**Flex Box:**

<body>

    <div class="container">

      <div class="el el--1">HTML</div>

      <div class="el el--2">and</div>

      <div class="el el--3">CSS</div>

      <div class="el el--4">are</div>

      <div class="el el--5">amazing</div>

      <div class="el el--6">languages</div>

      <div class="el el--7">to</div>

      <div class="el el--8">learn</div>

    </div>

  </body>

<style>

      .el--1 {

        background-color: blueviolet;

      }

      .el--2 {

        background-color: orangered;

      }

      .el--3 {

        background-color: green;

        height: 150px;

      }

      .el--4 {

        background-color: goldenrod;

      }

      .el--5 {

        background-color: palevioletred;

      }

      .el--6 {

        background-color: steelblue;

      }

      .el--7 {

        background-color: yellow;

      }

      .el--8 {

        background-color: crimson;

      }

      .container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

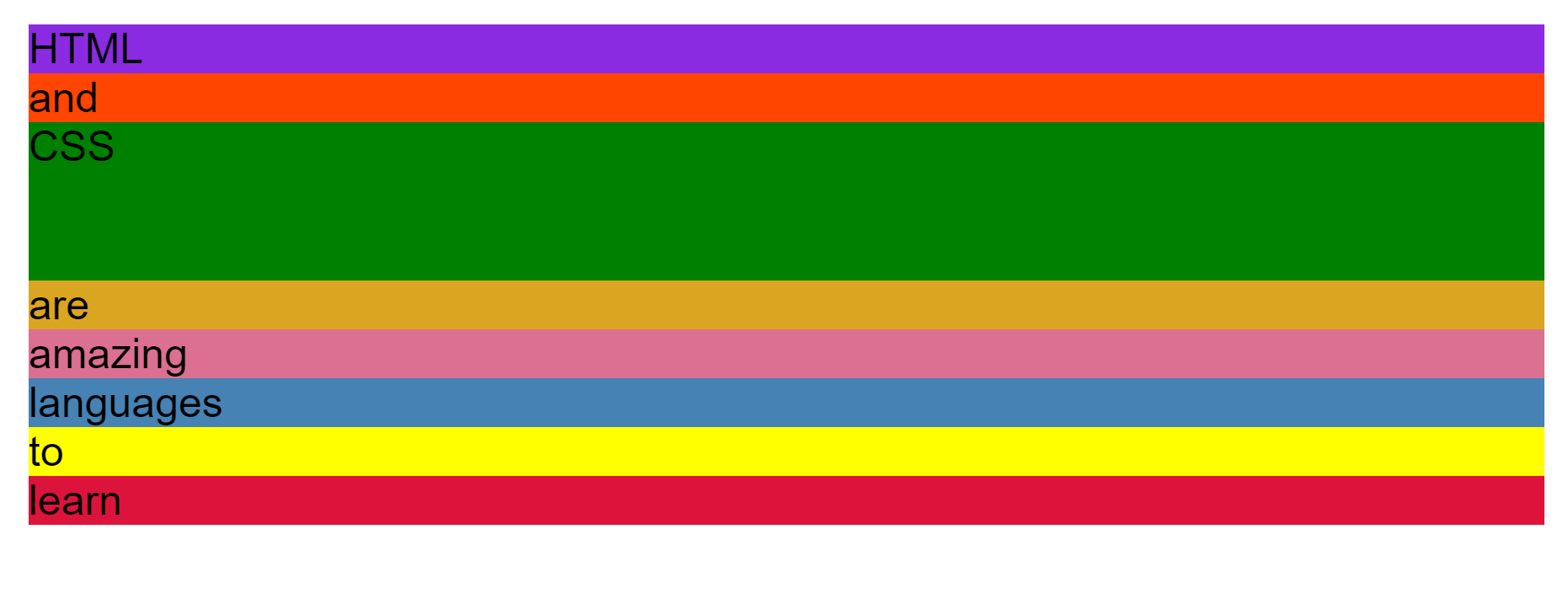
        margin: 40px;

        /\* FLEXBOX \*/

      }

    </style>

Before using FlexBox:



After setting display flex to container

.container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* FLEXBOX \*/

        display: flex;

      }



Vertical centering using align-items: center

.container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* FLEXBOX \*/

        display: flex;

        align-items: center;

      }



.container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* FLEXBOX \*/

        display: flex;

        align-items: flex-start;

      }



.container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

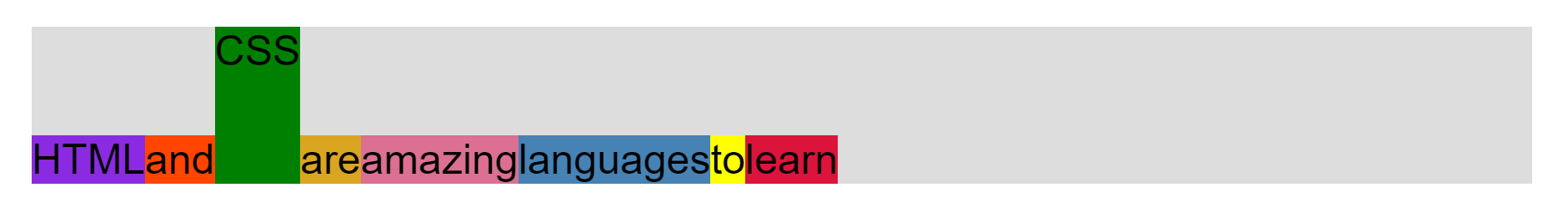
        margin: 40px;

        /\* FLEXBOX \*/

        display: flex;

        align-items: flex-end;

      }



 .container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* FLEXBOX \*/

        display: flex;

        align-items: stretch; /\* default \*/

      }



Horizontal centering using justify-content: center

 .container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* FLEXBOX \*/

        display: flex;

        align-items: center;

        justify-content: center;

      }



 .container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* FLEXBOX \*/

        display: flex;

        align-items: center;

        justify-content: space-between;

      }



.container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* FLEXBOX \*/

        display: flex;

        align-items: center;

        justify-content: space-evenly;

      }



Using align-self property on flex-items

.container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* FLEXBOX \*/

        display: flex;

        align-items: center;

        justify-content: flex-start;

      }

      .el--1 {

        align-self: flex-start;

      }

      .el--5 {

        align-self: stretch;

      }



Using order property on flex items

.el--1 {

        align-self: flex-start;

        order: 2;

      }

      .el--5 {

        align-self: stretch;

        order: 1;

      }

      .el--6 {

        order: -1;

      }

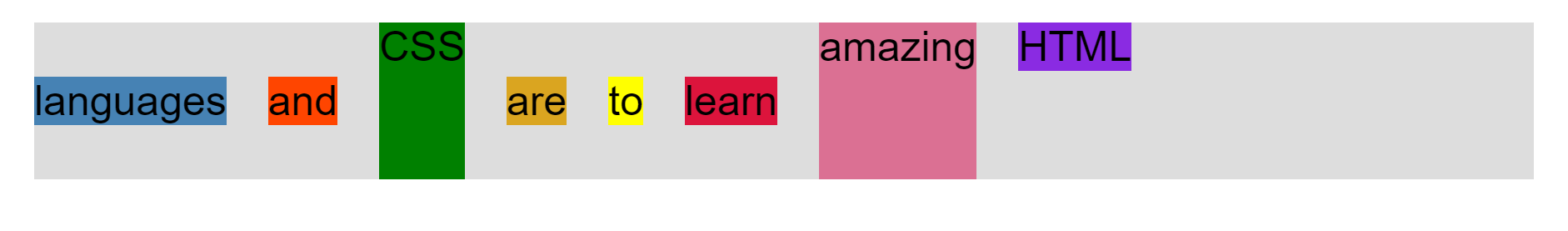


Using margin-right on flex items to manually define space

 .el {

        margin-right: 40px;

      }



Using gap property on flex container

 .container {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* FLEXBOX \*/

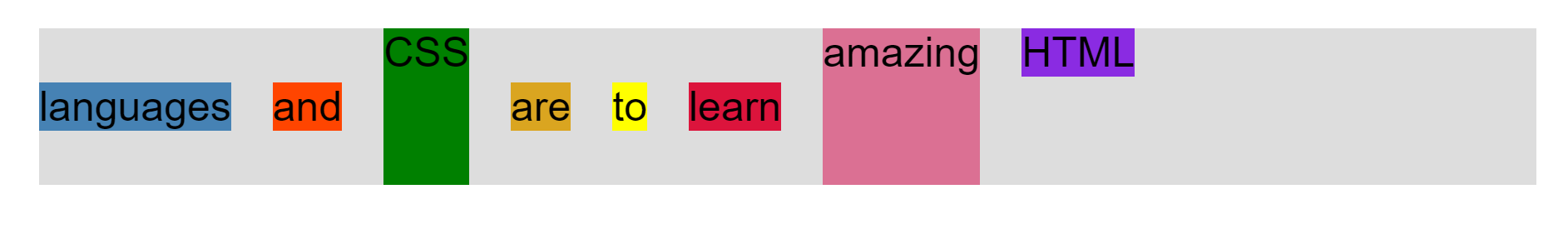
        display: flex;

        align-items: center;

        justify-content: flex-start;

        gap: 40px;

      }



**Flex Property**

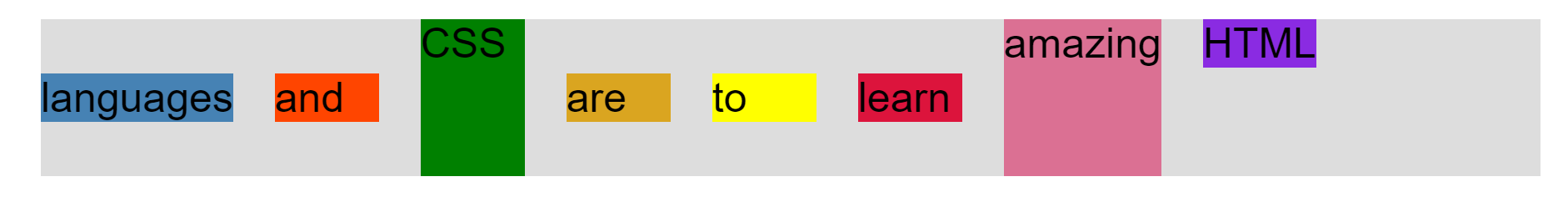
Specifying width using flex-basis to flex items

.el {

        /\* margin-right: 40px; \*/

        flex-basis: 100px;

      }



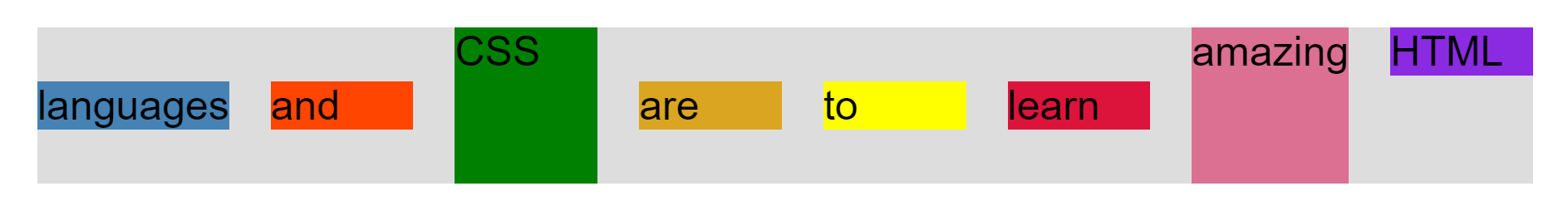
Using larger flex-basis (by default flex-shrink:1)

 .el {

        /\* margin-right: 40px; \*/

        flex-basis: 200px;

      }



Setting flex-shrink:0

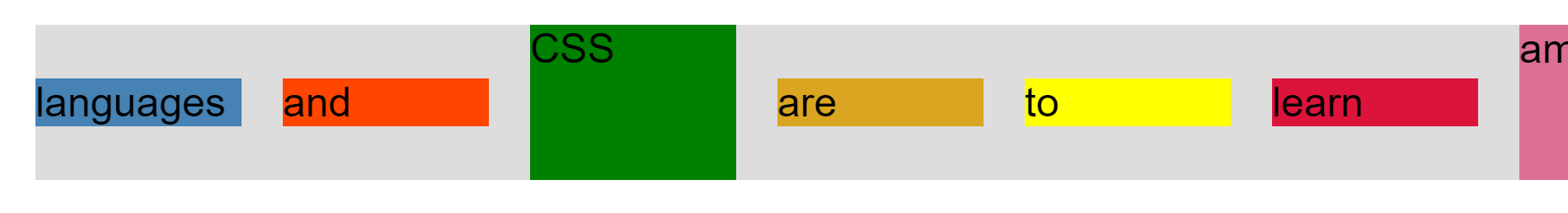
.el {

        /\* margin-right: 40px; \*/

        flex-basis: 200px;

        flex-shrink: 0;

      }



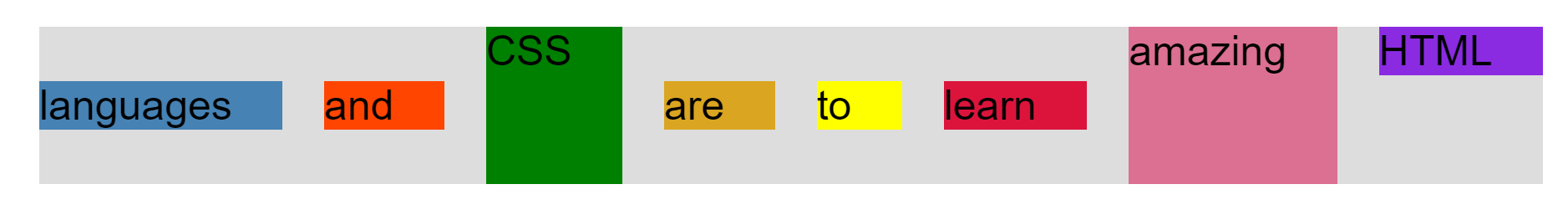
Setting flex-grow: 1 on all flex-items

      .el {

        /\* margin-right: 40px; \*/

        flex-grow: 1;

      }



How flex-grow works?

Setting flex-grow: 1 on single element

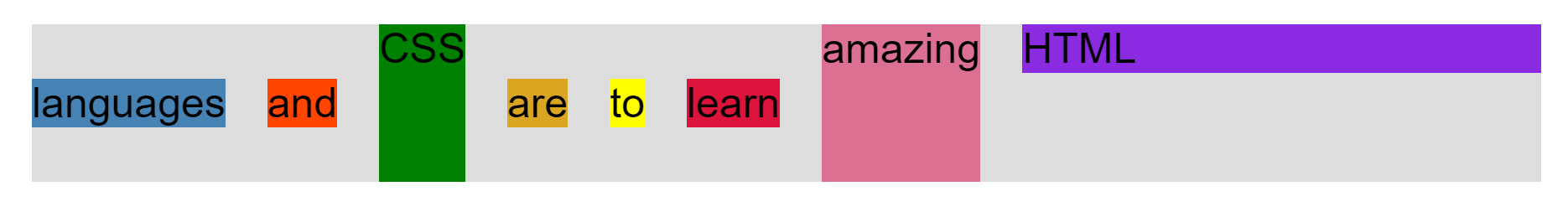
.el--1 {

        align-self: flex-start;

        order: 2;

        flex-grow: 1;

      }



Setting flex-grow: 2 on single element and flex-grow: 1 on all the elements

.el {

        /\* margin-right: 40px; \*/

        /\* flex-basis: 200px; \*/

        flex-grow: 1;

      }

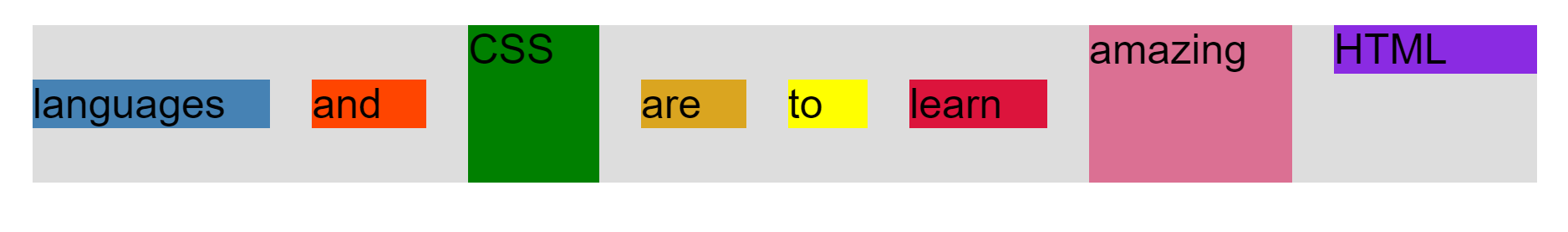
      .el--1 {

        align-self: flex-start;

        order: 2;

        flex-grow: 2;

      }



Instead of flex-grow: 1 we can use flex: 1 and it’ll figure out automatically

      .el {

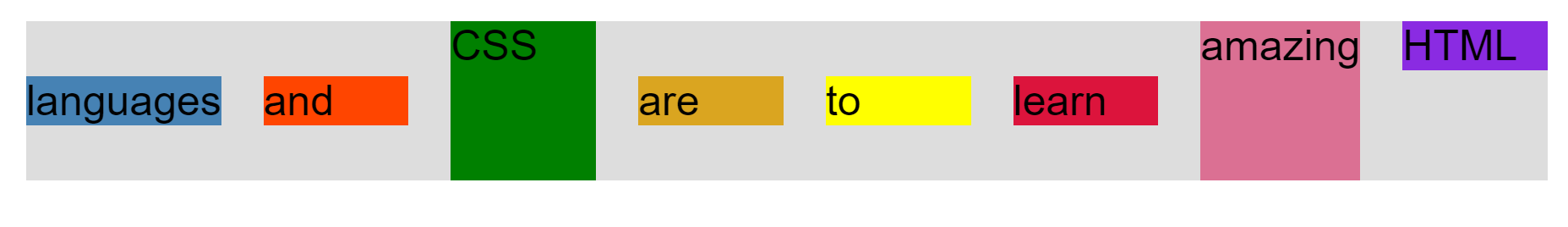
        /\* margin-right: 40px; \*/

        /\* flex-basis: 200px; \*/

        /\* flex-grow: 1; \*/

        flex: 1;

      }



Using flex as shorthand for flex-grow -shrink -basis

.el {

        /\* margin-right: 40px; \*/

        /\* flex-basis: 200px; \*/

        /\* flex-grow: 1; \*/

        /\* flex: 1; \*/

        flex: 0 1 100px; /\* shorthand for flex-grow -shrink -basis\*/

      }

