**1. Flex box**

1. flex-direction
   * flex-direction: row; /\* Default: Items arranged left-to-right \*/
   * flex-direction: column; /\* Items arranged top-to-bottom \*/
   * flex-direction: row-reverse; /\* Right-to-left \*/
   * flex-direction: column-reverse; /\* Bottom-to-top \*/
2. justify-content
   * justify-content: flex-start; /\* Default: Items start from the left \*/
   * justify-content: flex-end; /\* Items align to the right \*/
   * justify-content: center; /\* Items align in the center \*/
   * justify-content: space-between; /\* Equal space between items \*/
   * justify-content: space-around; /\* Equal space around items \*/
   * justify-content: space-evenly; /\* Equal space between and around items \*/
3. align-items
   * align-items: flex-start; /\* Items align to the top \*/
   * align-items: flex-end; /\* Items align to the bottom \*/
   * align-items: center; /\* Items align in the middle \*/
   * align-items: stretch; /\* Default: Items stretch to fill container \*/
   * align-items: baseline; /\* Align items along their text baselines \*/
4. flex-wrap
   * flex-wrap: nowrap; /\* Default: No wrapping \*/
   * flex-wrap: wrap; /\* Items wrap to the next line \*/
   * flex-wrap: wrap-reverse; /\* Wrapping occurs in reverse \*/
5. align-content
   * align-content: flex-start;
   * align-content: flex-end;
   * align-content: center;
   * align-content: space-between;
   * align-content: space-around;
6. Order : Specifies the order of the flex items:
   * order:4
7. Flex Grow : Specifies how much a flex item will grow relative to others, it occupy extra spaces:
   * flex-grow: 1;
8. flex-shrink : It determines how much a flex item **shrinks** when there isn't enough space in the flex container.
   * flex-shrink: 2; /\* Shrinks faster \*/ flex-shrink: 1; /\* Shrinks normally \*/

**2. Grid Layout**

* Introduction to CSS Grid.
* Grid container, grid items, grid lines.

1. Grid container
   * display: grid;
   * grid-template-columns: repeat(3,1fr) : Defines the number of columns.
   * grid-template-rows: 200px 200px
   * grid-template-columns: repeat(auto-fill, 250px) : For responsive design
   * grid-gap, row-gap and column-gap
2. Grid Item Placement
   * grid-column:1/3
   * grid-row:1/3
   * grid-area: 1/1/1/1
3. justify-content (use for Horizontal)
   * justify-content: start; /\* Default: Items start from the left \*/
   * justify-content: end; /\* Items align to the right \*/
   * justify-content: center; /\* Items align in the center \*/
   * justify-content: space-between; /\* Equal space between items \*/
   * justify-content: space-around; /\* Equal space around items \*/
   * justify-content: space-evenly; /\* Equal space between and around items \*/
4. align-items (use for vertical)
   * align-items: start; /\* Items align to the top of grid box \*/
   * align-items: end; /\* Items align to the bottom of grid box\*/
   * align-items: center; /\* Items align in the middle of grid box \*/
5. align-content (use for Vertical)
   * align-content: start; /\* Items align to the top of parent box\*/
   * align-content: end; /\* Items align to the bottom of parent box\*/
   * align-content: center; /\* Items align to the center of parent box\*/
   * align-content: space-between; /\* Equal space between items in vertical\*/
   * align-content: space-around; /\* Equal space around items in vertical\*/
   * align-content: space-evenly; /\* Equal space between and around items \*/

**🔸 Day 4: Responsive Layouts with Grid + Flexbox**

**Concepts:**

* CSS Grid Basics (grid-template-columns, grid-gap)
* Flexbox deep dive (justify-content, align-items, flex-wrap)
* Layout design tricks

**Hands-on:**  
✔ Convert birthday card layout to responsive using flex/grid  
✔ Build a simple responsive webpage