LISTADOS DE CSS GRID

**LISTADO1**

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<style>

.grid-container {

display: grid;

grid-template-columns: 1fr 2fr 1fr;

grid-template-rows: 100px auto 100px;

gap: 10px;

}

.header {

grid-column: 1 / 4;

background-color: lightblue;

}

.sidebar {

background-color: lightcoral;

}

.content {

background-color: lightgreen;

}

.footer {

grid-column: 1 / 4;

background-color: lightgray;

}

</style>

</head>

<body>

<div class="grid-container">

<div class="header">Header</div>

<div class="sidebar">Sidebar</div>

<div class="content">Content</div>

<div class="footer">Footer</div>

</div>

</body>

</html>

### **Grid básico con columnas y filas (LISTADO 2)**

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<style>

.grid-container {

display: grid;

grid-template-columns: 100px 200px 100px; /\* Tres columnas de distintos tamaños \*/

grid-template-rows: 100px 100px; /\* Dos filas de 100px \*/

gap: 10px; /\* Espacio entre las celdas \*/

}

.grid-item {

background-color: lightblue;

border: 1px solid #333;

text-align: center;

font-size: 20px;

display: flex;

justify-content: center;

align-items: center;

}

</style>

</head>

<body>

<div class="grid-container">

<div class="grid-item">1</div>

<div class="grid-item">2</div>

<div class="grid-item">3</div>

<div class="grid-item">4</div>

<div class="grid-item">5</div>

<div class="grid-item">6</div>

</div>

</body>

</html>

**Grid con elementos que abarcan varias columnas y filas** **(LISTADO 3)**

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<style>

.grid-container {

display: grid;

grid-template-columns: 1fr 1fr 1fr;

grid-template-rows: 100px 100px;

gap: 10px;

}

.grid-item {

background-color: lightcoral;

border: 1px solid #333;

text-align: center;

font-size: 20px;

display: flex;

justify-content: center;

align-items: center;

}

/\* El elemento 1 ocupa dos columnas \*/

.item1 {

grid-column: 1 / 3;

}

/\* El elemento 4 ocupa dos filas \*/

.item4 {

grid-row: 1 / 3;

}

</style>

</head>

<body>

<div class="grid-container">

<div class="grid-item item1">1</div>

<div class="grid-item">2</div>

<div class="grid-item">3</div>

<div class="grid-item item4">4</div>

<div class="grid-item">5</div>

</div>

</body>

</html>

Grid responsivo con repeat() y minmax() (**LISTADO** **4**)

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<style>

.grid-container {

display: grid;

grid-template-columns: repeat(auto-fill, minmax(150px, 1fr));

gap: 10px;

}

.grid-item {

background-color: lightgreen;

border: 1px solid #333;

text-align: center;

font-size: 20px;

padding: 20px;

}

</style>

</head>

<body>

<div class="grid-container">

<div class="grid-item">1</div>

<div class="grid-item">2</div>

<div class="grid-item">3</div>

<div class="grid-item">4</div>

<div class="grid-item">5</div>

<div class="grid-item">6</div>

<div class="grid-item">7</div>

<div class="grid-item">8</div>

</div>

</body>

</html>

Alineación de elementos en CSS Grid (**LISTADO 5**)

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<style>

.grid-container {

display: grid;

grid-template-columns: 200px 200px;

grid-template-rows: 200px 200px;

gap: 10px;

justify-items: center; /\* Alinea los elementos horizontalmente \*/

align-items: center; /\* Alinea los elementos verticalmente \*/

}

.grid-item {

background-color: lightyellow;

border: 1px solid #333;

text-align: center;

font-size: 20px;

padding: 20px;

}

</style>

</head>

<body>

<div class="grid-container">

<div class="grid-item">1</div>

<div class="grid-item">2</div>

<div class="grid-item">3</div>

<div class="grid-item">4</div>

</div>

</body>

</html>

Grid con nombres de áreas (grid-template-areas) (**LISTADO 6)**

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<style>

.grid-container {

display: grid;

grid-template-areas:

"header header header"

"nav content content"

"footer footer footer";

grid-template-columns: 150px 1fr 1fr;

grid-template-rows: 100px 1fr 100px;

gap: 10px;

}

.header {

grid-area: header;

background-color: lightblue;

}

.nav {

grid-area: nav;

background-color: lightgreen;

}

.content {

grid-area: content;

background-color: lightcoral;

}

.footer {

grid-area: footer;

background-color: lightyellow;

}

.grid-item {

border: 1px solid #333;

text-align: center;

font-size: 20px;

display: flex;

justify-content: center;

align-items: center;

}

</style>

</head>

<body>

<div class="grid-container">

<div class="grid-item header">Header</div>

<div class="grid-item nav">Nav</div>

<div class="grid-item content">Content</div>

<div class="grid-item footer">Footer</div>

</div>

</body>

</html>

**Control preciso con** grid-auto-flow (**LISTADO7)**

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<style>

.grid-container {

display: grid;

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

grid-auto-flow: dense; /\* Intenta llenar los huecos de manera eficiente \*/

gap: 10px;

}

.grid-item {

background-color: lightgray;

border: 1px solid #333;

text-align: center;

font-size: 20px;

padding: 20px;

}

.item1 {

grid-column: span 2; /\* Este elemento ocupará dos columnas \*/

}

</style>

</head>

<body>

<div class="grid-container">

<div class="grid-item item1">1</div>

<div class="grid-item">2</div>

<div class="grid-item">3</div>

<div class="grid-item">4</div>

<div class="grid-item">5</div>

</div>

</body>

</html>

MEZCLAR CSS GRID CON BOOTSTRAP 5.3 **(LISTADO 8)**

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">

<style>

.custom-grid {

display: grid;

grid-template-columns: 1fr 2fr 1fr; /\* Tres columnas con diferentes tamaños \*/

gap: 10px;

}

.grid-item {

background-color: lightgray;

border: 1px solid #333;

text-align: center;

font-size: 20px;

padding: 20px;

}

</style>

<title>Mezcla de Bootstrap y CSS Grid</title>

</head>

<body>

<!-- Usando el grid de Bootstrap para el layout principal -->

<div class="container mt-5">

<div class="row">

<div class="col-lg-4">

<div class="p-3 bg-primary text-white">Columna 1 (Bootstrap)</div>

</div>

<div class="col-lg-8">

<!-- Aquí usamos CSS Grid dentro de una columna de Bootstrap -->

<div class="custom-grid">

<div class="grid-item">1 (CSS Grid)</div>

<div class="grid-item">2 (CSS Grid)</div>

<div class="grid-item">3 (CSS Grid)</div>

</div>

</div>

</div>

</div>

<!-- Bootstrap JS (opcional) -->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>

</body>

</html>

OTRO EJEMPLO MEZCLAR CSS GRID CON BOOTSTRAP 5.3 **(LISTADO 9)**

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">

<style>

/\* CSS Grid aplicado solo a la galería \*/

.gallery {

display: grid;

grid-template-columns: repeat(auto-fit, minmax(150px, 1fr)); /\* Responsive grid \*/

gap: 15px; /\* Espacio entre las imágenes \*/

}

.gallery-item {

background-color: lightgray;

border: 1px solid #ccc;

padding: 10px;

text-align: center;

}

.gallery-item img {

width: 100%;

height: auto;

object-fit: cover;

}

</style>

<title>Combinación de Bootstrap y CSS Grid</title>

</head>

<body>

<!-- Contenedor principal de Bootstrap -->

<div class="container mt-5">

<div class="row">

<!-- Columna izquierda usando Bootstrap -->

<div class="col-lg-4">

<div class="p-3 bg-primary text-white">Menú lateral (Bootstrap)</div>

<p class="mt-3">Aquí puedes añadir el contenido de un menú o cualquier otro componente de Bootstrap.</p>

</div>

<!-- Columna derecha con Bootstrap y CSS Grid -->

<div class="col-lg-8">

<div class="p-3 bg-secondary text-white">Galería (Bootstrap + CSS Grid)</div>

<!-- Galería con CSS Grid dentro de Bootstrap -->

<div class="gallery mt-3">

<div class="gallery-item">

<img src="https://via.placeholder.com/150" alt="Imagen 1">

<p>Imagen 1</p>

</div>

<div class="gallery-item">

<img src="https://via.placeholder.com/150" alt="Imagen 2">

<p>Imagen 2</p>

</div>

<div class="gallery-item">

<img src="https://via.placeholder.com/150" alt="Imagen 3">

<p>Imagen 3</p>

</div>

<div class="gallery-item">

<img src="https://via.placeholder.com/150" alt="Imagen 4">

<p>Imagen 4</p>

</div>

<div class="gallery-item">

<img src="https://via.placeholder.com/150" alt="Imagen 5">

<p>Imagen 5</p>

</div>

<div class="gallery-item">

<img src="https://via.placeholder.com/150" alt="Imagen 6">

<p>Imagen 6</p>

</div>

</div>

</div>

</div>

</div>

<!-- Bootstrap JS (opcional) -->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>

</body>

</html>

**(LISTADO 10)**

@model YourProjectNamespace.Models.UserModel

@{

ViewBag.Title = "Register";

}

<h2>Register</h2>

@using (Html.BeginForm())

{

@Html.ValidationSummary(true, "", new { @class = "text-danger" })

<div class="grid-form">

<div class="form-group">

@Html.LabelFor(m => m.Nombre)

@Html.TextBoxFor(m => m.Nombre, new { @class = "form-control" })

@Html.ValidationMessageFor(m => m.Nombre, "", new { @class = "text-danger" })

</div>

<div class="form-group">

@Html.LabelFor(m => m.CorreoElectronico)

@Html.TextBoxFor(m => m.CorreoElectronico, new { @class = "form-control" })

@Html.ValidationMessageFor(m => m.CorreoElectronico, "", new { @class = "text-danger" })

</div>

<div class="form-group">

@Html.LabelFor(m => m.Contraseña)

@Html.PasswordFor(m => m.Contraseña, new { @class = "form-control" })

@Html.ValidationMessageFor(m => m.Contraseña, "", new { @class = "text-danger" })

</div>

<div class="form-group">

@Html.LabelFor(m => m.Telefono)

@Html.TextBoxFor(m => m.Telefono, new { @class = "form-control" })

@Html.ValidationMessageFor(m => m.Telefono, "", new { @class = "text-danger" })

</div>

<div class="form-group">

@Html.LabelFor(m => m.FechaNacimiento)

@Html.TextBoxFor(m => m.FechaNacimiento, new { @class = "form-control", type = "date" })

</div>

<div class="form-group">

<button type="submit" class="btn btn-primary">Enviar</button>

</div>

</div>

}

<style>

.grid-form {

display: grid;

grid-template-columns: 1fr 1fr;

grid-gap: 20px;

}

.form-group {

display: flex;

flex-direction: column;

}

.form-control {

padding: 10px;

font-size: 16px;

}

.btn {

padding: 10px 20px;

font-size: 16px;

cursor: pointer;

}

.text-danger {

color: red;

}

</style>

**LISTADO 11**

/\* Resetear estilos predeterminados \*/

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

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

height: 100vh;

}

.grid-container {

display: grid;

grid-template-columns: 200px 1fr 200px;

grid-template-rows: auto 1fr auto;

grid-template-areas:

"header header header"

"menu main-content advertisement"

"footer footer footer";

gap: 10px;

padding: 20px;

height: 100vh;

}

.header {

grid-area: header;

background-color: #4CAF50;

display: flex;

justify-content: center;

align-items: center;

font-size: 2rem;

color: white;

}

.menu {

grid-area: menu;

background-color: #FF5722;

padding: 20px;

color: white;

}

.main-content {

grid-area: main-content;

background-color: #3F51B5;

display: grid;

gap: 10px;

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

grid-auto-flow: dense; /\* Llenado de celdas vacías en orden denso \*/

padding: 20px;

color: white;

}

/\* Usamos subgrid en las secciones del contenido principal \*/

.content-grid {

display: grid;

grid-template-columns: subgrid;

grid-template-rows: subgrid;

gap: 10px;

}

.section {

padding: 20px;

background-color: rgba(255, 255, 255, 0.2);

text-align: center;

border-radius: 8px;

}

/\* Diferentes tamaños para las secciones \*/

.section1 {

grid-column: span 2;

}

.section2 {

grid-row: span 2;

}

.advertisement {

grid-area: advertisement;

background-color: #FFC107;

display: flex;

justify-content: center;

align-items: center;

font-size: 1.5rem;

color: black;

}

.footer {

grid-area: footer;

background-color: #607D8B;

display: flex;

justify-content: center;

align-items: center;

color: white;

font-size: 1.2rem;

}

/\* Alineación avanzada con CSS Grid \*/

.grid-container > div {

border-radius: 8px;

padding: 20px;

color: white;

font-weight: bold;

display: flex;

justify-content: center;

align-items: center;

}

/\* Media Query para pantallas más pequeñas \*/

@media (max-width: 768px) {

.grid-container {

grid-template-columns: 1fr;

grid-template-rows: auto;

grid-template-areas:

"header"

"main-content"

"menu"

"advertisement"

"footer";

}

.main-content {

grid-template-columns: 1fr;

}

}

**MINMAX(): LISTADO 12**

/\* Estilos generales \*/

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

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

height: 100vh;

padding: 20px;

background-color: #f5f5f5;

}

/\* Contenedor grid con minmax() y diferentes tamaños \*/

.grid-container {

display: grid;

grid-template-columns: repeat(2, minmax(150px, 1fr)) repeat(2, minmax(300px, 2fr));

grid-template-rows: repeat(2, minmax(150px, 1fr)) repeat(2, minmax(200px, 1fr));

gap: 15px;

padding: 10px;

}

.item {

background-color: #3F51B5;

color: white;

display: flex;

justify-content: center;

align-items: center;

font-size: 1.2rem;

border-radius: 8px;

text-align: center;

padding: 10px;

}

/\* Diferentes tamaños para algunos elementos \*/

.item1 {

grid-column: span 2;

grid-row: span 2;

}

.item2 {

grid-column: span 1;

grid-row: span 1;

}

.item4 {

grid-column: span 2;

grid-row: span 1;

}

/\* Responsividad \*/

@media (max-width: 768px) {

.grid-container {

grid-template-columns: repeat(auto-fill, minmax(150px, 1fr));

grid-template-rows: auto;

}

}

**GRID ANINADO (LISADO 13)**

/\* Estilos generales \*/

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

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

height: 100vh;

background-color: #f5f5f5;

display: flex;

justify-content: center;

align-items: center;

}

/\* Contenedor principal con grid \*/

.grid-container {

display: grid;

grid-template-columns: 1fr 3fr;

grid-template-rows: auto 1fr auto;

gap: 20px;

width: 90%;

max-width: 1200px;

height: 90vh;

}

/\* Estilos para los ítems del grid principal \*/

.header {

grid-column: 1 / -1;

background-color: #4CAF50;

color: white;

display: flex;

justify-content: center;

align-items: center;

padding: 20px;

font-size: 2rem;

}

.sidebar {

background-color: #607D8B;

color: white;

display: flex;

justify-content: center;

align-items: center;

padding: 20px;

font-size: 1.5rem;

}

.main-content {

display: grid;

grid-template-columns: repeat(2, 1fr);

grid-template-rows: repeat(2, 200px);

gap: 20px;

background-color: #f5f5f5;

padding: 20px;

}

.footer {

grid-column: 1 / -1;

background-color: #3F51B5;

color: white;

display: flex;

justify-content: center;

align-items: center;

padding: 20px;

font-size: 1.2rem;

}

/\* Estilos para los ítems anidados \*/

.nested-item {

background-color: #E91E63;

color: white;

display: flex;

justify-content: center;

align-items: center;

font-size: 1.2rem;

border-radius: 8px;

}

Layout Dinámico con grid-template-areas **(LISTADO 14)**

/\* Estilos generales \*/

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

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

background-color: #f5f5f5;

padding: 20px;

display: flex;

justify-content: center;

align-items: center;

}

/\* Definición del grid principal \*/

.grid-container {

display: grid;

grid-template-columns: 1fr 2fr;

grid-template-rows: auto 1fr auto;

grid-template-areas:

"header header"

"sidebar content"

"footer footer";

gap: 20px;

width: 90%;

max-width: 1200px;

background-color: white;

padding: 20px;

border-radius: 10px;

}

/\* Estilos para las áreas del grid \*/

.header {

grid-area: header;

background-color: #4CAF50;

color: white;

display: flex;

justify-content: center;

align-items: center;

padding: 20px;

font-size: 2rem;

}

.sidebar {

grid-area: sidebar;

background-color: #607D8B;

color: white;

display: flex;

justify-content: center;

align-items: center;

padding: 20px;

font-size: 1.5rem;

}

.content {

grid-area: content;

background-color: #E91E63;

color: white;

display: flex;

justify-content: center;

align-items: center;

padding: 20px;

font-size: 1.5rem;

}

.extra {

grid-column: 2 / 3;

grid-row: 2 / 4;

background-color: #FF9800;

color: white;

display: flex;

justify-content: center;

align-items: center;

font-size: 1.5rem;

padding: 20px;

border-radius: 8px;

}

.footer {

grid-area: footer;

background-color: #3F51B5;

color: white;

display: flex;

justify-content: center;

align-items: center;

padding: 20px;

font-size: 1.2rem;

}

/\* Responsividad \*/

@media (max-width: 768px) {

.grid-container {

grid-template-columns: 1fr;

grid-template-areas:

"header"

"sidebar"

"content"

"footer";

}

.extra {

grid-column: 1 / 2;

grid-row: auto;

}

}

Layout Dinámico con grid-column y grid-row**(LISTADO 15)**

/\* Estilos generales \*/

body {

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

background-color: #f5f5f5;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

}

/\* Grid dinámico \*/

.dynamic-grid {

display: grid;

grid-template-columns: repeat(6, 1fr);

grid-template-rows: 200px 150px 150px;

gap: 10px;

width: 90%;

max-width: 1200px;

}

/\* Estilos para los ítems \*/

.item {

background-color: #3F51B5;

color: white;

display: flex;

justify-content: center;

align-items: center;

font-size: 1.5rem;

padding: 20px;

border-radius: 8px;

}

/\* Control dinámico de grid-column y grid-row \*/

.item1 {

grid-column: 1 / 3;

grid-row: 1 / 2;

}

.item2 {

grid-column: 3 / 6;

grid-row: 1 / 2;

}

.item3 {

grid-column: 1 / 4;

grid-row: 2 / 3;

}

.item4 {

grid-column: 4 / 7;

grid-row: 2 / 4;

}

.item5 {

grid-column: 1 / 3;

grid-row: 3 / 4;

}

.item6 {

grid-column: 3 / 4;

grid-row: 3 / 4;

}

AUTO-FILL **(LISTADO16)**

/\* Estilos generales \*/

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

/\* Grid container \*/

.grid-container {

display: grid;

grid-template-columns: repeat(auto-fill, minmax(150px, 1fr));

gap: 20px;

width: 90%;

max-width: 1200px;

}

/\* Estilos para los ítems \*/

.item {

background-color: #4CAF50;

color: white;

display: flex;

justify-content: center;

align-items: center;

font-size: 1.5rem;

height: 150px;

border-radius: 8px;

}

AUTO-FIT **(LISTADO17)**

/\* Estilos generales \*/

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

/\* Grid container con auto-fit \*/

.grid-container {

display: grid;

grid-template-columns: repeat(auto-fit, minmax(150px, 1fr));

gap: 20px;

width: 90%;

max-width: 1200px;

}

/\* Estilos para los ítems \*/

.item {

background-color: #4CAF50;

color: white;

display: flex;

justify-content: center;

align-items: center;

font-size: 1.5rem;

height: 150px;

border-radius: 8px;

}

**ALINEACION AVANZADA DE ITEMS INDIVIDUALES(LISTADO18)**

/\* Estilos generales \*/

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

/\* Contenedor del grid \*/

.grid-container {

display: grid;

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

grid-template-rows: 200px 200px;

gap: 20px;

width: 90%;

max-width: 1200px;

}

/\* Estilos para los ítems \*/

.item {

background-color: #3F51B5;

color: white;

display: flex;

justify-content: center;

align-items: center;

font-size: 1.5rem;

border-radius: 8px;

}

/\* Alineación avanzada individual \*/

.item1 {

align-self: start; /\* Alineación superior (vertical) \*/

justify-self: start; /\* Alineación izquierda (horizontal) \*/

}

.item2 {

align-self: end; /\* Alineación inferior (vertical) \*/

justify-self: end; /\* Alineación derecha (horizontal) \*/

}

.item3 {

align-self: center; /\* Alineación centrada (vertical) \*/

justify-self: center; /\* Alineación centrada (horizontal) \*/

}

.item4 {

align-self: stretch; /\* Alineación para llenar el área en el eje vertical \*/

justify-self: stretch; /\* Alineación para llenar el área en el eje horizontal \*/

}

.item5 {

align-self: end; /\* Alineación inferior (vertical) \*/

justify-self: center; /\* Alineación centrada (horizontal) \*/

}

.item6 {

align-self: start; /\* Alineación superior (vertical) \*/

justify-self: end; /\* Alineación derecha (horizontal) \*/

}