**CSS FLEXOX**

**(Phumlani Mhlungu Summary)**

* One dimensional model (row or column).

**THE TWO AXIS OF FLEXBOX**

1. The main axis

* Defined by flex-direction, which has four possible values:
  + row
  + row-reverse
  + column
  + column-reverse

1. The cross axis

* runs perpendicular to the main axis, so if your main flex-direction (main axis) is set to row or row-reverse, the cross axis runs down the columns

**THE FLEX CONTAINER**

* display: flex or inline-flex;
* as soon as we do this the following applies:
  + Items display in a row (the flex-direction property’s default is row)
  + The items start from the start edge of the main axis
  + The items do not stretch on the main dimension, but can shrink.
  + The items will stretch to fill the size of the cross axis.
  + The flex-basis property is set to auto.
  + The flex-wrap is set to nowrap.

**MULTI-LINE FLEX CONTAINERS WITH FLEX-WRAP**

display :flex;

flex-wrap: wrap | nowrap | wrapreverse;

**THE FLEX-FLOW SHORTHAND**

* You can combine the two properties flex-direction and flex-wrap into the flex-flow
* flex-flow: column wrap;

**PROPERTIES APPLIED TO ITEMS**

* flex-grow,
  + if set to a positive integer, flex items can grow along the main axis from their flex-basis
  + This will cause the item to stretch and take up any available space on that axis, or a proportion of the available space if other items are allowed to grow too.
* flex-shrink,
* Controls how space is taken away.
* flex-basis, size of the item in terms of the space it leaves as available space
  + initial value is auto
  + if the items don’t have size, flex-bases would be the size of the container

**SHORTHAND VALUES FOR THE PROPERTIES**

Display: flex;

Flex: flex-grow, flex-shrink, flex-basis;

i.e flex: 1 1 70%;

**ALIGNMENT, JUSTIFICATION AND DISTRIBUTION OF SPACE BETWEEN ITEMS**

* Align-items, will align items on the cross axis
  + Stretch, default
  + Flex-start
  + Flex-end
  + Center
* Check align-self and align-content as well
* Justify-content
  + Flex-start
  + Flex-end
  + Center
  + Space-around
  + Space-between
  + Space-evenly

You can also change order of items too:

.container{

Display:flex;

}

Box1{

Order :2;

}

Box 2{

Order: 3;

}

Box3{

Order:1;

}

**EVEN-GRID-ISH, CONTENT AND SIDEBAR**

Check pdf “USEFUL FLEXBOX PATTERNS”