CSS Grid

<div class="container--1">

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

      <div class="el el--2">(2) and</div>

      <div class="el el--3">(3) CSS</div>

      <div class="el el--4">(4) are</div>

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

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

      <div class="el el--7">(7) to</div>

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

    </div>

<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--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

      }

</style>



After setting display to grid and defining columns

  .container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 250px 150px;

      }



.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

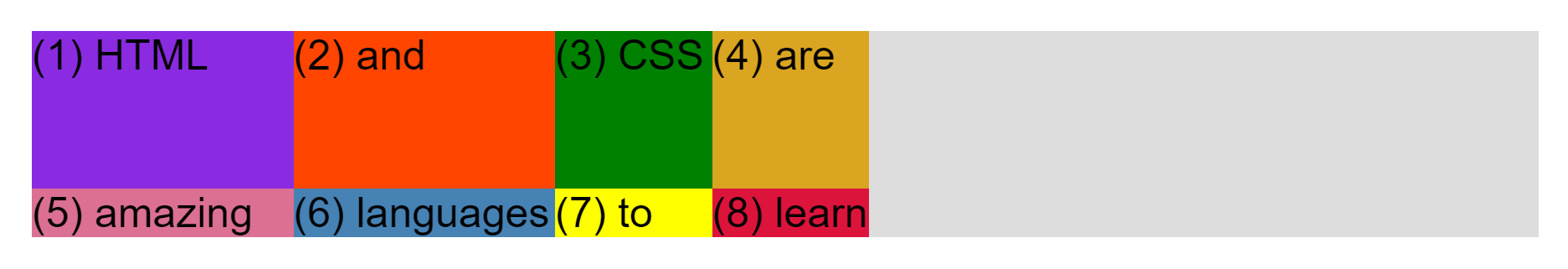
        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 250px 250px 150px 150px;

      }



Setting height of rows explicitly

.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

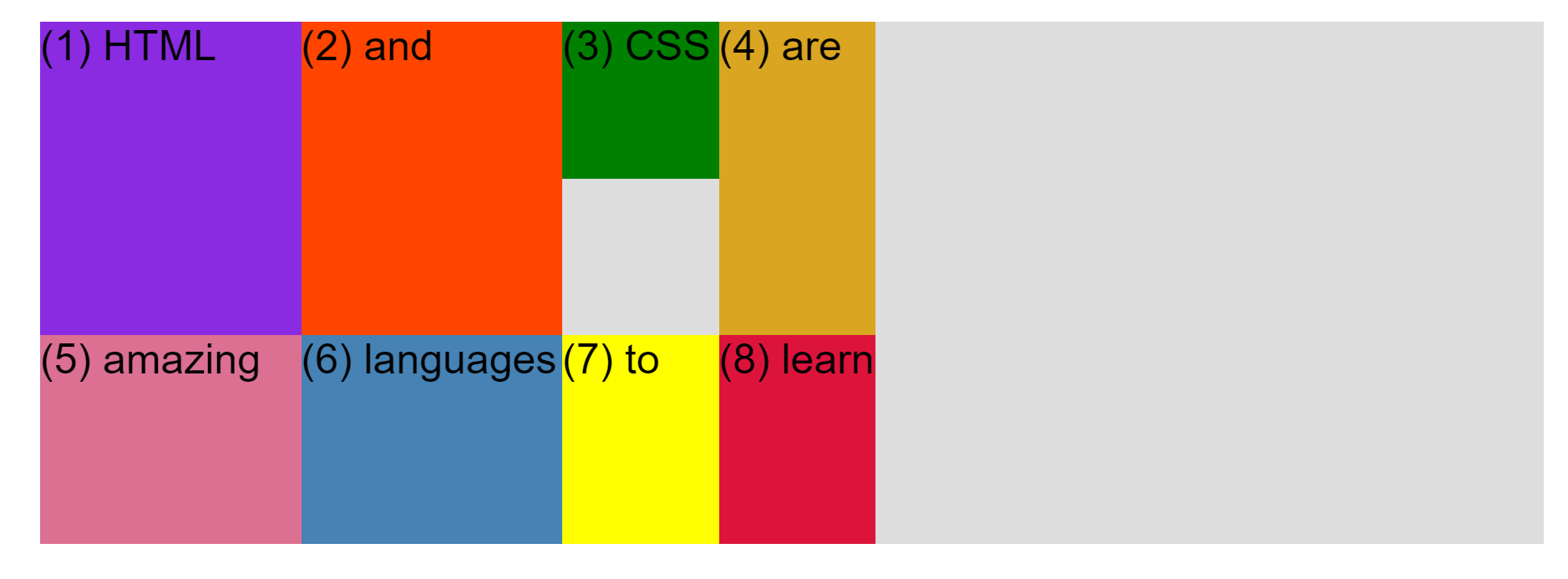
        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 250px 250px 150px 150px;

        grid-template-rows: 300px 200px;

      }



Setting gap between grid items

.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

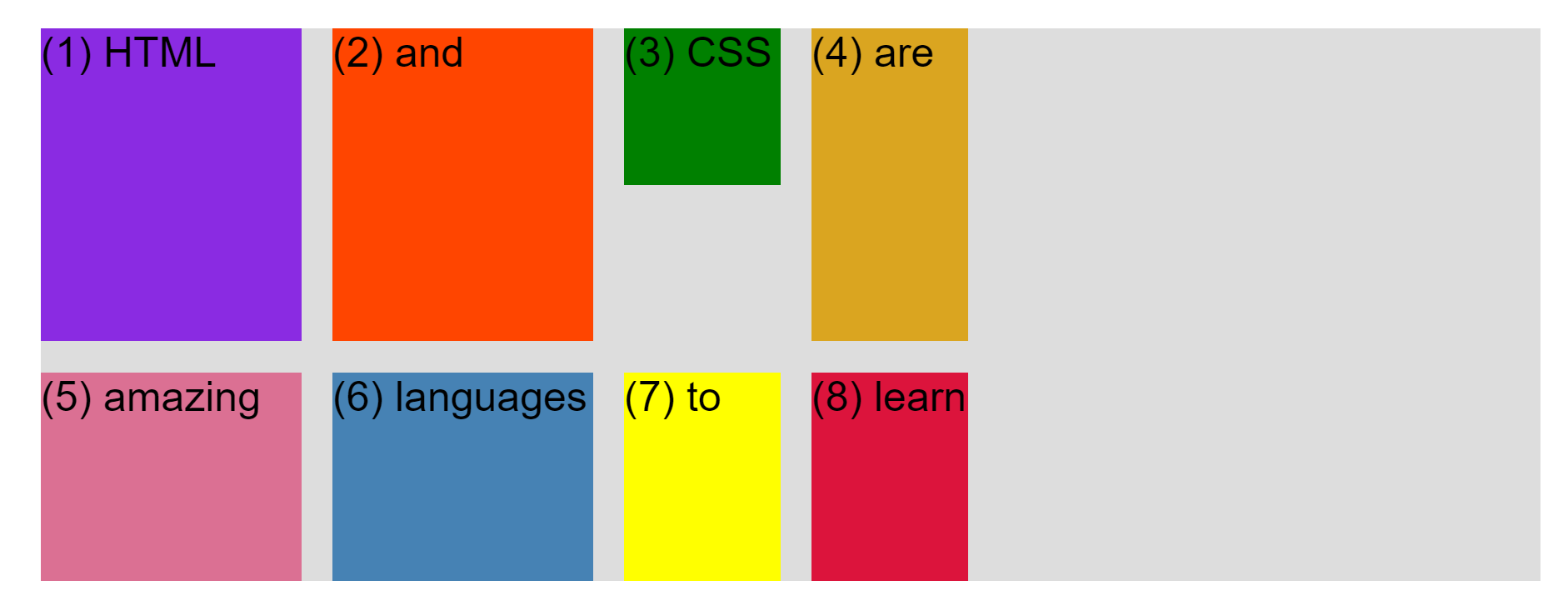
        display: grid;

        grid-template-columns: 250px 250px 150px 150px;

        grid-template-rows: 300px 200px;

        gap: 30px;

      }



Setting gap for rows and columns individually

 .container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 250px 250px 150px 150px;

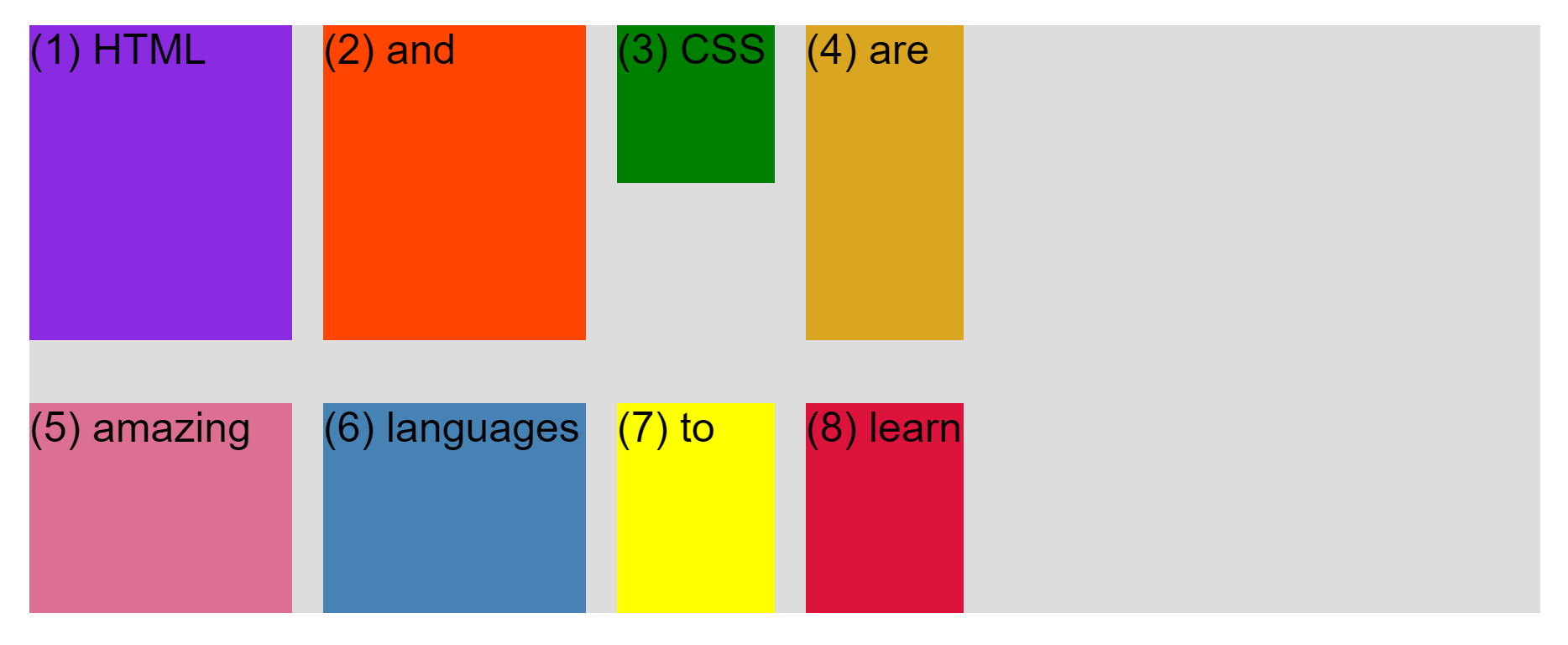
        grid-template-rows: 300px 200px;

        /\* gap: 30px; \*/

        column-gap: 30px;

        row-gap: 60px;

      }



Using fr for sizing rows and columns to create flexible rows/columns

.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 250px 250px 1fr 150px;

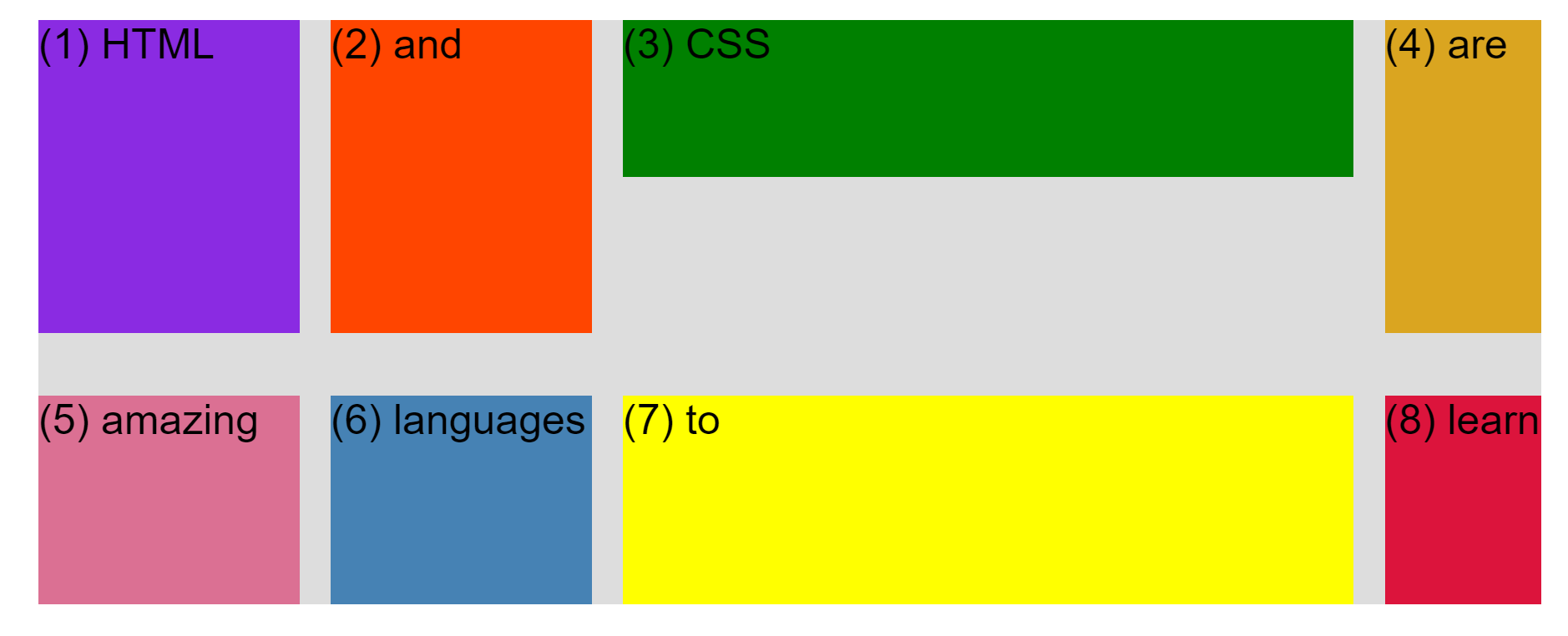
        grid-template-rows: 300px 200px;

        /\* gap: 30px; \*/

        column-gap: 30px;

        row-gap: 60px;

      }



.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 250px 250px 1fr 1fr;

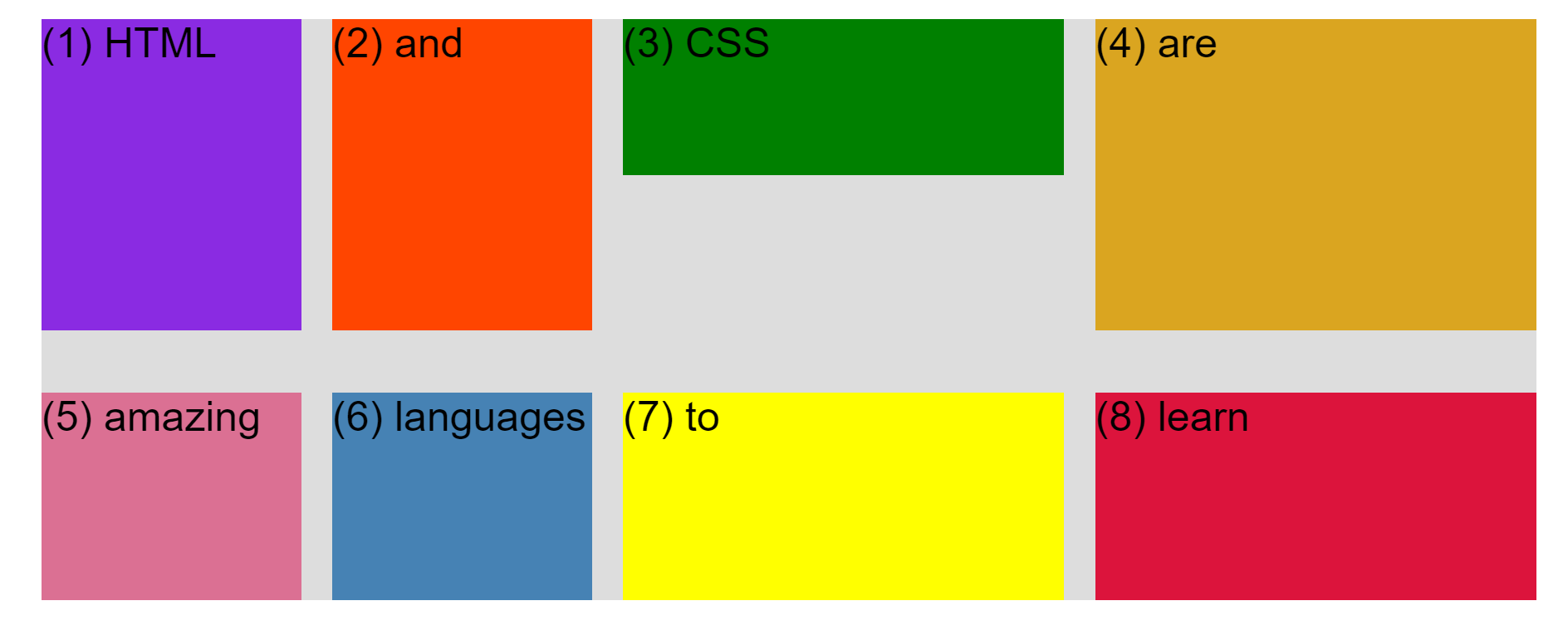
        grid-template-rows: 300px 200px;

        /\* gap: 30px; \*/

        column-gap: 30px;

        row-gap: 60px;

      }



Creating equal sized columns using fr

.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        /\* grid-template-columns: 250px 250px 1fr 1fr; \*/

        grid-template-columns: 1fr 1fr 1fr 1fr;

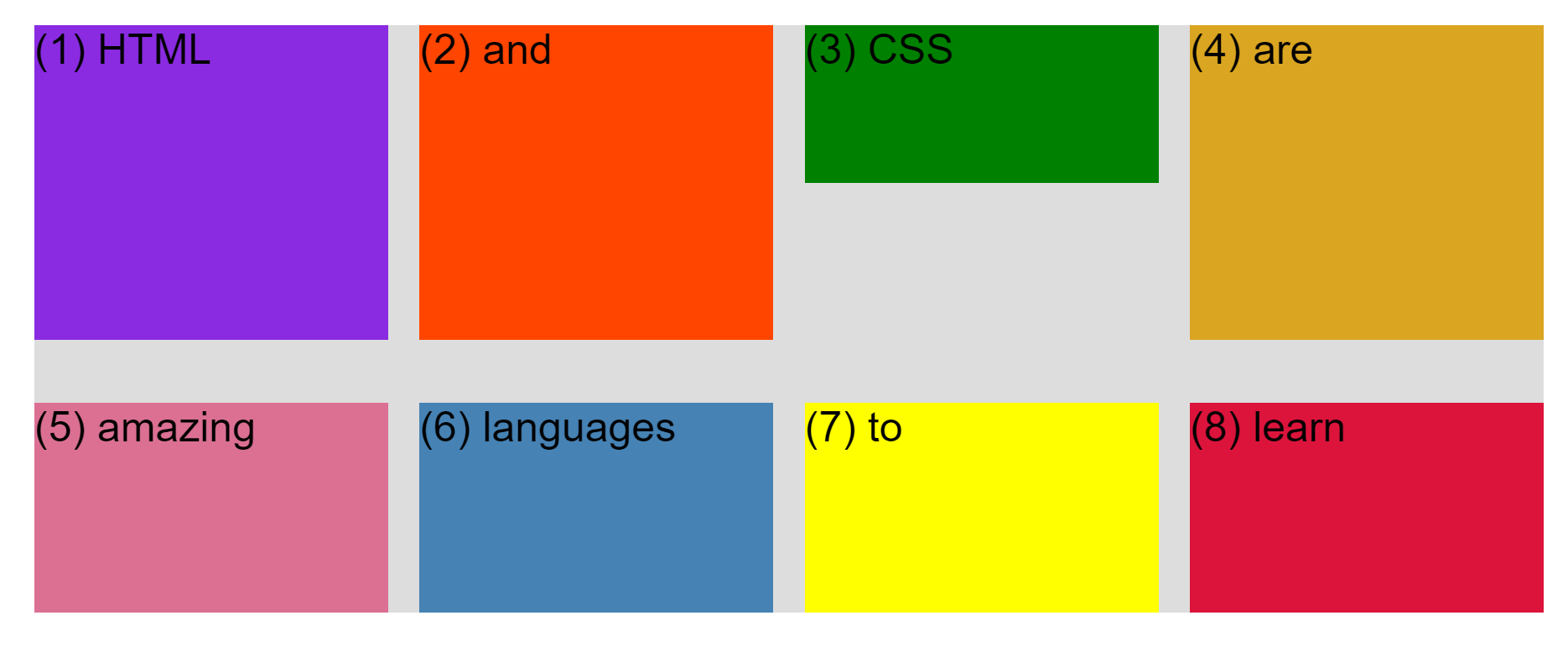
        grid-template-rows: 300px 200px;

        /\* gap: 30px; \*/

        column-gap: 30px;

        row-gap: 60px;

      }



 .container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        /\* grid-template-columns: 250px 250px 1fr 1fr; \*/

        grid-template-columns: 2fr 1fr 1fr 1fr;

        grid-template-rows: 300px 200px;

        /\* gap: 30px; \*/

        column-gap: 30px;

        row-gap: 60px;

      }



Using auto instead of fr

.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        /\* grid-template-columns: 250px 250px 1fr 1fr; \*/

        grid-template-columns: 1fr 1fr 1fr auto;

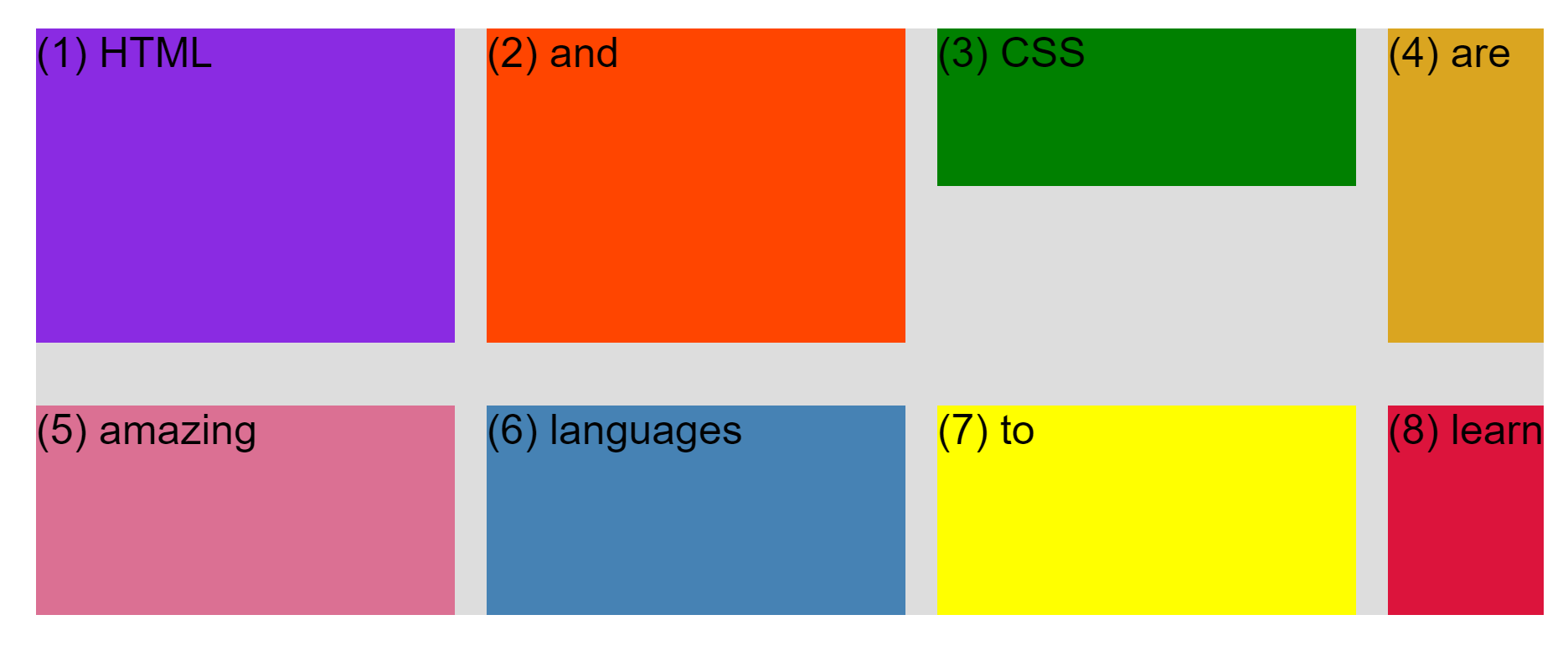
        grid-template-rows: 300px 200px;

        /\* gap: 30px; \*/

        column-gap: 30px;

        row-gap: 60px;

      }



Using repeat function

.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        /\* grid-template-columns: 250px 250px 1fr 1fr; \*/

        /\* grid-template-columns: 1fr 1fr 1fr 1fr; \*/

        /\* grid-template-columns: 1fr 1fr 1fr auto; \*/

        grid-template-columns: repeat(4, 1fr);

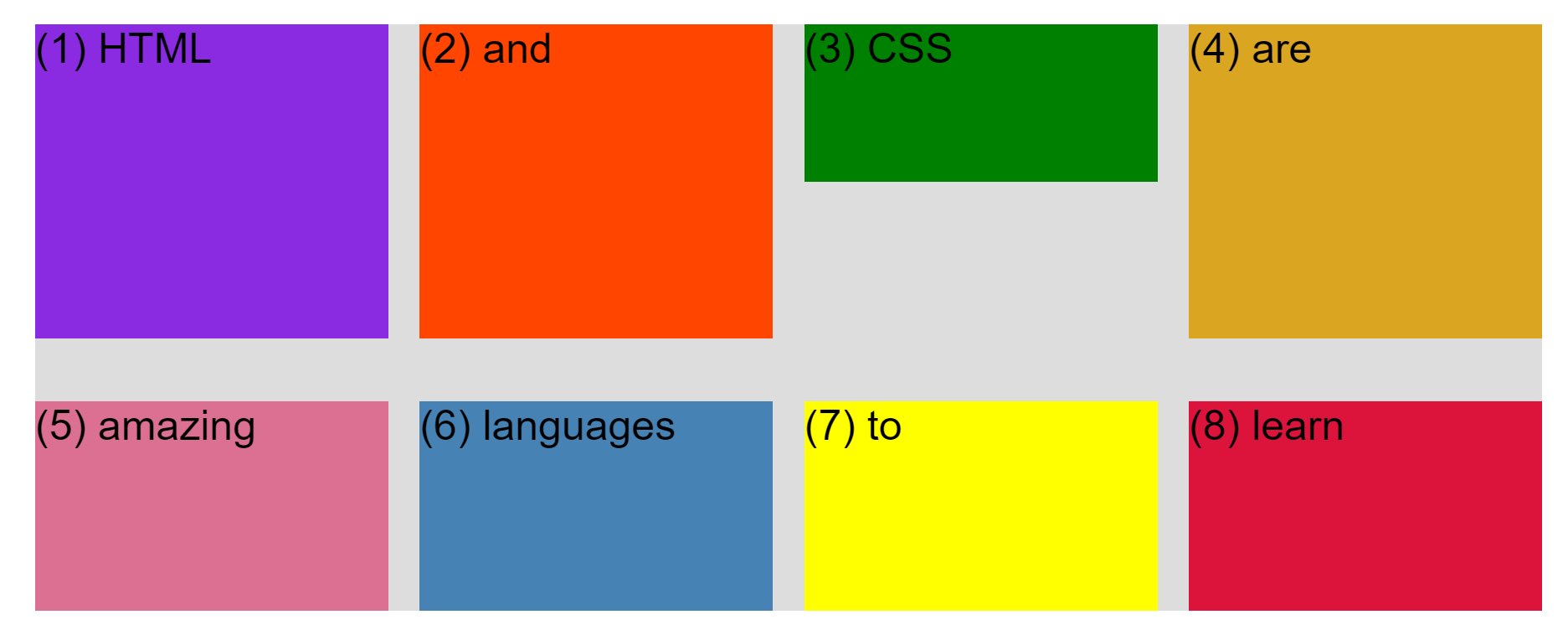
        grid-template-rows: 300px 200px;

        /\* gap: 30px; \*/

        column-gap: 30px;

        row-gap: 60px;

      }



Setting fr on sizing rows : Also removing 150px height on 3rd element

.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        /\* grid-template-columns: 250px 250px 1fr 1fr; \*/

        /\* grid-template-columns: 1fr 1fr 1fr 1fr; \*/

        /\* grid-template-columns: 1fr 1fr 1fr auto; \*/

        /\* grid-template-rows: 300px 200px; \*/

        grid-template-columns: repeat(4, 1fr);

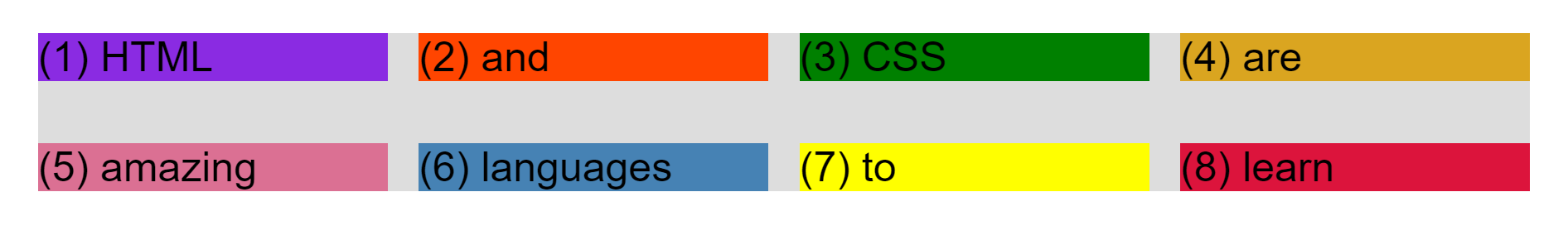
        grid-template-rows: 1fr 1fr;

        /\* gap: 30px; \*/

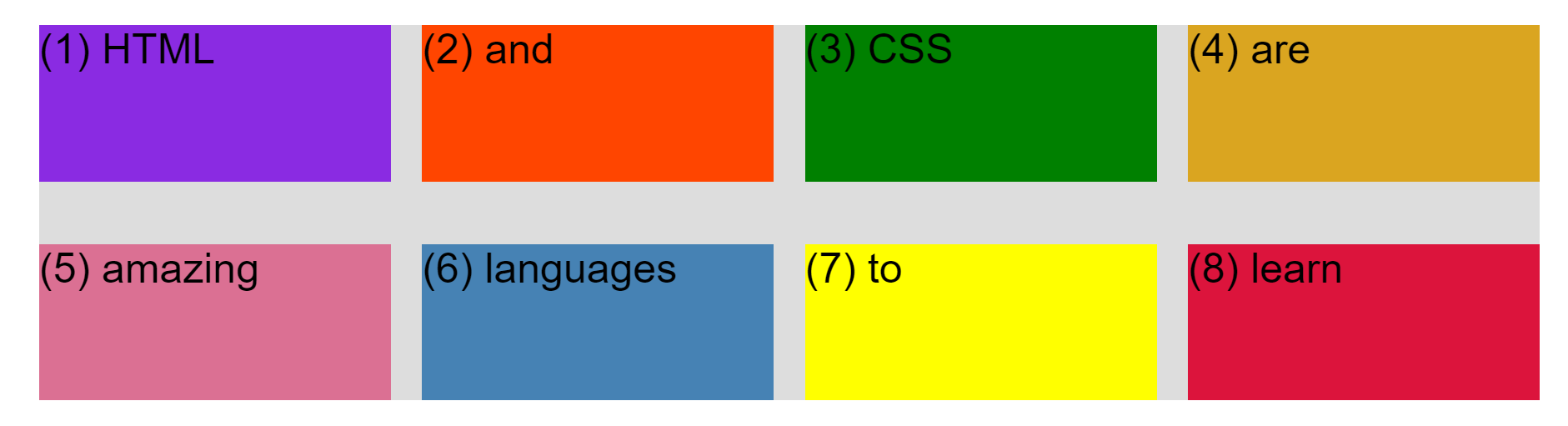
        column-gap: 30px;

        row-gap: 60px;

      }



After setting the 3rd element height to 150px



If we explicitly specify height for the container

 .container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        /\* grid-template-columns: 250px 250px 1fr 1fr; \*/

        /\* grid-template-columns: 1fr 1fr 1fr 1fr; \*/

        /\* grid-template-columns: 1fr 1fr 1fr auto; \*/

        /\* grid-template-rows: 300px 200px; \*/

        grid-template-columns: repeat(4, 1fr);

        grid-template-rows: 1fr 1fr;

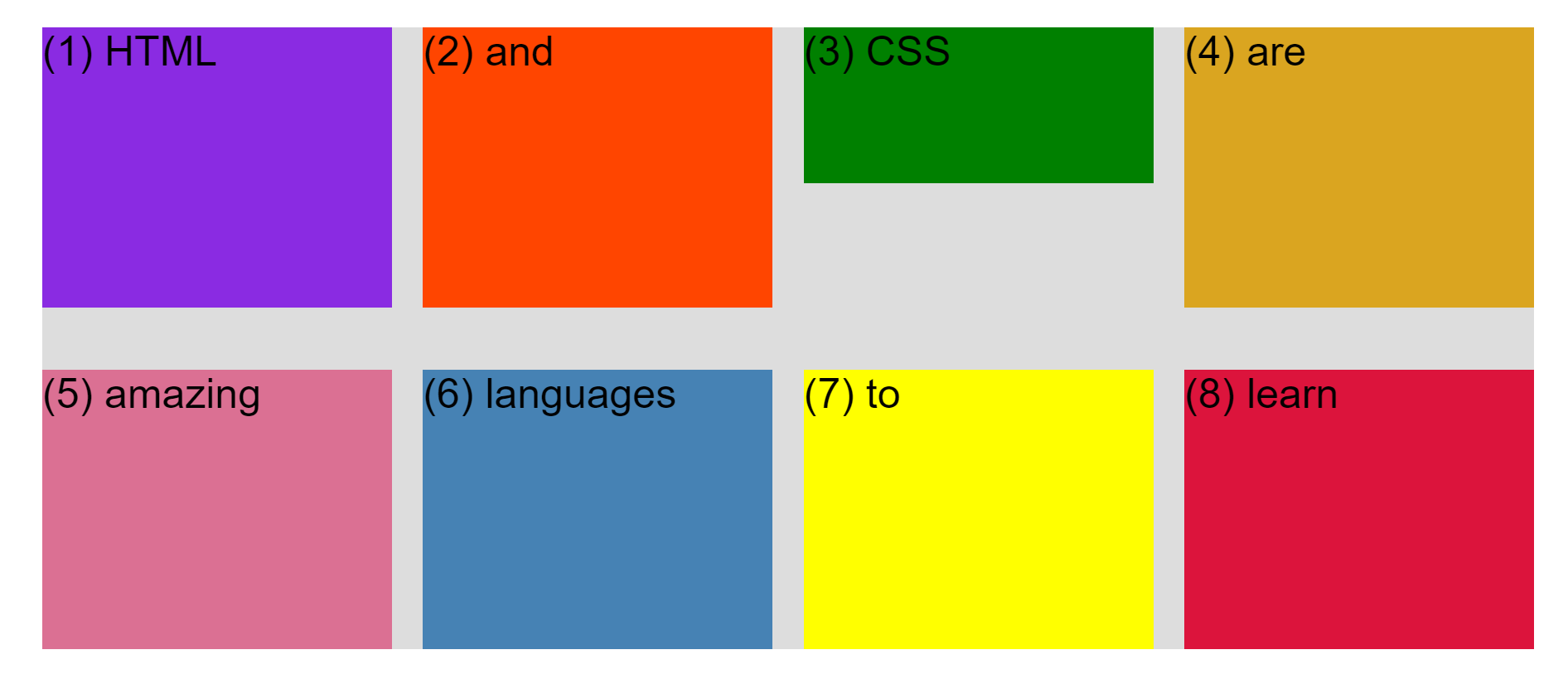
        height: 600px;

        /\* gap: 30px; \*/

        column-gap: 30px;

        row-gap: 60px;

      }



Specifying auto on sizing rows

.container--1 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: #ddd;

        font-size: 40px;

        margin: 40px;

        /\* CSS GRID \*/

        display: grid;

        /\* grid-template-columns: 250px 250px 1fr 1fr; \*/

        /\* grid-template-columns: 1fr 1fr 1fr 1fr; \*/

        /\* grid-template-columns: 1fr 1fr 1fr auto; \*/

        /\* grid-template-rows: 300px 200px; \*/

        grid-template-columns: repeat(4, 1fr);

        grid-template-rows: 1fr auto;

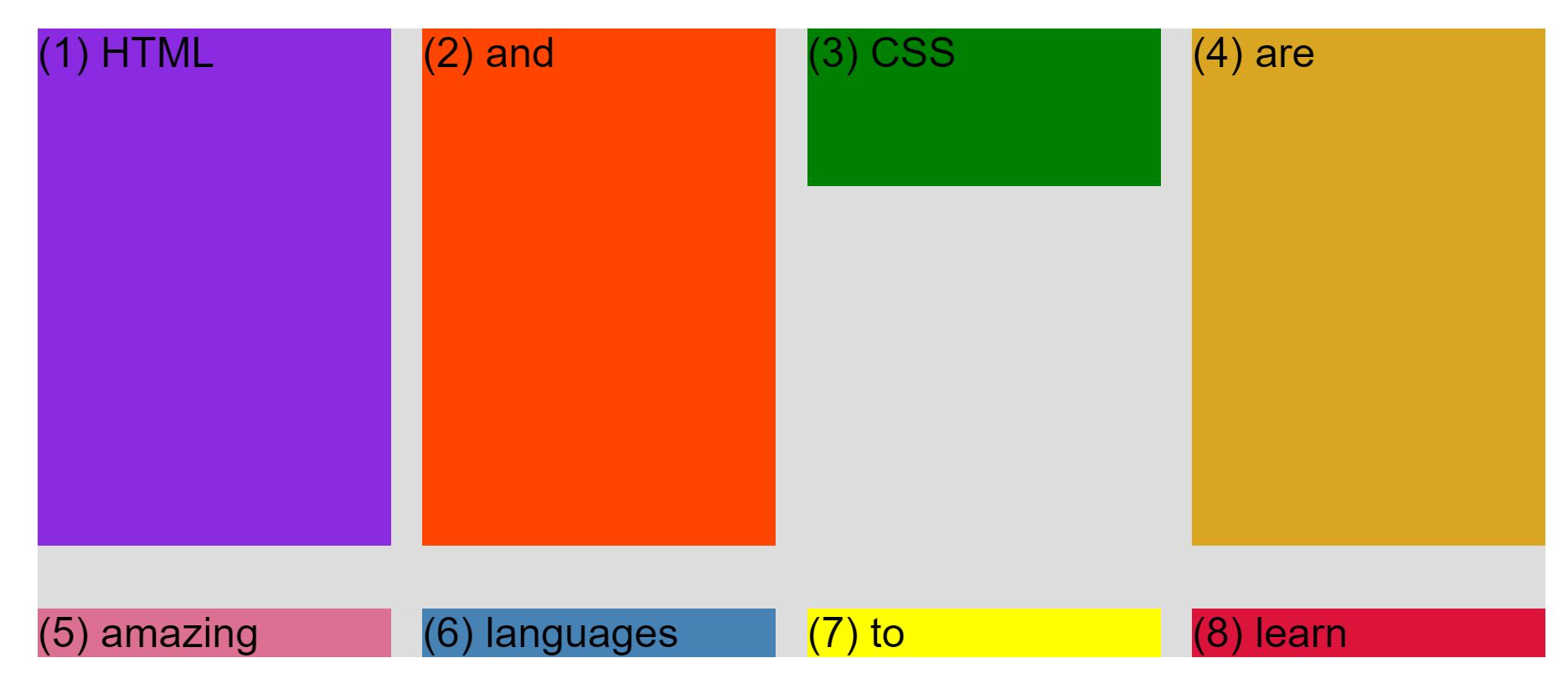
        height: 600px;

        /\* gap: 30px; \*/

        column-gap: 30px;

        row-gap: 60px;

      }



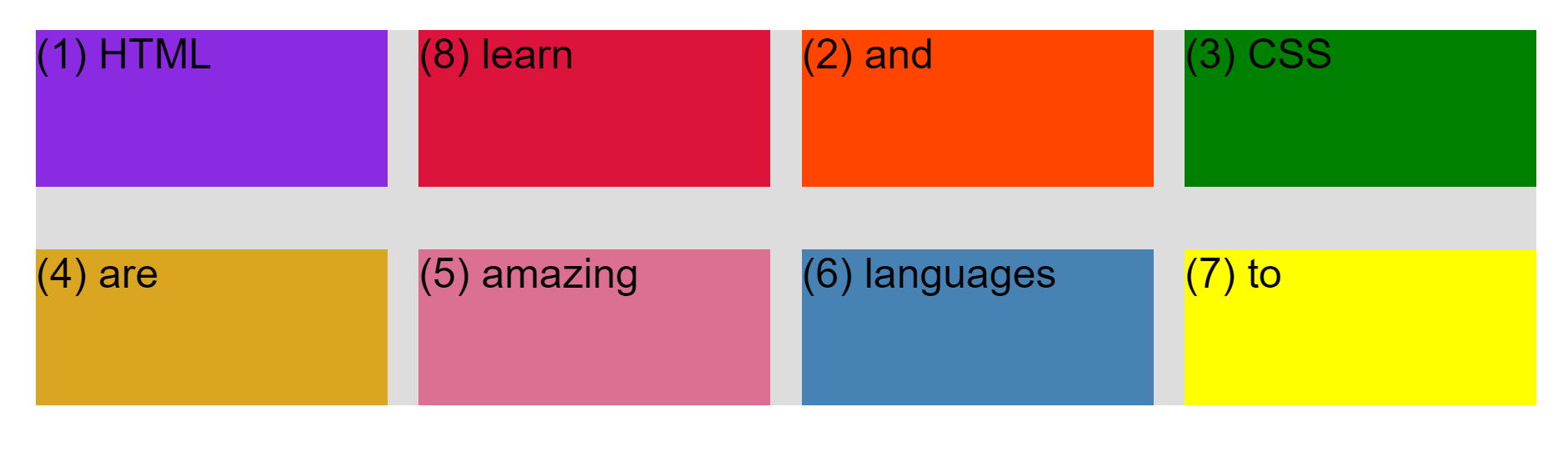
Placing a grid item in a particular grid cell

.el--8 {

        grid-column: 2 / 3;

        grid-row: 1 / 2;

      }



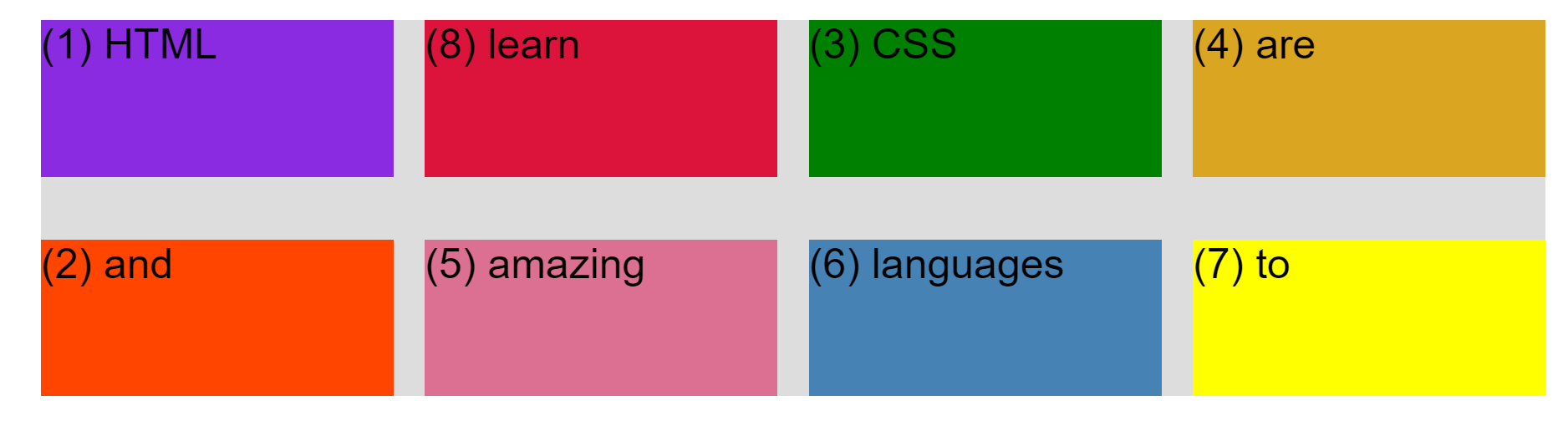
Omitting second value when it is just plus 1 than the first value

.el--2 {

        grid-column: 1;

        grid-row: 2;

      }



Spanning across two columns

.el--2 {

        grid-column: 1 / 3;

        grid-row: 2;

      }



Using span no.of.cells to span across certain cell

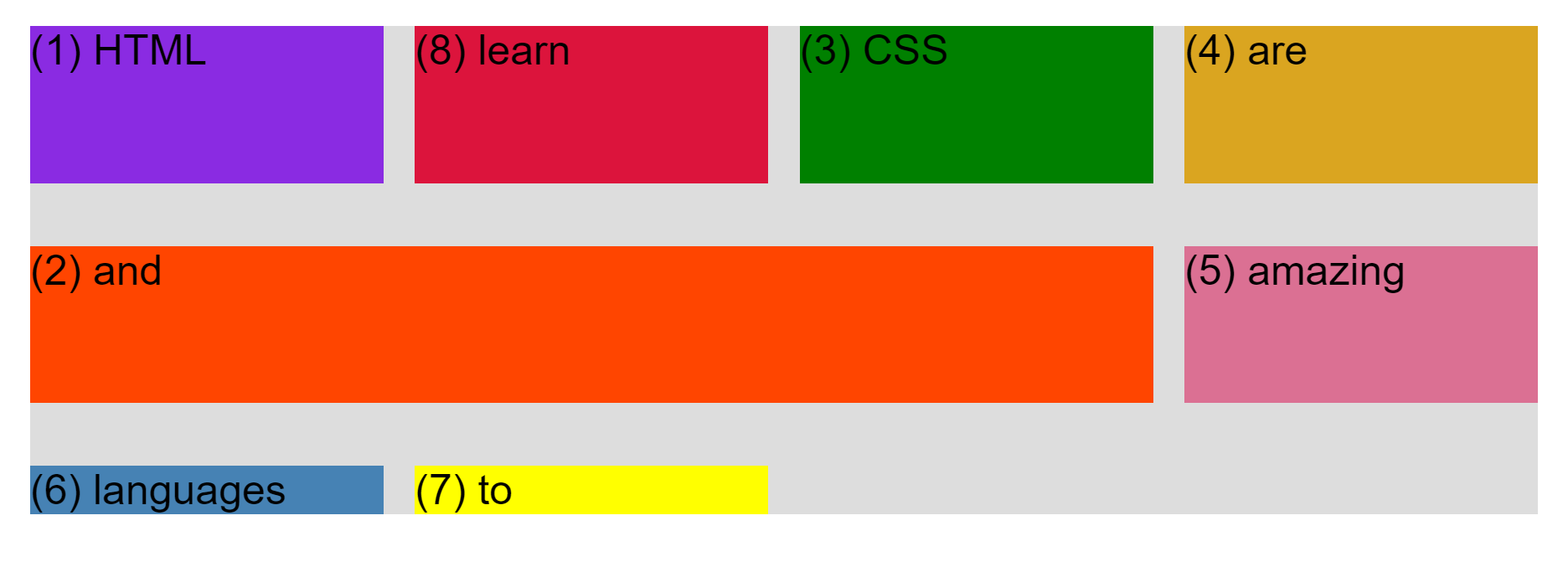
.el--2 {

        /\* grid-column: 1 / 3; \*/

        grid-column: 1 / span 3;

        grid-row: 2;

      }



Spanning one-grid item all the way to the end

.el--2 {

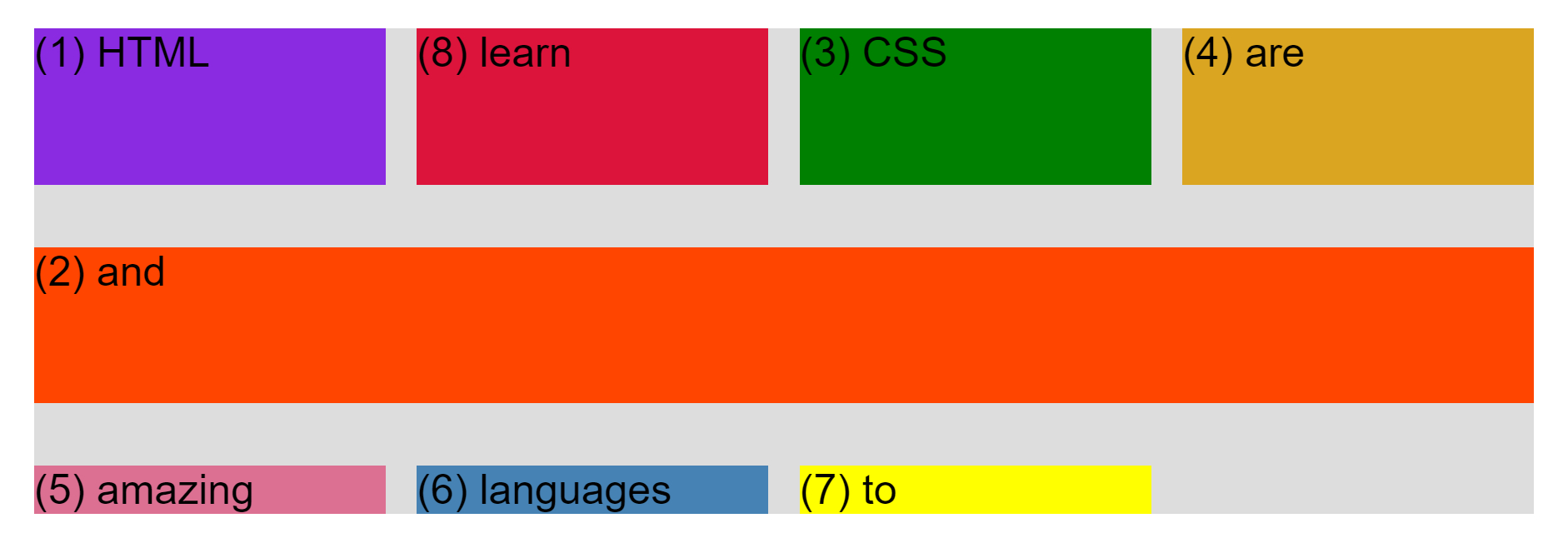
        /\* grid-column: 1 / 3; \*/

        /\* grid-column: 1 / span 3; \*/

        grid-column: 1 / -1;

        grid-row: 2;

      }



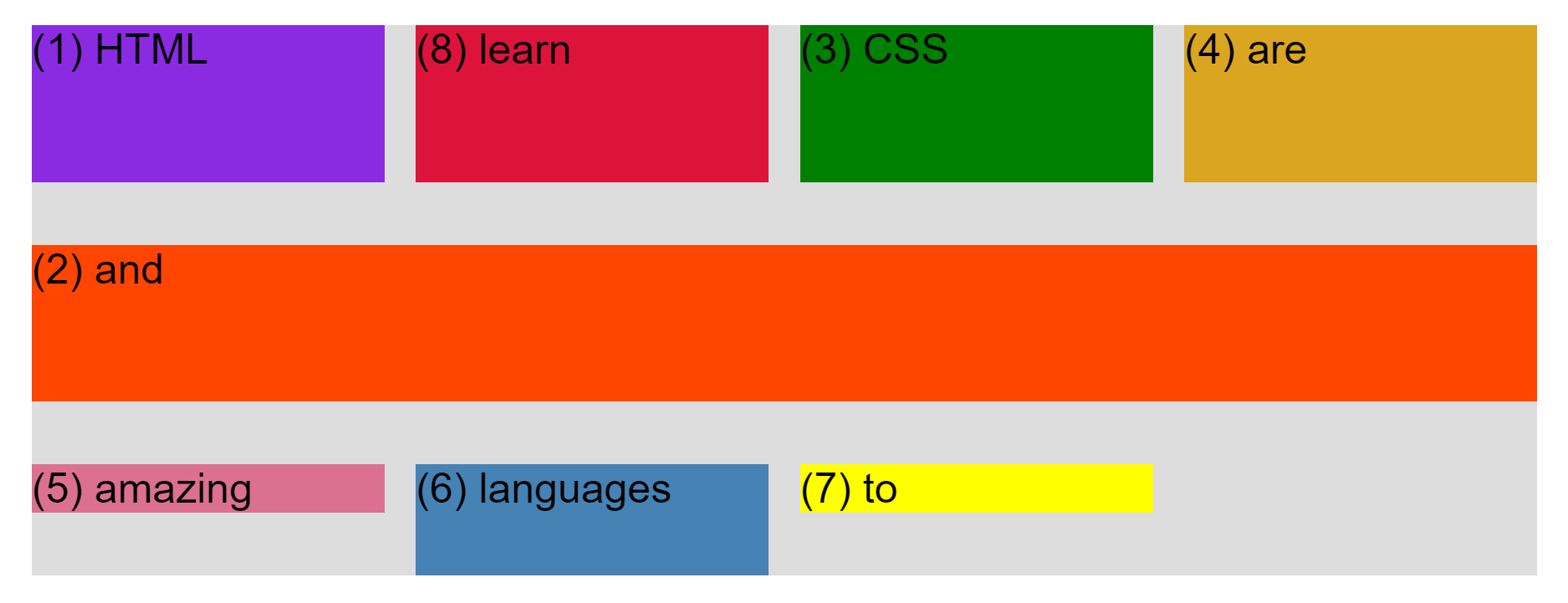
Spanning also works for rows

.el--6 {

        grid-row: 3 / span 2;

        grid-column: 2;

      }



Aligning grid tracks inside the container (only works if grid itself is smaller than the container)

Before aligning

.container--2 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: black;

        font-size: 40px;

        margin: 40px;

        width: 700px;

        height: 600px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 125px 200px 125px;

        grid-template-rows: 250px 100px;

        /\* Aligning grid tracks inside the container \*/

      }



  .container--2 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: black;

        font-size: 40px;

        margin: 40px;

        width: 700px;

        height: 600px;

        /\* CSS GRID \*/

        display: grid;

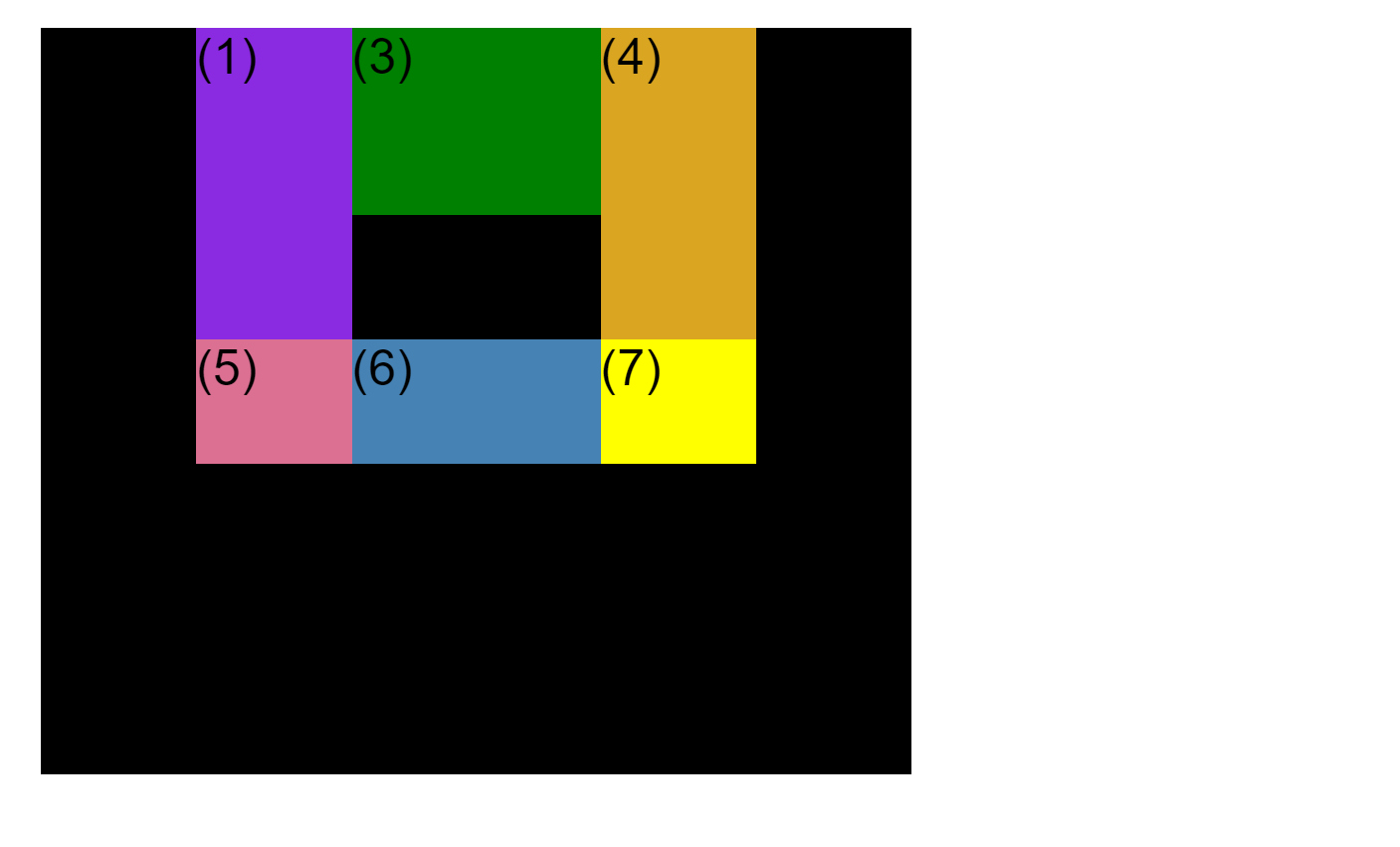
        grid-template-columns: 125px 200px 125px;

        grid-template-rows: 250px 100px;

        /\* Aligning grid tracks inside the container \*/

        justify-content: center;

      }



.container--2 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: black;

        font-size: 40px;

        margin: 40px;

        width: 700px;

        height: 600px;

        /\* CSS GRID \*/

        display: grid;

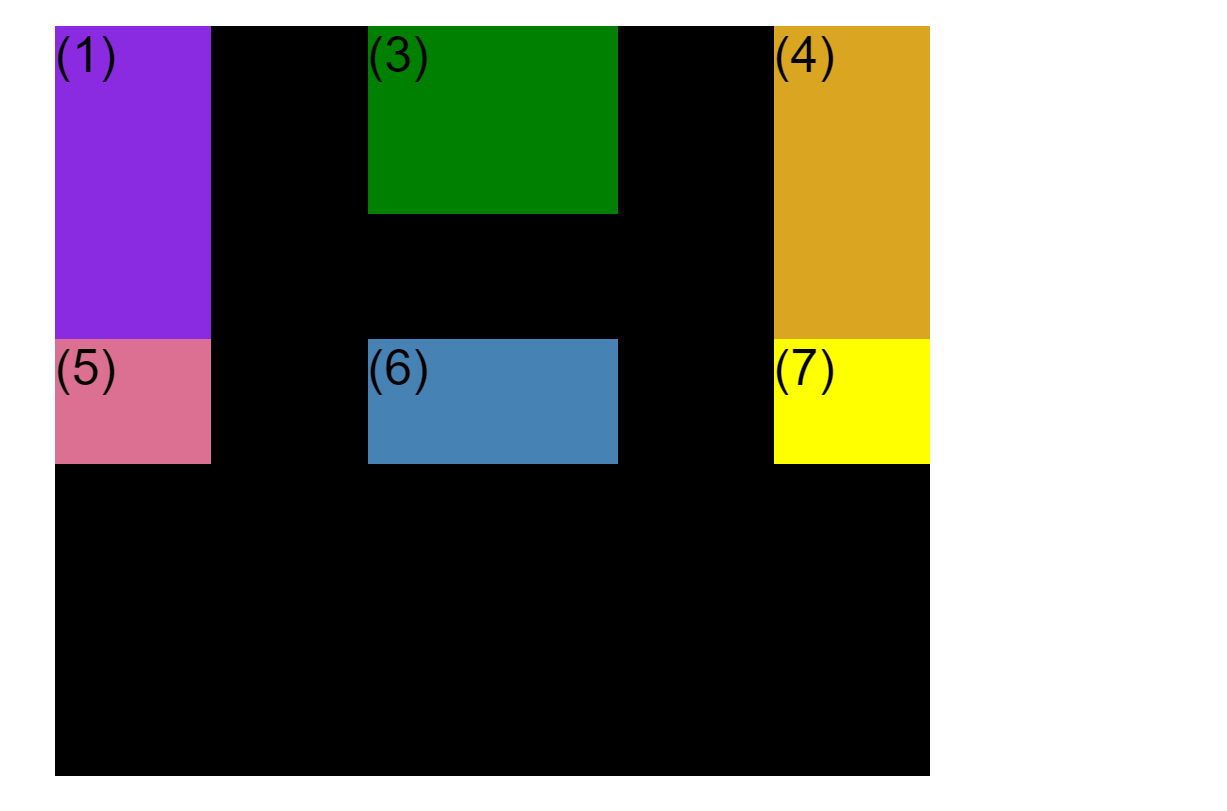
        grid-template-columns: 125px 200px 125px;

        grid-template-rows: 250px 100px;

        /\* Aligning grid tracks inside the container \*/

        justify-content: space-between;

      }



.container--2 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: black;

        font-size: 40px;

        margin: 40px;

        width: 700px;

        height: 600px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 125px 200px 125px;

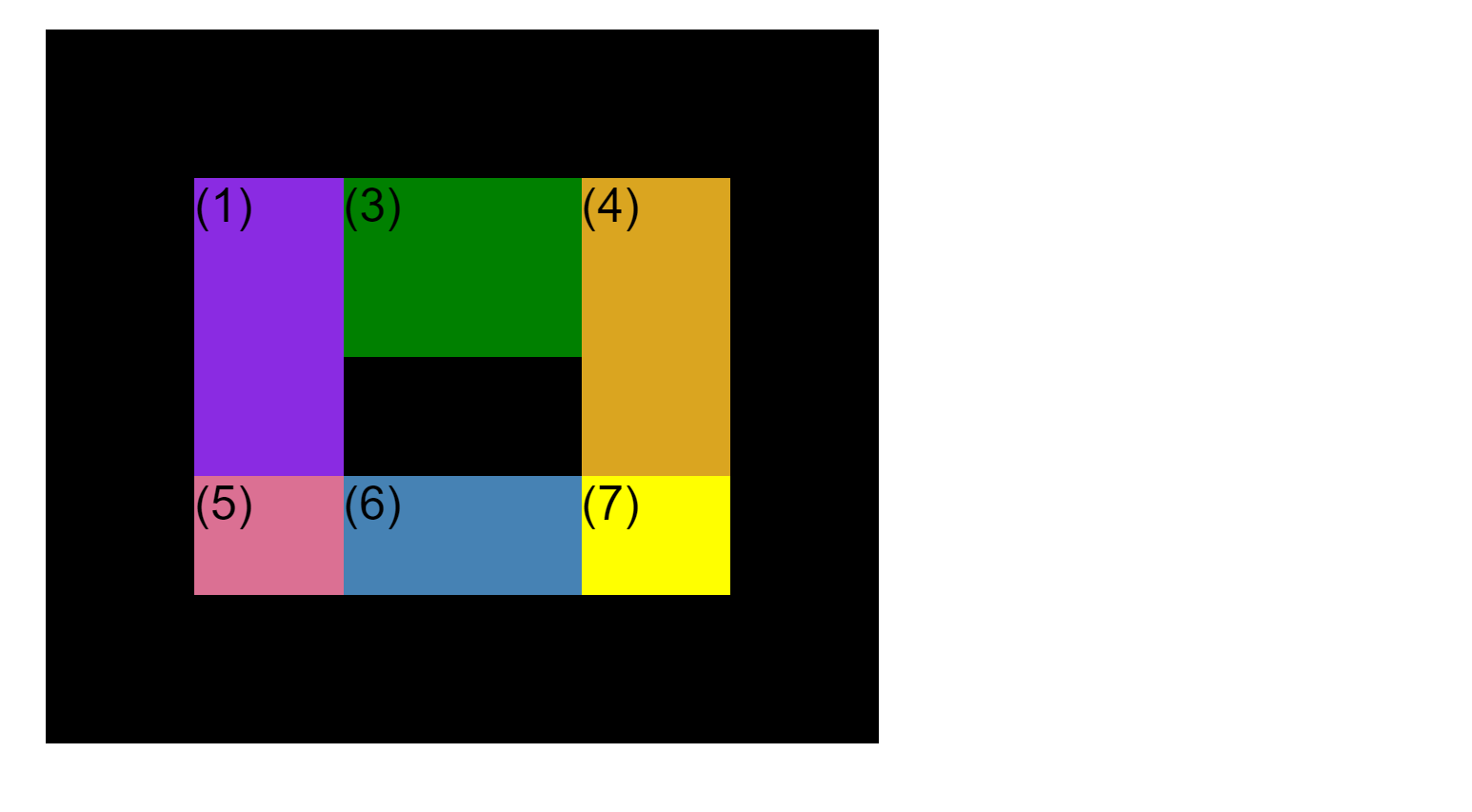
        grid-template-rows: 250px 100px;

        /\* Aligning grid tracks inside the container \*/

        justify-content: center;

        align-content: center;

      }



.container--2 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: black;

        font-size: 40px;

        margin: 40px;

        width: 700px;

        height: 600px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 125px 200px 125px;

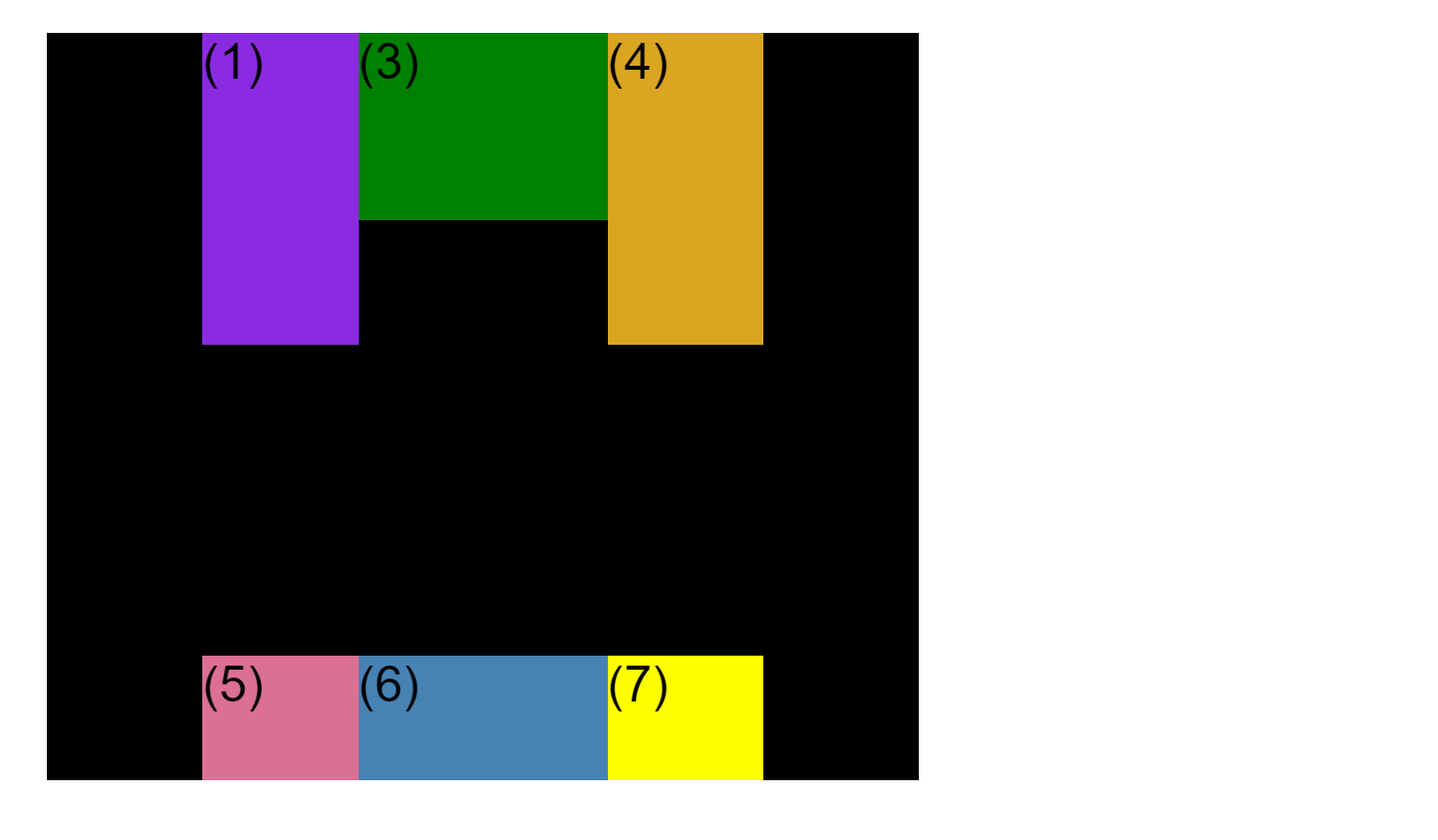
        grid-template-rows: 250px 100px;

        /\* Aligning grid tracks inside the container \*/

        justify-content: center;

        align-content: space-between;

      }



 .container--2 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: black;

        font-size: 40px;

        margin: 40px;

        width: 700px;

        height: 600px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 125px 200px 125px;

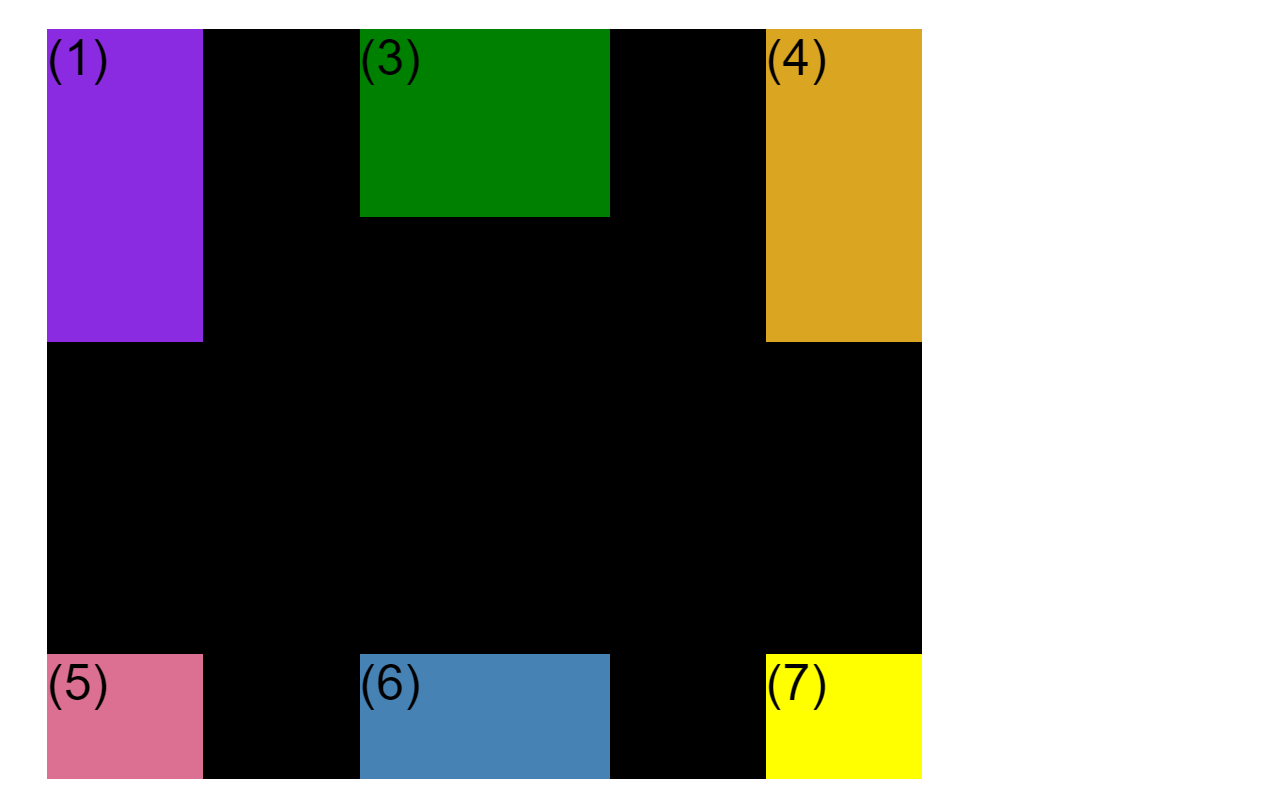
        grid-template-rows: 250px 100px;

        /\* Aligning grid tracks inside the container \*/

        justify-content: space-between;

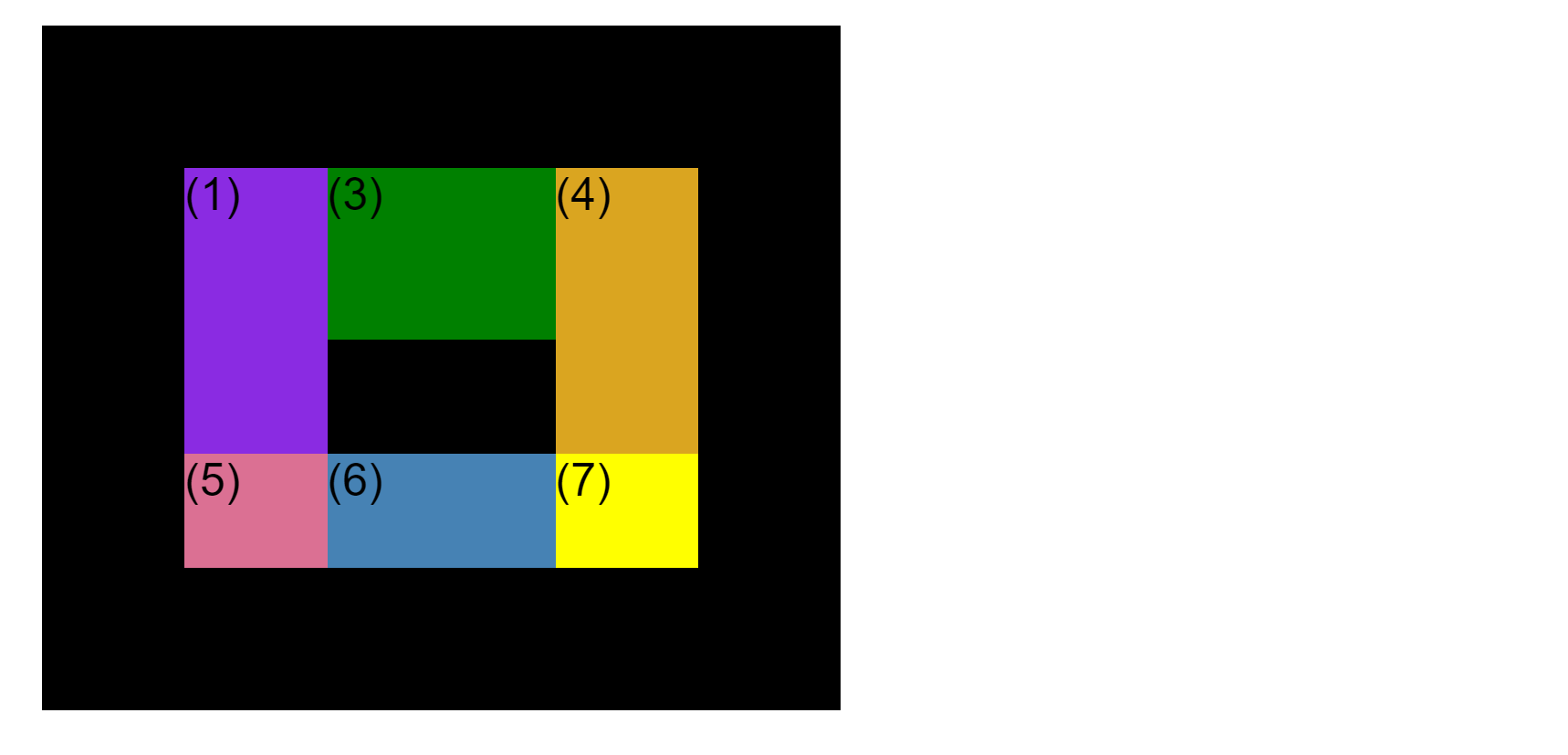
        align-content: space-between;

      }



Aligning grid items inside the cell

Before aligning



After aligning

.container--2 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: black;

        font-size: 40px;

        margin: 40px;

        width: 700px;

        height: 600px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 125px 200px 125px;

        grid-template-rows: 250px 100px;

        /\* Aligning grid tracks inside the container \*/

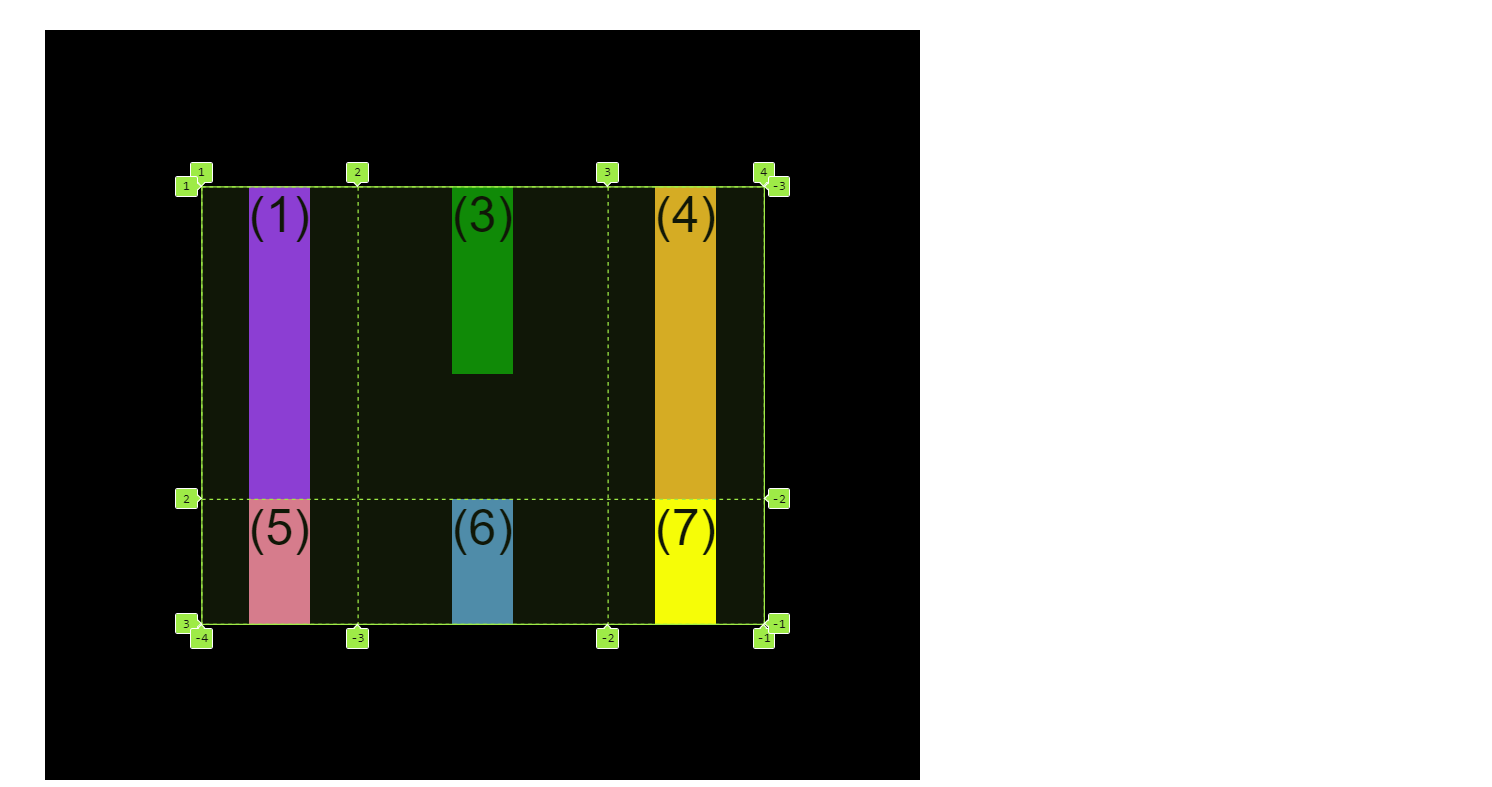
        justify-content: center;

        align-content: center;

        /\* Aligning grid items inside of the cell \*/

        justify-items: center;

      }



 .container--2 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: black;

        font-size: 40px;

        margin: 40px;

        width: 700px;

        height: 600px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 125px 200px 125px;

        grid-template-rows: 250px 100px;

        /\* Aligning grid tracks inside the container \*/

        justify-content: center;

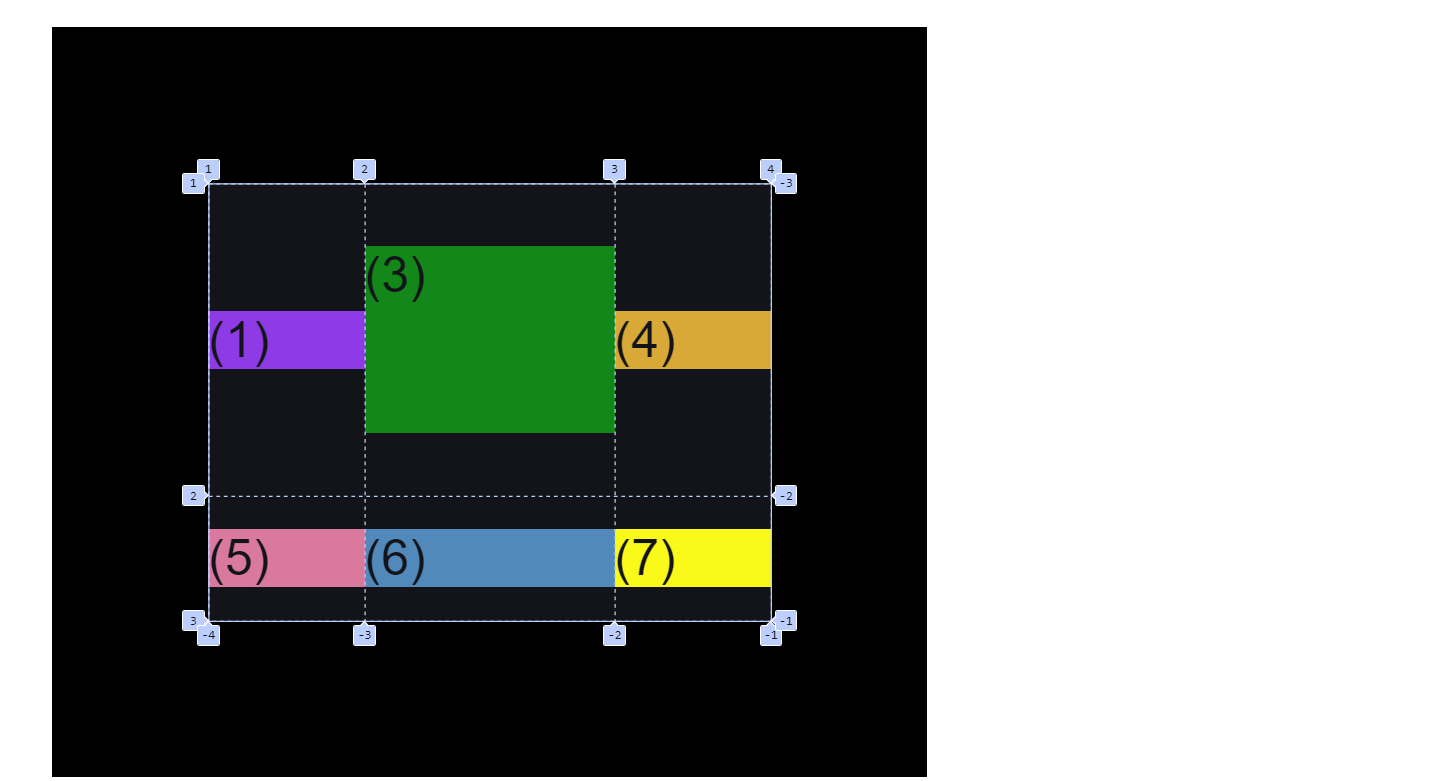
        align-content: center;

        /\* Aligning grid items inside of the cell \*/

        /\* justify-items: center; \*/

        align-items: center;

      }



 .container--2 {

        /\* STARTER \*/

        font-family: sans-serif;

        background-color: black;

        font-size: 40px;

        margin: 40px;

        width: 700px;

        height: 600px;

        /\* CSS GRID \*/

        display: grid;

        grid-template-columns: 125px 200px 125px;

        grid-template-rows: 250px 100px;

        /\* Aligning grid tracks inside the container \*/

        justify-content: center;

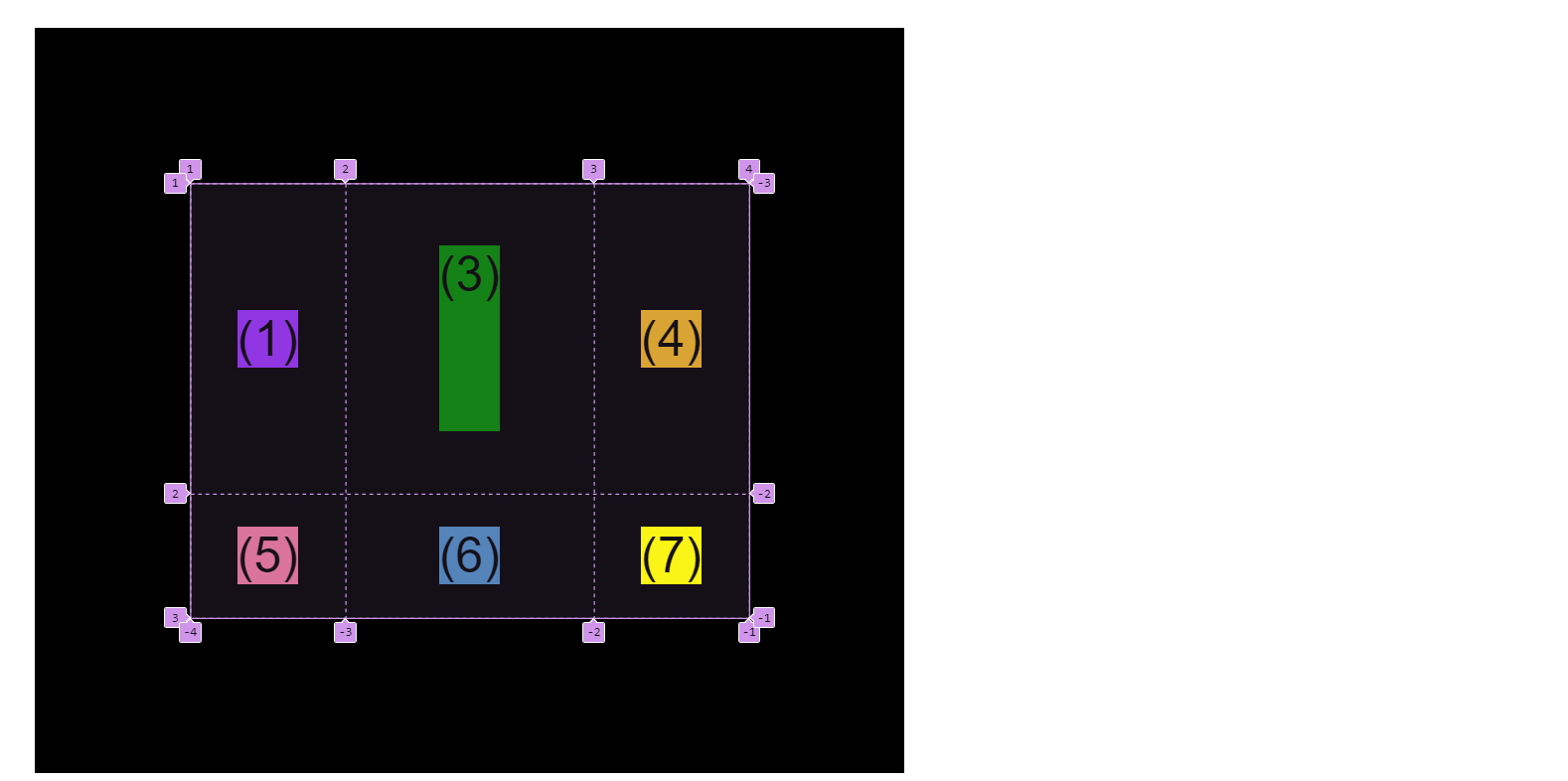
        align-content: center;

        /\* Aligning grid items inside of the cell \*/

        justify-items: center;

        align-items: center;

      }

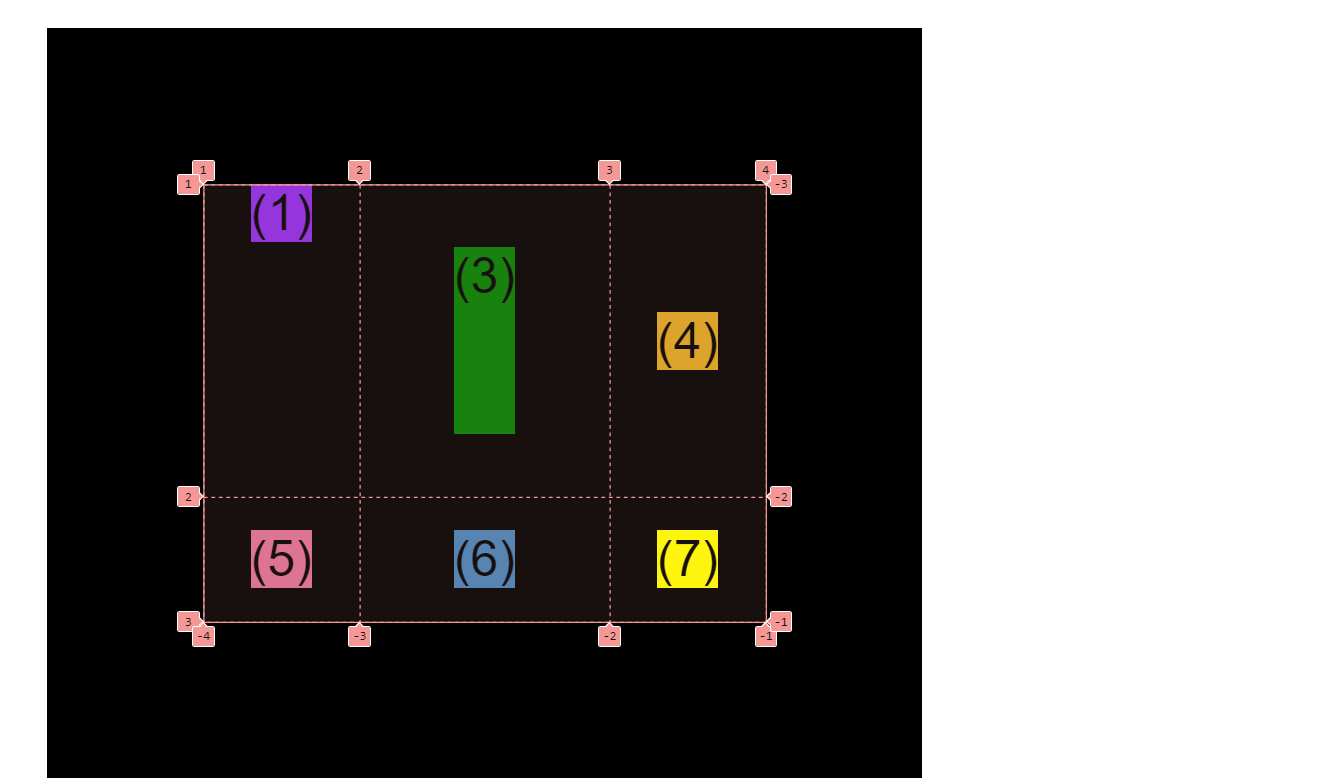


Aligining item individually

 .el--1 {

        align-self: start;

      }



 .el--1 {

        align-self: start;

        justify-self: start;

      }

