**Grid**

    display: grid;

**ativa o display grid no container**

    grid-template-columns: 1fr 1fr;

**divide as colunas em frações**

    grid-template-rows: 20vh 70h 10vh;

**divide as linhas da página (20 visão da altura...) nesse exemplo a página terá três linhas imaginárias**

    grid-row-gap: 20px;

**dá a distância entre os itens de um container (grid-colum-gap) para distância em colunas**

    grid-gap: 20px;

**aplica os tamanhos da coluna e das linhas em um só (se colocar só um aplica o mesmo valor aos dois)**

 grid-column-start: 1;

**em que coluna o item vai começar no grid**

grid-column-end: 3;

**em que coluna o item vai terminar**

    grid-column: 1/3;

**atalho do grid: o elemento vai da coluna um à coluna 3**

 grid-template-areas: "h h"

                         "m a"

                         "f f";

**1.1Determina como vão ser as posições dos items**

    grid-area: m;

**1.2utilizar a letra dentro do item**

**Alinhamentos**

**Justify-content e align-content nos permite alinhar o próprio grid, relativo ao espaço fora do grid.**

**7 valores:**

**Start**

**End**

**Center**

**Stretch**

**Space-between**

**Space-around**

**Space-evenly**

**Justify-items e align-items vai permitir alinhar os itens do gird, em qualquer espaço disponível, na célula que ele habitar.**

**4 valores:**

**Start**

**End**

**Center**

**Stretch**

**Justify-self nos permite alinhar o item em si – mesma coisa dos outros dois, mas é aplicado diretamente no item de um grid**