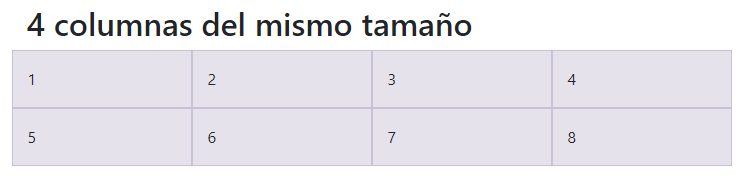
# TAREA 4: GRID

Ejercicio grid de bootstrap. Modifica el fichero 01-grid.html añadiendo las clases correspondientes para que se muestre como indican en los títulos.

1. 4 columnas del mismo tamaño:

Texto

Descripción generada automáticamente



Tenemos que definir los div hijos como clase col, de esta manera se generan 2 filas de 4 columas gracias a ese 5º elemento div con la clase w-100.

1. 4 columnas de un tamaño de 3:

Texto

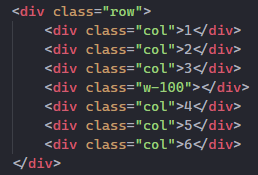
Descripción generada automáticamente

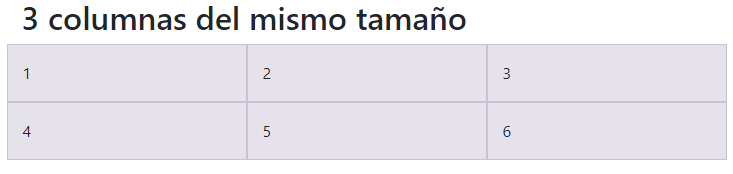
Tabla

Descripción generada automáticamente con confianza media

En este caso debemos fijar cada columna a un tamaño de 3 utilizando la clase col-3 que otorga 3 casillas (de las 12 existentes) a cada columna.

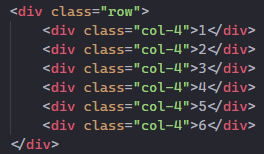
1. 3 columnas del mismo tamaño:

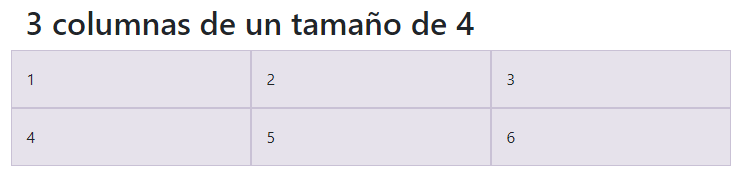




Tenemos que definir los div hijos como clase col, de esta manera se generan 2 filas de 3 columnas gracias a ese 4º elemento div con la clase w-100.

1. 3 columnas de un tamaño de 4:





En este caso debemos fijar cada columna a un tamaño de 4 utilizando la clase col-4 que otorga 4 casillas (de las 12 existentes) a cada columna.

1. 2 columnas del mismo tamaño:

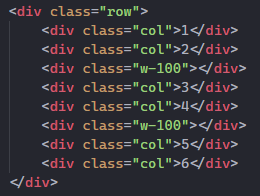


Gráfico de dispersión

Descripción generada automáticamente con confianza baja

Tenemos que definir los div hijos como clase col, de esta manera se generan 3 filas de 2 columnas gracias a esos 3º y 5º elementos div con la clase w-100.

1. 2 columnas de un tamaño de 6:

Texto

Descripción generada automáticamente

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

En este caso debemos fijar cada columna a un tamaño de 6 utilizando la clase col-6 que otorga 6 casillas (de las 12 existentes) a cada columna.

1. 1 columna de un tamaño de 12:

Texto

Descripción generada automáticamente

Tabla

Descripción generada automáticamente con confianza media

En este caso debemos fijar cada columna a un tamaño de 12 utilizando la clase col-12 que otorga 12 casillas (de las 12 existentes) a cada columna.

1. Fijando el tamaño de la columna central:

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente

En este debemos fijar el tamaño de la columna central, para ello debemos hacer que esa sea la única columna que tenga un valor fijo (col-10 en el primer caso, col-8 en el segundo caso, col-6 en el tercero y col-4 en el último) el resto de columnas se adaptan al espacio disponible con la clase col.

1. Ajustando el tamaño al contenido:

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Para conseguir esto debemos aplicar la clase col-auto en la columna que debe ajustar su tamaño al contenido. Las columnas de tamaño 2 reciben una clase col-2 y la que tiene un tamaño que se expande debe recibir la clase col para que su ancho se ajuste en todo momento al espacio disponible.

1. Multidispositivo normal:

Texto

Descripción generada automáticamente

En este caso he decido hacer lo siguiente:

1. En dispositivos de pantalla extragrande y grande cada columna recibe 2 espacios (col-xl-2 y col-lg-2):

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

1. En dispositivos de pantalla mediana cada columna recibe 4 espacios (col-md-4):

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

1. En dispositivos de pantalla pequeña cada columna recibe 6 espacios (col-sm-6):

Tabla

Descripción generada automáticamente

1. En el resto de los dispositivos (que serían los dispositivos de pantalla extrapequeña) cada columna recibe 12 espacios (col-12):

Imagen que contiene Texto

Descripción generada automáticamente

1. Alineación vertical:

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación, Tabla, Word

Descripción generada automáticamente

En las 3 primeras filas se debe usar la clase align-items-X donde X es start en el primer caso para alinear al inicio del contenedor a todos los elementos, center en el segundo caso para alinear al centro y end en el tercero para alinear los elementos al final del contenedor.

Por otro lado, la cuarta fila debe tener un alineamiento propio para cada columna por lo que hay que usar la clase align-self-X donde los valores start, center y end cumplen con lo mismo que en el anterior caso.

1. Alineación horizontal:

Texto

Descripción generada automáticamente

Gráfico

Descripción generada automáticamente

En este caso se debe hacer un alineamiento horizontal, es decir, una justificación. Para ello hay que usar la clase justify-content-X donde X es start si queremos alinear al inicio (fila 1), end si queremos alinear al final (fila 2), center si queremos alinear en el centro de la fila (fila 3), around si queremos generar un espaciado alrededor de los elementos (fila 4) o between si queremos que el espacio restante exista entre los elementos y los elementos de los extremos se vayan a inicio y fin de su contenedor (fila 5).

1. Huecos con offset:

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

En la primera fila se utilizan las clases col-md-4 y ml-auto con ello se consigue que la primera columna se coloque al inicio de la fila (col-md-4), mientras que la segunda columna se coloca al final de la fila (ml-auto), dejando un espacio entre ambas columnas.

En la segunda fila se utilizan las clases col-md-3 y ml-md-auto con esto se consigue una situación como la anterior, pero en este caso con un alineamiento hacia la derecha.

En la última fila, para colocar cada columna a un extremo del contenedor se deben usar las clases col-auto y mr-auto en la columna de la izquierda. Por otra parte en la segunda columna se debe utilizar la clase col-auto que coloca a la columna al final del contenedor.