);
  }
  to {
    opacity: 1;
    transform: scale(1.1) translateY(0);
  }
}
@keyframes slideDown {
  from {
    opacity: 0;
    transform: translateY(-1rem);
  }
  to {
    opacity: 1;
    transform: translateY(0);
  }
}
```

```css
@keyframes slideUp {
  from {
    opacity: 0;
    transform: translateY(1rem);
  }
  to {
    opacity: 1;
    transform: translateY(0);
  }
}
@keyframes zoomOut {
  from {
    opacity: 0;
    transform: scale(1.05);
  }
  to {
    opacity: 0.2;
    transform: scale(1);
  }
}
```

```css
.wallpaper,
.picture-shadow,
.picture-image,
.name,
.job,
.hr,
.description,
.contact,
.social li {
  animation-duration: 1s;
  animation-timing-function: cubic-bezier(0, 0.5, 0, 1);
  animation-fill-mode: both;
}

.contact {
  animation-name: bounceIn;
}

.picture-shadow,
.picture-image {
  animation-duration: 750ms;
  animation-timing-function: cubic-bezier(0, 0.5, 0.25, 1.25);
}

.wallpaper {
  animation-name: zoomOut;
  animation-timing-function: ease-out;
}
.picture-shadow {
  animation-name: picShadow;
}
.picture-image {
  animation-name: picImage;
}
.name {
  animation-name: slideDown;
}
.job {
  animation-name: slideUp;
}
.hr {
  animation-name: fillUp;
}
.description {
  animation-name: slideUp;
}
```

```css
.social li {
  animation-duration: 500ms;
  animation-name: slideUp;
  animation-timing-function: cubic-bezier(0.5, 0, 0.5, 1.5);
}
.name {
  animation-delay: 100ms;
}
.job {
  animation-delay: 200ms;
}
.hr {
  animation-delay: 300ms;
}
.description {
  animation-delay: 400ms;
}
.picture-image {
  animation-delay: 500ms;
}
.picture-shadow {
  animation-delay: 500ms;
}
.contact {
  animation-delay: 600ms;
}
.social li:nth-child(1) {
  animation-delay: 800ms;
}
.social li:nth-child(2) {
  animation-delay: 900ms;
}
.social li:nth-child(3) {
  animation-delay: 1s;
}
.social li:nth-child(4) {
  animation-delay: 1.1s;
}
.social li:nth-child(5) {
  animation-delay: 1.2s;
}
```

Fun 02: About me

- ให้ทำหน้าเว็บเพจ About me โดยมีองค์ประกอบดังนี้
  1. ชื่อ-นามสกุล (ชื่อเล่น) ความสนใจและอาชีพในฝัน

  2. Animation in CSS

  3. Art or Logo using CSS