Drag and Drop Shopping Cart – HTML5

**Day 1 : How the web page functions**

Try out the following:

1. Click on the each image of the jewelry. The image should open in a window of its own and be larger. Note that a description of the jewelry is included along with its price. I used the Paint program to add text to the enlarged image and saved it under the same name but added the letter b to the file name.

2. There is a roll over image. The coupon for 10%. The idea is that if you roll over the image, you will see a special promotional code that can be used later to give the buyer a 10% discount.

3. Each image is drag-able into the shopping cart area. The yellow box is the cart.

4. When item is dropped into the yellow box, a description appears in the white box below. Check boxes are marked next to the item and the price. In the orange box, Subtotal, sales tax, shipping and total are automatically calculated.

5. HTML5 has the capability of storing information when someone visits a web site. This feature was used to make a wish list. Delete the line that says (Enter the things you would like to purchase later). Key in the information on one of the items that you might want to purchase. Exit the page. Reload it and your item will still e listed. Trying clicking on the RESET Note link to clear the box.

6. When you hit the submit button all the information inside the form tags is sent to the email program.

6. The beginning form tag is <form name="productsForm" action="mailTo:jerrybelch@cox.net?cc=jerrybelch@hotmail.com&subject=Midas Touch Order Form" enctype="text/plain" method="post">

7. This example first names the form- productsForm. It includes all the information between the beginning and ending form tags.

8. The “action” word tells the browser to email the information between form tags to jerrybelch@cox.net. Here is where you substitute your company's sales department's email address.

9. The second part of the action that begins with a question make says to send a copy of this order to jerrybelch@hotmail.com. You would substitute the email address of your accounting department VP here so that they can also see the order,

10. The information following the & mark delineates the subject of the form. In this case it tis the company name and order form.

11 The encrypt type = text/plain tells the browser how to send the information. Without this tag, the whole form's contents are sent in a single string which makes it very difficult to read.

12. Post tells the browser to add this information to the body of an email. Without it the email is blank.

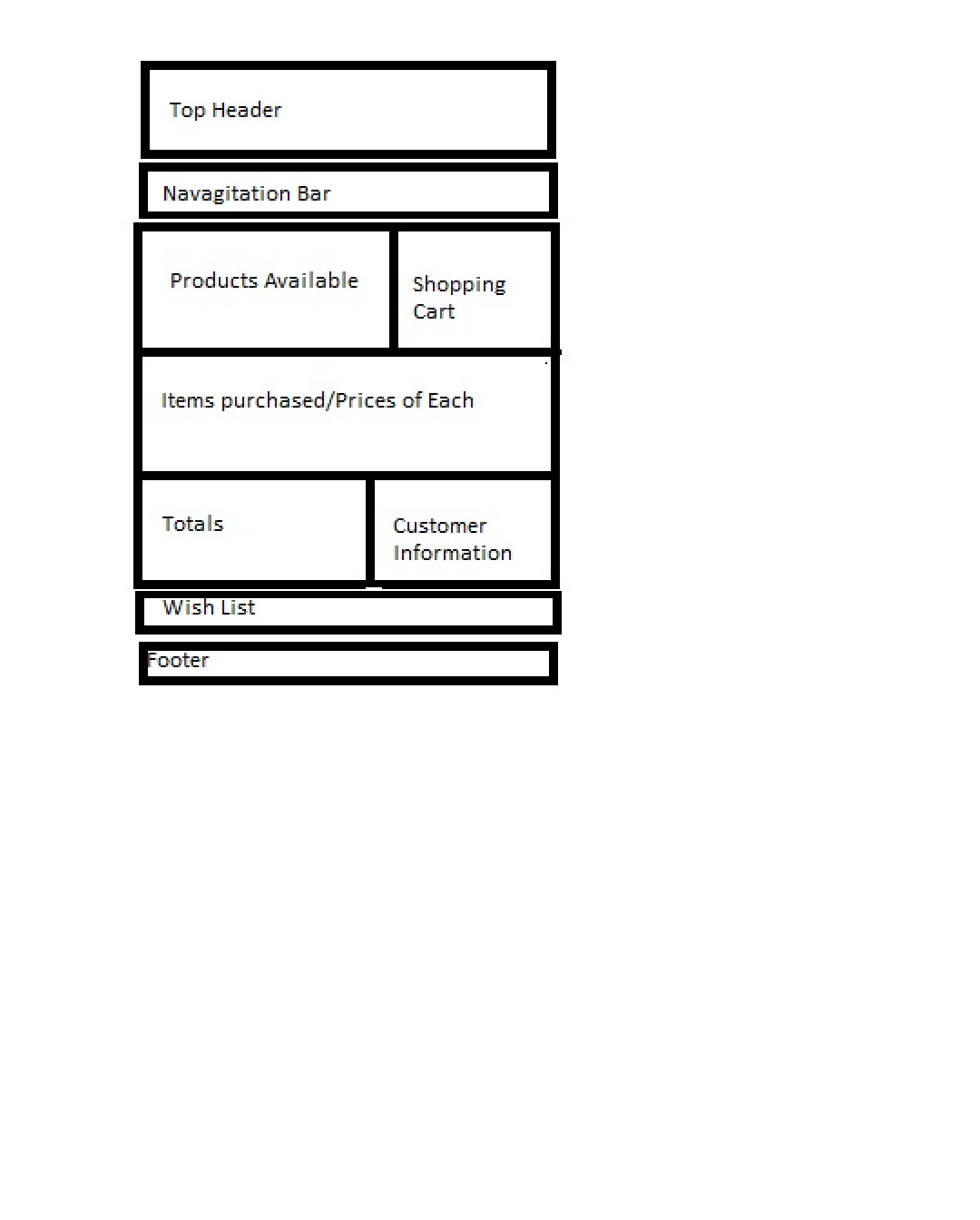
13. Watch where the quotation marks go.

[Shopping Cart Finished](http://janetbelch.com/DragAndDrop/ShoppingCartKeyFinished.html)

**Day 2: Web Page Elements - laying out the page**

We need to layout on a piece of paper how we want our shopping cart page to look like.

For this exercise we will use the drawing seen below.



We will be using the flexible box model for this exercise.

The flexbox model is a system which helps us layout our web page.

It creates a flexible layout for us in which the browser does all the calculations for us as to height, width, and placement.

White space is distributed equally between child elements.

The flexbox model is especially useful when working with different screen sizes like tablets and call phones.

Below you can see the html file template we will be working with.

!doctype html>

<html lang="en">

<head>

<meta charset="utf-8"/>

<title></title>

<!--

<link rel="stylesheet" href="parent.css"/>

-->

</head>

<body onLoad="initialize();">

<form name="productsForm" action="mailTo:jerrybelch@cox.net?cc=jerrybelch@hotmail.com&subject=Midas Touch Order Form" enctype="text/plain" method="post">

<div id="outside\_wrapper">

<Header id="top\_header">

<H1>Company Name</H1>

</Header>

<Nav id="top\_menu">

<ul>

<li><A href="Home.html">Home</A>

<li><A href="About.html">About</A>

<li><A href="Products.html">Our Products</A>

<li><A href="Contact.html">Contact Us</a>

</ul>

</Nav>

<section id="parent">

<article id="kid1">Products For Sale

<br>

</article>

<article id="kid2">Shopping Cart

<br>

</article> <!-- end of article 2 shopping cart -->

</section>

<section id="otherStuff">

<article id="kid3">Items Purchased/Prices ofItems

<br>

</article> <!-- end article 3-->

</section> <!-- otherStuff end-->

<section id="lastOne">

<article id="kid4">Totals

<br>

</article><!-- end of article 4 - totals-->

<article id="kid5">Customer Information

<br>

</article> <!-- end of kid 5 -->

</section>

</form>

<div id = "note" contenteditable = "true">

(Enter the things you would like to purchase later)

</br>

</section> <!-- Ends wish list or note section-->

<a href="#" onclick="reset\_note();return false">Reset Note</A>

<Footer id="the\_footer">

Copyright 2014

</Footer>

</body>

</html>

The very top tag tells the browsers that this is an html5 file

The second line tells the browsers to use English as the language

The third line is the beginning head tag. Inside the head tags you will place calls to other files, search engine key words, etc

The fourth line defines the character set to be used.

Other character sets include Latin, Greek, Chinese, etc.

The fifth line is where you will place the title of your document

Inside the title tags is what shows up on the browser tab and is useful in searching for your document by search engines.

The next line is a comment tag. It is put here to make the browser ignore the next line for now.

The next line links this html document to a cascading style sheet file called parent.css.

The next line is the ending comment tag. Which tells the browser to stop ignoring lines.

The css file formats the layout tags like headlines, section and article tags.

We will change these lines and remove the comment tags <!-- --> to link this html page to a cascading style sheet file and javaScript file.

Let's start making some changes to the html file.

First we need to get a copy of the file

Highlight the text in the box above by dragging the whole page

Press CTRL + C to put a copy on the clipboard.

Open up Notepad++. Press Window Key + R. Key in Notepad++ as the program to run. Click OK.

Select File

Select New

Press CTRL V to paste the document into Notepad++

Select location for the file

File - Save As and give the file a name.

Make sure that you save the file type as html

Making Changes to the HTML file

We need to supply the title of our page. We want it to be descriptive since it is a part of what search engines look for.

The title appears on the browser's tab so we want it to be distinctive.

Since we have a jewelry store, I selected the name The Midas Touch.

In Greek mythology, everything King Midas touched turned to gold.

Key in "The Midas Touch" in between the <title>The Midas Touch</title>

Save your file then select run and launch in Chrome to see your result.

You should see the title of your page, Midas Touch, located on the browser tab.

Without a title, the browser tab prints the name of the html page.

Without the link to the css file, here is what your screen should look like this.

Company Name

Home

About>

Our Products

Contact Us

Products For Sale

Shopping Cart

Items Purchased/Prices of Items

Totals

Customer Information

(Enter the things you would like to purchase later)

Reset Note

Copyright 2014

Before making any more changes it would be a good idea to explain some of the rest of the code on this page.

The ending head tag, &#60;head/&#62; is next. This ends the area where we call different files to help format and make our page interactive.

The beginning body tag is next and it contains an onLoad function called initialize. More on the function later,

The beginning form tag is next. It is extremely important, since it contains the information to send the items purchased by our customers to an email address.

The name of the form is listed first, productsForm.

The action we want it to take when the submit button is pressed is to mail the information contained between the form tags to jerrybelch@hotmail.com

In addition a copy is sent to another email account. I put in two different email accounts so that the information is sent to both. The first email should be your company email address, one sent to sales and the other to accounting.

The subject is also listed in this line so that when the email is received, you can see that it is and order for merchandise.

The method used is post.

Change the email addresses to ones in your company

**Layout tags**

The tag called <div id=outside\_wrapper"> surrounds the entire web page we are creating.

You can't see it yet, because we have not formatted it. We will use a border color so we can see where it is.

This is where we need a cascading style sheet file

Listed below is a portion of parent.css file that we will be making.

\*{

margin:0px;

padding: 0px;

}

h1{

font: bold 25px Tahoma;

}

h2{

font: bold 20px Tahoma;

}

h3{

display:inline-block;

font-size:15px;

padding:15px;

color:white;

margin-top:.5in;

}

header,section, footer, aside, nav, article, hgroup {

display:block;

}

#parent{

border:2px solid blue;

background:pink;

display:-webkit-box;

display:-mozkit-box;

display:-ms-flex;

-webkit-box-orient:horizontal;

//-mozkit-box-orient:horizontal;

//-msflex-orient:horizontal;

-webkit-box-direction:normal;

-webkit-box-pack:center;

-webkit-box-align:center;

-webkit-box-flex:1;

}

body{

width : 100%;

display:-webkit-box;

display:-mozkit-box;

display:-ms-flex;

-webkit-box-pack:center;

border-color:darkgreen;

border-width:5px;

border-style:ridge;

}

#outside\_wrapper{

max-width:1000px;

margin: 20px 0px;

display:-webkit-box;

border:5px solid aqua;

-webkit-box-orient:vertical;

}

#top\_header{

//background:rgb(255,228,225);

background:#FFE4E1;

//background:mistyrose;

border:3px solid black;

padding:20px;

}

#top\_menu{

border: 3px solid orange;

//background:#8fbc8f;

//background:darkseagreen;

background:rgb(143,188,143);

}

#top\_menu li{

display:inline-block;

list-style:none;

padding:5px;

font:bold 14px tahoma;

}

a:link{color:#FF0000;} /\* unvisited link red \*/

a:visited{color:#00ff00;} /\*visited link lime\*/

a:hover{color:#fffafa;} /\* mouse over link snow\*/

a:active{color:#0000ff;} /\*selected link blue\*/

#kid1{

border: 2px soild red;

background:green;

padding:20px;

margin:10px;

-webkit-box-flex:1;

//-mozkit-box-flex:1;

}

#kid2{

border: 2px soild red;

background:yellow;

padding:20px;

margin:10px;

width:220px;

}

#kid3{

border: 2px solid white;

background:red;

padding:20px;

margin:10px;

-webkit-box-flex:1;

//-mozkit-box-flex:1;

height:100px;

}

#kid4{

border: 2px solid red;

background:orange;

padding:20px;

margin:10px;

-webkit-box-flex:1;

//-mozkit-box-flex:1;

height:100px;

}

#kid5{

border: 2px solid red;

background:gray;

padding:20px;

margin:10px;

}

#otherStuff{

display:-webkit-box;

-webkit-box-orient:horizontal;

background:Orange;

}

#lastOne{

display:-webkit-box;

-webkit-box-orient:horizontal;

background:brown;

}

#note{

display:-webkit-box;

-webkit-box-orient:horizontal;

background:fuchsia;

border: 5px solid silver;

}

#the\_footer{

text-align: center;

padding:20px;

background:gray;

border-Top:5px solid navy;

}

Here is what the file looks like before the link to the cascading style sheet file.

<A href="ShoppingCartKey2.html">Web page without formatting file linked.</A>

[Web page without formatting](http://janetbelch.com/DragAndDrop/ShoppingCartKeyFinished.html)

Here is what the file looks like with the link to the css file.

<A href="ShoppingCartKey.html">Web page with formatting file linked.</A>

[Web page with formatting file linked](http://janetbelch.com/DragAndDrop/ShoppingCartKey.html)

The lines listed above from the css file reset the margins and padding to zero

Margins are spaces outside the border

padding are the spaces inside the border.

All HTML elements, like paragraphs and headings have default margins. Different browsers have different margins

To make our web site look good on all browsers we want to reset them to zero

Later we can specify what we want them to be.

You will note that the code for each item is enclosed in curly braces {}

Next list the item you want to set parameters for, like margins, background colors, padding, fonts.

The colon, : acts like an equal sign.

Next tell the browser what you want the setting to be.

In the case of margins we use pixels. The abbreviations is px. We set the pixels for both margins and padding to 0px.

The last item on the line is the semicolon, which tells us that it is the end of that particular line.

**Setting the Headlines**

h1{

font: bold 25px Tahoma;

}

Tells the browser that we want headline 1 to be bold, 25 pixels and Tahoma font.

You can see that there are formatting options for headline 2 and headline 3.

The only difference is that each headline gets progressively smaller in terms of font size.

**Day 4:**

New HTML5 Elements in css file

header,section, footer, aside, nav, article, hgroup {

display:block;<br>

}

These are the new HTML5 Elements for laying out your web page.

The lines above are telling older browsers that do not recognize these elements to display them as a block of text.

**Parents and Children in cs file**

#parent{

border:2px solid blue;

background:pink;

display:-webkit-box;

display:-mozkit-box;

display:-ms-flex;

-webkit-box-orient:horizontal;

//-mozkit-box-orient:horizontal;

//-msflex-orient:horizontal;

-webkit-box-direction:normal;

-webkit-box-pack:center;

-webkit-box-align:center;

-webkit-box-flex:1;

}

Children inherit traits or properties from the parent, like display:-webkit-box is a property of article 1/kid 1 and article 2/kid 2.

To be considered a child or kid, they must be located between section or div tags.

For example, Articles 1 and 2 are the children of parent. Article 1 will contain the pictures of our products and article two will be our shopping cart

Article 3 is all by itself and will show what items we have dragged and dropped into the shopping cart, including the prices

Article 4 and 5 are children of &#60;section id="last\_one"&#62; and will contain totals and customer information.

The wish list or note is all by itself.

The children of parent will have a pink background and a solid blue border

**Laying out the body format**

body{

width: 100%;

display:-webkit-box;

-webkit-box-pack:center;

border:5px solid green;

}

The first line of this code specifies that the element we want to format is the body.

We can specify the width in either percentages or pixels. We specified 100%

The display for the body element is formatted as a flexible box

We want to put a box with a 5 pixel, solid, green border around the body of our page.

The -webkit prefix is vendor specific. The -webkit prefix is for Chrome, and Safari.

-moz prefix is used for Firefox.

They are used to add features to their browsers

The Firefox and Internet Explorer options are commented out. Some of these functions are not yet implemented in these browsers.

The box-pack property specifies where child elements of a box are placed when the box is larger than the size of the children.

This property specifies the horizontal position in horizontal boxes, and the vertical position in vertical boxes.

The box-pack property can take the following properties:start means left. End means right. Center, centers the child boxes in the parent box.

**Outside Wrapper**

#outside\_wrapper{

max-width:1000px;

margin: 20px 0px;

display:-webkit-box;

border:5px solid red;

-webkit-box-orient:vertical;

}

The outside wrapper is a container around the entire page, but inside the body.

These divisions will be clear to you soon when we link the style sheet to the html file

The border of the body is green and the border for the outside wrapper is red. Click on formatted web page sample link located above.

We want the width to be a maximum of 1000 pixels

We want a left and right margin of 0 and a top an bottom of 20

The margin property can be handled in a number of ways

***Example 1***

margin:10px 5px; - Top and bottom are 10. Right and left are 5

***Example 2***

margin:10px 5px 15px; -Top = 10px. Right and left=5px. Bottom is 15px;

***Example 3***

margin:10px; - all margins are the same 10 pixels

***Example 4***

margin:10px 5px 15px 20px; - Top = 10px. Right=5px. Bottom= 15px. Left=20px;

The display:-webkit-box puts a box on your screen. This represents a section of your screen as is used to layout your web form.

We want the boxes aligned vertically not horizontally.

**Laying Out the top Header**

#top\_header{

//background:rgb(255,228,225);

background:#FFE4E1;

//background:mistyrose;

border:3px solid black;

padding:20px;

{

The top header has a mistyrose background with a 3 pixel solid black border.

Colors can be specified in three different ways: name of color, RGB color combination and hexadecimal.

The example above uses hexadecimal. The other options for the same color are listed, but commented out.

Click on sample above to find this section?

This is where your company name and logo will go

The padding refers to the space around any text or images in the box.

<Header id="top\_header">

<H1>Company Name</H1>

</Header>

Above are the HTML tags which are formatted by the cascading style sheet file**.**

**Top Menu/ Nav tag plus ID**

Cascading Style Sheet Code

#top\_menu{

border: 3px solid orange;<br>

//background:#8fbc8f;<br>

//background:darkseagreen;<br>

background:rgb(143,188,143);<br>

}

HTML Code

<Nav id="top\_menu">

<ul>

<A href="Home.html">Home</A>

<A href="About.html">About</A>

<A href="Products.html">Our Products</A>

<;A href="Contact.html">Contact Us</A>

</ul>

</Nav>

The next section want to cover is the navigation bar. This is how we get from one page to another.

As you can see from the code above, we gave the **nav tag** and ID equal to top\_menu. This id is then referenced in the cascading style sheet file.

The html tag is then formatted according to the code or instructions in the cascading style sheet file.

This section of our page has a dark sea green background. The code used RGB/ The other ways of describing the color are commented out.

The link or menu options are in an unordered list. That means they are not numbered 1,2,3.

When the list is complete there is an unordered ending tag,</ul>

<li>This section is terminated with and ending nav tag,</nav>

**Day 5:**

**Laying out the menu in a horizontal fashion/hover code**

Cascading Style Sheet Code

#top\_menu li{

display:inline-block;<br>

list-style:none;<br>

padding:5px;<br>

font:bold 14px tahoma;<br>

}

a:link{color:#FF0000;} /\* unvisited link red \*/

a:visited{color:#00ff00;} /\*visited link lime\*/

a:hover{color:#fffafa;} /\* mouse over link snow\*/>

a:active{color:#0000ff;} /\*selected link blue\*/

In the css code we want the menu items laid out horizontally instead of vertically

That is what inline-block does

We do not want a list style. These are bullets

We want some padding between the words

We are using a larger font to make the links to the other pages stand out.

The a:link lines come into effect when you pass the cursor over the links. They change colors

**HTML Code for a new section**

< section id = "parent">

<section id="parent">

<article id="kid1">Products For Sale

<br>

</article>

<article id="kid2">Shopping Cart

<br>

</article>; <!-- end of article 2 shopping cart -->

</section><br>

We started this new section since it is separate from the top part of our page:The heading and navigation boxes.

We want this box to be oriented horizontally

You can see from the html code, that this section called parent and contains two articles.

The first article will be used to contain pictures of all products featured in the drag and drop shopping cart.

It is from this section where the user will click on the product and drag it to article 2: the shopping cart.

**Cascading Style Sheet Code for Kid1 and Kid 2**

#kid1{

border: 2px solid red;

background:green;

padding:20px;

margin:10px;

-webkit-box-flex:1;

//-mozkit-box-flex:1;

}

#kid2{

border: 2px solid red;

background:yellow;

padding:20px;

margin:10px;

width:320px;

height:190px;

}

First we will look at kid1 and kid2. Remember they inherit some traits from the parent: the section that contains them has a pink background with a blue border.

The kids also inherit the display property of the parent since they are inside the parent section tags.

The display property for all children or kids is a flexbox.

kid 1 has a solid red border with a background color of green. All margins are 10px and padding is 20px.

Kid 1 is a flexible box. There is also a reference to coding for mozkit : Firefox browser that is commented out.

Kid 2 also has s solid red border, but has a yellow background.

Kid 2 is not a flexible box. You can also set a width. The code indicates a width of 220 pixels.

**Cascading Style Sheet Code for Kid3 and Kid 4 and Kid 5**

#kid3{

border: 2px solid red;

background:white;

padding:20px;

margin:10px;

-webkit-box-flex:1;

//-mozkit-box-flex:1;

height:100px;

}

#kid4{

border: 2px solid red;

background:orange;

padding:20px;

margin:10px;

-webkit-box-flex:1;

//-mozkit-box-flex:1;

height:100px;

}

#kid5{

border: 2px solid red;

background:gray;

padding:20px;

margin:10px;

}

Kid 3 box is used for the items purchased. It will show each item dropped into the shopping and its price.

Kid 3 is a flexible box with a solid red 2 px border and green background. Click on the link above to see this.

On flexible boxes, heights can be set. Non flexible boxes can contain a set width.

Ten pixel margins are set all the way around with 20 pixels of padding.

Kid 4 has identical properties with the exception of background color. This box will contain the totals of the order.

Kid 5 will contain the customer information. It is a fixed box with a gray background