1. Problem Statement

Create a simple webpage with a header, a main content section, and a footer. Apply a media query that changes the background color of the header to blue when the screen width is less than 600px.

Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Simpal Web page</title>

<style>

h1{

background-color:#a7f5f5;

padding: 30px;

text-align: center;

}

@media screen and (max-width: 600px) {

h1{

background-color: blue;

}

}

</style>

</head>

<body>

<header>

<h1>Responsive Header</h1>

</header>

<main>

<p>Main contant</p>

</main>

<footer>

<p>Footer</p>

</footer>

</body>

</html>

Output:



Width less than 600 px



1. Problem Statement

Create an image gallery with three images in a row. Use media queries to adjust the layout to two images in a row for screens smaller than 800px and one image in a row for screens smaller than 500px.

Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>image Gallary</title>

<style>

.image-gallery {

display: flex;

flex-wrap: wrap;

justify-content: space-between;

}

.image-gallery img {

width: 30%;

margin: 10px;

}

@media screen and (max-width: 800px) {

.image-gallery img {

width: 45%;

}

}

@media screen and (max-width: 500px) {

.image-gallery img {

width: 100%;

}

}

</style>

</head>

<body>

<div class="image-gallery">

<img src="/Media query Assignment/image.jpg" alt="image 1">

<img src="/Media query Assignment/image.jpg" alt="image 2">

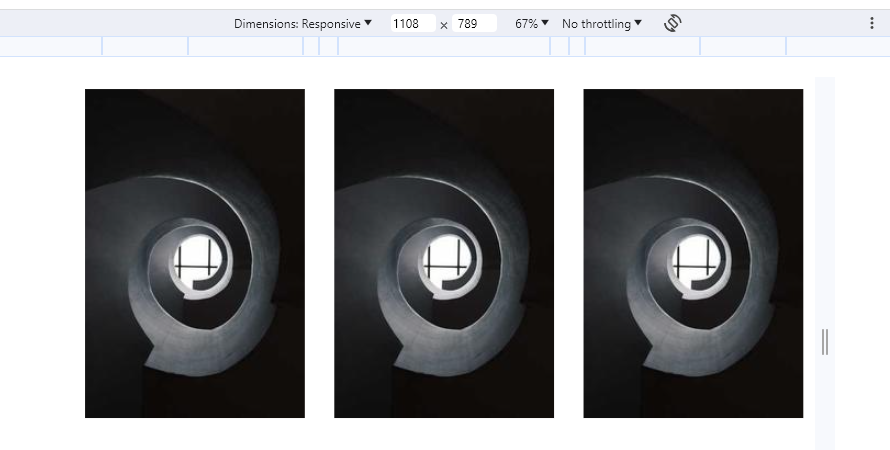
<img src="/Media query Assignment/image.jpg" alt="image 3">

</div>

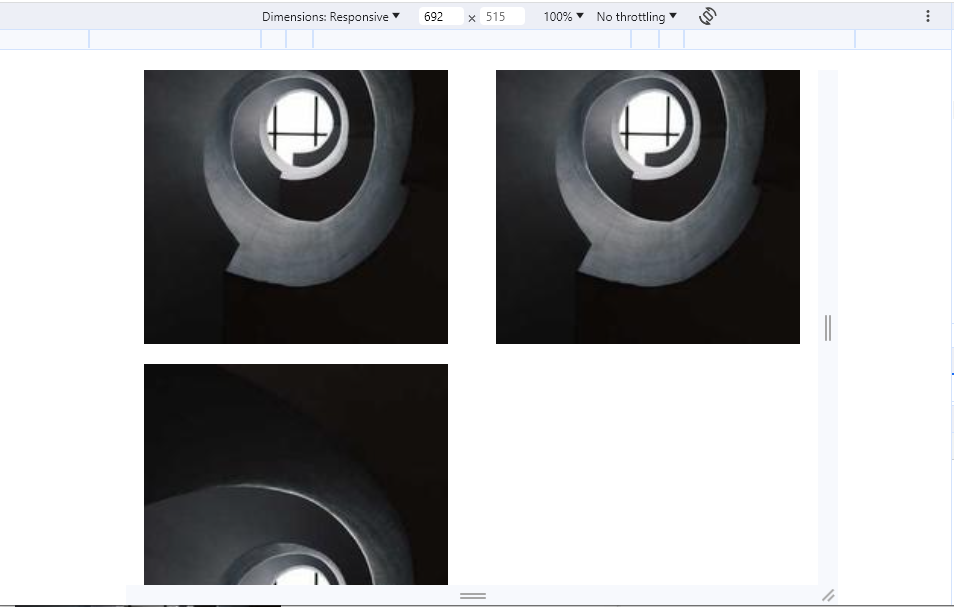
</body>

</html>

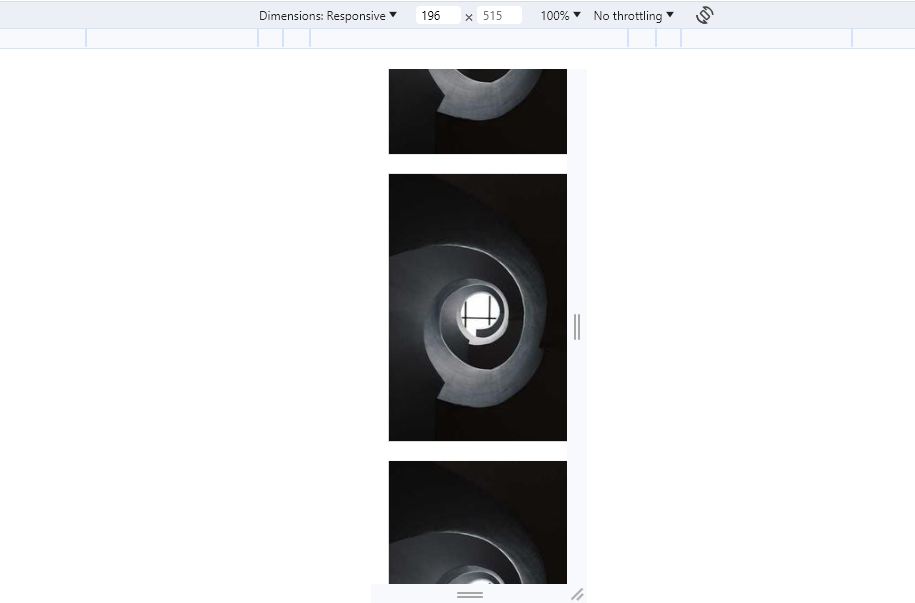
First output:



Second output smaller than 800px:



Third output less than 500px:



1. Create a navigation bar with five links. Apply a media query that converts the navigation into a vertical list for screens smaller than 768px and also changes the background color to gray.

Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Navbar</title>

<style>

.container{

display: flex;

justify-content: space-around;

background-color: black;

}

.container a{

text-decoration: none;

padding: 15px;

color: white;

}

@media screen and (max-width: 786px) {

.container{

flex-direction: column;

align-items: center;

background-color: gray;

}

}

</style>

</head>

<body>

<nav class="container">

<a href="">Home</a>

<a href="">About</a>

<a href="">Services</a>

<a href="">Portfolio</a>

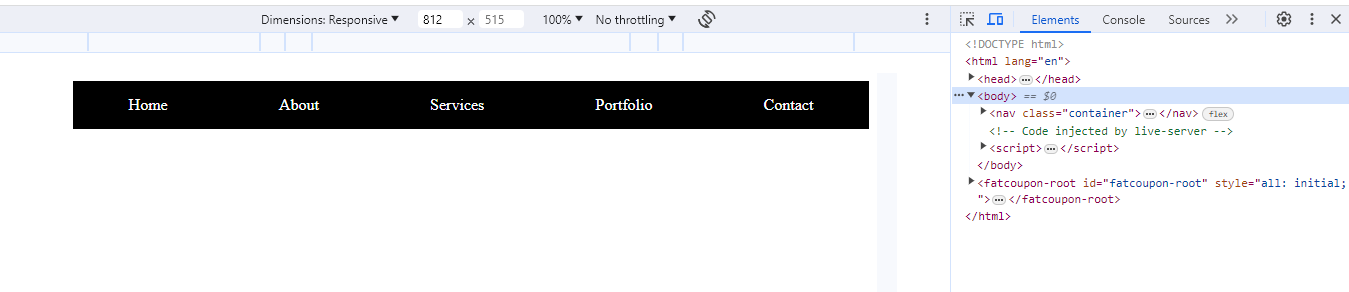
<a href="">Contact</a>

</nav>

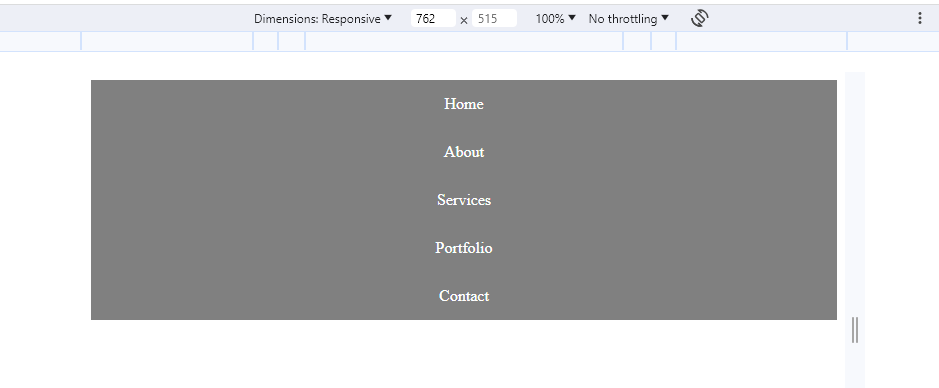
</body>

</html>

First output:



Less than 786px;



1. Problem Statement

Create a grid based layout containing four grid items which arranges them as the screen size changes.

Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Document</title>

<style>

body {

margin: 0;

font-family: Arial, sans-serif;

}

.grid-container {

display: grid;

grid-template-columns: 1fr;

gap: 20px;

padding: 20px;

background-color: #f5f5f5;

min-height: 100vh;

}

.box {

background-color: #ddd;

padding: 20px;

text-align: center;

}

.box1 {

background-color: red;

color: white;

}

.box2 {

background-color: green;

color: white;

}

.box3 {

background-color: blue;

color: white;

}

.box4 {

background-color: chocolate;

color: white;

}

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

.grid-container {

grid-template-columns: repeat(3, 1fr);

grid-template-areas:

"box1 box1 box1"

"box2 box3 box3"

"box4 box4 box4";

}

}

</style>

</head>

<body>

<div class="grid-container">

<div class="box box1">Box 1</div>

<div class="box box2">Box 2</div>

<div class="box box3">Box 3</div>

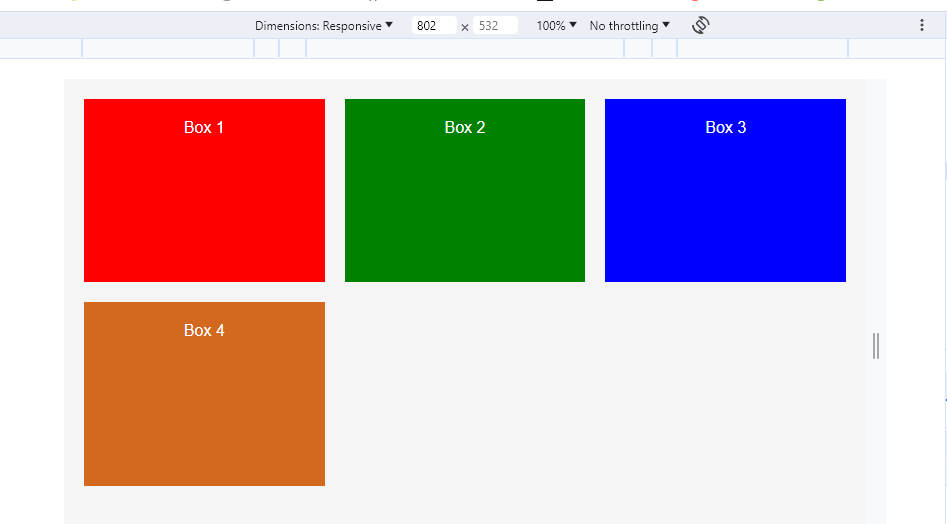
<div class="box box4">Box 4</div>

</div>

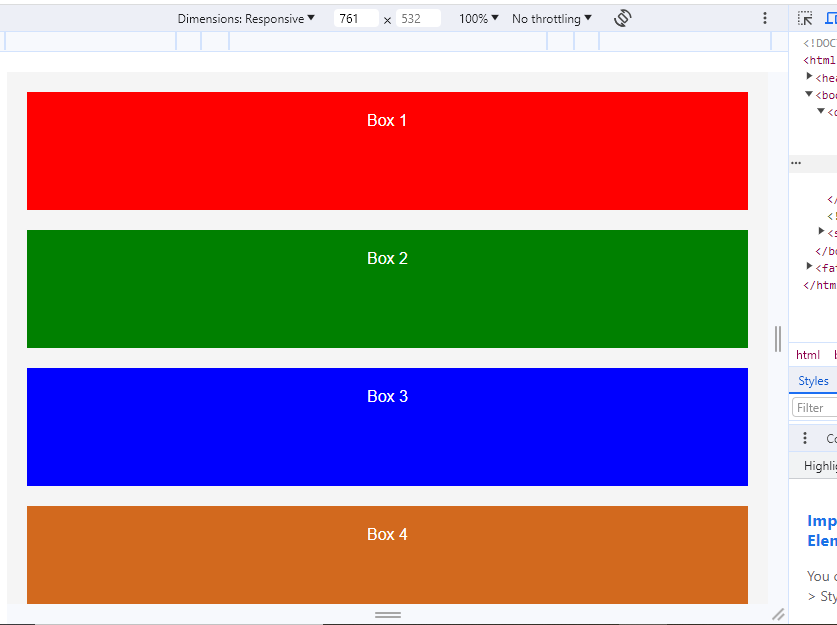
</body>

</html>

Output:



Responsive Output:



1. Problem Statement

Create the below given layout using the flexbox in css, which should adapt itself on mobile screen as per given below output.

Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Document</title>

<style>

body{

margin: 0;

font-family: Arial, sans-serif;

}

.container {

display: flex;

flex-direction: column;

min-height: 100vh;

background-color: #f5f5f5;

}

.box1,

.box4 {

background-color: #333;

color: white;

padding: 20px;

text-align: center;

}

.main {

display: flex;

flex-direction: column;

flex: 1;

}

.box2,

.box3 {

background-color: #fae8e8;

padding: 20px;

text-align: center;

}

.box2 {

background-color: #66cc99;

color: white;

}

.box3{

background-color: orangered;

color: white;

}

@media screen and (min-width:786px) {

.container{

flex-direction: row;

}

.box2,

.box3{

flex: 1;

}

}

</style>

</head>

<body>

<div class="container">

<div class="box1">Box 1</div>

<div class="main">

<div class="box2">Box 2</div>

<div class="box3">Box 3</div>

</div>

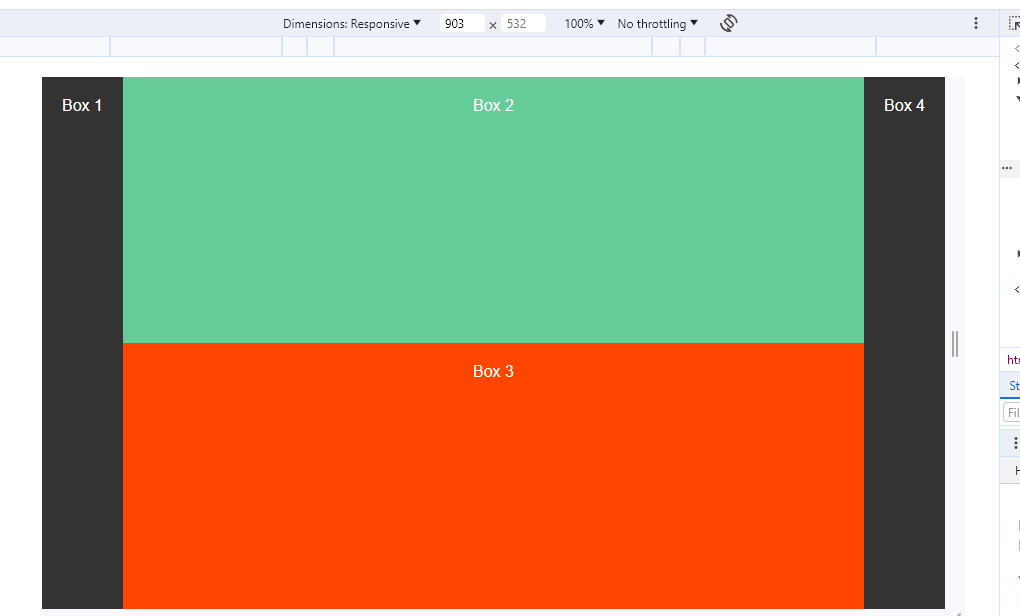
<div class="box4">Box 4</div>

</div>

</body>

</html>

Output:



Responsive Output:

