

CSS



CSS

Advanced styling and Animations

COS216
AVINASH SINGH
DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF PRETORIA

CSS – OVERVIEW

- CSS has two main components
- Styling rules
 - Details on **how** to style elements
- Selectors
 - Details on **which** elements to style



CSS – BACKGROUND

- Backgrounds can be set using
 - Colors (RGB, HSL, HEX, Name)
 - Images (URL)
- Backgrounds can be
 - Repeated
 - Positioned
 - Resized

CSS Syntax

```
background: bg-color bg-image position/bg-size bg-repeat bg-origin bg-clip bg-attachment initial|inherit;
```

CSS – BACKGROUND

```
body
{
    background: #FF44AA url("back.png") no-repeat right top;
    background: red; /* set any parameter */
    background-color: red; /* set a color */
    background-image: url("images/back.jpg"); /* set an image */

    background-repeat: no-repeat; /* do not repeat */
    background-repeat: repeat-x; /* repeat x-axis */
    background-repeat: repeat-y; /* repeat y-axis */

    background-size: auto; /* use image resolution */
    background-size: 100%; /* resize to 100% of parent */
    background-size: 100px 200px; /* resize width and height of image */

    background-position: center; /* center on x-axis and y-axis */
    background-position: center top; /* center on x-axis and top on y-axis */
}
```

CSS – SHADOWS

- Add shadows/glows on the outside of an element
- Shadows are determined by
 - Horizontal offset (required 1st parameter)
 - Vertical offset (required 2nd parameter)
 - Blur radius (optional 3rd parameter)
 - Spread radius (optional 4th parameter)
 - Color (optional 5th parameter)

```
div {  
    width: 100px;  
    height: 100px;  
    border: 1px;  
    box-shadow: 10px 10px 8px #1144AA;  
}
```



CSS – CURSORS

- Set the mouse cursor of an element
 - Pointer
 - Crosshair
 - Wait
 - Zoom in/out
 - Many more

```
div:hover  
{  
    cursor: pointer;  
}
```

CSS – Z-INDEX

- The stack order of elements
- Is used to stack one element on top or below another one
- Requires the element to be positioned (relative, absolute, fixed)
- Higher values will move to the top, lower values moves to the bottom

```
/* The tree will show in front of the house */
#tree
{
    z-index: 10;
}

#house
{
    z-index: 5;
}
```

CSS – OPACITY

- Make elements transparent by setting the opacity
- Similar to the alpha channels in `rgba()` colors, but can be applied to any elements/attributes/images not only the color
- In range [0,1]
- 0.0 is fully transparent, 1.0 is fully opaque

```
/* Slightly transparent div */  
div  
{  
    opacity: 0.9;  
}
```

CSS – COLOR

- Sets the text color

```
p {  
    color: rgb(142, 256, 230);  
    color: #A4B5C0;  
    color: red;  
}
```

CSS – TEXT ALIGNMENT

- Align text
 - Left
 - Right
 - Center
 - Justify
- The property can be used to align other elements in some cases

```
p  
{  
    text-align: right;  
    text-align: justify;  
}
```

CSS – VERTICAL ALIGNMENT

- Vertical align elements
 - Top
 - Bottom
 - Middle
- Typically has to be applied on the parent (eg: vertical align property on a table cell so that its content is vertically aligned)

```
td  
{  
    vertical-align: middle;  
}
```

CSS – FONTS

- Set the font of text
 - Style (normal, italic, oblique)
 - Weight (normal, bold, bolder, lighter, custom boldness)
 - Size (size of font)
 - Family (the font type/family, names with spaces should be put in quotes)

CSS – WEB SAFE FONTS

- Browsers and OSs don't have all fonts installed
- Some default fonts are available in all browsers, called web safe fonts
 - Georgia
 - Times New Roman
 - Arial
 - Comic Sans
 - Lucida
 - Tahoma
 - Verdana
 - Courier
 - More ...

CSS – FONTS

```
p {  
    font: 15px arial, sans-serif;  
    font: italic bold 12px Georgia, serif;  
  
    font-family: "Courier New", Courier, monospace;  
    font-family: Tahoma, Geneva, sans-serif;  
  
    font-size: 12px;  
    font-size: 5em;  
  
    font-weight: bold;  
    font-weight: 900;  
  
    font-style: normal;  
    font-style: italic;  
}
```

CSS – CUSTOM FONTS

- If a font is not installed, you can provide a custom font file
- Certain font formats are well supported (TTF, OTF, WOFF)
- Others are not widely supported (WOFF2, EOT, SVG)
- Multiple formats can be provided, browsers will pick the one they support

CSS – CUSTOM FONTS

```
/* Link to a font file on the server */
@font-face
{
    font-family: MyCustomFont;
    src: url(fonts/customfont.ttf);
}

/* Use the custom font */
div
{
    font-family: MyCustomFont;
}
```

CSS – ANIMATIONS

- Since CSS3 various animations are possible
- Before CSS3, animations had to be done in JavaScript
 - Create a HTML element
 - Use a JS timer (`setInterval`) and update/move/resize the element every few milliseconds
- Or a low-resolution animated image (GIF, APNG) had to be used
- This can now be done more easily and efficiently in CSS

CSS – ANIMATIONS

- Examples

CSS – ANIMATIONS

- Animations typically consist of
 - Timer to execute changes in intervals
 - Keyframes to determine the transitions
 - CSS properties to specify the type of change
- Keyframes are a predefined set of frames to transition between and can be reused by different elements
 - Example: move element from position X to position Y
 - Example: change from color A to color B
 - Example: rotate 360 degrees in 5 degree steps

CSS – ANIMATIONS

- The general syntax of animations with optional parameters is as follows

```
animation: name duration timing-function delay iteration-count direction fill-mode play-state;
```

CSS – ANIMATIONS

- Animate a box to move horizontally from left to write over 5 seconds, if the box is at the right-hand side it will respawn at the left-hand side

```
@keyframes horizontalMove
{
    from {left: 0px;}
    to {left: 500px;}
}

div
{
    width: 50px;
    height: 50px;
    background: black;
    position: relative;
    animation: horizontalMove 5s infinite;
}
```

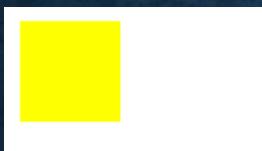
CSS – ANIMATIONS

- To continuously move from left-to-right and the right-to-left

```
/* Move from left-to-right, respwan on the  
left, and move from left-to-right again */  
@keyframes horizontalMove1  
{  
    from {left: 0px;}  
    to {left: 500px;}  
}  
  
/* Move from left-to-right, then in the  
opposite direction from right-to-left */  
@keyframes horizontalMove2  
{  
    0% {left: 0px;}  
    50% {left: 500px;}  
    100% {left: 0px;}  
}
```

CSS – ANIMATIONS

- Change colors in a rainbow fashion (blue » green » yellow » red)

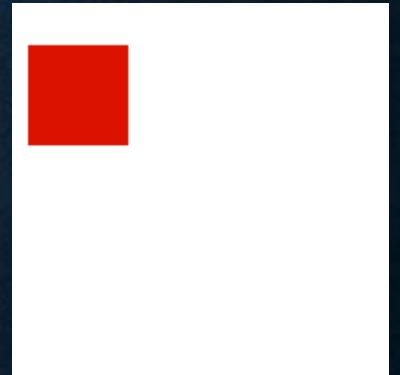


```
@keyframes colorChanger
{
    0%   {background-color: blue;}
    33%  {background-color: green;}
    66%  {background-color: yellow;}
    100% {background-color: red;}
}
```

CSS – ANIMATIONS

- Change multiple attributes at the same time

```
@keyframes multiChanger
{
    0%   {top: 0px; left: 0px; background: red;}
    25%  {top: 0px; left: 100px; background: blue;}
    50%  {top: 100px; left: 100px; background: yellow;}
    75%  {top: 100px; left: 0px; background: green;}
    100% {top: 0px; left: 0px; background: red;}
}
```



CSS – ANIMATIONS

- Various animation options are available
 - **animation-delay**: wait before starting an animation
 - **animation-direction**: play the animation forward, backward, or alternate
 - **animation-duration**: how long one cycle takes in seconds or milliseconds
 - **animation-fill-mode**: style for the element if the animation is not running
 - **animation-iteration-count**: the number of times (or infinite) to execute the animation
 - **animation-name**: the name for the animation
 - **animation-play-state**: whether the animation is playing or paused
 - **animation-timing-function**: the speed curve/function to change from one style to the next

CSS – MEDIA QUERIES

- What about mobile design? Different device sizes? How do we make a website more responsive?
- The `@media` rule was created to allow CSS for specific device sizes
- Media queries can be used to determine viewport width and height, orientation and resolution

```
@media not|only mediatype and (expressions) {  
    ...  
}
```

CSS – MEDIA QUERIES

```
/* Set the background color of body to tan */  
body {  
    background-color: tan;  
}  
  
/* On screens that are 992px or less, set the  
background color to blue */  
@media screen and (max-width: 992px) {  
    body {  
        background-color: blue;  
    }  
}  
  
/* On screens that are 600px or less, set the  
background color to olive */  
@media screen and (max-width: 600px) {  
    body {  
        background-color: olive;  
    }  
}
```

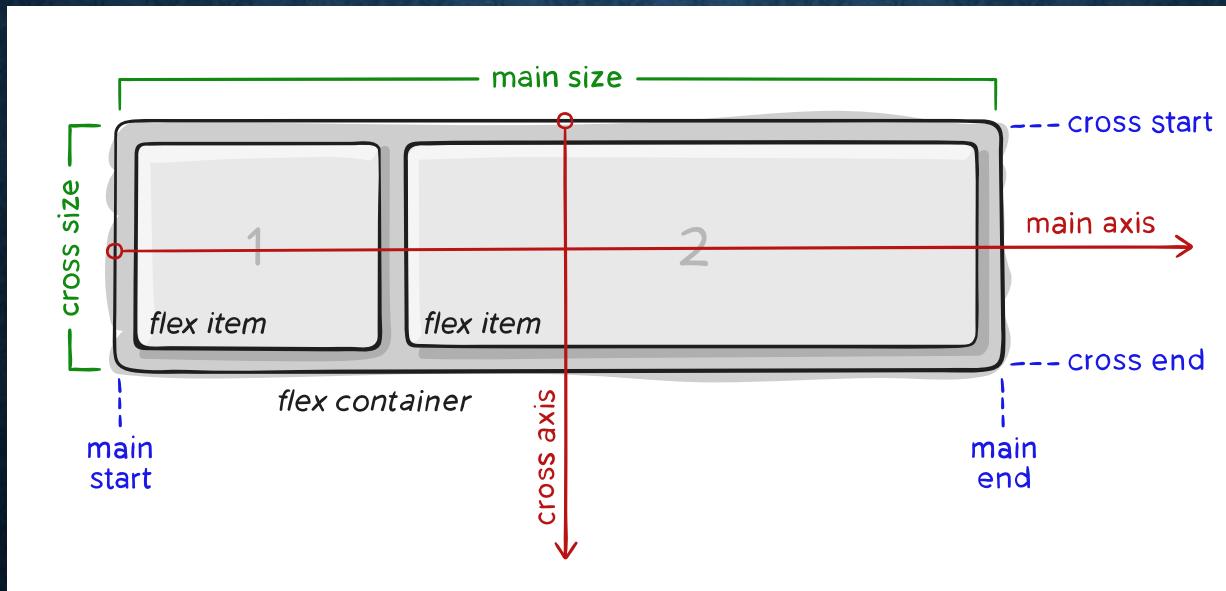


CSS – MEDIA QUERIES

- https://www.w3schools.com/css/css3_mediaqueries.asp
- https://www.w3schools.com/css/css3_mediaqueries_ex.asp

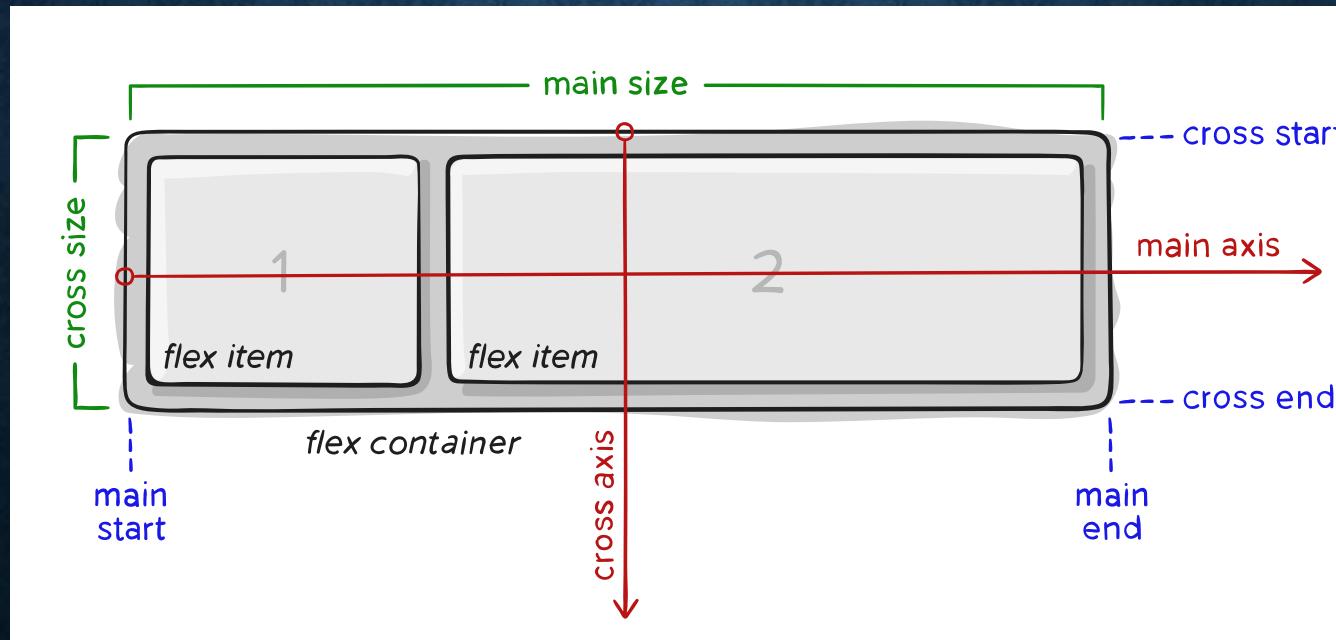
CSS – FLEXBOX MODULE

- Flexbox was introduced in CSS3 in 2017 that provided easier ways to align content
- Easier responsive design
- Ability for elements to be in a container for better design and grouping



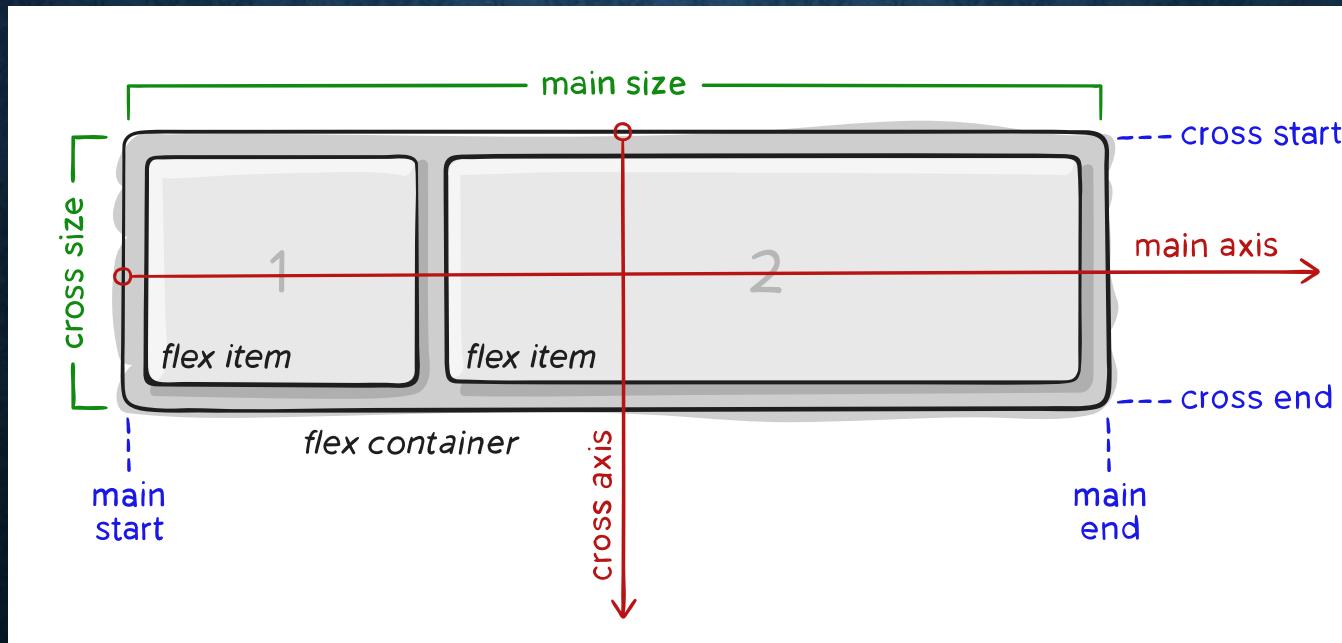
CSS – FLEXBOX MODULE

- main axis – provides the flex container with the axis / direct.
- main-start | main-end – The start and end positions of the flex container.
- main size – defines the height or width depending on the axis.

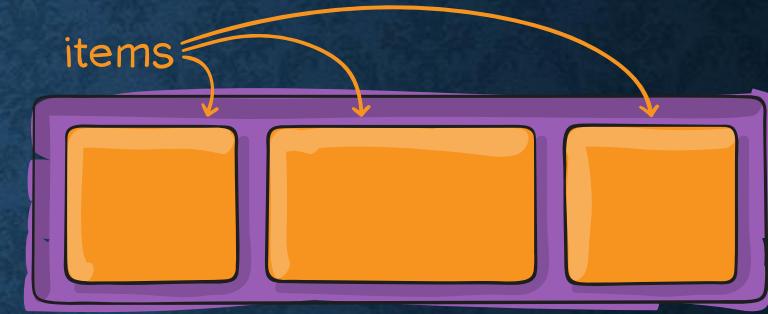
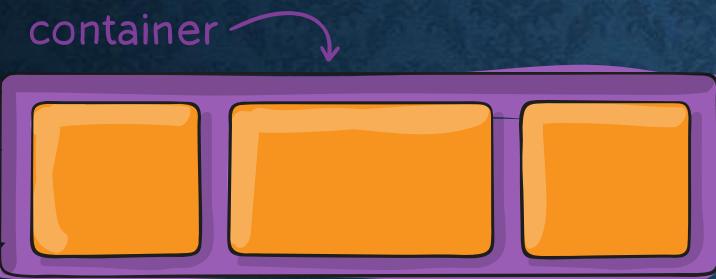


CSS – FLEXBOX MODULE

- cross axis – an axis perpendicular to the main axis.
- cross size – the height or width depending on the cross axis.



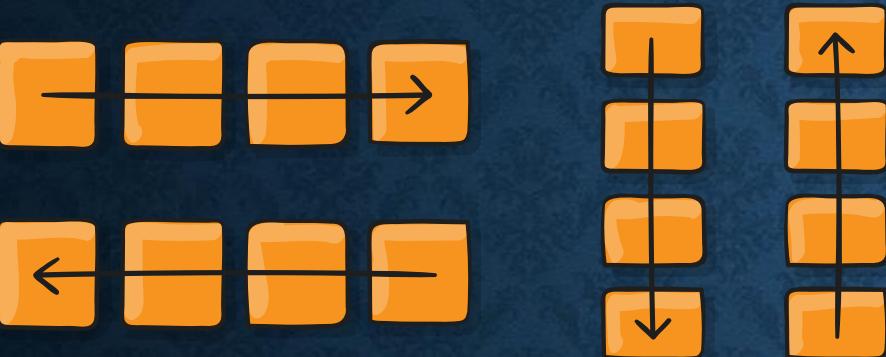
CSS – FLEXBOX



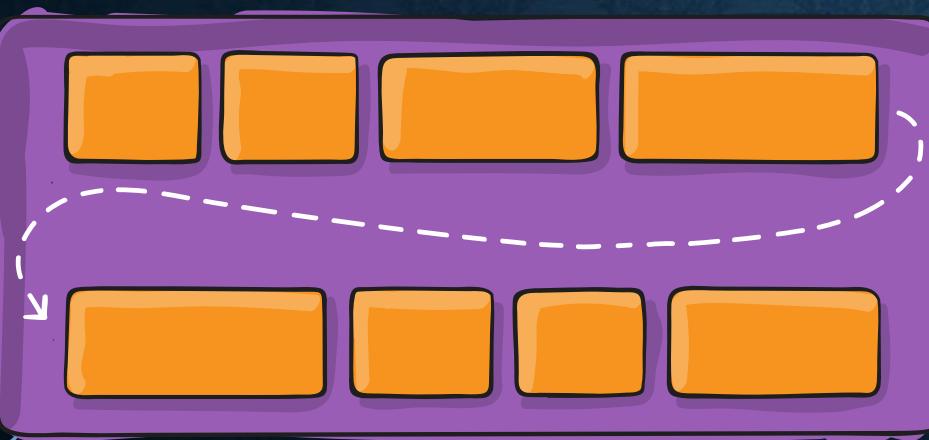
```
.container  
{  
    display: flex; /* or inline-flex */  
}
```

```
.item {  
    order: 5; /* default is 0 */  
}
```

CSS – FLEXBOX



```
.container {  
    flex-direction: row | row-reverse | column | column-reverse;  
}
```



```
.container {  
    flex-wrap: nowrap | wrap | wrap-reverse;  
}
```

CSS – FLEXBOX

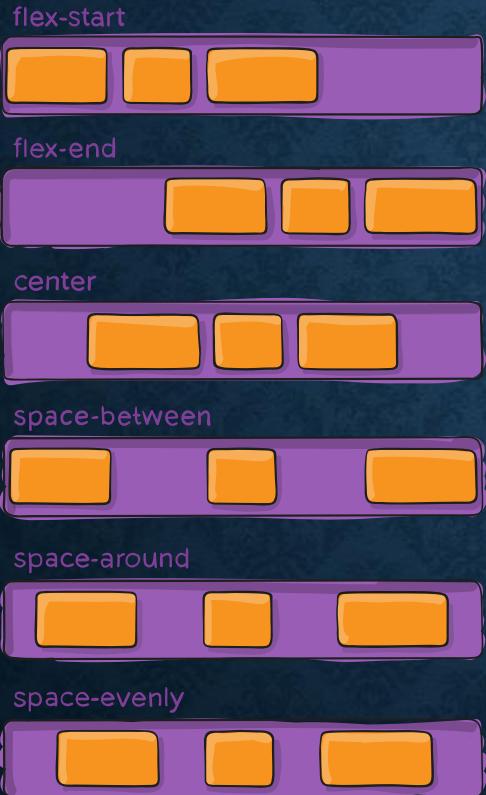


```
.item {  
    flex-grow: 4; /* default 0 */  
}  
  
.item {  
    flex-shrink: 4; /* default 1 */  
}
```



```
.item {  
    order: 5; /* default is 0 */  
}
```

CSS – FLEXBOX



```
.container {  
    justify-content: flex-start | flex-end | center |  
    space-between | space-around | space-evenly |  
    start | end | left | right ... + safe | unsafe;  
}
```

CSS – FLEXBOX GUIDE

- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

CLICK

CSS – ANIMATIONS

Css3 Spinners

