<meta charset="utf-8" name="viewport" content="width=device-width, initial-scale=1">

<link rel="stylesheet" media="(max-width:360px)" type="text/css" href="flex4.css">

<link rel="stylesheet" media="(min-width:360px)" type="text/css" href="flex3.css">

.flex-container {

display: flex;

background-color: darkblue;

/\* object otomatis turun. \*/

flex-wrap: wrap;

/\* Other options: wrap , wrap-reverse \*/

/\* column, turun kebawah \*/

/\*flex-direction: column;\*/

flex-flow: row wrap;

/\*align-items: flex-start; \*/

justify-content: center;

align-items: center;}

[data-icon]:before {

font-family: icons; /\* BYO icon font, mapped smartly \*/

content: attr(data-icon);

speak: none; /\* Not to be trusted, but hey. \*/

}

@media screen and (max-width: 700px) {.respo {width: 100vw;}}

@media screen and (min-width: 700px) {.respo {width: 50vw; }}

.imagespot {

width: 640px;

height: 240px; /\* exact dimension of image \*/

background: #ffffff url(images/500.jpg) top left no-repeat;}

<img class="resp" src="daihatsu.jpg">

<!-- unicode, icon fonts -->

<h3>

<span aria-hidden="true" data-icon="&#x21dd;"></span>

Stats</h3>

<!-- &#77; &#68; &#69; https://unicode-table.com/en/#control-character -->

<!-- https://css-tricks.com/examples/IconFont/ -->

<!-- srcset -->

<img class='respo' srcset="images/1600.jpg 1600w, images/1000.jpg 1000w, images/800.jpg 800w, images/500.jpg 500w">

<picture>

<source media="(min-width: 1000px)" srcset="images/1000.jpg 1x, images/1600.jpg 2x">

<source media="(min-width: 500px)" srcset="images/800.jpg 1x, images/500.jpg 2x">

<img src="daihatsu.jpg" alt="A Small Picture">

</picture>

<!-- text dalam image -->

<div class="imagespot">

text here

</div>

<!-- FIGCAPTION -->

<figure>

<!-- text selalu berada dibawah gambar, seperti penjelasan untuk gambar. -->

<img src="daihatsu.jpg" class="gambar">

<figcaption>I'm skiing</figcaption>

</figure>

<!-- SVG -->

<svg width="100" height="100">

<circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow">

</svg>

<button onclick="startup()">Initialize</button><br>

function startup() {

document.getElementById('para1').addEventListener('touchstart', handle\_touchstart);

document.getElementById('para1').addEventListener('touchmove', handle\_touchmove);

document.getElementById('para1').addEventListener('touchend', handle\_release);

document.getElementById('canvas').addEventListener('touchstart', handle\_touchstart);

document.getElementById('canvas').addEventListener('touchend', handle\_release);

console.log('initialized')}

function handle\_touchstart(ev) {console.log(ev.touches, ev.type); }

function handle\_touchmove(ev) {console.log(ev.touches, ev.type); }

function handle\_release(ev) {console.log(ev.touches, ev.type); }

// API

const ul = document.getElementById('authors');

const url = 'https://randomuser.me/api/?results=5';

function createNode(element) {

return document.createElement(element);

}

function append(parent, element) {

return parent.appendChild(element);

}

function getUsers() {

fetch(url).then((resp) => resp.json())

.then(handleReponse)

.catch(error);

}

function handleReponse(data) {

let users = data.results;

for (var i in users) {

users\_name = users[i].name;

console.log(users\_name['first'] + " " + users\_name['last']);

let li = createNode('li');

let img = createNode('img');

let span = createNode('span');

img.src = users[i].picture.large;

span.innerHTML = `${users\_name.first} ${users\_name.last}`;

append(li, img);

append(li, span);

append(ul, li);

}

}

function error(e) {console.log(e);}