Теория к практическому занятию

Количество и ширина рядов в CSS гридах

Элементы грида можно также разместить по рядам. Для этого предназначено свойство *grid-template-rows*. Это свойство через пробел принимает размеры рядов. В нем можно применять те же значения, которые мы изучали и для столбцов.

```
<div id="parent">
     <div>1</div>
3
      <div>2</div>
     <div>3</div>
 4
     <div>4</div>
 6 </div>
 1 #parent {
     display: grid;
     grid-template-rows: 50px 100px 50px 50px;
     border: 2px solid #696989;
     padding: 10px;
6
     width: 600px;
   #parent > div {
     padding: 10px;
10
      border: 1px solid #696989;
11 }
Результат выполнения кода:
     4
```

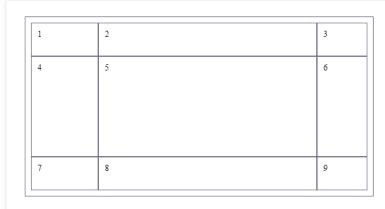
Сокращение для рядов и столбцов в CSS гридах

Бывают ситуации, когда удобно указать размеры рядов и столбцов в краткой форме. Для этого мы используем свойство *grid-template*, которое указывается в элементе-родителе и представляет собой сокращенную форму записи двух свойств *grid-template-rows* и *grid-template-columns*.

Ряды и столбцы для грид-контейнера указываются через слеш, их размеры определяются в единицах измерения.

```
1 <div id="parent">
    <div>1</div>
   3
4
5
7
    <div>7</div>
8
    <div>8</div>
9
10
    <div>9</div>
11 </div>
1 #parent {
    display: grid;
    grid-template: 60px 1fr 60px / 20% 1fr 15%;
border: 2px solid #696989;
3
4
   padding: 10px;
5
6
    width: 600px;
    height: 300px;
7
8 }
   #parent > div {
9
10
    padding: 10px;
11
     border: 1px solid #696989;
12 }
```

Результат выполнения кода:



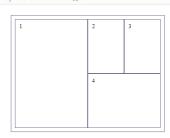
Объединение рядов в CSS гридах

Для того, чтобы элемент занял более одного ряда, ему следует задать свойство *grid-row*. В этом свойстве через слеш задаются числа, указывающее начальную и конечную позицию элементов в гриде.

При этом элемент растянется от первой позиции до второй позиции (не включая ее). То есть значение 1/2 заставит элемент занять одну первую ячейку, а значение 1/3 заставит элемент занять первую и вторую ячейку (но не включая третью).

```
<div id="parent">
                                      3
        1 | #parent {
                                           display: grid;
grid-template-columns: 2fr 1fr;
                                        gradualities and solutions and solutions and solutions and solutions and solutions are solutions and solutions and solutions and solutions are solutions. 211 height: 300px; width: 300px; padding: 10px; padd
          4
  14 grid-row: 1 / 2;
    16 #elem2 {
17 grid-row
     17 grid-row: 2 / 3;
18 }
19 #elem3 {
    20 grid-row: 3 / 4;
21 }
22 melem4 {
    23 grid-row: 1 / 4;
24 }
Результат выполнения кода:
                                           1
                                           3
```

Комбинации объединений по рядам и столбцам в CSS гридах



Расстояние между столбцами и рядами в CSS гридах

С помощью свойства *дар* можно одновременно задать расстояние между столбцами и рядами грида. Можно передать либо одно значение, либо два через пробел. Если передано одно значение, оно задает одновременно расстояние между столбцами и рядами. Если же передано два значения, то первое задает расстояние между рядами, а второе - между столбцами.

```
1 | <div id="parent">
      <div>1</div>
      <div>2</div>
4
      <div>3</div>
      <div>4</div>
6
      <div>5</div>
     <div>6</div>
<div>7</div>
7
 8
 9
      <div>8</div>
      <div>9</div>
10
11 </div>
1 | #parent {
      display: grid;
     gap: 10px;
    grid-template: 1fr 1fr 1fr / 1fr 1fr 1fr;
padding: 10px;
border: 2px solid #696989;
width: 600px;
6
7
8
      height: 200px;
 9 }
10 #parent > div {
11
      padding: 10px;
      border: 1px solid #696989;
12
13 }
Результат выполнения кода:
      1
                             2
                                                     3
                              5
      4
                                                      6
                              8
                                                     9
```