

Нашия файл трябва да е най отдолу за да можем да стилизираме файлове заредени от други библиотеки!!!

 img {

    width: 100%;

    height: 100%;

    border-radius: 6px;

    box-shadow: 1px 1px 13px -1px black;

    object-fit: cover;

 }

Изрязва снимките колкото е необходимо



Reset file

/\* Box sizing rules \*/

\*,

\*::before,

\*::after {

  box-sizing: border-box;

}

/\* Remove default margin \*/

body,

h1,

h2,

h3,

h4,

p,

figure,

blockquote,

dl,

dd {

  margin: 0;

}

/\* Remove list styles on ul, ol elements with a list role, which suggests default styling will be removed \*/

ul[role='list'],

ol[role='list'] {

  list-style: none;

  margin: 0;

  padding: 0;

}

/\* Set core root defaults \*/

html:focus-within {

  scroll-behavior: smooth;

}

/\* Set core body defaults \*/

body {

  min-height: 100vh;

  text-rendering: optimizeSpeed;

  line-height: 1.5;

}

/\* A elements that don't have a class get default styles \*/

a:not([class]) {

  text-decoration-skip-ink: auto;

}

/\* Make images easier to work with \*/

img,

picture {

  max-width: 100%;

  display: block;

}

/\* Inherit fonts for inputs and buttons \*/

input,

button,

textarea,

select {

  font: inherit;

}

/\* Remove all animations, transitions and smooth scroll for people that prefer not to see them \*/

@media (prefers-reduced-motion: reduce) {

  html:focus-within {

   scroll-behavior: auto;

  }

  \*,

  \*::before,

  \*::after {

    animation-duration: 0.01ms !important;

    animation-iteration-count: 1 !important;

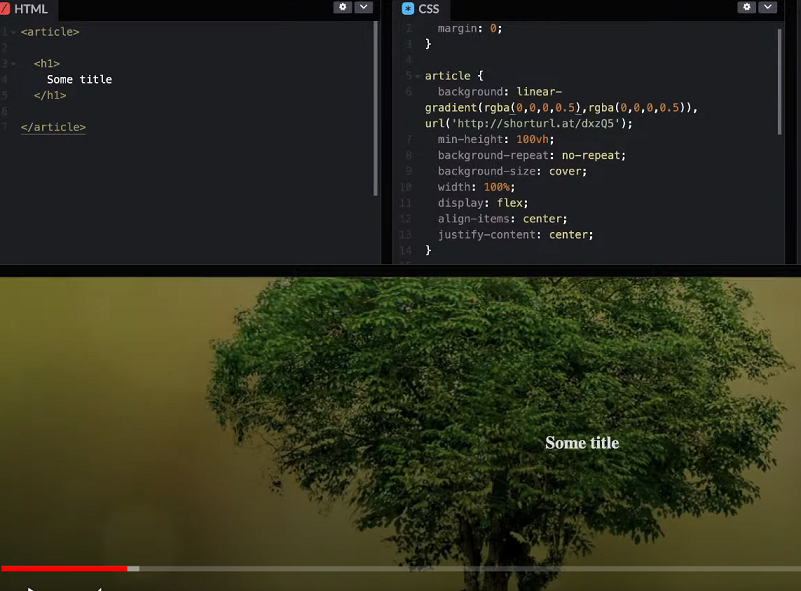
    transition-duration: 0.01ms !important;

    scroll-behavior: auto !important;

  }

}

Overlay efect with background-image



Cover fit image.

.thumbnail { това е контейнера 

    height: 210px; на снимката

    width: 210x;

    border-bottom: 2px solid red;

.thumbnail\_\_img {

width: 100%;

  height: 100%; това е снимката

  object-fit: cover;

}

Font Awesome

@import url('https://use.fontawesome.com/releases/v5.6.3/css/all.css');

@import url('https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.3.0/css/all.min.css');

Overlay efect with ::before

body {

    font-family: Arial, Helvetica, sans-serif;

}

.hero {

    width: 100vw;

    height: 100vh;

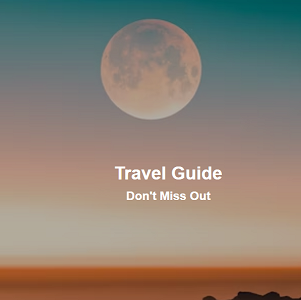
    position: relative;

}

.hero::before {

    content: "";

    width: 100%;

    height: 100%;

    position: absolute;

    top: 0;

    left: 0;

    background-color: rgb(0, 0, 0, 0.3);

}

.hero\_\_img {

    width: 100%;

    height: 100%;

    object-fit: cover;

}

.wraper {

    color: white;

    position: absolute;

    top: 50%;

    left: 50%;

    transform: translate(-50%, -50%);

    text-align: center;

}

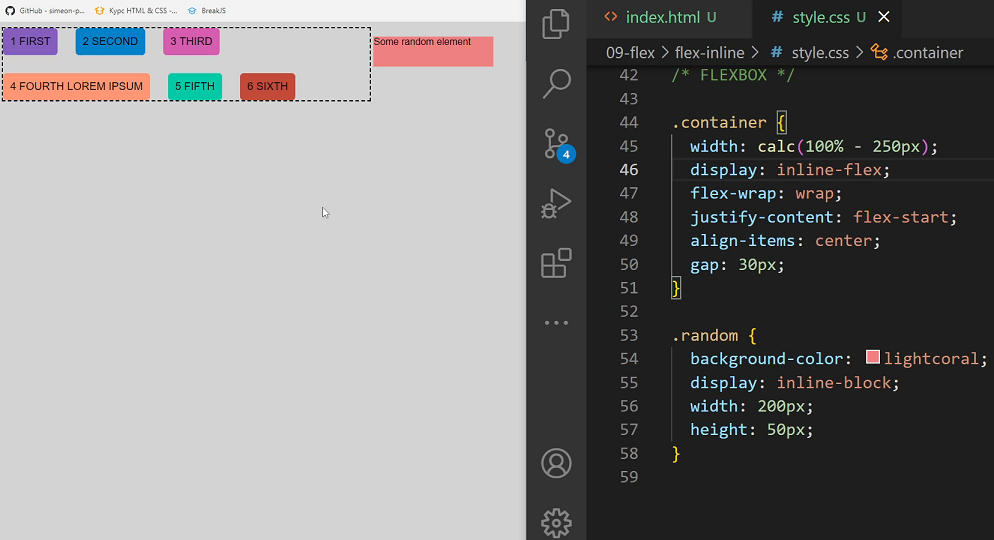
Unset all defolt property from buttons

.btn {

    all: unset;

}

Inline-flex property



след като зададем inline-flex на контейнера

той започва да се държи като inline-block

елемент и освобождава мястото до него за

друг елемент , а вътре в конейнера се запазват

всички flex свойства

вертикална линия

 <header class="header">

        <div class="logo">

            <h1>WIREWORK</h1>

            <div class="vl"></div> създаваме празен div с клас vl

            <p>BY TEMPLATED</p>

        </div>

    </header>

.vl {

    border-left: 1px solid grey; стилизираме класа vl

    height: 30px;}

media querie min-width and max width

@media  (min-width:415px) and (max-width:768px) {

    html {

        background-color: aqua; този код ще се изпълнява в диапазона от 415px

    } до 768px

}

@media  (max-width:414px) { този код ще се изпълнява от 0px до 414px

    html {

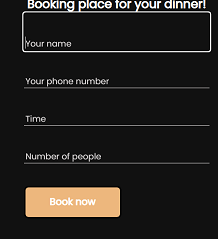
        background-color: red;

    }

}

Move placeholder to bottom

input {

    background-color: transparent; 

    border: none;

    border-bottom: 0.5px solid #d5d2d2;

    font-size: 0.7rem;

    height: 50px;

    position: relative;

    outline: none;

    cursor: text; премахва

    color:  #d5d2d2; бордера

}

input::placeholder {

    position: absolute;

    bottom: 0;

    color: #d5d2d2;

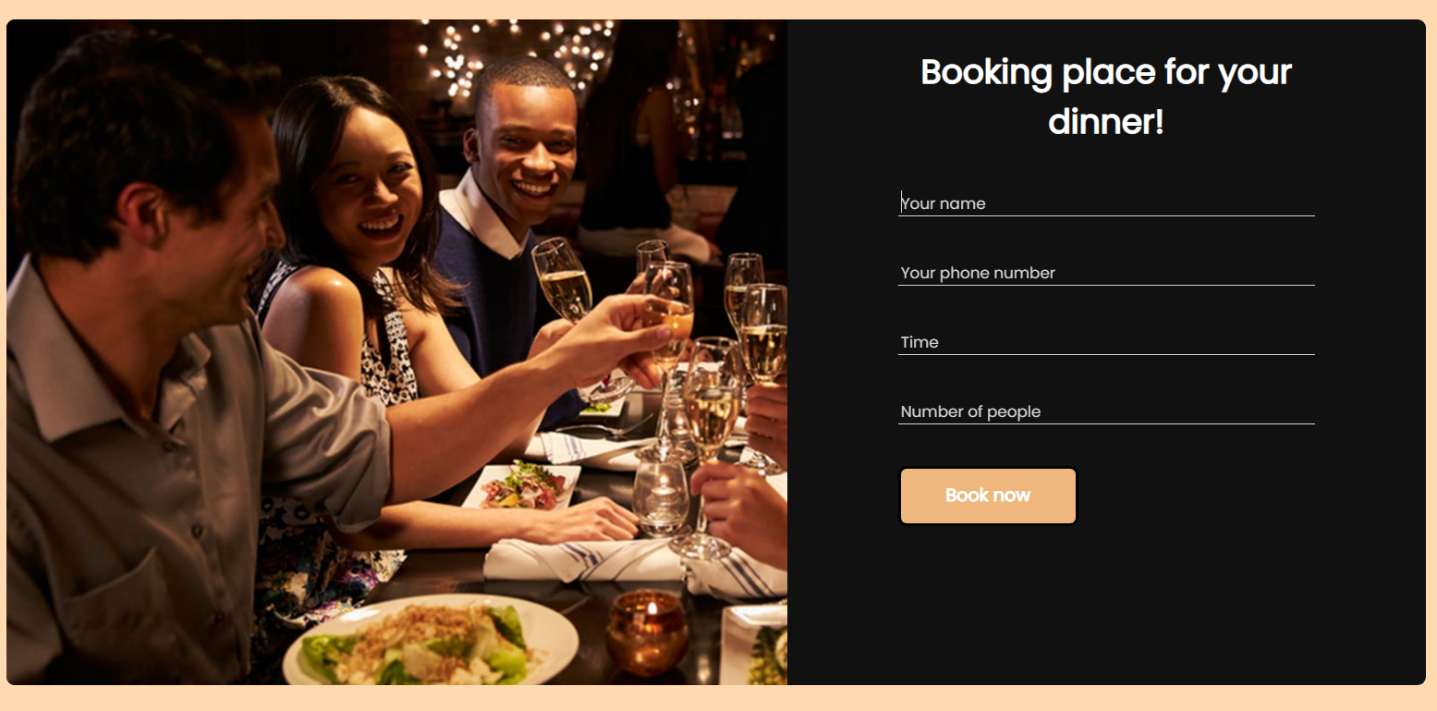
}

Move the text in input field to the bottom

input[type=text] {

    padding-top: 30px;

}



Background img with object-fit

<body>

    <header class="header">

        <img src="assets/logo\_icon.png" alt="logo">

        <button class="btn">Sign In</button>

    </header>

    <section id="hero">

        <img class="hero\_\_bg" src="assets/bg.jpg" alt="bg">

        <h1 class="title">Unlimited movies, TV shows, and more.</h1>

        <h2 class="subtitle">Watch anywhere. Cancel anytime.</h2>

        <h2 class="subtitle">Ready to watch? Enter your email to create or restart your membership.</h2>

        <form action="" class="form">

            <input type="text" placeholder="Email address">

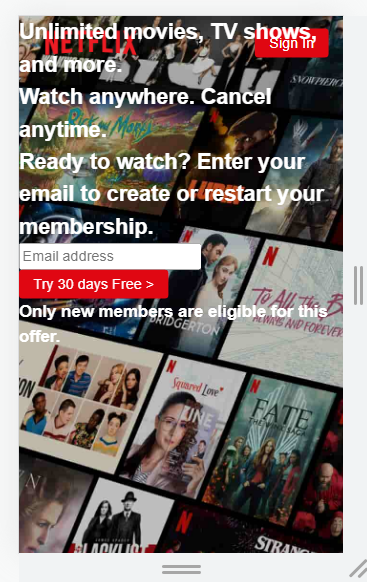
            <button class="btn">Try 30 days Free ></button>

        </form>

<h3 class="desc">Only new members are eligible for this offer.</h3>

</section>

</body>

.header {

    position: fixed;

    top: 0;

    left: 0;

    width: 100%;

    height: 60px;

    display: flex;

    justify-content: space-between;

    align-items: center;

    overflow: hidden;

    padding: 0 1rem;

}

/\* Hero \*/

#hero {

   height: 100vh;

   position: relative;

}

#hero .hero\_\_bg {

    position: absolute;

    z-index: -1;

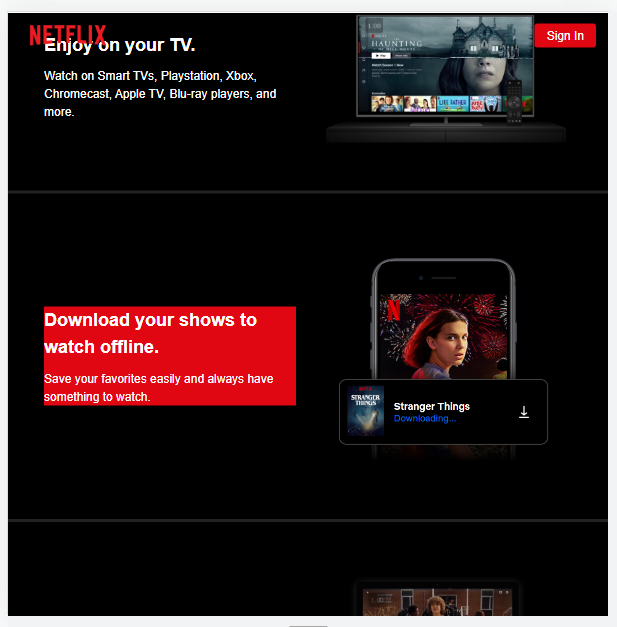
    height: 100%; отива като най заден слой

    width: 100%;

    object-fit: cover;

}

Change the position for two elements in flex or grid

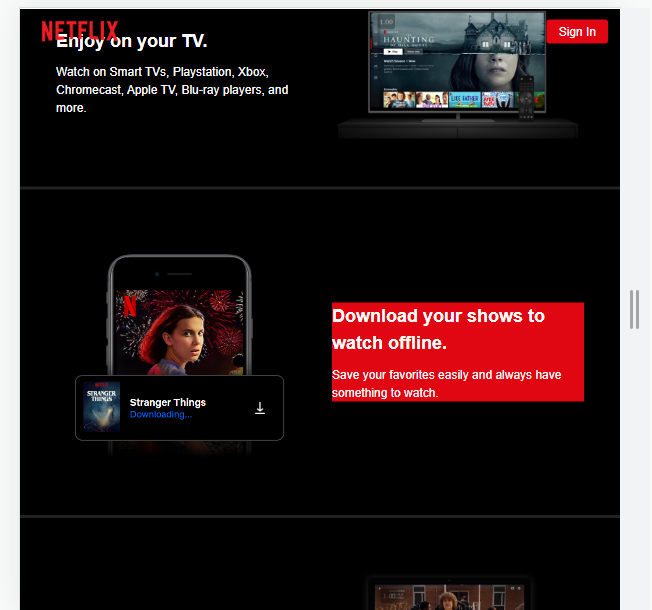
 before

    .content\_\_section:nth-of-type(odd) .content {

        background-color: #e00712;

        order: 1; задаване на ордър който измества червения

    } елемент на предна позиция в случая надясно

 after