1. 1) Свойство text-align (используется, но реже) · 2) Свойство margin: auto (актуальное) · 3) Свойство position: absolute (устаревшее)

2. **Фракция** — единица доступного пространства, которое можно разделить на сколько угодно частей. Когда мы задаём столбцу размер 1fr, он смотрит, есть ли вокруг другие фракции. Если других фракций нет, забирает всё свободное пространство. Если другие фракции есть, смотрит, сколько места они занимают и сколько пространства осталось ему.

3. Grid-template-columns: repeat (5, 20%);

Grid-template-columns: 20% 20% 20% 20% 20%;

4. Auto-fit- Пытается поместить наибольшее количество повторяющихся элементов в колонку, сколько их может уместиться без переполнения. Когда места для добавления другого элемента не хватает, следующий элемент будет размещён на следующей строке, а это место, которое он не смог заполнить, останется пустым.

Auto-fill Ведёт себя почти какauto-fit, за исключением того, что любая пустая область будет схлопываться и растягивать элементы в этой строке – напоминая поведение флексбоксов, когда по мере уменьшения доступного пространства колонки схлопываются.

5. Display: grid;

Grid-template-columns: 100px 30% auto;

Grid-template-rows: 200px 100px auto;

6. Justify content: space-between;

7. Свойство grid-area определяет размер и расположение элемента сетки в макете сетки и является сокращенным свойством для следующих свойств:

• **grid-row-start**

**• grid-column-start**

**• grid-row-end**

**• grid-column-end**

HAПример с указанием на именованную область: блок item1 займёт область content внутри грид-сетки.

.item1 {

grid-area: content;

}

8. Свойство **grid-template-areas** определяет шаблон сетки ссылаясь на имена областей, которые заданы с помощью свойства grid-area.

Повторение названия области приводит к тому, что содержимое охватывает эти ячейки. Точка означает пустую ячейку. Сам синтаксис предоставляет визуализацию структуры сетки.

grid-template-areas:

"a a a"

"b c c"

"b c c";

9. С помощью свойства auto fill

10. В Grid Layout мы можем дать наименование каждой линии грида, присвоив ей какое-либо имя в квадратных скобках и затем, используя это имя, позиционировать элементы. Например, определим следующую веб-страницу. При именовании линий их имена заключаются в квадратные скобки, а между для именами указывается ширина столбца или высота строки, которые находятся между этими линиями. Затем, используя эти названия, мы можем позиционировать элементы между определенными линиями.

11. Grid-template column: repeat (12, 1fr);

12. Иногда требуется позиционировать элемент не относительно родителя, а относительно окна браузера вне зависимости от вложенности. Для решения подобной задачи подходит position: fixed. Свойство так же, как и предыдущее, работает с указанием смещения left, right, top, bottom или inset.

13. Большие блоки текста лучше обернуть в div и работать с этими блоками, т.е блочная верстка.