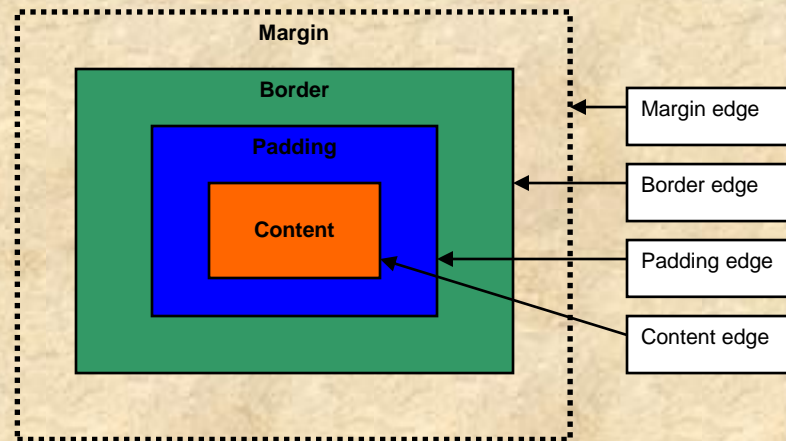

Additional HTML Elements

- `<div>` and `` -
 - Allow you to delimit a section of the HTML body
- ``
 - Used to wrap inline content (e.g., text sequence)
- You can apply style to the sections defined by `span` and `div`.
- **Example:** `spanDiv.html`, `spanDiv.css`

Box Model

- Each block element (e.g., p) contains four edges (top, bottom, right, and left) defining a box
- Four sections can be identified with a block element
 - Content – what lies in the middle of the box (text, image, etc.)
 - Padding – surrounds the content
 - Border – surrounds the padding and represents the box border
 - Margin – surrounds the border



<div> is a block element

- <div> Defines a block-level entity
 - Browser starts a div element's content on its own line
- <body> also defines a block-level entity
- You can use your box model knowledge to add more style to your pages.
- **Example:** boxModel1.html, boxModel1.css

Box Model

- Let's explore more of the box model with the following examples
- **Example:** boxModel2.html, boxModel2.css
- **Example:** padding.html, padding.css
- The margins, borders, padding, and background properties of block elements (e.g., body, p, etc.) are not passed to its child block-level elements

Setting Size

- Percentages – size of the font is based on the size of the parent element
- Length units
 - centimeters (cm)
 - millimeters (mm)
 - points (pt) - 1 pt → 1/72 inch
 - picas(pc) - 1pica → 12 pts
 - inches (in)
- Relative
 - ex – height of the lowercase x in the font
 - px – pixels
 - em – refers to font size of parent element
 - Example: 2 em → twice the font size of the parent element
 - If the parent is body tag and no font size is specified then the size is looked in the user's preferences specified in the browser
 - Allow you to define scalable style sheets

Shorthand Property

- Shorthand Property- allows you to specify several properties by using only one.
- If you don't specify one of the properties a default value will be used.
- Commonly used shorthand properties
 - background
 - font
 - list-style
 - margin
 - border
 - padding
- **Example:** noShorthandProp.html, noShorthandProp.css, shorthandProp.html, shorthandProp.css

Font-size Keywords

- xx-small
- x-small
- small
- medium
- large
- x-large
- xx-large
- larger
- smaller
- **Example:** fontsizeKeywords.html, fontsizeKeywords.css

Background

- Background properties
 - background-color
 - background-image – location of image
 - background-repeat – how image repeats. Possible values
 - no-repeat – one instance of the image
 - repeat – tile
 - repeat –y → repeats on the y-axis
 - repeat –x → repeats on the x-axis
 - background-attachment – indicates attachment of the image to the containing element. Possible values are:
 - scroll → default value.
 - fixed → image will stay stationary as the scrolling takes place
 - background-position – Possible values (combination of are valid)
 - top, bottom, center, left, right
- Background images can be used in elements other than body
- **Example:** background.html, background.css
- Shorthand property: backgroundShorthand.html, backgroundShorthand.css

Generic Font Families

- sans-serif – (e.g., Verdana, Helvetica, Arial)
- serif – (e.g., Times New Roman, Georgia, Times)
- monospace – (e.g., Courier, MS Courier New)
- cursive – (e.g., Lucida Handwriting)
- fantasy – (e.g., Whimsey, Comic Sans)

JavaScript

- JavaScript – programming language that can appear in html pages.
- It allow us to:
 - To dynamically create web pages
 - To control a browser application
 - Open and create new browser windows
 - Download and display contents of any URL
 - To interact with the user
 - Ability to interact with HTML forms
 - You can process values provided by checkbox, text, textarea buttons
- **Example:** SqrTable.html

Execution of JavaScript Programs

- HTML parser – Takes care of processing an html document
- JavaScript interpreter – Takes care of processing JavaScript code
- HTML parser – must stop processing an html file when JavaScript code is found (JavaScript interpreter will then be running)
 - This implies a page with JavaScript code that is computationally intensive can take a long time to load
- **Example:** Template – TemplateJS.html

JavaScript

- Unlike html, JavaScript is a case-sensitive language
- JavaScript relies on the Unicode character set
- Let's go over several basic constructs that allow us to define JavaScript programs.
- Some definitions
 - string – Any set of characters in double quotes (“ “)
 - function/method – An entity that completes a particular task for us. It can takes values necessary to complete the particular task and it can return values.
- Generating output with the document.writeln method
 - Allow us to add text to the html file (see **Example:** WriteIn.html) by providing the required text in “ “
 - You can specify html code and results of JavaScript constructs

JavaScript (Output)

- **Example:Table.html**
 - Illustrates how we can create a table using document.writeln
 - Notice how we can use the Date() to specify a particular date format. Date() is part of JavaScript and it is a method.
 - The + allow us to concatenate strings
 - Example: “Mary” + “Land” → “MaryLand”
 - Example: “Time is: “ + new Date()
 - Notice how we have specified the border size. If you use “ “ then the table borders will not be generated. You need to use single quotes.
 - Keep in mind that this example could have been written without using JavaScript. However you will see how by extending code similar to the one provided you can dynamically decide what your final html will look like.

JavaScript (Variables)

- Variable – A memory location that can store a value. In JavaScript variables are declared using **var**
var temperature;
- Variables names must start with a letter, underscore or dollar sign and can be followed by any number of letters, underscores, dollar signs or digits.
- Variables must be declared before they are used.
- A variable can hold different type of values
- Values we can assign to variables
 - Integer – 0, 10, 40, 6, -7
 - Floating-point – 3.2, .67, 1.48E-20
 - String literals – “hello”, “goodbye”
- Operators
- Assignment operator (=)
 - Typical arithmetic operators (+, -, *, /)
- **Example:** Variables.html