



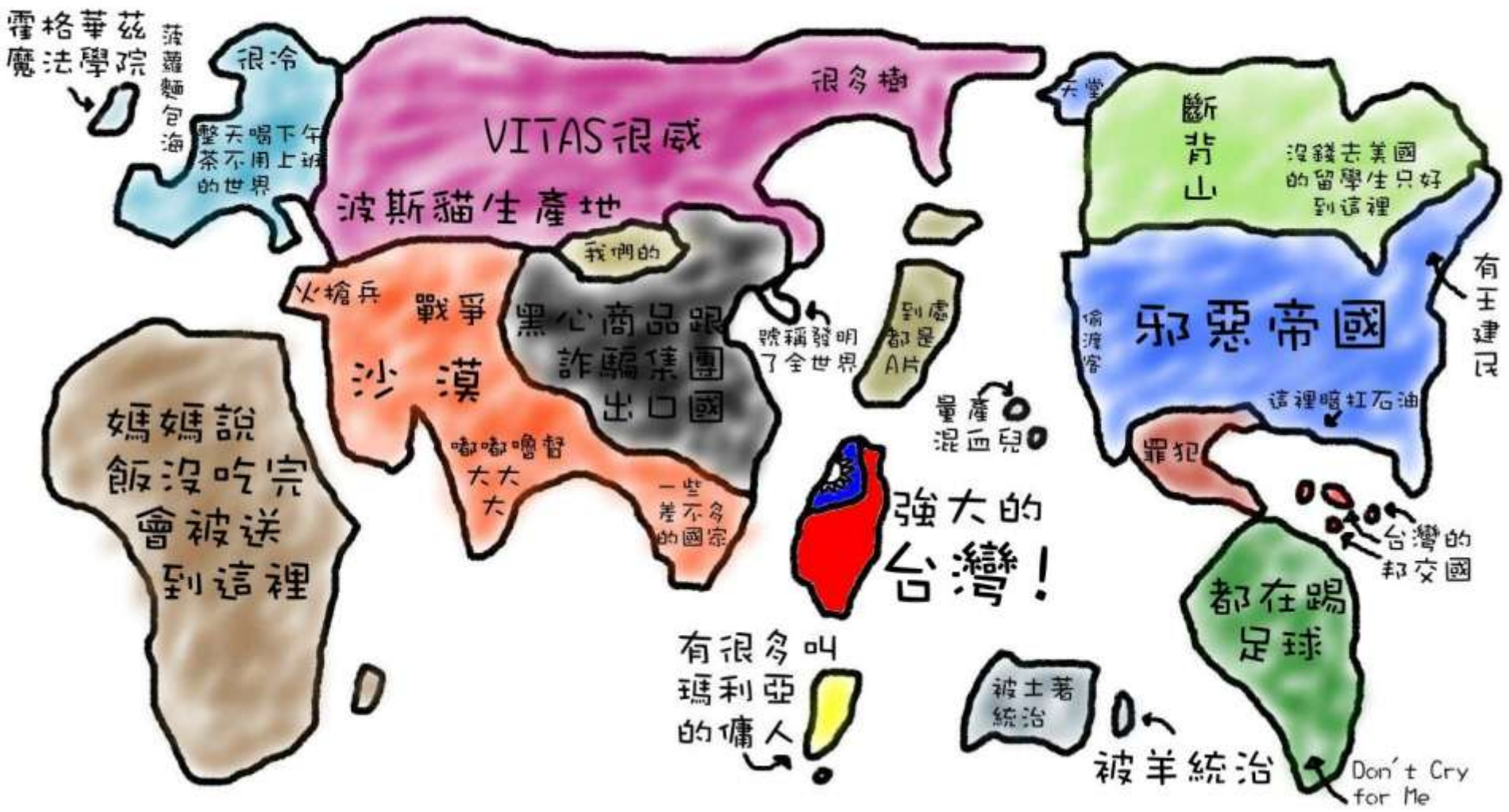
Molecular Biomedical Informatics  
分子生醫資訊實驗室

Web Programming  
網際網路程式設計

# HTML+CSS

# HyperText Markup Language (HTML)

- HyperText
  - contains images, sounds and links to other documents
- Markup Language
  - define how to mark things
  - what is marking?



# HTML is plain text

- The content is divided into two types
  - tag, used to mark things
  - text which can be seen in the web page
  - `<b>Text in tag 'b' is usually shown in bold</b>`
- A tag can contain, but not cross, other tags
  - `<b><i>bold and italic</i></b>`
  - `<b><i>what the hell?</b></i>`
- That's it! We are done.

# So, what actually to learn for HTML?

- Available tags
- Suppose that there is a BOOKML language, there might be <book>, <chapter>, <section>, <title>, <paragraph>, <image>, <table>...
- Everyone knows which elements may appear in a web page
  - links, images, tables, forms...
  - [HTML Tutorial](#)

# Only tags are not enough

- The same link tag, different destinations
  - `<a href="http://here/">here</a>`
  - `<a href="http://there/">there</a>`
- Attribute
  - href in the above example is an attribute of `<a>`
  - Just consider tag as **type**. For example, phone and camera are two types of things. Then attribute is **specification**. For example, two cameras could have different resolutions.
    - yes, two different tags could have similar attributes
  - attributes are various, since they are not visible
    - `<a href="here/" target="_blank" title="cursor here">`
- Considerable tag-attribute combinations

# But we don't teach here

Too much time

Consulting is always needed



# What to learn for HTML in this class?

- You should know
  - what HTML is
  - which elements are available
  - where to consult
- Developing HTML is not programming but designing how to mark your document. There are several guidelines when marking.
  - **Correct:** in addition to look up services, there are validation services.
  - **Reasonable:** since the document is yours, there is no standard answer of marking. However, please make it reasonable. Many frequently used facilities have a common practice (see the next slide).
  - **Cooperating with CSS:** today HTML is just for architecture.
    - To represent a book, for example, HTML should be responsible for only the boundaries of chapters. Leave the color and typeface to CSS.

# Zen Garden

The Beauty of CSS Design

*A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.*

*Download the sample [html file](#) and [css file](#)*

## *The Road to Enlightenment*

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.



select a design:

**Under the Sea!** by Eric Stoltz

**Make 'em Proud** by Michael McAgnon and Scotty Reifsnnyder

**Orchid Beauty** by Kevin Addison

**Oceanscape** by Justin Gray

**CSS Co., Ltd.** by Benjamin Klemm

**Sakura** by Tatsuya Uchida

**Kyoto Forest** by John Politowski

# How to cooperate with CSS

- General tags (without logical meaning)
- `<div>`
  - block
  - `<div class="footer">`
- `<span>`
  - inline
  - `<span class="date">`



# Cascading Style Sheets (CSS)

- Sheets
  - list
- Style
  - how to render elements
- Cascading
  - the famous CSS selector
  - understanding this is a big step in web design

# CSS selector

Selector	Description	Example
<code>a</code>	select tag	<code>&lt;a&gt;</code>
<code>.date</code>	select class	<code>&lt;a class="date"&gt;</code> , <code>&lt;span class="date"&gt;</code> , ...
<code>a.date</code>	mixed	<code>&lt;a class="date"&gt;</code>
<code>#footer</code>	select id (faster)	<code>&lt;div id="footer"&gt;</code>
<code>div#footer</code>	(unnecessary)	<code>&lt;div id="footer"&gt;</code>
<code>#footer a.date</code>	cascading	<code>&lt;div id="footer"&gt;</code> <code>&lt;a class="date"&gt;selected&lt;/a&gt;</code> <code>&lt;a&gt;not selected&lt;/a&gt;</code> <code>&lt;/div&gt;</code> <code>&lt;a class="date"&gt;not selected&lt;/a&gt;</code>

# Selected something, then?

- Change almost anything on the selected elements
  - background color, border, text color, typeface, font size, margin, padding...
  - [CSS Tutorial](#)
  - [CSS 語法教學](#)
- Even switch between block and inline (**display**)
- More position schemes that HTML does not provide (**float** and **position**)
  - this ~~advanced~~ (very common today) layout is only in CSS
- It means you can use arbitrary tags in HTML. But in practice people would use correct tags and use such attributes for special purposes.

 Any Questions?



# Demo 示範

# Layout tips

## ■ Table is dead

- [A List Apart: Articles: Practical CSS Layout Tips, Tricks, & Techniques](#)

## ■ Responsive design

- [Responsive Web Design: What It Is and How To Use It](#)
- [Responsive Web Design: 50 Examples and Best Practices – DesignModo](#)
- [A List Apart: Articles: Responsive Web Design](#)
- [Responsive Web Design - Wikipedia, the free encyclopedia](#)

# Today's assignment

## 今天的任務

# Design static pages of your site

- Plan pages and the containing blocks. Use HTML to construct them. Use CSS to beautify them. You may add some sample text to make the designs look real.
  - inspiration → design → coding
  - at least a homepage containing links to all pages
  - remember to include your previous about me page
- Reference
  - Ultimate Guide to Website Wireframing
  - 30 Fresh Web UI, Mobile UI and Wireframe Kits
  - Google 文件範本
  - Free CSS Templates
- Your web site (<http://merry.ee.ncku.edu.tw/~xxx/cur/>) will be checked not before 23:59 10/7 (Sun). You may send a report (such as some important modifications) to me in case I did not notice your features.

# Tiddly Résumé

CONAN DOYLE

NORWOOD, LONDON  
T: 03 069 94072  
E: conan.doyled@...



OBJECTIVES

EXPERIENCE

JAN 1890 - Present  
FREELANCE, LONDON

JAN 1870 - DEC 1889  
COAL TRIMMER, DOYLE+SONS, BRIGHTON

MODERN

LEGAL  
ARCHITECT  
WEB DEVELOPER

OPTIONS >>

- SAVEBACKUPS
- NORD (RTF)
- PLAINTEXT

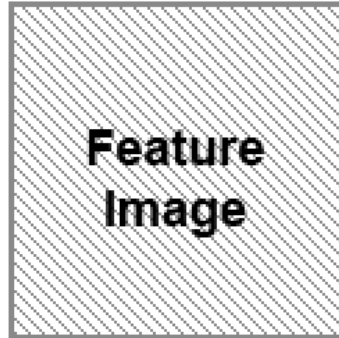
BACK  
STAGE

STAGE  
CSS

TYPE,  
CRE-AT

**My App**

[Sign Up](#) [Login](#)



**Feature  
Image**

## **App Tagline**

Description  
of feature

Description  
of feature

Description  
of feature

Description  
of feature

**Box 1 heading**

**Box 2 heading**

**Box 3 heading**

**Box 4 heading**

Logo

Primary Nav

Column 1

Column 2

Aside

# Tip

- The first thing you do in each exercise
  - `cd public_html`
  - `cp -r ex1 ex2` ← clone ex1 to ex2
  - `ln -fsT ex2 cur` ← make cur point to ex2
- Indent / 縮排 (in vi)
  - `gg=G`



# Appendix

## 附錄

# HTML5

- Very advanced technology: won't be standard before 2012
- Multimedia: `<audio>`, `<canvas>`, `<video>`...
- Interactive form (without Javascript)
- Storage
- Backward compatibility: compatible older HTML and even XHTML
- Semantic: `<header>`, `<footer>`, `<nav>`, `<aside>`...
- 今天就用 HTML5 的新語意標籤
- On the other hand, CSS3 is just more fancy

# CSS specificity

ul#nav li.active a

Style  
attribute

0

ID

1

Class,  
psuedo-class,  
attribute

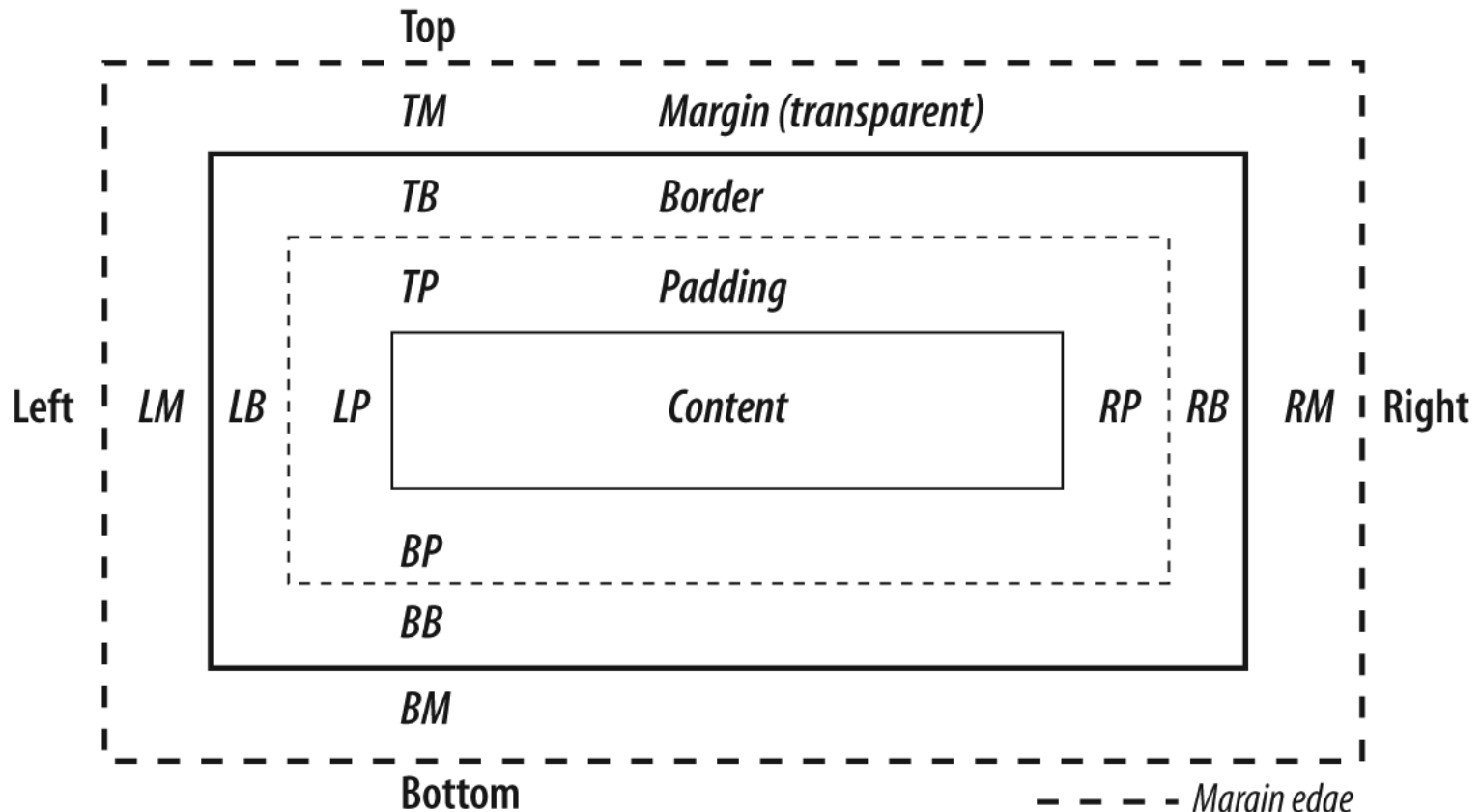
1

Elements

3

<http://css-tricks.com/855-specifics-on-css-specificity/> <http://net.tutsplus.com/tutorials/html-css-techniques/quick-tip-understanding-css-specificity/>

# CSS box model



--- Margin edge  
—— Border edge  
- - - - Padding edge  
—— Content edge

[http://ofps.oreilly.com/titles/9780596516482/language\\_and\\_browser\\_utilities.html](http://ofps.oreilly.com/titles/9780596516482/language_and_browser_utilities.html) <http://spyrestudios.com/css-in-depth-margins-padding-the-box-model/>  
<http://www.w3.org/TR/css3-box/> <http://www.w3.org/TR/CSS2/box.html>

# CSS float and position

- float is neither block nor inline
  - #sidebar { float: left; width: 100px; }
- In general, block and inline elements obey a main flow
  - #element\_id { position: relative; left: 100px; top 100px; }
- Use absolute to position elements precisely
  - #element\_id { position: absolute; left: 100px; top 100px; }
- Use fixed to pin elements on screen
  - #element\_id { position: fixed; left: 100px; top 100px; }
- [CSS In Depth Part 2: Floats & Positions](#)
- [CSS Float Theory: Things You Should Know](#)
- Please practice to experience the above concepts