1：等宽队列布局



类似图中这种采用flex布局：

1：给父元素设置flex

2：子元素width：100%即可

3：整个的外边距可以用padding，行与行间距可以用子元素的margin

第一种方法：

<div class="wrapper">

            <div class="row">

                <dl>

                    <dt>86</dt>

                    <dd>原创</dd>

                </dl>

                <dl>

                    <dt>62</dt>

                    <dd>粉丝</dd>

                </dl>

                ...

            </div>

            <div class="row">

                <dl>

                    <dt>86</dt>

                    <dd>原创</dd>

                </dl>

                <dl>

                    <dt>62</dt>

                    <dd>粉丝</dd>

                </dl>

                ...

            </div>

        </div>

.wrapper {

            width: 600px;

            border: 1px solid #000;

        }

        .row{

            display: flex;

        }

        dl {

            width: 100%;

            text-align:center;

        }

该方法需要定义个row，好处是结尾行个数与前面不一致不需要特殊处理，缺点是动态数据渲染的情况下数据结构需要相应处理下。

第二种方法

不需要row，但是需要设定每个元素100%/n 的宽度，n为每行元素的个数，同时父元素要设置flex-wrap

<style>

        \*{

            margin: 0;

            padding: 0;

        }

        .wrapper {

            width: 600px;

            border: 1px solid #000;

            display: flex;

            flex-wrap: wrap;

        }

        .row{

            display: flex;

        }

        dl {

            width: 25%;

            text-align:center;

        }

    </style>

<div class="wrapper">

                <dl>

                    <dt>86</dt>

                    <dd>原创</dd>

                </dl>

                <dl>

                    <dt>62</dt>

                    <dd>粉丝</dd>

                </dl>

                <dl>

                    <dt>86</dt>

                    <dd>原创</dd>

                </dl>

                <dl>

                    <dt>62</dt>

                    <dd>粉丝</dd>

                </dl>

                <dl>

                    <dt>62</dt>

                    <dd>粉丝</dd>

                </dl>

                <dl>

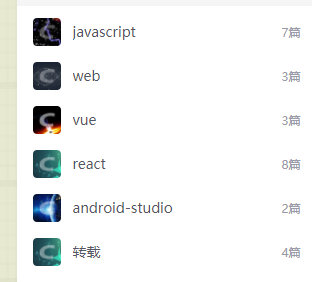
                    <dt>62</dt>

                    <dd>粉丝</dd>

                </dl>

        </div>

2：文字和图标相对居中的布局



1：如果图标是已知固定大小的，则后面包裹文字的父元素（inline-block）设置line-height值等于图标高度即可

2：如果图标是自适应大小，则将图标套一层div，然后在最外层父元素设置flex垂直居中

.wrapper{

            display: flex;

            align-items: center;

        }

        .imgw{

        }

<div class="wrapper">

            <div class="imgw">

                <img src="./app icon\_60.png" alt="">

            </div>

            <div>图标和文字垂直居中</div>

        </div>