PW SKILLS

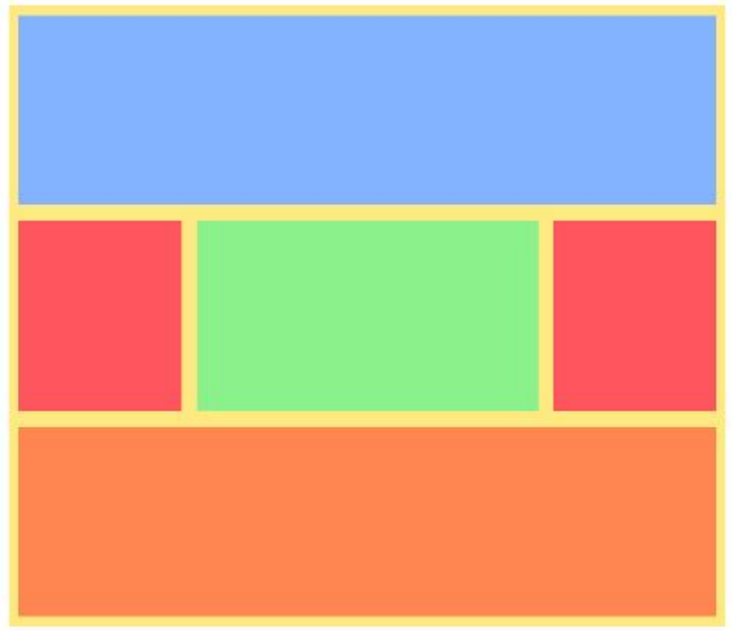
What is grid and Difference Between Grid and Flex

List of content :

1. What is grid
2. Terminologies associated with grid
3. Grid container
4. Example
5. Grid items
6. Example
7. Benefits of grid

What is grid

- An intersection of vertical and horizontal lines is known as a grid.
- Major sections of a page are separated using CSS Grid layout.
- A grid-based layout system with rows and columns is provided by the grid attribute. It eliminates the need for positioning and floating, making web page creation simple.
- The grid layout enables us to construct grid structures represented in CSS rather than HTML.
- The user can align the pieces into rows and columns like a table. But using the CSS grid to construct a layout is simpler than using tables.

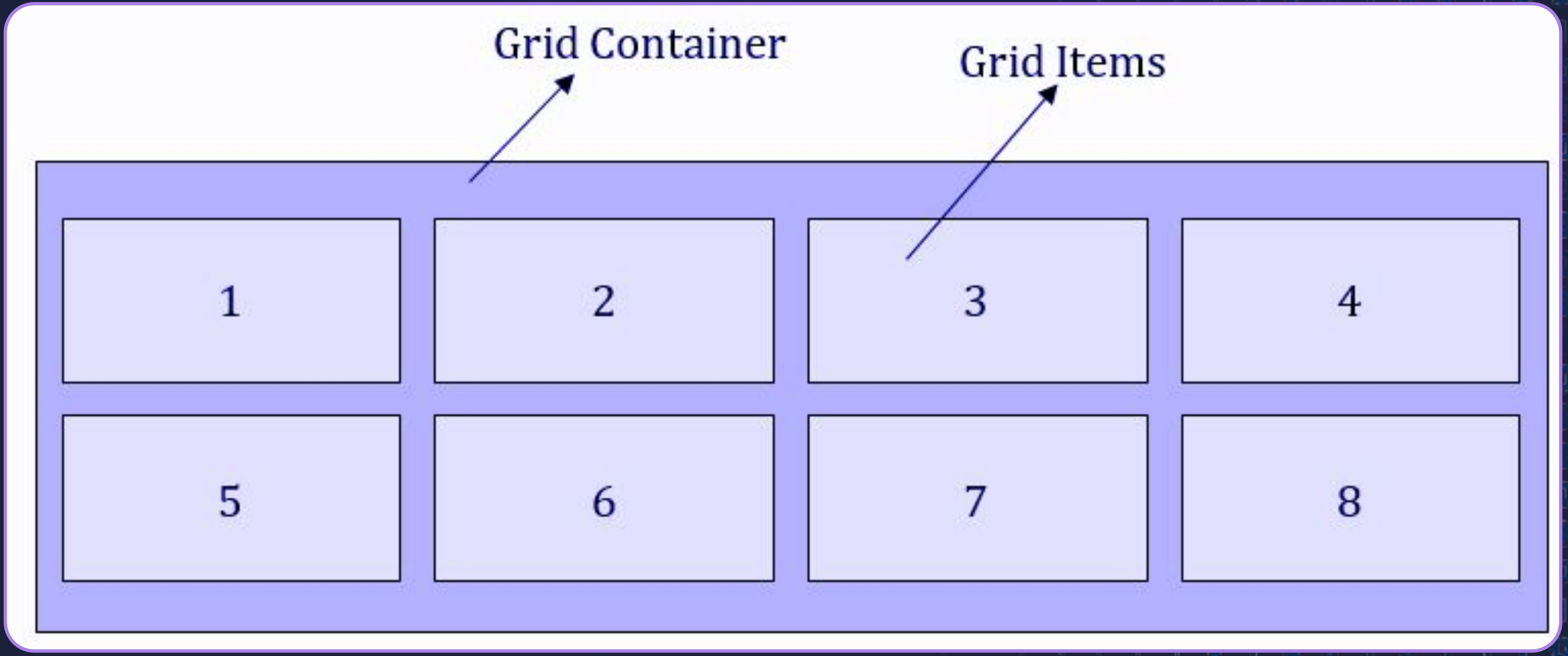


Terminologies associated with Grid:

1. Element
2. Row
3. Column
4. Gap
5. Line

Grid Container:

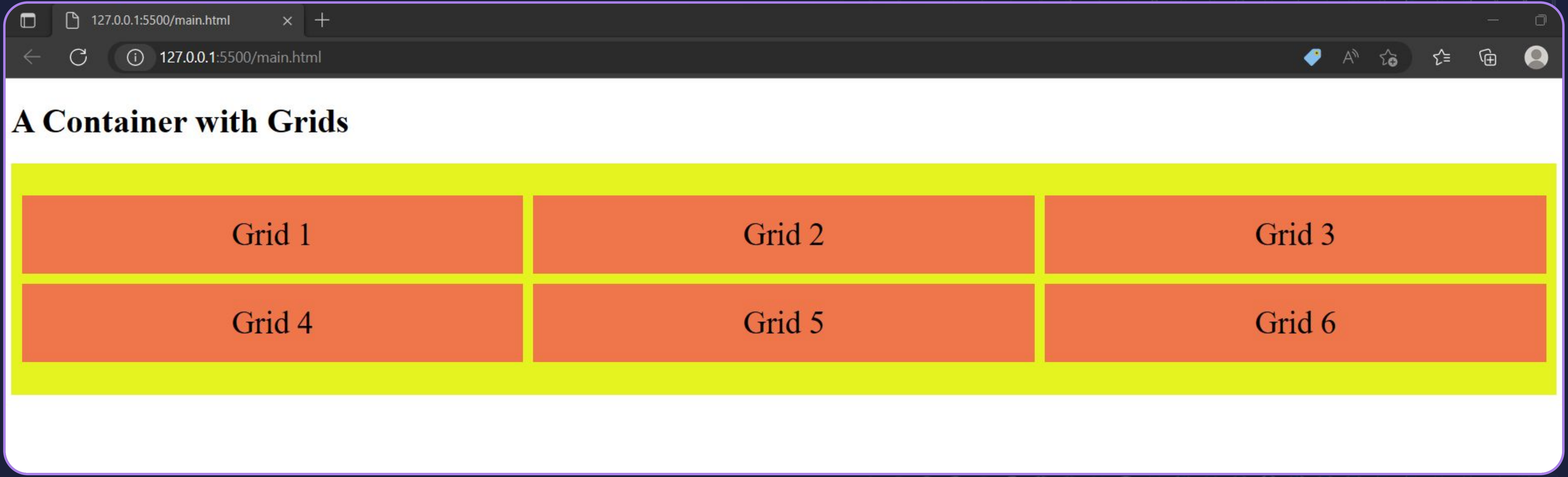
- Grid containers are made up of grid elements that are arranged in columns and rows.
- This is a beautiful and organized arrangement for any webpage.



Properties associated with grid container

- **Grid-template-rows :** It simply states the height of each row.
- **Grid-template-columns :** It states the number of columns in the grid and it can define the width of each column. Here, there is a provision to use the specification as "auto" if all columns should have the same width.
- **Align-content :** vertically aligns the whole grid inside the specified container.It can be start, center, end, space evenly, space around, space between.
- **Justify-content :** it aligns the whole grid inside the container. It can be start, center, space evenly, space around, space between.

Example



A Container with Grids

Grid 1	Grid 2	Grid 3
Grid 4	Grid 5	Grid 6

Grid items

- Grid items are actually the child elements within a container.
- A grid item appears in each column and each row of a container by default, but you can customize the grid items to span several columns and/or rows.

Properties associated with grid items

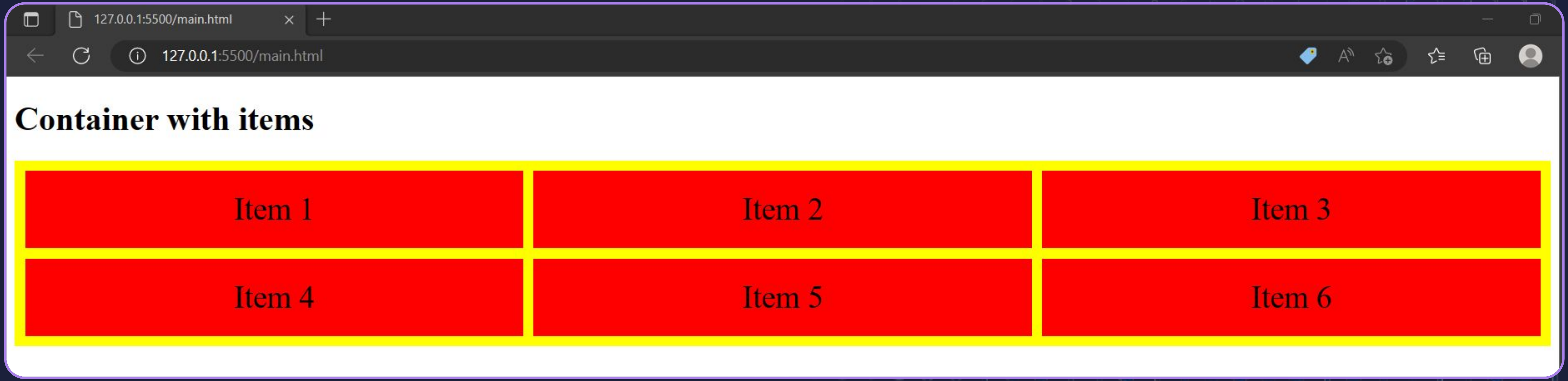
- **Grid-row:** defines on which row to place an item. You can also state here, where the item will start, and where the item will end. To place an item, you can either use line numbers, or use span to define how many columns the item will span.

- **Grid-column:** defines on which column the item is to be placed. Like a row, here too, it can be stated where the item will start and end. Here again, you can either use line numbers, or use span to define how many columns the item will span.

- **Grid-area:** For the attributes grid-row-start, grid-column-start, grid-row-end, and grid-column-end, the grid-area property can be used as a shorthand.

- **Naming grid:** grid-area property can also be used to assign names to grid items.

Example



Container with items

Item 1	Item 2	Item 3
Item 4	Item 5	Item 6

Benefits of grid :

- Grid Layout enables us to clearly distinguish between the visual display of elements and their order in the source.
- Code reduction
  The columns, rows, and grid tracks for your grid are built within your CSS rather than as separate HTML elements.
- Smaller file size
  There is no requirement for huge frameworks like Bootstrap in your projects because CSS Grid is native.
- Developmental pace
  CSS Grid makes prototyping quick and easy after you master the syntax.
- Two-dimensional layouts
  Rows and columns are respected in a two-dimensional grid. If an element's cell cannot accommodate it, the row and/or column will expand to accommodate it. For the layout of pages and forms, a grid is excellent.
- Grids in Nests
  Smaller components like content sections with blog articles or the blog entries themselves can be styled using the CSS Grid Layout.

SKILLS

THANK YOU