1. Единица измерения для распределения пространства fr позволяет делить всю длину или ширину элемента на равные пропорциональные части, затем, указав, сколько таких частей нужно взять для определенного элемента мы можем задать этому элементу размер. Размер элемента будет складываться из сумм длин fr частей.
2. 2.1. Grid-template-columns: 20% 20% 20% 20% 20%

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

2.3. Grid-template-columns: 20% 1fr 20% 1fr 20%

1. Auto-fill и Auto-fit с указанием определенной длины элемента позволяют заменить явное указание количества элементов в строке, т.е. с помощью функции repeat() они оба будут повторять элемент столько раз, сколько уместится в строку. Если же добавить минимальное и максимальное значение элемента, то они будут вести себя различно: Auto-fill позволит заполнить строку частями, содержащими контент или грид-элементы, затем он позволить добавление таких же частей в строку при ее растягивании, даже если эти части ничего не содержат и все вместе части будут одинаковые, если не указано другое. Auto-fit при растягивании строки позволяет заполнить строку грид-элементами или элементами с контентом, и если остается пустое место, оно распределяется между этими элементами, позволяя им увеличиваться, тем самым визуально в строке не будет пустого места. При этом части без контента все равно будут добавляться, но не учитываться при разделении пространства.
2. .container {

display: grid;

grid-template-columns: 100px 30% 1fr;

grid-template-rows: 200px 100px;

gap: 10px;

}

1. С помощью свойства грид-контейнера justify-content: space-between;
2. Пространство внутри грид-контейнера, содержащее n-ое количество грид-элементов. Может задаваться границами, указанными в линиях контейнера, либо, указав размер контейнера и задав названия нужным областям имен, создать макет в виде матрицы, где повторение названий областей определяет количество частей контейнера, которые будут занимать эти области.
3. grid-template-areas: "1a 1a 1b 1b"

"2a 2a 2c 2d"

"3a 3a 2e 2e"

1. grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
2. 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. Имена заключаются в квадратные скобки, а между именами указывается ширина столбца или высота строки, которые находятся между этими линиями. С помощью этих названий можно размещать элементы между определенными линиями. Полезным является создание названия какой-либо области, а затем линии-границы этой области могут состоять из нескольких частей *название области-суффикс start/end-номер(если несколько линий имеют одинаковое название*.
3. grid-template-columns: repeat (12, 1fr)