# MIS3690 WEB TECHNOLOGIES

BABSON COLLEGE

TOIM DIVISION

# AGENDA

- Introduction to the course, syllabus and deliverables
- Term Project – mentioned
- Software – Visual Studio Code
- Architecture of the web
- HTML 5 – Introduction
- Your first web page!
- More HTML

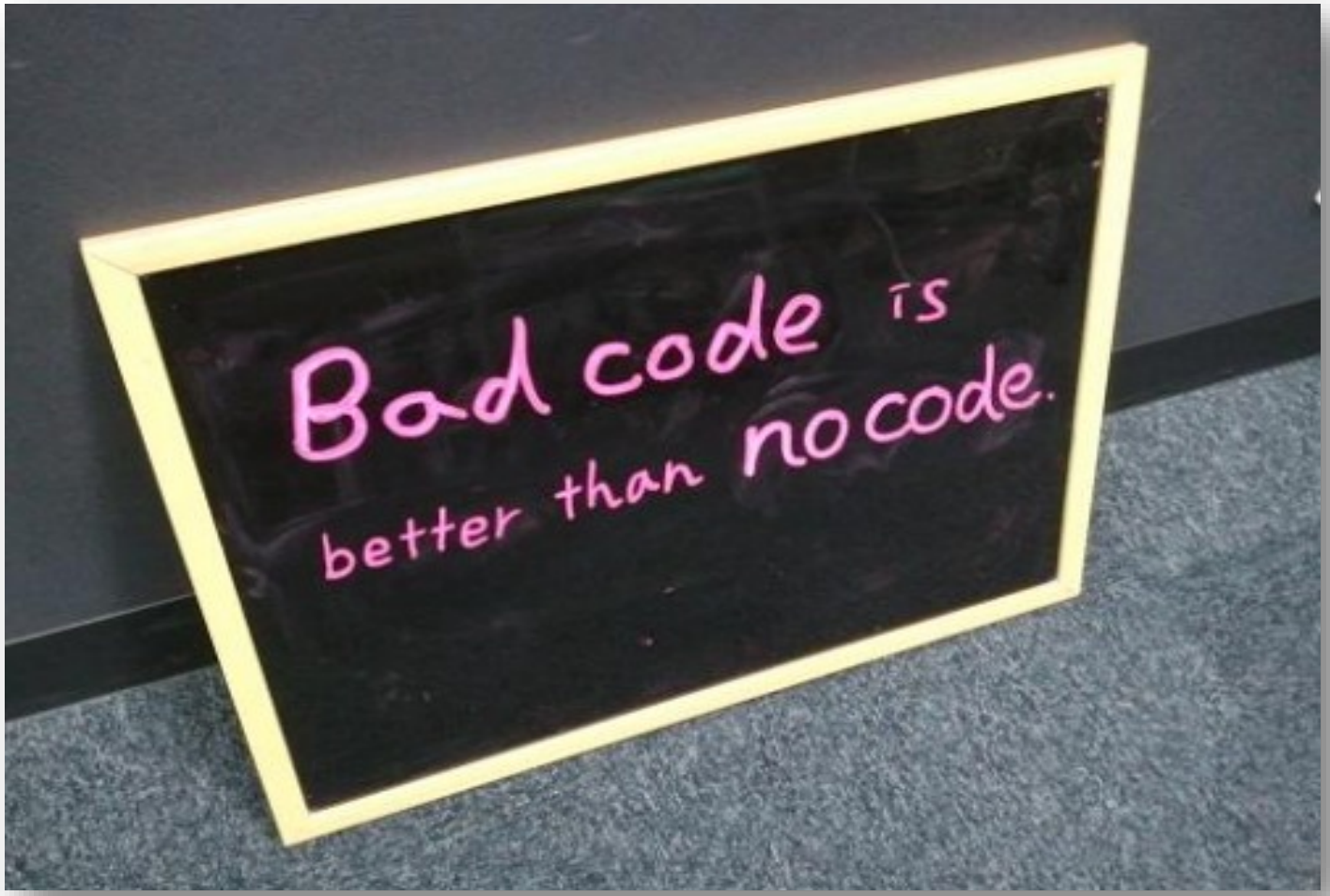
# SYLLABUS

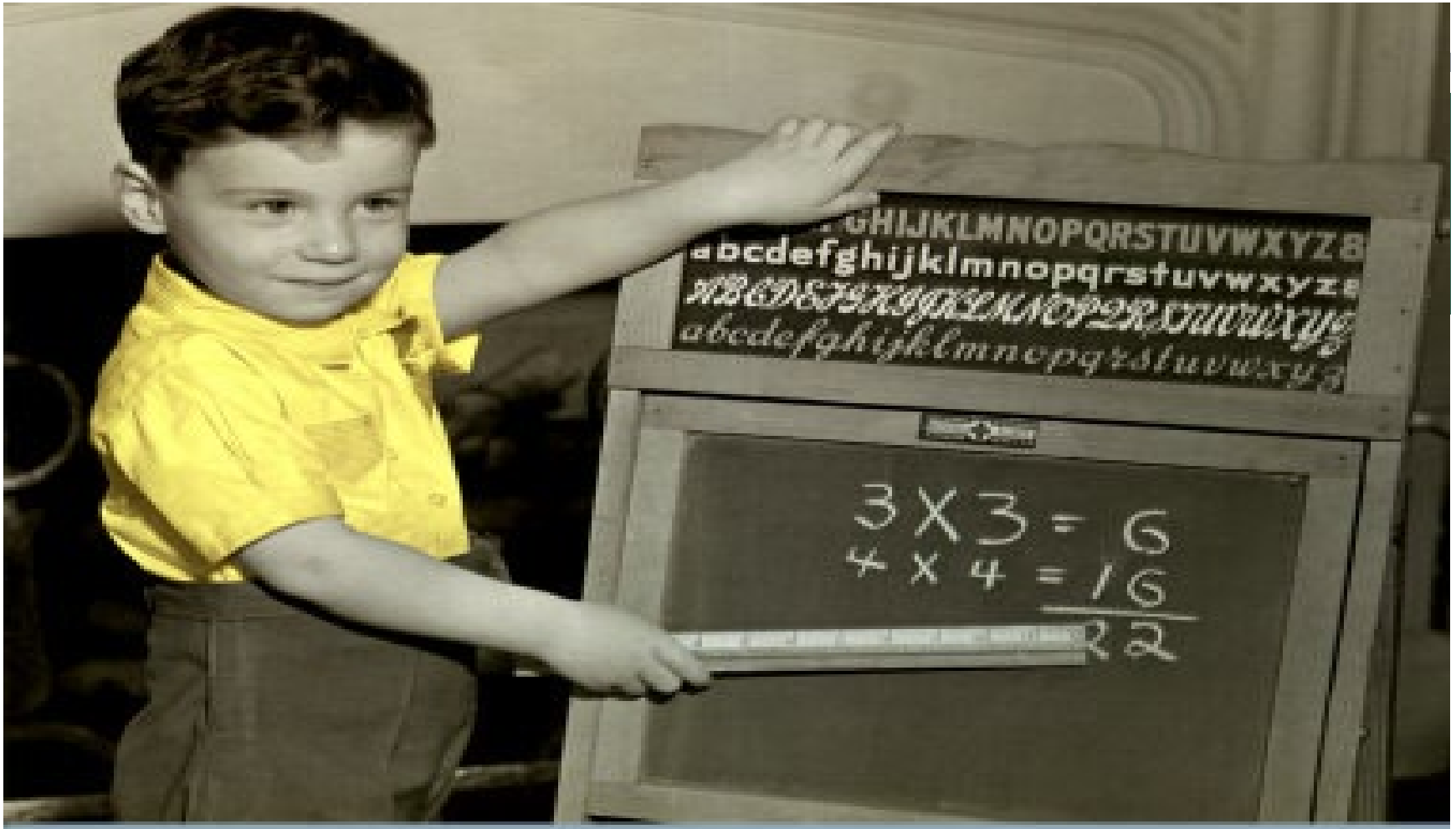
- **Instructor:** Zhi Li ( 李直 )
- **Office:** Babson Hall 216D
- **E-mail:** zli@babson.edu
- **Phone:** (781)239-5915
- **Office Hour:** Tuesday, Thursday, 4:40 PM-6:00 PM

# SYLLABUS

- Course Objectives
- Prerequisites and Textbooks
- Software
- Term Project
- Personal Website for the Course
- Graded Homework
- In-Class Exercises
- Midterm Exam
- Grading
- Course Policies

# HOW TO LEARN WEB TECHNOLOGIES?






There are

# NO STUPID QUESTIONS

or stupid answers.

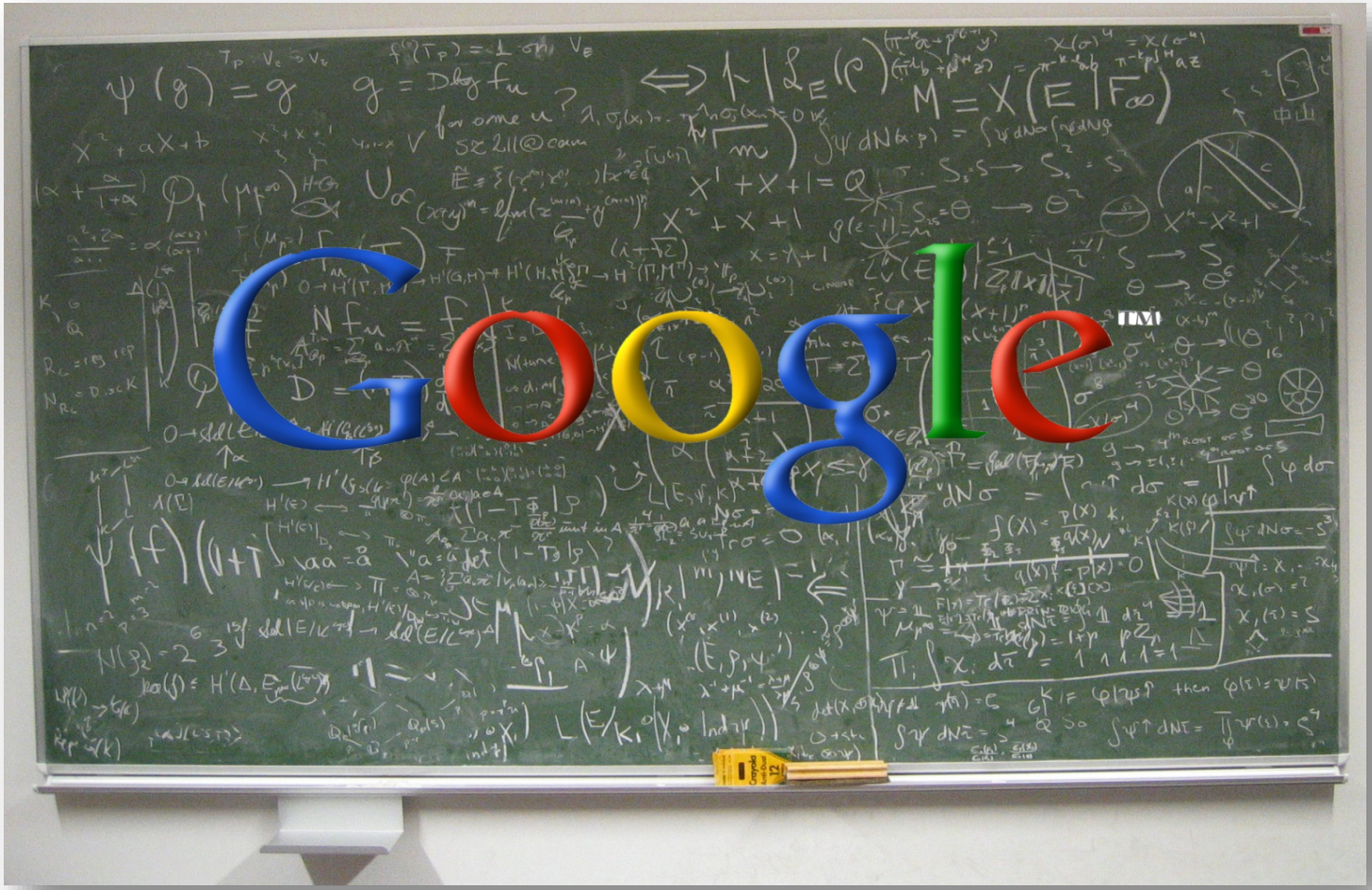
Coding is hard.

It takes time to
get good at coding.

# HOW TO SUCCEED IN THIS COURSE

- Read the error!
- Search Google
- Ask for help

# GOOGLE IS YOUR (SECOND) BEST TEACHER!

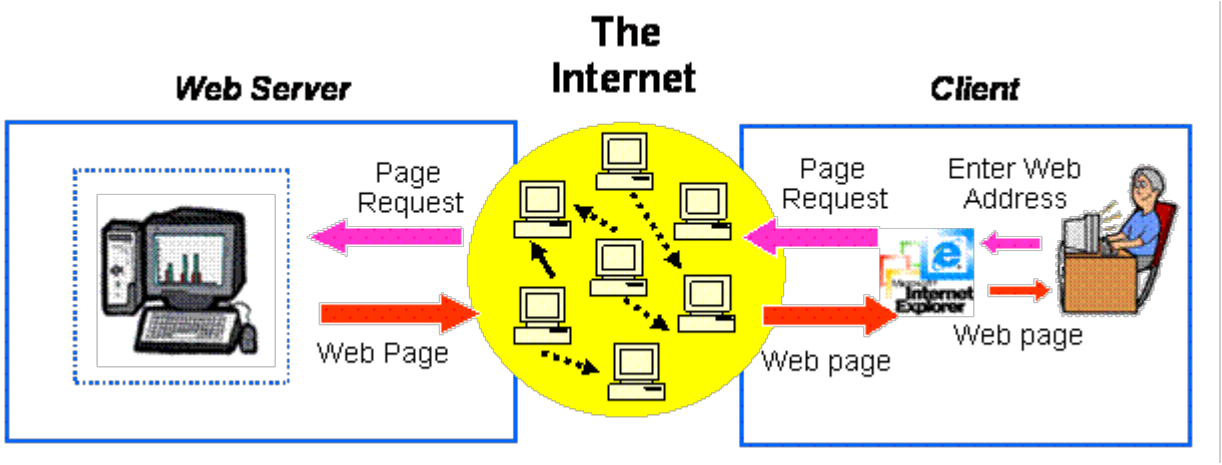


# OR STACKOVERFLOW.COM…

# MORE NOTES ON HOW TO SUCCEED IN THIS COURSE

- Do not take the "couch potato" approach
- Think of my code as one of the "answers"
- You need to code by yourself
- This force you to think in certain ways and ask specific questions
- After you've attempted it yourself, look at my code and think about what you were missing and why

# WEB ARCHITECTURE OVERVIEW




- A Client is a machine where users can submit requests to other computers on the network (Servers on the Internet or on the local network).
  - A Client can be any device, such as your PC, Mac, iPhone, Android or any Smartphone.
- A Server is a computer that can accept requests from clients and process a response or contact other servers
  - There are many kinds of servers: File Servers, Print Servers, Mail Servers, Database Servers, Web Servers….

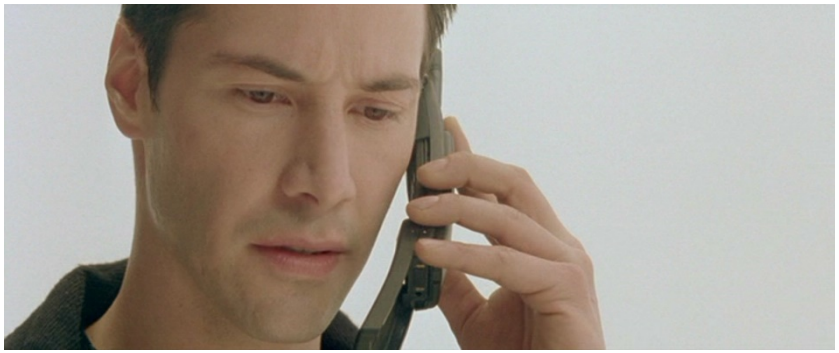
# WEB SOFTWARE COMPONENTS

- The Client Browser
  - Internet Explorer, Firefox, Safari, Chrome…

- The Web Server
  - Apache, Microsoft IIS, nginx…

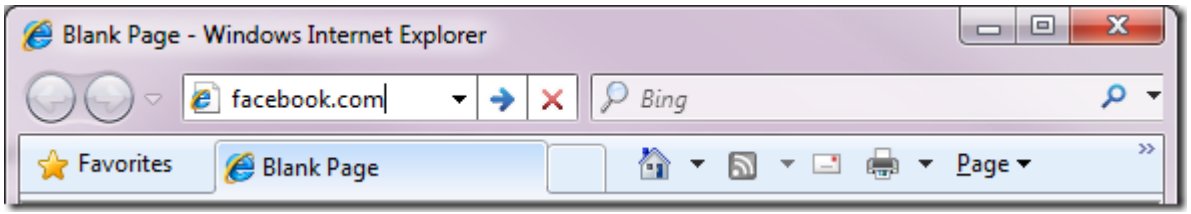


# HOW DO THE BROWSER AND SERVER REALLY TALK?

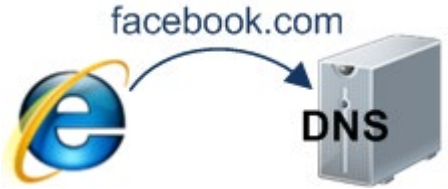


# WHAT REALLY HAPPENS WHEN YOU NAVIGATE TO A URL?

- You type an URL (Uniform Resource Locator, e.g., www.facebook.com) into the browser or click on a link.

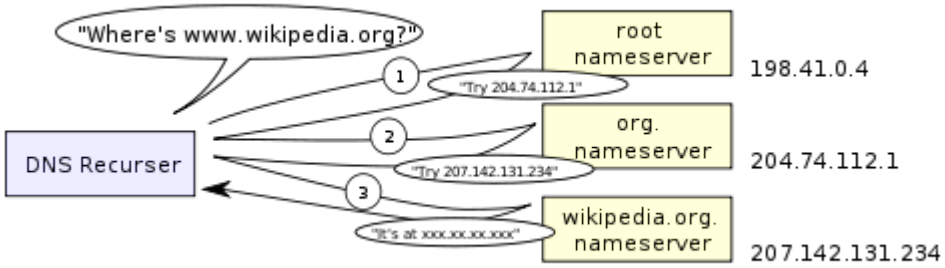



- Either way, the browser sends a request to the IP address specified in the URL.
- The request goes to the local Server (within the company, your service provider (ISP), or the Internet).
- A translator, DNS (Domain Name Service/System), translates the URL to an IP Address.



# WHAT REALLY HAPPENS WHEN YOU NAVIGATE TO A URL? (CONT.)

- The Server determines that the address (in this case, Facebook) is outside somewhere and forwards the request to the Internet – together with the return address (your computer or client)
- The Internet has a number of machines (computers) called Routers. The router, routes the request to its destination. It may go through several routers before it gets to its destination (in this case, Facebook's Web Server).




- Try to type `tracert www.facebook.com` in Command Prompt in your laptop and see what happens.

# WHAT REALLY HAPPENS WHEN YOU NAVIGATE TO A URL? (CONT.)

- The browser sends a HTTP request to the web server.


- Facebook's server then tries to locate the specific page (specified in the URL) that your requested. Sometimes this is the home page – it could also be some specific page within Facebook's web site.

- The resource (the requested page) is then sent back. It goes through several routers on the Internet and eventually gets to the local server (in this case, Babson's Web Server).

# WHAT REALLY HAPPENS WHEN YOU NAVIGATE TO A URL? (CONT.)

- This server determines the IP Address to which the resource must be sent (the IP address of your computer).
- Your browser receives the resource. The resource is coded in HTML. The browser interprets the HTML.
- The browser also determines if additional resources are needed to display the page (e.g., images, multi-media, etc.) and sends additional requests for each resource needed.


- Once the browser has all the resources, it interprets the HTML-coded resource and displays the page on your client computer.

# THE WEB IS STATELESS

- Web servers have no concept of a session or ongoing conversation
- Once the server responds the conversation ends
- We will learn some tricks to create long conversations and multi-step transactions when we learn JavaScript

# INTRODUCING HTML(5)

# INTRODUCING HTML

- Stands for **H**yper **T**ext **M**arkup **L**anguage
  - Hyper Text = the content of a web page
  - Markup = tags used to define how content is to be displayed
  - Language = syntax and vocabulary associated with marking-up.

# STANDARD HTML5 TEMPLATE

```
<!DOCTYPE html >

<html>
    <head>
        <meta charset="UTF-8">
        <title>Page title goes here</title>
        <style type="text/css">
                /* Your styles go here */
        </style>
        <script type="text/javascript">
                /* Your JavaScript goes here */
        </script>
    </head>

    <body>
        <!-- Your HTML goes here -->
    </body>
</html>
```

<html> and </html> tell the browser where the page begins and where it ends

<head> and </head> tags define where the meta information starts and ends

<title> and </title> tells the browser what to display on the top of the page. This is required.

<style></style> and the <script></script> tags are optional. Use these in the future classes.

<body> and </body> tell the browser where the content of the page start and end.

# EVERYTHING IN HTML IS DEFINED USING <TAGS>

```
<h1> Welcome to MIS 3690 </h1>

<p> This is the first paragraph that is the least bit
interesting </p>

<p> This is the second paragraph that is even less
interesting than the first. It is longer than the first
paragraph. It also shows how the paragraph is formatted on
the page. </p>
```

# MORE SIMPLE TAGS – PART 1

```
<h1> Big Text </h1>
<h2> Relatively smaller text </h2>
<h3> Even smaller text </h3>
<h4> Well, you </h4>
<h5> get the </h5>
<h6> picture </h6>
```

- Most tags come in pairs – a opening tag and a closing tag.
- Tags are enclosed in tag markers (the < and >)
- Closing tags include a forward slash (/)

# SIMPLE TAGS – PART 2

-  and  - italicize the content
-  and  - make the content bold
- <u> and </u> - underline content
- <hr> - a thematic break in an HTML page (e.g. a shift of topic)
  - Note: does not come in pairs – single tag
  - Note: Cannot be placed inside a paragraph (syntax of the language)
- <br> - provides a line break

# MORE TAGS…. (WHAT ELSE…)

- `<div>` and `</div>` - helps divide a page into sections for formatting
- `<span>` and `</span>` - helps format multiple occurrences of a specific word or phrase the same way
- Will use more of these when dealing with formatting and CSS
- Comments
  - `<!- - Your comment goes here - - >`
  - For use by the web designer (you)
  - Will not be displayed

# NESTING TAGS

- Some tags may be nested.
- Some tags should not be nested.
- We will learn more about "nesting" as we go through the course.

```
<p> This is a <i> good </i> example of how to do nesting
</p>
```

```
<p><i> Bad idea </p></i>
```

```
<p> This is a really bad <hr/> example of nesting </p>
```

# SPECIAL CHARACTERS IN HTML

- **For HTML5, the default character encoding is UTF-8.**
  - This has not always been the case. The character encoding for the early web was ASCII.
- Reserved characters in HTML must be replaced with character entities.
  - For example, if you use the less than (<) or greater than (>) signs in your text, the browser might mix them with tags.
- To use "special" characters we need additional codes. For example:
  - > (greater than) is done using `>`
  - < (less than) is done using `<`
  - & (ampersand) is done using `&`
  - " (quotation) is done using `&quot;`
- Reference
  - https://www.w3schools.com/html/html_entities.asp

# IN CLASS 01 – CREATING YOUR FIRST WEB PAGE

- Download the template.
- Save it.
- Open the template file using *VS Code* (you can right-click and choose which the software to open the file with)
- Add a title for the page, say, "Web Technologies – ".
  - This is in the `<head>` section of the template.
- Save the file as "CS01-InClass.htm" under folder 'WebTech/classwork'

# IN CLASS 01 – ADDING CONTENT

- Add a heading to the page "All About ME" (replace ME with your name)
- Add a paragraph on why you chose Babson and/or about what you want to be when you graduate.
- Add a paragraph on your favorite sport (anything but poker).
- Place a horizontal rule between the heading and the first paragraph and another one at the end of the second paragraph.

# IN CLASS 01 – FINALLY…….

- Save the file (remember you called it "CS01-InClass.htm")
- Go to windows explorer, right click on the file and open it with Chrome or Firefox or Internet Explorer.
- There is your first web page!!!!!!

- Don't forget to **commit** and **push** in GitHub Desktop.