

Introduction to Web Design

Lecture_2

Dr. Sarah mohamed



HTML Tables

- HTML tables were created for instances when you need to add tabular material (data arranged into rows and columns) to a web page. Tables may be used to organize schedules, product comparisons, statistics, or other types of information. A table is a grid organized into columns and rows, much like a spreadsheet, as shown in following figure:

Books about Computing		
Title	Publisher	Author
IBM PC Assembly Language and Programming	Prentice-Hall	Abel, P
Object-Oriented Analysis and Design	Addison-Wesley	Booch, G
Demonstration of a Fixed Table		

- In visual browsers, the arrangement of data in rows and columns gives readers an instant understanding of the relationships between data cells and their respective header labels.
- The elements that make up the basic structure of a table

`<table>`

`<tr>` `<th>Menu item</th>` `<th>Calories</th>` `<th>Fat</th>` `</tr>`

`<tr>` `<td>Chicken noodle soup</td>` `<td>120</td>` `<td>2</td>` `</tr>`

`<tr>` `<td>Caesar salad</td>` `<td>400</td>` `<td>26</td>` `</tr>`

`</table>`

`<table>...</table>`

Tabular content (rows and columns)

`<tr>...</tr>`

Table row

`<th>...</th>`


Table header

`<td>...</td>`

Table cell data

Table Example

```
<table>
  <tr>
    <th>Menu item</th>
    <th>Calories</th>
    <th>Fat (g)</th>
  </tr>
  <tr>
    <td>Chicken noodle soup</td>
    <td>120</td>
    <td>2</td>
  </tr>
  <tr>
    <td>Caesar salad</td>
    <td>400</td>
    <td>26</td>
  </tr>
</table>
```



Start and end table tags identify the beginning and end of the tabular material. The table element may directly contain only some number of tr (row) elements, a **caption** and, optionally, the row and column group elements listed in the “Row and Column Groups” section. The only thing that can go in the tr element is some number of td or th elements. In other words, there may be no text content within the table and tr elements that isn’t contained within a td or th.

TABLE HEADERS

```
<table>
  <tr>
    <th>Burgers</th>
    <td>Organic Grass-fed Beef</td>
    <td>Black Bean Veggie</td>
  </tr>
  <tr>
    <th>Fries</th>
    <td>Hand-cut Idaho potato</td>
    <td>Seasoned sweet potato</td>
  </tr>
</table>
```

Burgers	Organic Grass-fed Beef	Black Bean Veggie
Fries	Hand-cut Idaho potato	Seasoned sweet potato

Spanning Columns

- The colspan attribute is used to make **a cell span** (merge) more than one column.
- HTML code that's spans two columns in the first row, and the output is shown in figure:

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<table border="1" align="center" width="100%" height="400px">
  <tr>
    <th>ID</th>
    <th colspan="2">Name</th>
  </tr>
  <tr>
    <td align="center">1</td>
    <td align="center">Bill</td>
    <td align="center">Gates</td>
  </tr>
</table>

</body>
</html>
```

ID	Name	
1	Bill	Gates

Spanning Columns example2

```
<table>
  <tr>
    <th colspan="2">Fat</th>
  </tr>
  <tr>
    <td>Saturated Fat (g)</td>
    <td>Unsaturated Fat (g)</td>
  </tr>
</table>
```

Fat	
Saturated Fat (g)	Unsaturated Fat (g)

HTML Table with Colspan

Name	Age	Country
Sarah	22	Egypt
End of Table		

```
3 </head> </head>
4 <body>
5     <h2>HTML Table with Colspan</h2>
6
7 <table border="1" cellpadding="10">
8     <tr>
9         <th>Name</th>
10        <th>Age</th>
11        <th>Country</th>
12    </tr>
13
14    <tr>
15        <td>Sarah</td>
16        <td>22</td>
17        <td>Egypt</td>
18    </tr>
19
20    <tr>
21        <!-- This cell spans across 3 columns -->
22        <td colspan="3" align="center">End of Table</td>
23    </tr>
24 </table>
25
26 </body>
```

```
<!DOCTYPE html>
<html>
  <head> </head>
<body>
  <h2>HTML Table with Colspan</h2>

  <table border="1" cellpadding="10">
    <tr>
      <!-- This cell spans across 3 columns -->
      <td colspan="3" align="center">Header of Page</td>
    </tr>
    <tr>
      <th>Name</th>
      <th>Age</th>
      <th>Country</th>
    </tr>

    <tr>
      <!-- This cell spans across 3 columns -->
      <td colspan="3" align="center">Footer of page</td>
    </tr>
  </table>
```

HTML Table with Colspan

Header of Page		
Name	Age	Country
Footer of page		

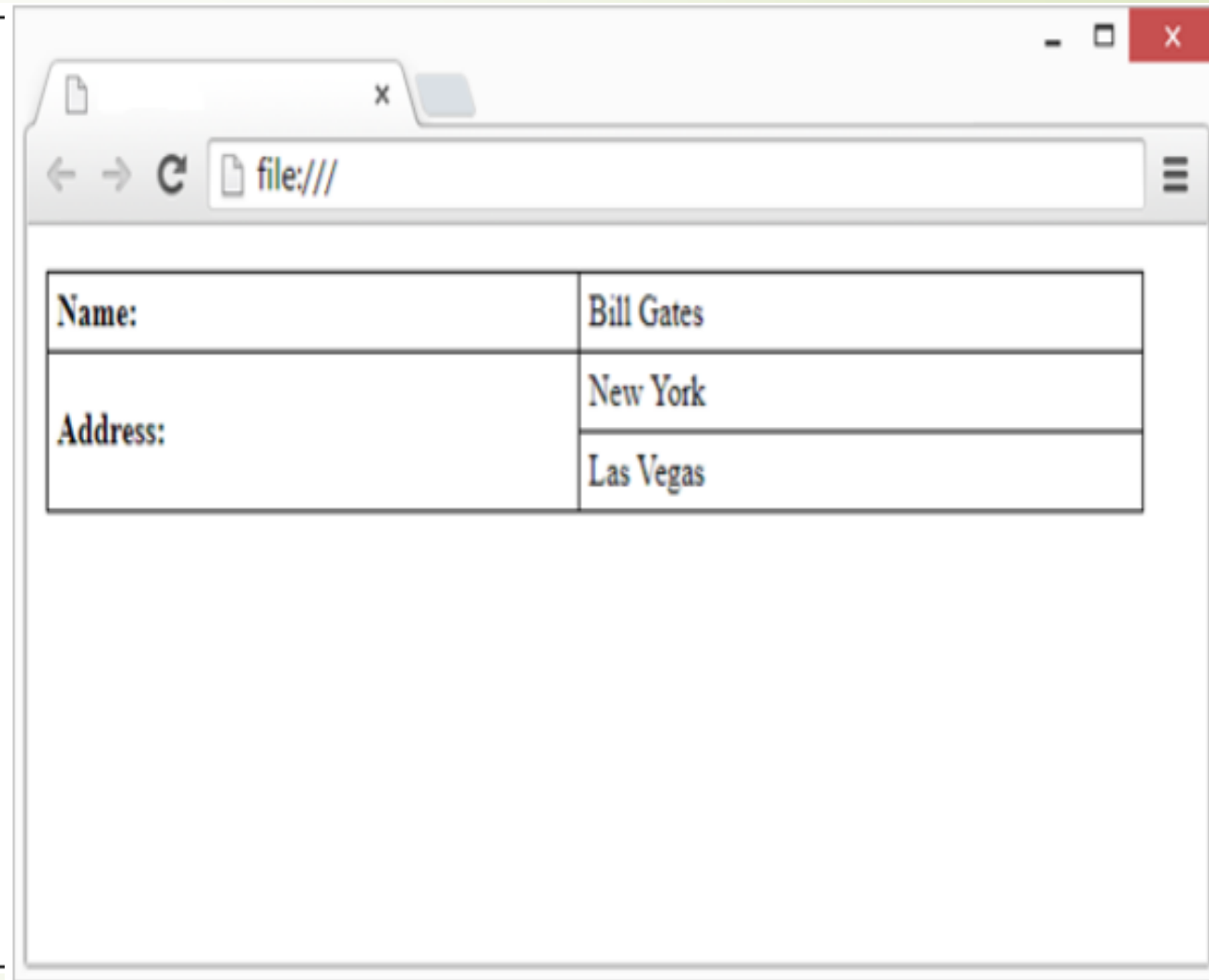
Spanning Rows

- Row spans, created with the rowspan attribute, work just like column spans, but they cause the cell to span **downward** over several rows:

```
<table border="1" align="center" >
  <tr>
    <th rowspan="3">Serving Size</th>
    <td>Small (8oz.)</td>
  </tr>
  <tr>
    <td>Medium (16oz.)</td>
  </tr>
  <tr>
    <td>Large (24oz.)</td>
  </tr>
</table>
```

Serving Size	Small (8oz.)
	Medium (16oz.)
	Large (24oz.)

```
<table style="width:100%">
  <tr>
    <th>Name:</th>
    <td>Bill Gates</td>
  </tr>
  <tr>
    <th rowspan="2">Address:</th>
    <td>New York</td>
  </tr>
  <tr>
    <td>Las Vegas</td>
  </tr>
</table>
```

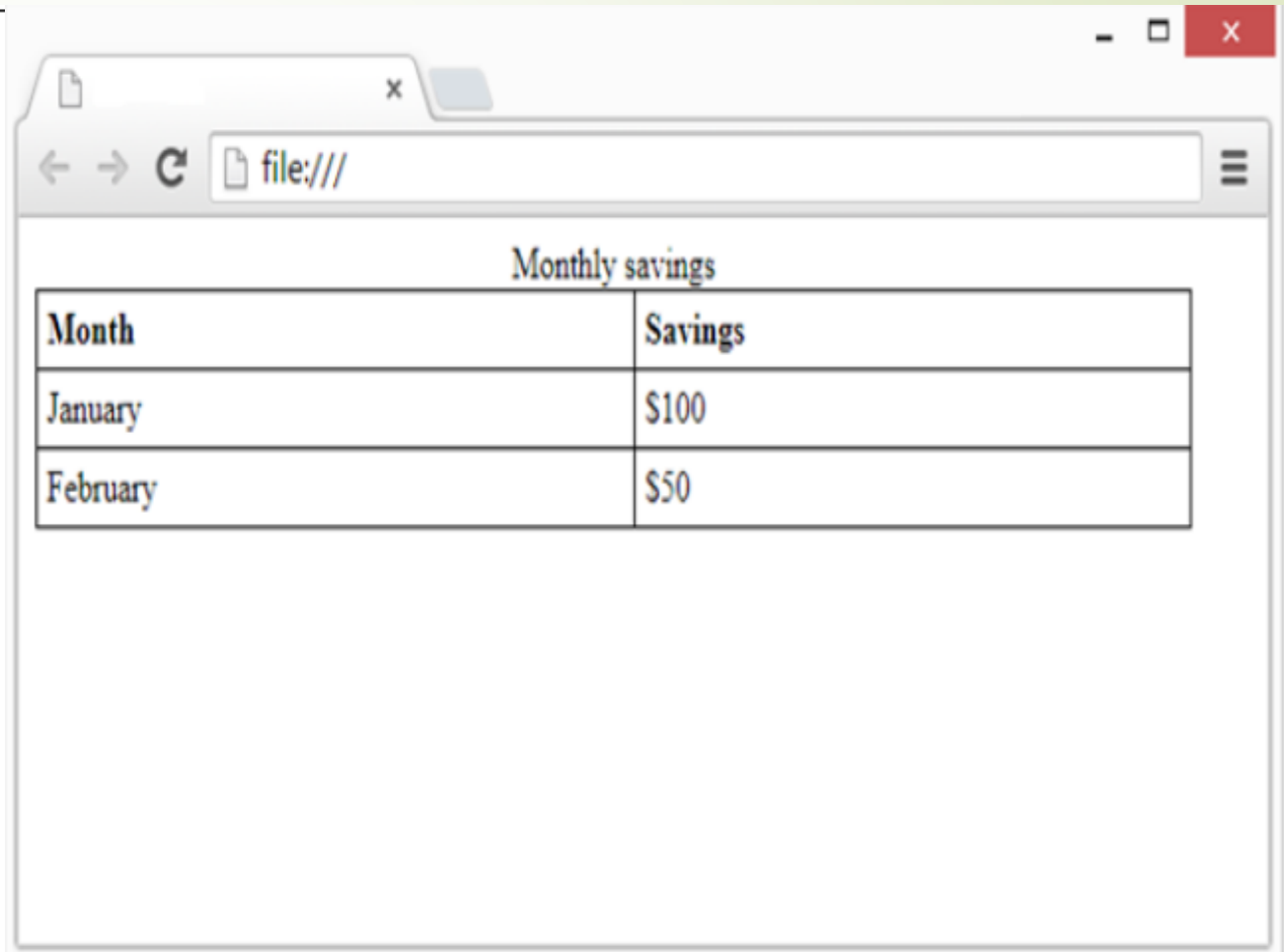


A screenshot of a web browser window. The address bar shows 'file:///'. The browser displays the rendered HTML table from the code on the left. The table has three rows: a header row with 'Name:' and 'Bill Gates', and two data rows under a rowspan=2 header 'Address:' with values 'New York' and 'Las Vegas'.

Name:	Bill Gates
Address:	New York
	Las Vegas

Adding a Caption to HTML Tables

```
<table style="width:100%">
  <caption>Monthly savings</caption>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$50</td>
  </tr>
</table>
```



A screenshot of a web browser window showing the rendered HTML table. The browser's address bar displays 'file:///'. The table is titled 'Monthly savings' and contains three rows: a header row with 'Month' and 'Savings', and two data rows for 'January' (\$100) and 'February' (\$50).

Month	Savings
January	\$100
February	\$50

HTML Tables Border and Background

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>
<table border="2" bgcolor="#efefef">
<tr>
<th width="150"> Acount</th>
<th>Withdrawn</th>
<th>Credit</th>
<th width="150">Balance</th>
</tr>
</table>

</body>
</html>
```

Acount	Withdrawn	Credit	Balance
--------	-----------	--------	---------

With Padding

hello	hello	hello
hello	hello	hello
hello	hello	hello

With Spacing

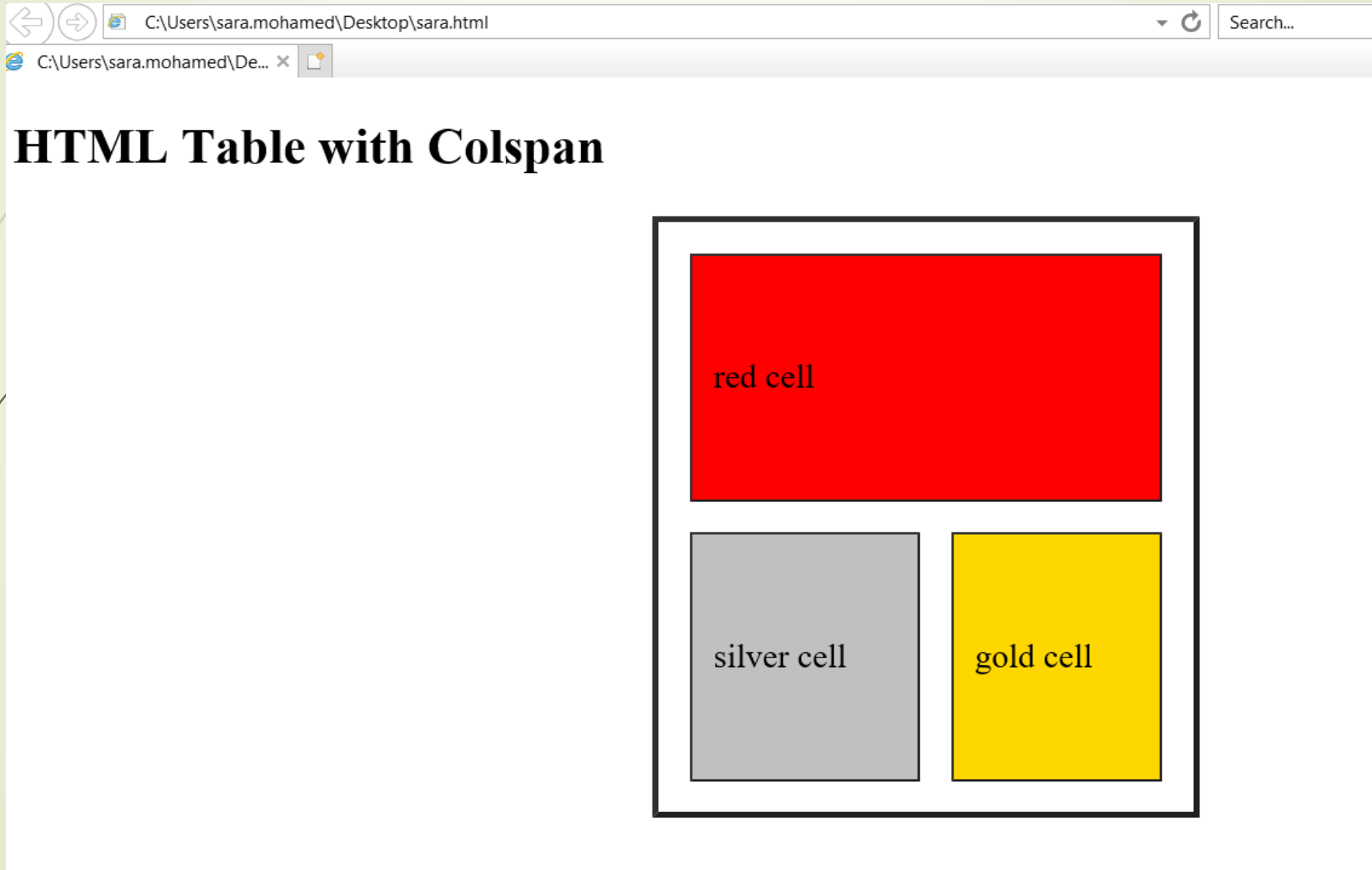
hello	hello	hello
hello	hello	hello
hello	hello	hello

cellspacing and cellpadding

- The **cellspacing** attribute in HTML defines the space between table cells.
- Defines the space between table cells (the **outer** gap).
- **Cell padding** is the space between the cell edges and the cell content. HTML tables can adjust the padding inside the cells

```
1  <!DOCTYPE html>
2  <html>
3      <head> </head>
4  <body>
5      <h2>HTML Table with Colspan</h2>
6  <TABLE style= "width: 30%" border=3 align = "center"
7      cellpadding= 10 cellspacing = 15>
8  <tr>
9      <td colspan=2 height=100 bgcolor="red">red cell</td>
10 </tr>
11 <tr >
12 <td bgcolor = "silver" height =100> silver cell</td>
13 <td bgcolor = "gold" height =100> gold cell</td>
14 </tr>
15 </TABLE>
16
17 </body>
18
19 </html>
```

Output





➡ ROW AND COLUMN GROUPS

Row Group Elements

- You can describe rows or groups of rows as belonging to a header, footer, or the body of a table by using the **thead**, **tfoot**, and **tbody** elements, respectively. Some user agents (another word for a browser) may repeat the header and footer rows on tables that span multiple pages. For example, the head and foot rows may print on every page of a multipage table. Authors may also use these elements to apply styles to various regions of a table

<h1>The thead, tbody, and tfoot elements</h1>

```
<table>
  <thead>
    <tr>
      <th>Month</th>
      <th>Savings</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>January</td>
      <td>$100</td>
    </tr>
    <tr>
      <td>February</td>
      <td>$80</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Sum</td>
      <td>$180</td>
    </tr>
  </tfoot>
</table>
</body>
</html>
```

The thead, tbody, and tfoot elements

Month	Savings
January	\$100
February	\$80
Sum	\$180

Using Tables in Page Design

- Tables are useful for laying out text and images on in Web page.
- While a Web browser can manage to layout a document at any resolution, different resolutions do affect the layout and presentation of an HTML document. Resolution is measured in picture elements, called **pixels**.
- Resolution and monitor size are independent of one another: a large monitor can have a low resolution, while a small monitor may have a high resolution. Resolution is determined by the hardware, the user, and the video card driver installed on the computer.
- A single monitor may have a choice of resolutions.
- By default, all Web pages are designed with flexibility in mind.
- Flexibility can be defined as a Web page's ability to resize and adapt to the available resolution, monitor and window sizes.



The ways for web layout design

- Flexible Design
 - Fixed Design
- 

Flexible Design

- While HTML is flexible by default, it should not be confused with thinking a flexible document is disorganised, poorly managed with an unstructured layout. A flexible HTML document can still be structured and organised by using, for instance, tables to create columns of text (as in newspapers), and provide layout design.
- **Flexible** layout can be achieved by using percentage measurements for table dimensions. As an example, view the table by changing the size of your browser's window (i.e. the live space). **Observe that as the window size changes, so does the table size.** The table measurements used in this example are called '**relative**' measurements, as the sizes are expressed in terms of a percentage of the screen space.

Fixed Design

- Fixed design expresses all dimensions in pixels: the dimensions remain fixed regardless of the size of the device it is viewed on. Such an approach has advantages and disadvantages.
- The chosen fixed size may be too large for a user's available live space, causing the user to scroll in order to view the whole page; a fixed Web page may also appear too small, leaving unsightly blank spaces.
- To develop a fixed Web page using tables, supply all measurements in pixels.



Example_1 of using table to design the
page layout

```

<!DOCTYPE html>
<html>

<head> <title>home</title></head>

<body>

    <table border="1" bgcolor="silver" align="center" width="100%" >

        <!--banner-->

        <tr>

<td height="150" > </td>

        </tr>

        <!--navigation bar-->

        <tr>

            <td height="50" >

<nav align="center">
    <font size="6">
<a href="http://www.facebook.com" target="_blank"> Facebook</a>|
<a href="http://www.yahoo.com" target="_blank"> yahoo</a>|
<a href="http://www.youtube.com" target="_blank"> youtube</a>
</font>
</nav>

</td>
    </tr>

```

```

<!--content and sidebar-->

        <tr>

        <td>

            <table width="100%" height="600" border="1" align="center">

                <tr>

                    <td width="60%" bgcolor="red">content</td>

                    <td width="40%" bgcolor="yellow">sidebar</td>

                </tr>

            </table>

        </td>

    </tr>

    <tr>

        <td height="50" bgcolor="gray" align="center">@copyright</td>

    </tr>

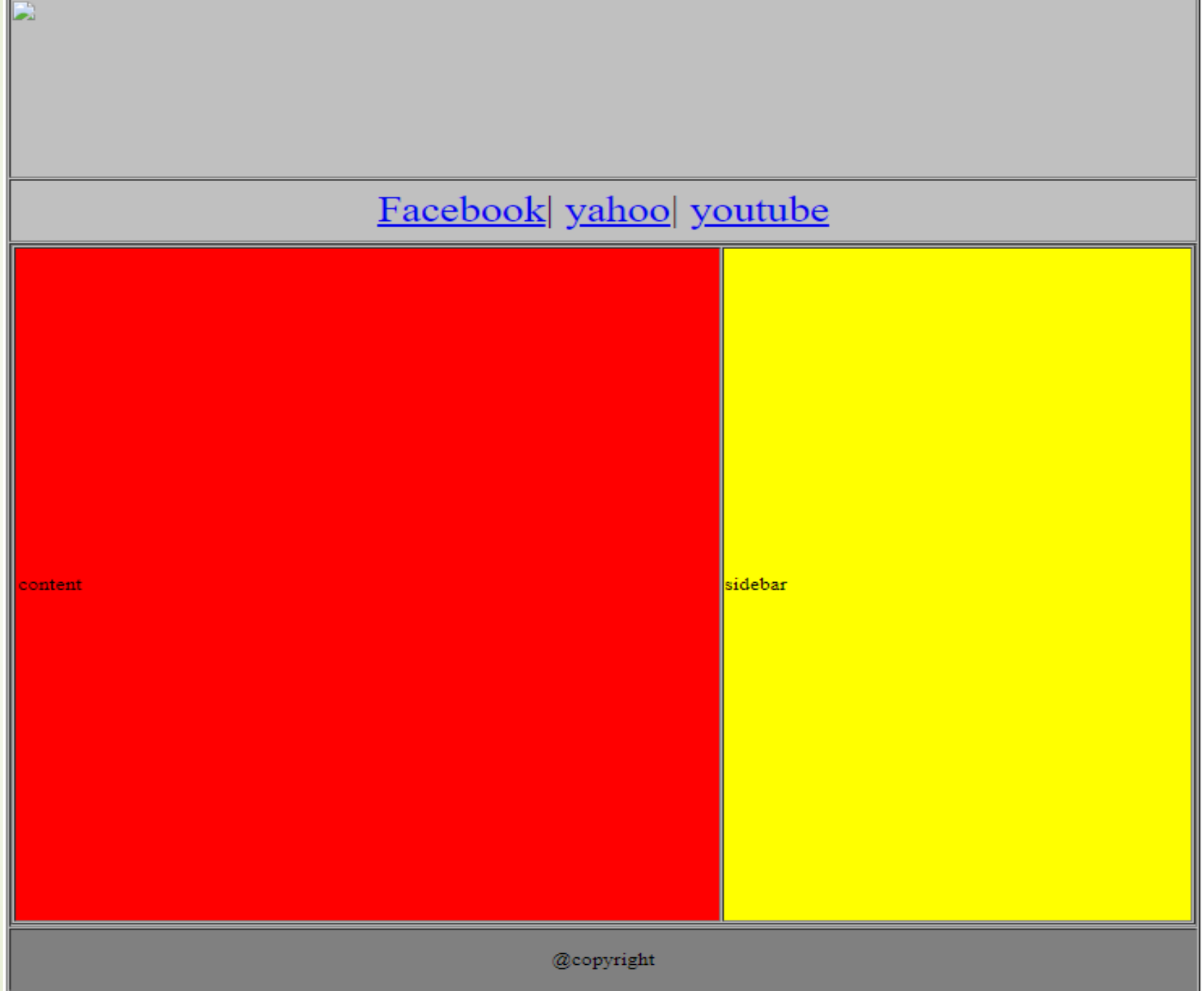
</table>

</body>

</html>

```

Output



Do the following task



References

- Jennifer Niederst Robbins, LEARNING WEB DESIGN A BEGINNER'S GUIDE TO HTML, CSS, JAVASCRIPT, AND WEB GRAPHICS, O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol 2018.
- Lance Douglas Jackson, INTRODUCTION TO THE INTERNET AND WEB PAGE DESIGN, Master of Arts in Professional Communication 2009.
- https://www.w3schools.com/html/html_table_padding_spacing.asp
- https://www.cs.uct.ac.za/mit_notes/web_programming/pdfs/chp04.pdf