**Aulão Tiktok**

Parte 1: Introdução

Começamos com uma estrutura HTML básica que contém algumas informações na header:

* Tag meta com caracteres padrão;
* Tag meta com compatibilidade para o navegador Edge;
* Tag meta configurando a responsividade e o zoom padrão;
* Tag title com o título “Tiktok Clone”;
* Tag link com a folha de estilos;
* Tag link com a biblioteca de ícones do Google;

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>TikTok Clone</title>

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

<link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet" />

</head>

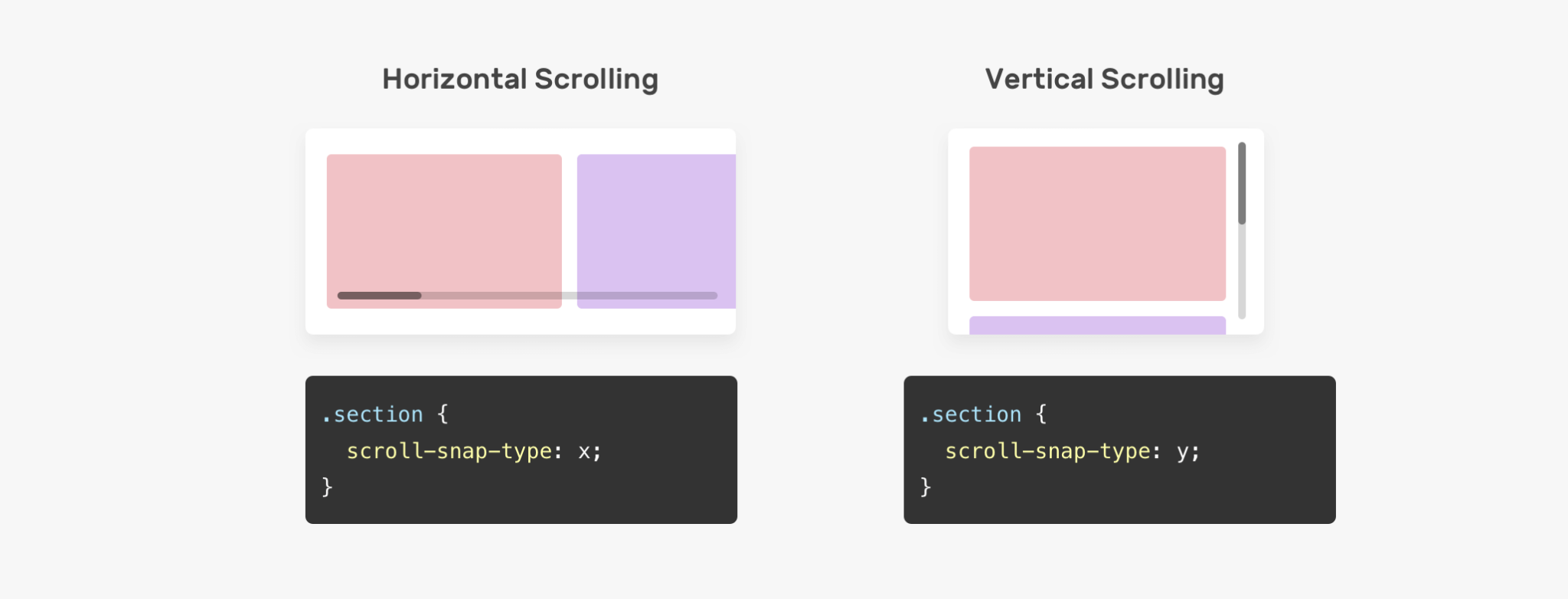
<body>

<!--Conteúdo da aula-->

</body>

</html>

E na sequência configuramos o CSS, se atente a propriedade scroll-snap-type:



Código CSS:

/\* Reseta a margem de todos os elementos, bordas são consideradas \*/

\* {

margin: 0;

box-sizing: border-box;

}

/\* Define que a rolagem será vertical \*/

html {

scroll-snap-type: y mandatory;

}

/\* Cor de textos branca, cor de fundo preta, altura vh\*/

/\* A medida vh é igual a 1/100 da altura da viewport \*/

/\* se a altura do navegador é 900px, 1vh equivale a 9px.\*/

/\* display em grade, com itens centralizados\*/

body {

color: white;

background-color: black;

height: 100vh;

display: grid;

place-items: center;

}

Parte 2: Área dos vídeos

Dentro de <body> crie a divisão com a classe “app\_\_videos”, que vai comportar nossos vídeos:

<body>

<div class="app\_\_videos"></div>

</body>

E agora vamos configurar o código CSS para essa divisão:  
/\* posição relativa, altura de 750px, fundo branco \*/

/\* se o conteúdo estourar a tela, é cortado e ganha barra de rolagem \*/

/\* largura 100% do que estiver disponível, máximo de 400px \*/

/\* rolagem vertical e arredondado em 20px \*/

.app\_\_videos {

position: relative;

height: 750px;

background-color: white;

overflow: scroll;

width: 100%;

max-width: 400px;

scroll-snap-type: y mandatory;

border-radius: 20px;

}

/\* barra de rolagem invisível \*/

.app\_\_videos::-webkit-scrollbar {

display: none;

}

/\* é uma propriedade CSS proprietária,

específica do Internet Explorer e do Microsoft Edge,

que controla o comportamento das barras de rolagem quando o

conteúdo de um elemento transborda. \*/

/\* A propriedade scrollbar-width é usada para definir a largura

ou espessura da barra de rolagem de um elemento quando mostrado. \*/

.app\_\_videos {

-ms-overflow-style: none;

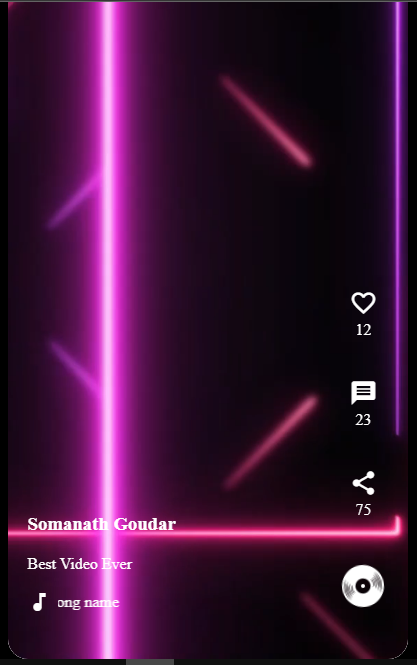
scrollbar-width: none;

}

Parte 3: Criando um vídeo

Nosso vídeo será composto de três partes:

* A <div> com a classe “vídeo” que carrega o conteúdo;
* A <div> com a classe “videoSidebar” que carrega os ícones laterais;
* A <div> com a classe “footer” com o título do vídeo, descrição e trilha;



Comece abrindo a div com a classe “video” e vamos construir a primeira parte, que carrega o conteúdo:

<div class="video">

<video class="video\_\_player" src="video1.mp4"></video>

</div>

Logo depois, vamos criar a segunda parte, com os ícones:

<div class="videoSidebar">

<div class="videoSidebar\_\_button">

<span class="material-icons"> favorite\_border </span>

<p>12</p>

</div>

<div class="videoSidebar\_\_button">

<span class="material-icons"> message </span>

<p>23</p>

</div>

<div class="videoSidebar\_\_button">

<span class="material-icons"> share </span>

<p>75</p>

</div>

</div>

E finalizando, criamos o footer com as informações do vídeo:

<div class="videoFooter">

<div class="videoFooter\_\_text">

<h3>Codi Academy</h3>

<p class="videoFooter\_\_description">O Programa Aprendizado Estratégico está disponível no Youtube e Spotify.</p>

<div class="videoFooter\_\_ticker">

<span class="material-icons videoFooter\_\_icon"> music\_note </span>

<marquee>Nome da Música</marquee>

</div>

</div>

<img

src="https://static.thenounproject.com/png/934821-200.png"

alt=""

class="videoFooter\_\_record"

/>

</div>

Agora vamos configurar o CSS, parte a parte:

/\* o espaço do video terá posição relativa, altura e altura preenchendo

a tela \*/

/\* cor de fundo branca\*/

/\* A scroll-snap-align especifica a posição de encaixe da

caixa como um alinhamento de sua área de encaixe

(como o assunto de alinhamento) dentro do snapport de seu recipiente

de encaixe (como o recipiente de alinhamento).\*/

.video {

position: relative;

height: 100%;

width: 100%;

background-color: white;

scroll-snap-align: start;

}

/\* Encaixa o conteúdo na parte de vídeos, com altura e largura total\*/

.video\_\_player {

object-fit: cover;

width: 100%;

height: 100%;

}

/\* posição relativa, a 48% do topo e a 10px da direita\*/

.videoSidebar {

position: absolute;

top: 48%;

right: 10px;

}

/\* ícones com tamanho 28px e o cursor vira uma mão em cima deles \*/

.videoSidebar .material-icons {

font-size: 28px;

cursor: pointer;

}

/\* padding de 20px em volta, texto alinhado do centro\*/

.videoSidebar\_\_button {

padding: 20px;

text-align: center;

}

/\* posição relativa, a 150px do fim da tela, margem esquerda de 20px

textos da cor branca e display flexível\*/

.videoFooter {

position: relative;

bottom: 150px;

margin-left: 20px;

color: white;

display: flex;

}

/\* função para girar o disco \*/

@keyframes spinTheRecord {

from {

transform: rotate(0deg);

}

to {

transform: rotate(360deg);

}

}

/\* configura a animação do disco, chamando a função anterior \*/

.videoFooter\_\_record {

animation: spinTheRecord infinite 5s linear;

height: 50px;

filter: invert(1);

position: absolute;

bottom: 0;

right: 20px;

}

/\* textos ocupam 100% do espaço flexível \*/

.videoFooter\_\_text {

flex: 1;

}

/\* Texto h3 com padding inferior de 20px \*/

.videoFooter\_\_text h3 {

padding-bottom: 20px;

}

/\* ícone do disco com posição absoluta \*/

.videoFooter\_\_icon {

position: absolute;

}

/\* Espaço com nome da música e disco terá 400px de largura \*/

/\* display flexível e itens alinhados ao centro \*/

.videoFooter\_\_ticker {

width: 400px;

display: flex;

align-items: center;

}

/\* nome da música com altura ajustada, margem esquerda de 30px \*/

/\* e ocupará 60% da largura disponível \*/

.videoFooter\_\_ticker marquee {

height: fit-content;

margin-left: 30px;

width: 70%;

}

/\* descrição do vídeo com padding de 20px em todas as direções \*/

.videoFooter\_\_description {

padding-bottom: 20px;

}

/\* ajuste da mídia \*/

@media (max-width: 425px) {

.app\_\_videos {

width: 100%;

height: 100%;

max-width: 100%;

border-radius: 0;

}

}

Agora adicione um segundo vídeo, trocando o conteúdo para video2.mp4.

Em seguida, verifique se a rolagem funciona.

Parte 4: Funções Pause e Play com JavaScript

Agora vamos criar um script JavaScript para pausar ou dar play em um vídeo.

A constante “videos” recebe uma referência aos vídeos disponíveis, varremos todos com o for e se clicarmos, se estiver pausado do play, se estiver rodando pausa.

<script>

const videos = document.querySelectorAll('video');

for (const video of videos) {

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

console.log('clicked');

if (video.paused) {

video.play();

} else {

video.pause();

}

});

}

</script>