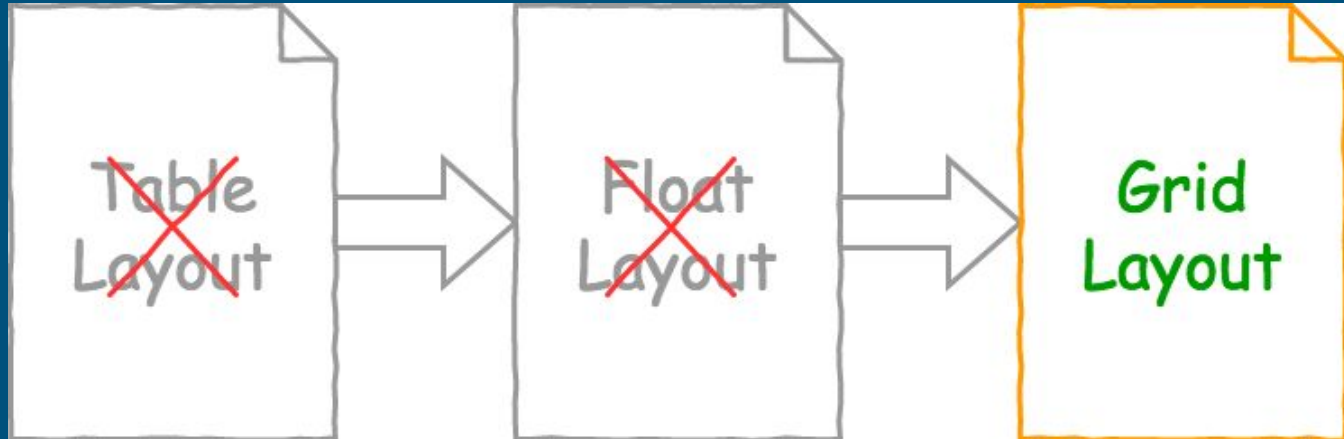[▶] [Learn CSS Grid in 20 Minutes](#)

[▶] [CSS Grid Layout Crash Course](#)

1

CSS Grid *Tutorials and Examples*

- Tutorial: CSS Grid Layout Module ✔
- Tutorial: A Complete Guide to Grid ✔
- A lot of grid examples in Grid by Example
- Altres fonts:
    - Tutorial: A Complete Guide to CSS Grid Layout
    - CSS Grid Layout Module (W3C) - Work in Progress

Design of layouts has evolved from using <u>HTML tables</u> to CSS <u>floated </u>elements, and finally to <u>CSS Grid</u>

CSS Grid _What is it?_

- **CSS Grid Layout** creates a grid where we can put our containers

- Tables, floats, positioning and inline-block were essentially _hacks_ that left out a lot of important functionality

- **Flexbox** helped out, but it's intended for simpler one-dimensional layouts

- **Grid** is the very first CSS module created specifically to solve layout problems

4

CSS Grid   template-areas

- ○ **grid-template-areas**: Establim un grid amb els noms de les seccions

    "header header header"

    "main main aside"

    "main main aside"

    " footer footer footer " ;

- ○ **grid-area**: El nom de cada secció s'aplica a un element amb CSS

CSS Grid   Introduction

- ## The container

  - **display**   grid   or   inline-grid

  - **grip-template-columns**   width of every column of the grid (and name of lines)

  - **grip-template-rows**   height of every row of the grid (and name of lines)

  - **grid-column-gap   grid-row-gap**   gap between rows - between columns

- ## The item

  - **grid-column**   columns where the item starts / ends

  - **grid-row**   row where the item starts / ends

CSS Grid   **The container**

Grid combines with **Flexbox Properties** to create layouts.

- ○ **justify-content:**   To horizontally align inside the container. Values:

    space-around   space-evenly   space-between   center   start   end

- ○ **align-content:**   To vertically align inside the container

    space-around   space-evenly   space-between   center   start   end

CSS Grid: Extra contents

- [CSS Grid vs. other layout methods](#)

  - Flex works mainly in 1-dimension

  - Grid is better for 2-dimension layouts

- [CSS Templates](#)

  - Differences between layouts created using Floats, Flexbox and Grid