



INTRODUCTION TO INTERNET

1. Internet

Internet គឺជាការតភ្ជាប់គ្នារវាងកុំព្យូទ័ររាប់លានជុំវិញពិភពលោកតាមរយៈ Protocol TCP/IP ដើម្បីទំនាក់ទំនងគ្នា ផ្ទេរធនធានឬឯកសារ ផ្ញើសារ ជជែកកំសាន្ត ។ ល។ ការភ្ជាប់ពី PC ទៅកាន់ Internet គឺយើងអាចប្រើប្រាស់ វិធីជាច្រើនអាស្រ័យទៅលើក្រុមហ៊ុនផ្តល់សេវា Internet (ISP) ។ បច្ចុប្បន្ននេះមានក្រុមហ៊ុនជាច្រើនកំពុងផ្តល់សេវា Internet នៅក្នុងព្រះរាជាណាចក្រកម្ពុជា ក្រុមហ៊ុននីមួយៗមានប្រព័ន្ធ និងវិធីផ្តល់សេវា Internet ខុសគ្នា ចំពោះតម្លៃវិញក៏ខុសគ្នាដែរ ដោយត្រូវបានគេបែងចែកទៅតាមវិធីនៃរបៀបទទួលសេវា ឬគិតទៅតាមល្បឿន ឬជាកញ្ចប់សេវា ។ សម្រាប់អ្នកប្រើគិតតាមចំនួន Kilobyte, ឬគិតជាល្បឿន តម្លៃមានភាពខុសគ្នាច្រើនគួរសម ។

ដើម្បីឲ្យមានភាពងាយស្រួលក្នុងការសម្រេចចិត្ត និងជ្រើសរើសសេវារបស់ក្រុមហ៊ុនណាមួយមកប្រើនោះ សូមពិនិត្យមើលទៅលើសេវាផ្សេងៗ របស់ក្រុមហ៊ុននីមួយៗ និងបញ្ហាបច្ចេកទេសខ្លះៗទាក់ទងដល់ការប្រើសេវា Internet របស់ក្រុមហ៊ុនទាំងនោះ ។

វិធីសាស្ត្រនៃការផ្តល់សេវា Internet

សេវា Internet ត្រូវបានផ្តល់ចែកចាយដោយវិធីជាច្រើនរួមមាន:

- ខ្សែកាបអុបទិច
- ខ្សែទូរស័ព្ទ (ខ្សែទូរស័ព្ទអចល័ត)
- ប្រើអង់តែន
- ខ្សែ Cable digital
- តាមរយៈ: wifi (ប្រព័ន្ធ wireless)
- តាមរយៈ: Mobile Broad band (ប្រើប្រព័ន្ធ ទូរស័ព្ទចល័ត)
- តាមរយៈ: Lease line និងតាមវិធីច្រើនទៀត

ក្រុមហ៊ុនខ្លះ មានផ្តល់វិធីច្រើនបែបសម្រាប់អ្នកជាវឲ្យអាចជ្រើសរើសបានទៅតាមសេចក្តីត្រូវការ ។ ចំណែកក្រុមហ៊ុនខ្លះទៀតបានរៀបចំជាកញ្ចប់សេវាជូនជាស្រេច ដោយរួមមានទាំងការកំណត់តម្លៃ និងកំណត់ចំនួននៃការប្រើប្រាស់ទៅតាមតម្លៃដែលគេបានរៀបចំហើយ ។ មុនពេលជាវសេវាណា មួយសូមពិនិត្យមើលញ្ញាខាងក្រោម:

* ចំនួន Kilobyte ដែលយើងទទួលបានគឺត្រូវបានគេគិតជាការប្រើដោយមានកំណត់ ឬគ្មានកំណត់ ។ ឧទាហរណ៍ បើគេកំណត់ឲ្យយើងចំនួន 1500 Mb ឲ្យយើងសម្រាប់រយៈពេល១ខែ យើងចាប់ត្រូវធ្វើការគណនាមួយពីចំនួនដែលយើងប្រើប្រាស់តើច្រើន ឬតិចជាងចំនួនកំណត់របស់គេនោះដែរឬទេ បើសិនជាការប្រើប្រាស់របស់យើងច្រើនជាងចំនួនដែលគេបានកំណត់នោះ យើង

ចាប់ត្រូវជ្រើសរើសយកកញ្ចប់សេវា ឬជ្រើសយកការប្រើដោយគ្មានកំណត់ជាជំនួស ិញ ដើម្បី ការពារការប្រើលើសពីចំនួនកំណត់ ។

មានក្រុមហ៊ុនជាច្រើនបានកំពុងធ្វើ Promotion ផ្តល់សេវា Internet គ្មានកំណត់ ការប្រើប្រាស់ ប្រើប្រាស់ ដោយមានបញ្ចុះតម្លៃយ៉ាងទាប មកនៅត្រឹមតែក្រោម ៣៥ដុល្លារ ក្នុងខែ ចំណែកល្បឿននៃ ការប្រើប្រាស់ ិញគឺចន្លោះ 128kb/64kb (ទំហំនេះត្រូវយកមកចែកនឹង៨ នៅយើងនឹងបានតួលេខ ល្បឿនពិតប្រាកដមិនខាន) ។

- 128kb÷8= 16kb (upload) (អាចតិចឬច្រើនជាងនេះ)
- 64kb÷8= 8kb (Download) (អាចតិចឬច្រើនជាងនេះ)

ទោះជាល្បឿនកំណត់គេបានដាក់មកចំនួនប៉ុន្មានក៏ដោយ វានឹងមិនពិតទេ ដើម្បីឲ្យដឹងពី ចំនួនពិត គឺត្រូវចែកចំនួនល្បឿននោះជាមួយ ៨ នៅយើងនឹងបានតួលេខល្បឿនមិនខាន តែវា មិនមែនស្ថិតនៅក្នុងកម្រិតល្បឿន ដែលយើងចែកឃើញនោះទាំងស្រុងឡើយ គឺវាឡើងវាចុះច្រើន ជាង ឬតិចជាងនេះរកកំណត់មិនបានឡើយ ។ យើងពិបាកកំណត់ថា វានឹងមានល្បឿនមួយស្មើ បានណាស់ពីព្រោះល្បឿនរបស់ Internet អាស្រ័យលើភក្តាច្រើនយ៉ាងណាស់។

ខាងក្រោមនេះគឺជាក្រុមហ៊ុនផ្តល់សេវា Internet



2. History of Internet

ការចំណាត់ចែងដំបូងរបស់អ៊ីនធើណែត គឺនៅដើមទសវត្សទី៧០ដោយក្រុមភ្នាក់ងារ ស្រាវជ្រាវ គំរោងជាន់ខ្ពស់ស្ថិតនៅក្នុងក្រសួងការពារជាតិរបស់សហរដ្ឋអាមេរិចដែល យើងហៅ កាត់ថា DARPA (Defense Advanced Research Projects Agency) ពីដំបូងពួកគេគ្រាន់តែសិក្សា សុវត្ថិភាពបច្ចេកវិទ្យាទំនាក់ទំនងពីកុំព្យូទ័រមួយទៀត តែប៉ុណ្ណោះ។

២០ ឆ្នាំក្រោយមក អ៊ីនធើណែតត្រូវបានគេភ្ជាប់ជាបណ្តាញ Network សំរាប់ប្រើប្រាស់ផ្នែក យោធា សាកលវិទ្យាល័យ និង ស្ថាប័ន នានា ការភ្ជាប់បណ្តាញអ៊ីនធើណែតជាដំបូងគឺនៅក្នុង ប្រទេសអូស្ត្រាលីក្រោយមកក៏បានពង្រីកទូទាំងពិភពលោកដែលធ្វើអោយក្លាយទៅជាអ៊ីនធើណែតពិត ប្រាកដមួយមានអ្នកប្រើប្រាស់ច្រើននិងល្បីស្មោះសាយពាសពេញពិភពលោក។

3. សារប្រយោជន៍នៃការប្រើប្រាស់អ៊ីនធឺណិត

- ស្រាវជ្រាវឯកសារ

នៅលើអ៊ីនធឺណិត កន្លែងព័ត៌មានជាច្រើនគ្រប់ផ្នែកទាំងអស់ដូចជាឯកសារដែលទាក់ទងទៅនឹងមុខវិជ្ជានីមួយៗតាមឯកទេសរបស់អ្នកសិក្សា ស្រាវជ្រាវ ជាពិសេសទៅទៀតវាក៏ផ្តល់ធនធានជាច្រើនដូចជា Software សំរាប់អោយយើង Download ដោយបង់លុយ ឬ Download ដោយឥតលុយ ។ ឯកសារនៅលើអ៊ីនធឺណិតមានលក្ខណៈទូលំទូលាយនិងទាក់ទិតជាង ឯកសារដែលពេញមួយ ដោយក្រដាសព្រោះទោះបីជាអ្នកស្រាវជ្រាវនៅទីណាក៏ដោយ ក៏អាចស្រាវជ្រាវបានពេញ ២៤ ម៉ោង អោយតែកុំព្យូទ័រនៅភ្ជាប់ជាមួយប្រព័ន្ធ អ៊ីនធឺណិត។

- កំសាន្ត (Entertainment)

អ៊ីនធឺណិតមិនត្រឹមតែផ្តល់ព័ត៌មានអោយយើងប៉ុណ្ណោះទេថែមទាំងអាចអោយយើងលេង Game ដោយឥតគិតថ្លៃដូចជា លេង អុក បាវ់ទាត់ ប្រណាំងឡាន ម៉ូតូ ទស្សនា រូបភាព ស្តាប់បទចម្រៀង ស្តាប់វិទ្យុ មើលឃុតខ្លះៗនៃខ្សែភាពយន្ត និង រូបភាព ស្រស់ស្អាតជាច្រើន។ល។

- ផ្ញើសារ

ការភ្ជាប់បណ្តាញអ៊ីនធឺណិត អាចអោយយើងផ្ញើសារទៅមនុស្សជុំវិញពិភពលោកដោយគ្រាន់តែស្គាល់ Email address របស់អ្នកដែលយើងផ្ញើទៅ។ បច្ចុប្បន្នការផ្ញើអាចអោយ យើងផ្ញើពី Email Address មួយទៅ Email address មួយទៀតទៅទូរស័ព្ទដៃ។ Web Site បង្កើត Email Address ដោយឥតគិតថ្លៃ មានដូចជា:

- <http://www.yahoo.com>
- <http://www.hotmail.com>
- <http://www.gmail.com>
- <http://www.mail.com>

- ការពិភាក្សា (Discussion)

អ៊ីនធឺណិតអាចអោយលោកអ្នក ធ្វើការពិភាក្សាជាមួយប្រជាជនជុំវិញពិភពលោក អំពីបញ្ហាអ្វីមួយ ដោយការប្រើប្រាស់ បណ្តាញសង្គមដូចជា: Facebook, twitter រឺក៏ Video Calling (Skype) ។

- បំរើសេវាកម្ម

នៅប្រទេសជប៉ុនល្បីនាអ៊ីនធឺណិតជួយសំរួលការងារជាច្រើនដូចជាការប្រើប្រាស់នូវប្រព័ន្ធ E_Commerce, Online Shopping, E_business, E_learning..... ទាំងនេះអាចកាត់បន្ថយក្នុងការធ្វើដំណើរដោយមធ្យោបាយណាមួយ។

4. ធាតុប្រឡូកទេស

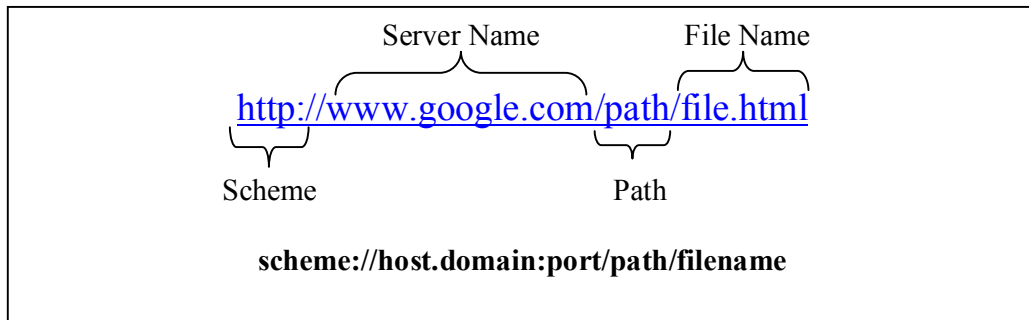
- Web Browser គឺជាកម្មវិធីសំរាប់បង្ហាញ Web Page



	Firefox Firefox is a web browser from Mozilla. It was released in 2004 and is one of the most popular browsers today.
	Netscape Netscape was the first commercial web browser. Netscape was released in 1994, but gradually lost its popularity to Internet Explorer. The development of Netscape officially ended in February 2008.
	Opera Opera is a free, fast, small, and standards-compliant web browser. Opera is the preferred browser for a number of small devices like mobile phones and hand-held computers.
	Google Chrome Google Chrome is a free, open-source web browser developed by Google. It was released in 2008, and has grown to be one of the most popular browsers today.
	Apple Safari Safari is the default web browser for Mac systems. Safari is known for its sleek design.

- **Web site** គឺជាអសយដ្ឋាន (Address) ដែលស្ថិតនៅលើ Internet ឧទាហរណ៍ <http://www.usea.edu.kh> ។ រាល់ Website ទាំងអស់កំណត់ដោយប្រើប្រាស់អាសយដ្ឋានពិតប្រាកដ ត្រូវបានហៅថា Uniform Resource Locator (URL) ។

Uniform Resource Locator (URL) គឺសំដៅលើទីតាំង (path) របស់ឯកសារនៅលើប្រព័ន្ធ Internet ។ នៅពេលអ្នកប្រើប្រាស់រង់ចូលទៅ Website ណាមួយ រឺចង់រក ឯកសារណាមួយនៅលើ Internet នៅត្រូវវាយ នូវ Address (<http://www.google.com/>) ដែល Address នេះយើងហៅថា URL ។



- **scheme** - defines the **type** of Internet service. The most common type is **http**
- **host** - defines the **domain host** (the default host for http is **www**)
- **domain** - defines the Internet **domain name**, like w3schools.com
- **:port** - defines the **port number** at the host (the default port number for http is **80**)
- **path** - defines a **path** at the server (If omitted, the document must be stored at the root directory of the web site)
- **filename** - defines the name of a document/resource

Common URL Schemes

The table below lists some common schemes:

Scheme	Short for....	Which pages will the scheme be used for...
http	HyperText Transfer Protocol	Common web pages starts with http://. Not encrypted
https	Secure HyperText Transfer Protocol	Secure web pages. All information exchanged are encrypted
ftp	File Transfer Protocol	For downloading or uploading files to a

	website. Useful for domain maintenance
file	A file on your computer

- **Home Page** គឺជា Page ដែលងាយដំបូងគេ នៅពេលដែលយើងវាយ Web site ។ ជាទូទៅវាជា Page ដំបូងដែលងាយដំបូងអំពីព័ត៌មានសំខាន់ៗសំរាប់ធ្វើអោយអ្នកស្រាវជ្រាវមានការចងចាំមូលដ្ឋាននិងអាចចូលទៅកាន់ Page បន្តបន្ទាប់ទៀត។
- **Web Page** គឺជា Page នីមួយៗដែលស្ថិតនៅក្នុង Web Site ។ វាជា File ដែលមានឈ្មោះនិង extension យ៉ាងច្បាស់លាស់ សំរាប់ផ្ទុកកន្លែងព័ត៌មាន និងយាយូរមទៅវាប្រៀបបានទៅនឹងទំព័រនីមួយៗនៅក្នុងសៀវភៅ
- **Web Server** គឺជាកម្មវិធីដែលដំណើរការនៅលើ Server ដើម្បីរង់ចាំទូលសំណើពី browser និងផ្តល់ទិន្នន័យភាពទៅ អោយ browser វិញ។
- **Intranet** គឺជាការតភ្ជាប់គ្នារវាង កុំព្យូទ័រជាច្រើនតាម TCP/IP ដែលមានលក្ខណៈ : ជាឯកជន
- **Internet Service Provider** គឺជាក្រុមហ៊ុនសំរាប់ផ្តល់សេវាអ៊ីនធឺណែត។
- **E-Mail** មកពីពាក្យថា Electronic Mail មានន័យថាគឺជាសារអេឡិចត្រូនិក។
- **Host** :គឺជាកុំព្យូទ័រមួយដែលធ្វើការតភ្ជាប់ជាមួយប្រព័ន្ធ អ៊ីនធឺណែត។
- **File Transfer Protocol** គឺជា Protocol ប្រើសំរាប់ផ្ទេរ File ទៅ និង មករវាង Client និង Remote Server ដោយឆ្លងកាត់បណ្តាញ Network មួយប្រើ TCP/IP
- **What is the World Wide Web?**
 - The Web is a network of computers all over the world.
 - All the computers in the Web can communicate with each other.
 - All the computers use a communication protocol called HTTP.
- **How does the WWW work?**
 - Web information is stored in documents called web pages.
 - Web pages are files stored on computers called web servers.
 - Computers reading the web pages are called web clients.
 - Web clients view the pages with a program called a web browser.
 - Popular browsers are Internet Explorer and Firefox.
- What is a Domain Name គឺជា A domain name is a unique name for a web site ដូចជា **usea.edu.kh, fbis.edu.kh, khmerweekend.com** ។



INTRODUCTION TO HTML

1. តើ HTML គឺជាអ្វី?

HTML គឺជាភាសា សំរាប់បង្កើតគេហទំព័រ (Website)។ HTML មកពីពាក្យ Hyper Text Markup Language វាមិនមែនជាភាសា Programming Language ទេ តែវាជាភាសា Markup Language ។ Markup Language គឺជាសំនុំនៃ Markup Tags ហើយដែល HTML ប្រើប្រាស់ markup tags ដើម្បីកសាងគេហទំព័រ ហើយធ្វើការបង្ហាញដោយ Web Browser ។ Web page កើតឡើងដោយទំនាក់ទំនងរវាង Page ជាច្រើន Link ទៅវិញទៅមកសំរាប់បង្ហាញ Contents ។ HTML ជាភាសាមួយដែលងាយស្រួល ក្នុងការសិក្សា និង ប្រើប្រាស់ ។

2. HTML Syntax and Rules

2.1 អ្វីជា HTML Tag?

HTML Tag ឬ HTML Markup គឺជាកំរងពាក្យ (keywords) ដែលរុំព័ទ្ធ ដោយ angle brackets “<” និង “>” ។

ឧទាហរណ៍ <html>,<head>,<body>

Tag មានពីរប្រភេទ គឺ

- Empty tag : មានតែ tag ចាំផ្តើមពុំមាន tag បញ្ចប់ទេ

ឧទាហរណ៍:
, <hr />, ...

Tip: In XHTML, all elements must be closed. Adding a slash inside the start tag, like
, is the proper way of closing empty elements in XHTML (and XML).

- Containers tag ជា tag ពេញលេញដែលមាន tag ចាំផ្តើម និង បញ្ចប់

ឧទាហរណ៍: <p> This is a paragraph </p>

- <p> tag ចាំផ្តើម

- </p> tag បញ្ចប់

Tip: ប្រើប្រាស់អក្សរតូចសំរាប់ Tags

HTML tags មិនប្រកាន់តួអក្សរទេ: <P> មានន័យស្មើ <p>។ web sites ភាគច្រើនគឺប្រើប្រាស់ HTML tags អក្សរធំ។ យើងគួរតែប្រើប្រាស់ tag អក្សរតូចព្រោះ World Wide Web Consortium (W3C) បានផ្តល់អនុសាសន៍អោយប្រើអក្សរតូចក្នុង HTML 4, ហើយក៏ទាមទារអោយប្រើ tag អក្សរតូច ក្នុង XHTML ។

2.2 លក្ខណៈរបស់ tag (Attribute)

Attribute វាស្ថិតនៅក្នុង Empty ឬ Beginning tag(Start tag) តែប៉ុណ្ណោះ។ គ្រប់ប្រភេទរបស់ tag សុទ្ធ តែមាន លក្ខណៈ (attribute) ដើម្បីធ្វើការកែតម្រូវទ្រង់ទ្រាយ តាមតម្រូវការដែលគេចង់បាន។

```

<tag syntax: name = "value"> .....</tag>

```

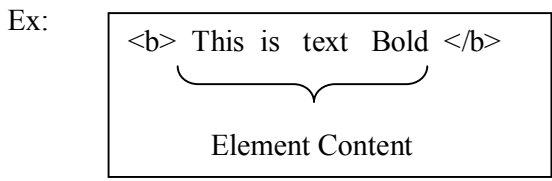
ឧទាហរណ៍ USEA Website
 ↑ ↑
 name = "value"

Tip: ប្រើប្រាស់អក្សរតូចសំរាប់ Attributes

- ➔ Attribute គឺជានិច្ចកាលមកព្រមគ្នារវាង name និង value
- ➔ Attribute names and attribute values are មិនប្រកាន់តួអក្សរ. However, the World Wide Web Consortium (W3C) បានផ្តល់អនុសាសន៍អោយប្រើប្រាស់អក្សរតូចសំរាប់ attributes និង attribute values ក្នុង HTML 4 ។

➔ Attribute Values គួរតែ ដាក់ Quote

2.3 Element Content: ចំបាប់ត្រូវតែមាន start tag និង ending tag ។ ព័ត៌មាននៅចន្លោះ tag ទាំងពីរយើងហៅថា Element Content ។

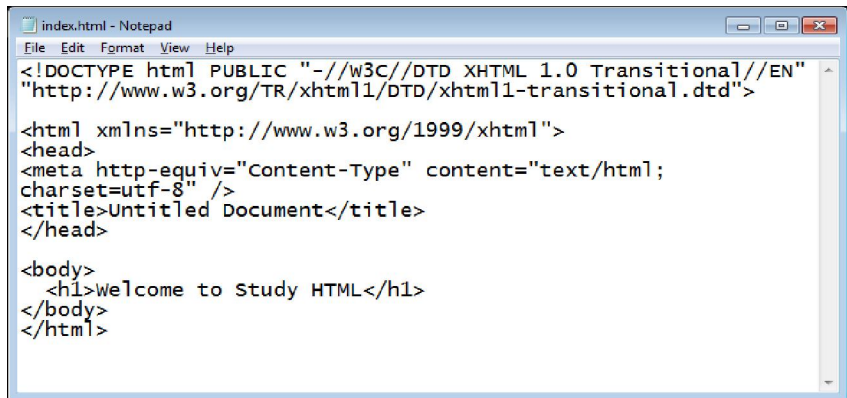


3. ការបង្កើតសេរីសរសេរក្នុងភាសា HTML

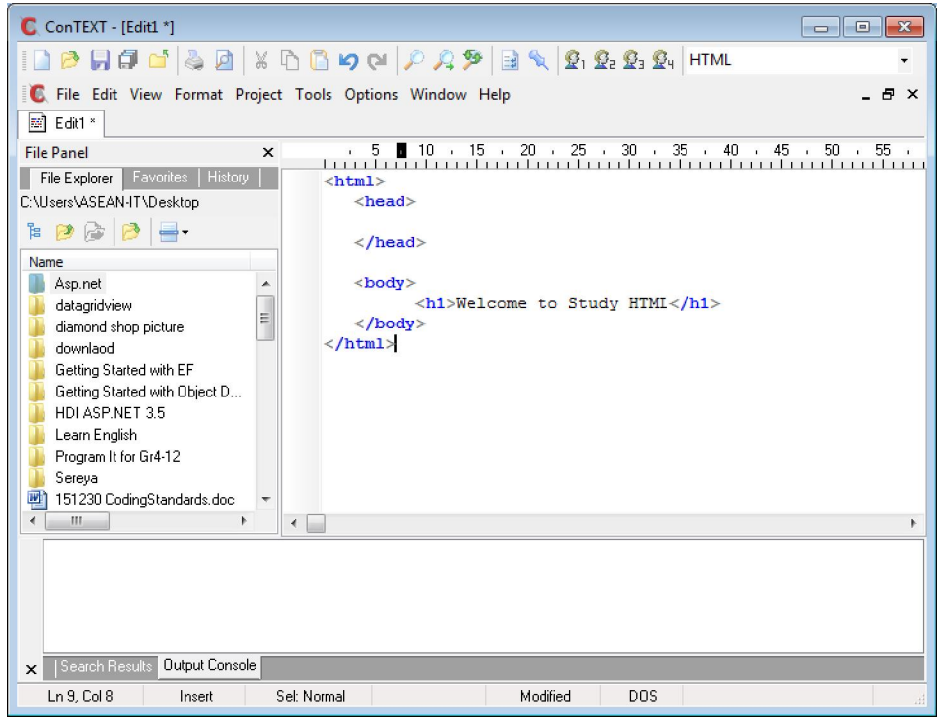
ភាសា HTML គេសរសេរនៅក្នុង Text Editor ដូចជា Notepad , Wordpad, ConTEXT ...។ Text Editor ដែលគួរឲ្យយកចិត្តទុកដាក់ប្រើ និង មានភាពងាយស្រួលគឺ Notepad ។ យើងក៏អាចកសាងគេហទំព័រតាម Web Editor ដែលមានភាពងាយស្រួលព្រោះវាមាន Tool ស្រាប់ៗដោយមិនចាប់ចំណុច នេះទេ Web Editor មានដូចជា Microsoft Frontpage, Macromedia Dreamweaver and Visual Studio

➔ របៀបបើកកម្មវិធី Notepad

ចុច Start → Programs → Accessories → Notepad



➔ សេសភាសា HTML តាមកិច្ចវិធី ConTEXT Editor (<http://www.contexteditor.org/>)

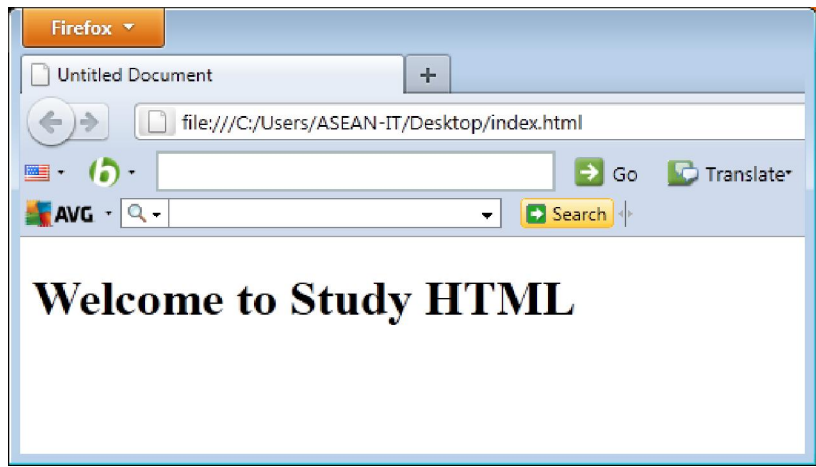


➔ របៀបរក្សាឯកសារថ្មីដើម្បីបង្ហាញលើ Browser

នៅក្នុង Program Notepad or ConTEXT ចុចFile ចុចលើ Save As

- ជ្រើសរើសទីតាំងសំរាប់អោយ File ស្ថិតនៅ
- នៅក្នុងប្រអប់ File Name: វាយឈ្មោះ File និង extension .html ឬ .htm
ឧទាហរណ៍ index.html
- ចុច Save

➔ បង្ហាញ web page លើ Browser



4. Structure of HTML

```

<html>
  <head>
    .....
  </head>
  <body>
    .....
  </body>
</html>

```

- ❖ <html> គឺជា Tag ដំបូងដែលយើងប្រើប្រាស់នៅក្នុង HTML Document ។ គ្រប់ HTML Document ត្រូវចាប់ផ្តើមដោយ <html> និងបញ្ចប់ដោយ </html>
- ❖ <head> section ជាកន្លែងសំរាប់ ផ្ទុកចំណងជើងនៃ Page និងបង្កើតទំនាក់ទំនងរវាង HTML Document និង file ដូចជា Script និង CSS ។ វាចាប់ផ្តើមដោយ <head> និង បញ្ចប់ដោយ </head>
ឧទាហរណ៍: <link href="main.css" rel="stylesheet" type="text/css" /> សំរាប់ភ្ជាប់ ទៅកាន់ Stylesheet ខាងក្រៅ។
- ❖ <body> section ជាផ្នែកខ្លួននៃ HTML Document សំរាប់ផ្ទុក កន្លែង:
 - អត្ថបទ(text)
 - Link
 - រូបភាព picture,
 - វីដេអូ (video)
 - ចំរៀង(mp3)។



CREATING A WEB PAGE

1. ការបង្កើត អត្ថបទ HTML

1.1 ការប្រើប្រាស់ Header

ក្នុង HTML យើងអាចដាក់ចំណងបាន ៦ យ៉ាងគឺ <h> ដែល n មានតំលៃថា ១ ដល់ ៦
n: កាន់តែតូច អត្ថបទកាន់តែធំ

Syntax:
<h align="left | right | center"></h>

ឧទាហរណ៍:

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Untitled Document</title>
</head>
<body>
  <h1>Welcome to Study HTML</h1>
  <h2>Welcome to Study HTML</h2>
  <h3>Welcome to Study HTML</h3>
  <h4>Welcome to Study HTML</h4>
  <h5>Welcome to Study HTML</h5>
  <h6>Welcome to Study HTML</h6>
</body>
</html>
```



ចំណាំ: Browsers automatically add some empty space (a margin) before and after each heading.

1.2 ការប្រើប្រាស់ Paragraph

ជាធម្មតានៅក្នុងអត្ថបទជាភាសាអង់គ្លេសគឺចែកចេញជា paragraph ដែលមានន័យជាភាសាខ្មែរថា កថាខ័ណ្ឌ ចំណែកក្នុងភាសា HTML វិញក៏អនុញ្ញាតអោយយើងបែងចែកអត្ថបទនៅលើ website ជា paragraph បានផងដែរដោយប្រើប្រាស់ tag <p> ។

Syntax:
<p align="left | right | center | justify"></p>

ឧទាហរណ៍:

```

<html>
<head>
<title>Using Paragraph</title>
</head>
<body>
  <h1>C# Coding Standards and Best Programming
Practices</h1>
  <p>
    If you have a team of different skills and tastes, you
are going to have a tough time convincing everyone to
follow the same standards. The best approach is to have a
team meeting and developing your own standards
document. You may use this document as a template to
prepare your own document.
  </p>

```



1.3 ការចុះដើមបន្ទាត់

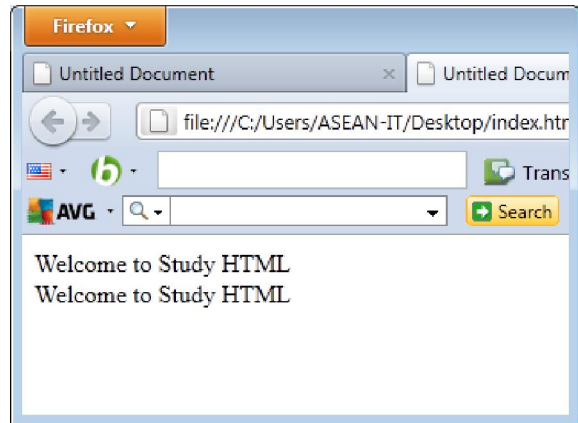
❖
 យើងប្រើសំបាប់ចុះបន្ទាត់នៅក្នុង អត្ថបទ។ ពេលសរសេរអត្ថបទនៅកន្លែង HTML Code រួចចុះបន្ទាត់តែម្តងនៅពុំមានឥទ្ធិពលនៅពេលបង្ហាញនៅទេ ព្រោះ Web Browser ធ្វើការបកប្រែអត្ថបទដោយមិនគិតការចុះបន្ទាត់ទេវានឹងបង្ហាញអត្ថបទបន្តបន្ទាប់។

ឧទាហរណ៍

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Break line test</title>
</head>
<body>
  Welcome to Study HTML<br/>
  Welcome to Study HTML
</body>
</html>

```



❖ <nobr> Tag

<nobr>.....</nobr> វាមិនអនុញ្ញាតិអោយចុះបន្ទាត់ទោះបីជាប្រវែង អត្ថបទរបស់យើងវែងជាប្រវែងរបស់ Window ក៏ដោយ។

1.4 ការគូសនូវ

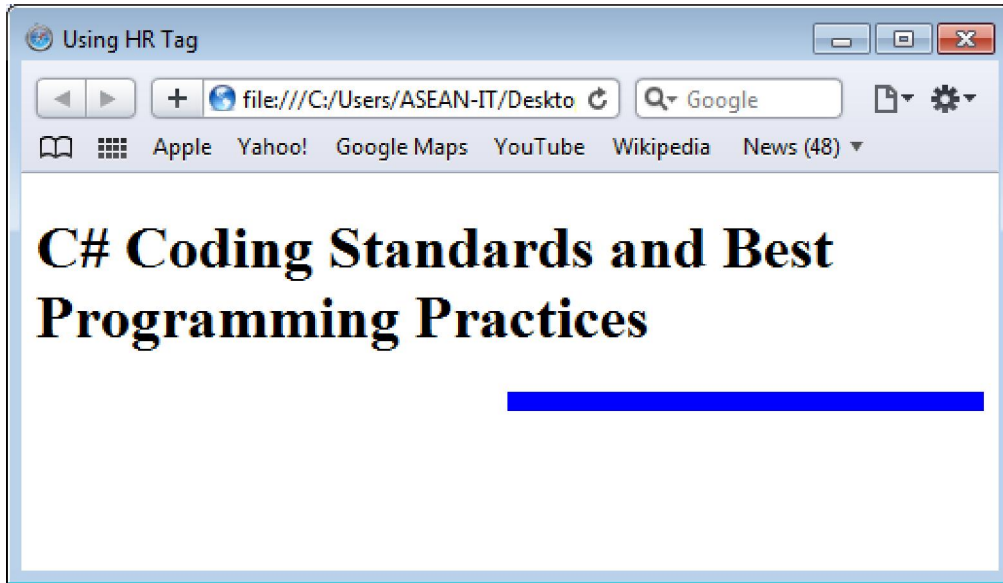
ដើម្បីគូសនូវដេកយើងប្រើប្រាស់ <hr /> tag

Attributes របស់ <hr /> tag

- align = "left | right | center"
- size = "x" ដែល x គឺជាកំរាស់នូវ គិតជា pixel
- width = "y" ដែល y គឺជាប្រវែងនូវ គិតជា pixel រឺ ភាគរយ(%)

- color = "color_name" | "color_code" ប្រើសំរាប់ដាក់ពណ៌អោយបន្ទាត់

Ex: <hr size="10" width="50%" color="blue" align="right" />



Note: ប្រវែងបន្ទាត់ Default របស់វាឆ្លើងប្រវែង paragraph(គិតជា pixel ឬ % ដោយប្រៀបធៀបជាមួយប្រវែង Window)

1.5 ការប្រើប្រាស់ PREFORMATTED TEXT(PRE)

<pre>.....</pre> ប្រើសំរាប់រៀបចំទំព័រនៃអត្ថបទតាម HTML file 1 នៅពេលដែល browser ជួបប្រទះ tag នេះវាបកប្រែអត្ថបទដោយរក្សានូវភាពដើមដំដែល

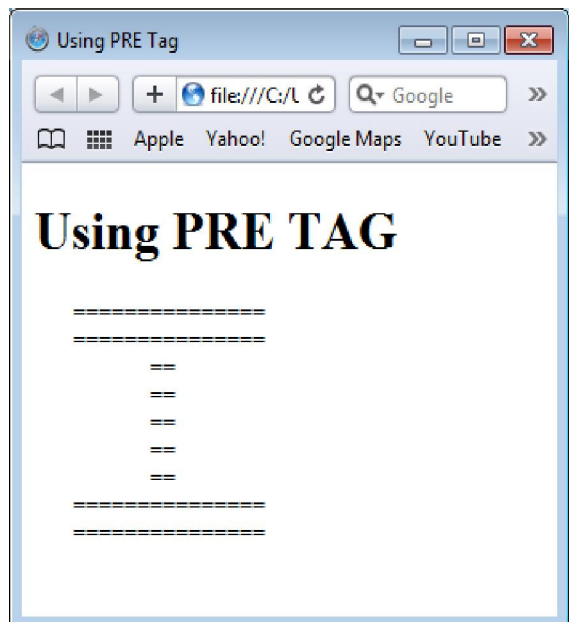
ឧទាហរណ៍

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Using PRE Tag</title>
</head>

<body>
  <h1>Using PRE TAG</h1>
  <pre>
  =====
  =====
  ==
  ==
  ==
  ==
  ==
  =====
  =====
  </pre> </body>
</html>

```



❖ HTML "Computer Output" Tags

Tag	Description
<code>	Defines computer code text
<kbd>	Defines keyboard text
<samp>	Defines sample computer code
<tt>	Defines teletype text
<var>	Defines a variable
<pre>	Defines preformatted text

❖ HTML Citations, Quotations, and Definition Tags

Tag	Description
<abbr>	Defines an abbreviation
<acronym>	Defines an acronym
<address>	Defines contact information for the author/owner of a document
<bdo>	Defines the text direction
<blockquote>	Defines a long quotation
<q>	Defines a short quotation
<cite>	Defines a citation
<dfn>	Defines a definition term

1.10 HTML Fonts

ក្នុងភាសា HTML version 4 និង ក្រោម 4 tag ត្រូវបានគេប្រើសំរាប់ផ្លាស់ប្តូរ ប្រភេទអក្សរ, ទំហំអក្សរ, ពណ៌អក្សរ។ ចំពោះ HTML ជំនាន់ 5 tag ត្រូវបានលុបចោល ក្នុង HTML 4, style sheets (CSS) ត្រូវបានប្រើប្រាស់កំណត់ទំរង់ website និងការបង្ហាញដូចជា ការផ្លាស់ប្តូរ អក្សរ font, text color, size, text align... ជាដើម។

ឧទាហរណ៍:

```
<font face="arial" size="5" color="red">
  This paragraph is in Arial, size 5, and in red text color.
</font>
```

- face = "font_name" កំណត់ប្រភេទអក្សរដូចជា "Arial", "Times New Roman"
- size = n សំរាប់កំណត់ទំហំអក្សរដែលមានចាប់ពី 1 ដល់ 7 ទំហំធំជាងគេគឺ 7
- color = color_name or color_code កំណត់ពណ៌អក្សរ
 - color_name = "red", "blue"
 - color_code = "#FFFFFF", "#000000"

1.11 Page Layout ខាងក្រោមនេះគឺជា Attribute សំរាប់កំណត់អោយ Page ទាំងមូល

```
<body bgcolor="color_name" | "color_code"
  background = "image_url"
  bgproperties = "fixed | scroll"
```

```

leftmargin = "x"
topmargin = "y"
marginwidth = "x"
marginheight = "y"
text = "color_name" | "color_code"
link = "color_name" | "color_code"
vlink = "color_name" | "color_code"
alink = "color_name" | "color_code">

```

</Body>

```

bgcolor = "color_name" | "color_code" កំណត់ពណ៌ផ្ទៃគេហទំព័រ
background = "image_url" កំណត់ផ្ទៃគេហទំព័រជារូបភាព
bgproperties = "fixed | scroll"
    fixed : កំណត់ទីតាំងនៃផ្ទៃទំព័រអោយនៅមួយកន្លែងទោះបីបំលាស់ទី Scrollbar ក៏ដោយ
    scroll : (default) ផ្ទៃទំព័រចល័តនៅពេលយើងបំលាស់ទី Scrollbar
leftmargin = "x" តំរិមទំព័រផ្នែកខាងឆ្វេងគិតជា Pixel
topmargin = "y" តំរិមទំព័រផ្នែកខាងលើគិតជា Pixel
marginwidth = "x" កំណត់ប្រវែងបណ្តាយទំព័រគិតជា Pixel
marginheight = "y" កំណត់ប្រវែងទទឹងទំព័រគិតជា Pixel
text = "color_name" | "color_code" កំណត់ពណ៌អោយអត្ថបទ
នៅក្នុងទំព័រទាំងមូល
link = "color_name" | "color_code" កំណត់ពណ៌អោយ Hypertext
vlink = "color_name" | "color_code" កំណត់ពណ៌អោយ Hypertext
ក្រោយពីយើងចុចលើរូបភាពហើយ។
alink = "color_name" | "color_code" កំណត់ពណ៌អោយ
Hypertext ក្នុងខណៈពេលកំពុងចុច។

```

Deprecated Tags and Attributes

ក្នុងភាសា HTML 4 មាន tags និង attributes ជាច្រើនត្រូវបានលុបចោល Deprecated
មានន័យថាវាទាំងនេះត្រូវបានដកចេញពីការប្រើប្រាស់ HTML ។

☀️ចំណាំ: ជៀសវាងការប្រើប្រាស់ deprecated tags and attributes!

ខាងក្រោមនេះគឺជា tags and attributes ដែលត្រូវជៀសវាងកុំប្រើ :

Tags	Description
<center>	Deprecated. កំណត់ centered content
 and <basefont>	Deprecated. កំណត់ HTML fonts

<code><s></code> and <code><strike></code>	Deprecated. កំនត់ strikethrough text
<code><u></code>	Deprecated. កំនត់ underlined text
Attributes	Description
<code>align</code>	Deprecated. កំនត់ the alignment of text
<code>bgcolor</code>	Deprecated. កំនត់ the background color
<code>color</code>	Deprecated. កំនត់ the text color

For all of the above: Use styles instead!



LISTS

Lists ជាធាតុមួយយ៉ាងសំខាន់ប្រើសំរាប់រៀបចំអត្ថបទរបស់យើង មានរបៀបរៀបរបរយ មានការបែងចែកតាមប្រភេទ ទិន្នន័យដើម្បីអោយ អ្នកអានងាយយល់ងាយមើលអត្ថបទ។ នៅក្នុង HTML គេចែក List សំខាន់ៗមានបីប្រភេទ គឺ

- Definition List
- Order List : List មានលេខរៀង
- Unorder List : List ប្រភេទជានិមិត្តសញ្ញា

1. **Definition Lists** : យើងប្រើ Definition List នៅពេលដែលយើងធ្វើការ អោយ និយមន័យ អោយទៅឃ្លានរឺក៏ពាក្យណាមួយ ព្រោះនៅពេលដែលយើងប្រើ <dl> tag អ្វីដែលនៅបន្ទាប់ពី <dl> tag គឺត្រូវចូលដើម បន្ទាត់ ដែលទំរង់បែបនេះគឺយើងប្រើច្រើនជាមួយនឹងការអោយនិយមន័យ។ គេអាច ហៅម្យ៉ាងទៀតថាជា Glossary List ព្រោះវាផ្តល់ទំរង់ដូចជា អត្ថបទក្នុង ហា នុក្រម។

រចនាសម្ព័ន្ធ នៃគំរូ List មានដូចខាងក្រោម

```

<dl>
  <dt>.....</dt>
  <dd>.....</dd>
  <dt>.....</dt>
  <dd>.....</dd>
</dl>

```

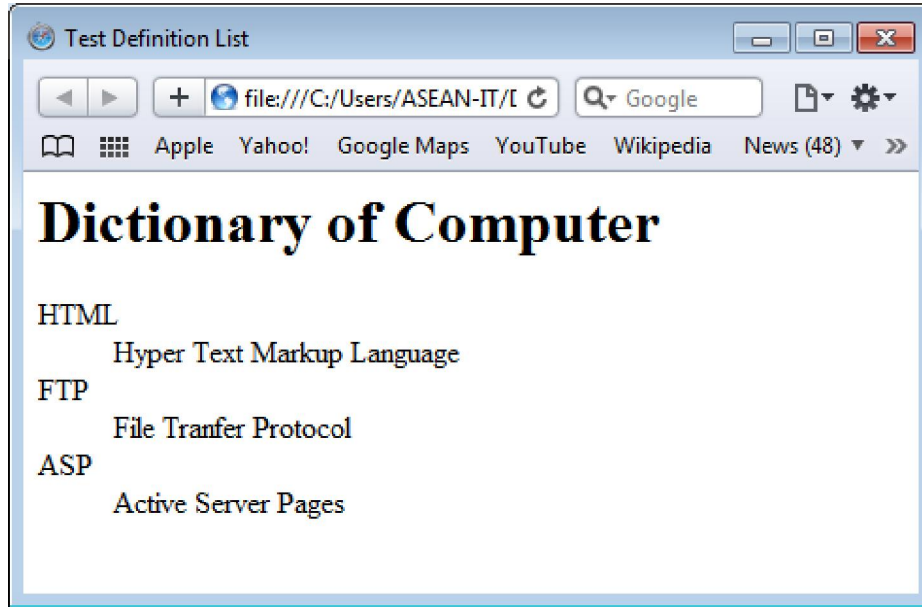
- <dl> : Definition List
- <dt> : Definition term
- <dd> : Definition data

ឧទាហរណ៍

```

<html>
  <head><title>Test Definition List </title></head>
  <body>
    <h1>Dictionary of Computer</h1>
    <dl>
      <dt> HTML </dt>
      <dd> Hyper Text Markup Language </dd>
      <dt> FTP </dt>
      <dd> File Tranfer Protocol </dd>
      <dt> ASP </dt>
      <dd> Active Server Pages </dd>
    </dl>
  </body>
</html>

```



2. Order List

 ប្រើចំពោះ List មានលេខរៀង ឬ លំដាប់ ។

រចនាសម្ព័ន្ធនៃគំរូ Order List មានដូចខាងក្រោម

```

<ol type = "x" start = " n" >
  <li type = "x" value = "m"> Item 1 </li>
  <li type = "x" value = "m"> Item 2 </li>
  .....
  .....
  <li type = "x" value = "m"> Item N </li>
</OL>

```

Attribute នៅក្នុង Order list មានដូចខាងក្រោម

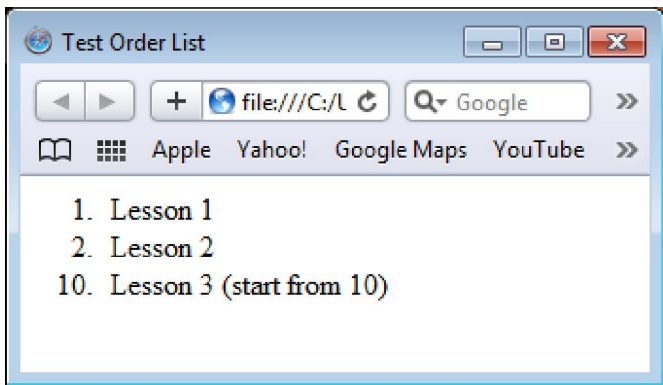
- type = "x"
 - x = 1 កំនត់លេខរៀងតាមលំដាប់ 1,2,3,....., n (Default)
 - x =A កំនត់លំដាប់ជាអក្សរក្រមធំ A, B, C,, Z
 - x = a កំនត់លំដាប់ជាអក្សរក្រមធម្មតា a, b, c,....., z
 - x = I កំនត់លំដាប់លេខរៀង រ៉ូម៉ាំង I, II, III,
 - x = i កំនត់លំដាប់លេខរៀង រ៉ូម៉ាំង i, ii, iii,
- start = "n" កំនត់ការចម្លើមដំបូង តាមប្រភេទ។
- value = "m" កំនត់ទីតាំងតំលៃរបស់Item តាមតំរូវការរបស់យើង។

ឧទាហរណ៍

```

<html>
<head>
  <title>Test Order List</title>
</head>
<body>
  <ol>
    <li> Lesson 1 </li>
    <li> Lesson 2 </li>
    <li value= 10> Lesson 3 (start from 10) </li>
  </ol>
</body>
</html>

```



3. Unordered List

ការប្រើប្រាស់ tag មានលក្ខណៈដូចគ្នាទៅនឹង ការប្រើប្រាស់ Ordered List ដែរ ខុសគ្នា ត្រង់ប្រភេទលំដាប់ Item ព្រោះ tag ប្រើប្រាស់និមិត្តសញ្ញាជំនួសលេខវិញ។

រចនាសម្ព័ន្ធ នៃគំរូ List មានដូចខាងក្រោម

```

<ul type = "value">
  <li type = "value"> Item 1 </li>
  <li type = "value"> Item 2 </li>
  .....
  .....
  <li type = "value"> Item N </li>
</ul>

```

Attribute នៅក្នុង Unorder List មានដូចខាងក្រោម
type = "value"

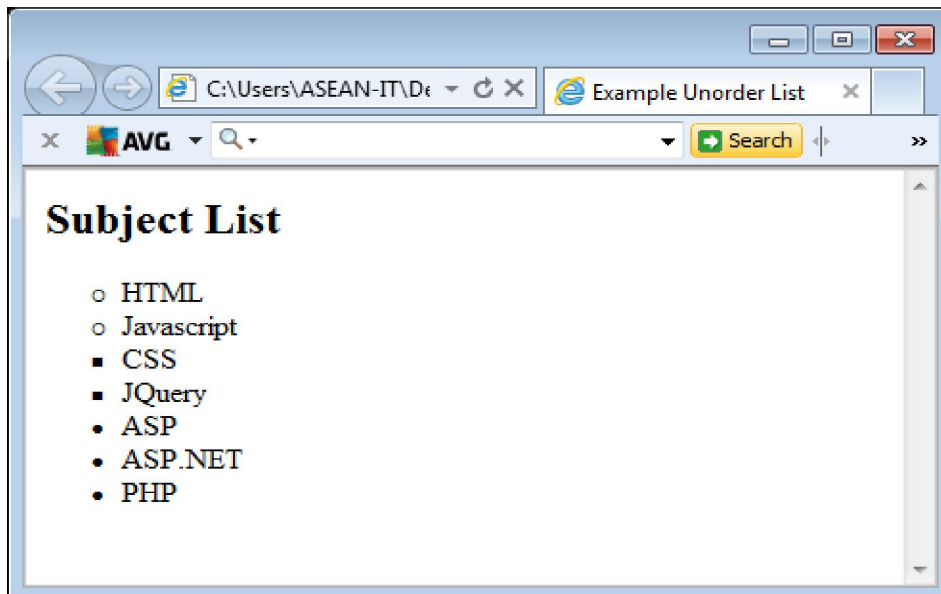
- value មានដូចខាងក្រោម
 - square រូបភាពជាការ៉េ
 - circle រូបភាពជារង្វង់ទទេ
 - disc រូបភាពជារង្វង់ពេញ

ឧទាហរណ៍

```

<html>
  <head>
    <title>
      Example Unorder List
    </title>
  </head>
  <body>
    <h2>Subject List</h2>
    <ul type="circle">
      <li> HTML </li>
      <li> Javascript </li>
      <li type="square"> CSS </li>
      <li type="square"> JQuery </li>
      <li type="disc"> ASP </li>
      <li type="disc"> ASP.NET </li>
      <li type="disc"> PHP </li>
    </ul>
  </body>
</html>

```



List Order

- disc
- circle
- square

4. បង្កើត List ក្នុង List (Nested List)

ទំរង់ទូទៅ:

```

<ol type= "1 | A | a | I | I" start= "n" >
  <li> ----- </li>
  <ul type= "disc | square | circle">
    <li> ----- </li>
    <li> ----- </li>
    <li> ----- </li>
  </ul>
  <li> ----- </li>
  <ol type= "1 | A | a | I | I" start= "n" >
    <li> ----- </li>
    <li> ----- </li>
    <li> ----- </li>
  </ol>
</ol>

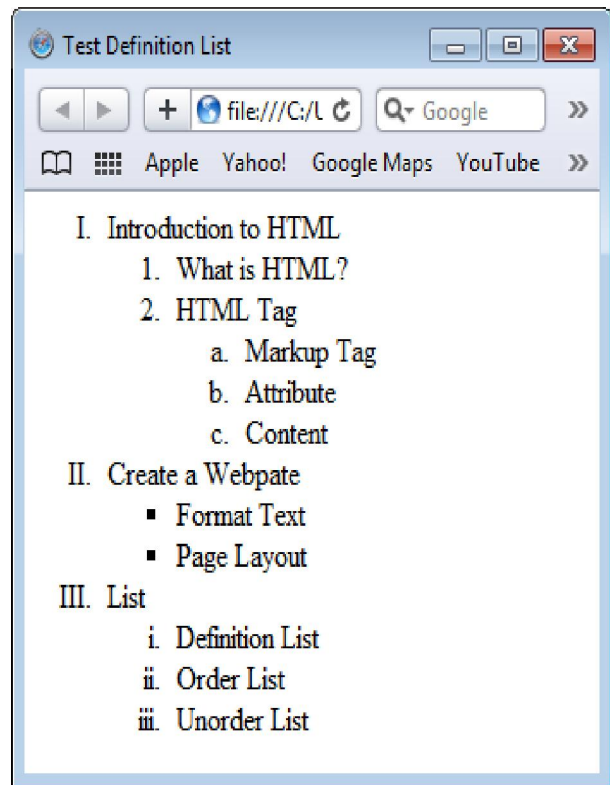
```

ឧទាហរណ៍:

```

<html>
  <head><title>Nested List </title>
  </head>
  <body>
    <ol type="I">
      <li>Introduction to HTML</li>
    </ol>
    <li>What is HTML?</li>
    <li>HTML Tag</li>
    <ol type="a">
      <li>Markup Tag</li>
      <li>Attribute</li>
      <li>Content</li>
    </ol>
    <li>Create a Webpate</li>
    <ul type="square">
      <li>Format Text</li>
      <li>Page Layout</li>
    </ul>
    <li>List</li>
    <ol type="i">
      <li>Definition List</li>
      <li>Order List</li>
      <li>Unorder List</li>
    </ol>
  </body>
</html>

```

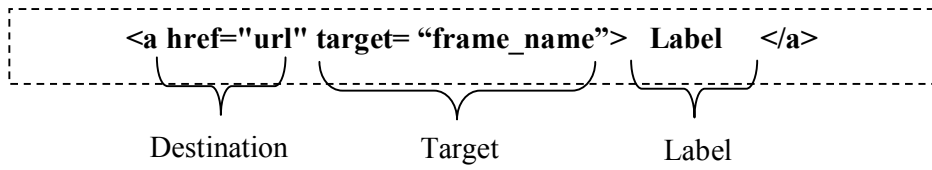




Links គឺឃើញមានភ្លើងគ្រប់ web pages ទាំងអស់។ links អនុញ្ញាតអោយអ្នកទស្សនា (visitor) ចុចលើ Links ដើម្បីទៅកាន់ page ណាមួយផ្សេងទៀត។ A Hyperlink or Link គឺជា ពាក្យ, បណ្តុំនៃពាក្យ រឺជា រូបភាពដែលអាចអោយយើងចុចលើវា ហើយនាំទៅកាន់ គេហទំព័រថ្មី ឬ new section ក្នុង current document។

Links គឺជា ចំនុចដ៏សំខាន់មួយនៅក្នុងការរចនាគេហទំព័រ គឺ វាអនុញ្ញាតអោយយើង ធ្វើការភ្ជាប់ពី ចំនុចមួយ ទៅចំនុចមួយទៀតនៅក្នុង Page តែមួយ មិនតែប៉ុណ្ណោះ យើងអាចធ្វើការ link ទៅកាន់ website មួយ ឬក៏ចំនុចណាមួយនៅក្នុង website មួយផ្សេងទៀតបាន។ ដើម្បីអាចប្រើប្រាស់ Links នៅក្នុង HTML បានយើងត្រូវការឈ្មោះ: File ឬ URL ដែលយើង ចង់ Link ទៅកាន់។

Syntax



Link មានបីផ្នែកសំខាន់គឺ

- Destination: គឺជា URL រឺ File ដែលភ្ជាប់ទៅនៅពេលអ្នកប្រើប្រាស់ចុច
- Label: ជាផ្នែកមួយដែលអ្នកប្រើប្រាស់ឃើញ និង អាចចុចបានដើម្បីអោយវាភ្ជាប់ទៅកាន់ Destination ។ វាអាចជាអត្ថបទ រឺរូបភាព
- Target : កំនត់ទីកន្លែងដែល Destination នឹងបង្ហាញ

1 ប្រើប្រាស់ HTML Link Tag <a>

1.1 ការ Link ទៅកាន់ page

នៅក្នុងការ Link ទៅកាន់ page ណាមួយ ចំនុចសំខាន់ដែលយើងត្រូវគិតដំបូងគឺ តើ page ដែលត្រូវ link ទៅមានឈ្មោះអ្វី? មានទីតាំងនៅកន្លែងណា? ព្រោះបើសិនជាយើងមិនស្គាល់ឈ្មោះ និង ទីតាំងទេនោះយើងប្រាកដជាមិនអាច link ទៅកាន់ page នោះបានឡើយ។

```
<a href = "filename.html" > Zone Click</a>
```

ឧទាហរណ៍

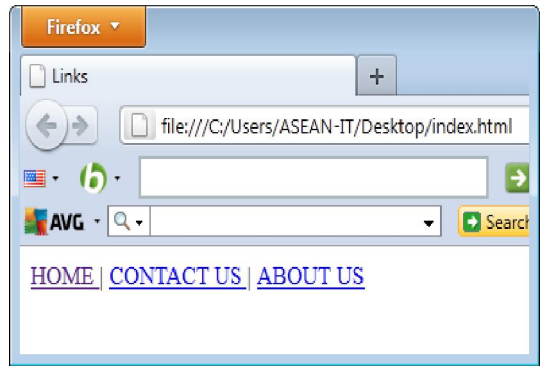
```
<a href = "aboutus.html"> About Us </a>
```

href = "aboutus.html" មានន័យថា link ទៅកាន់ page មួយមានឈ្មោះថា aboutus.html ។

```

<html>
  <head><title>Links </title>
  </head>
  <body>
    <a href="index.html">HOME </a> |
    <a href="contactus.html">CONTACT US </a> |
    <a href="/aboutus.html">ABOUT US </a>
  </body>
</html>

```



1.2 ការ Link ទៅកាន់គេហទំព័រណាមួយ

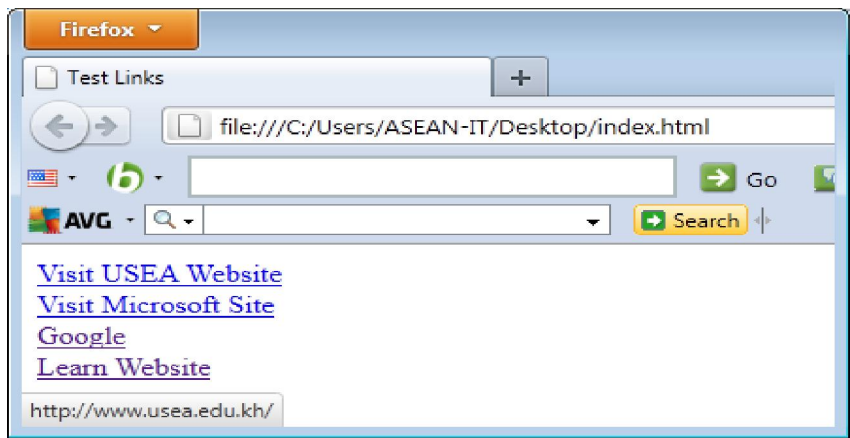
គឺជាការ link ទៅកាន់គេហទំព័រខាងក្រៅ។

ឧទាហរណ៍

```

<html>
  <head><title>Test Links </title>
  </head>
  <body>
    <a href="http://www.usea.edu.kh/"> Visit USEA Website </a> <br/>
    <a href="http://www.microsoft.com"> Visit Microsoft Site </a> <br/>
    <a href="http://www.google.com/"> Google </a> <br/>
    <a href="http://www.w3schools.com/"> Learn Website </a> <br/>
  </body>
</html>

```



1.3 ការ Link នៅក្នុង page

Anchor អាចជួយយើងអោយធ្វើការ link ទៅកាន់តំបន់ណាមួយជាក់លាក់ក្នុង page តែមួយបាន យ៉ាងលឿនទោះបីជាការ link នៅពីលើចុះមកក្រោម ឬក៏ពីក្រោមឡើងលើយ៉ាងណាក៏ដោយ។ ប្រសិនបើ យើងមានអត្ថបទច្រើន ហើយយើងមិនបានប្រើប្រាស់ anchor ជាជំនួយនោះទេ វានឹងនាំអោយ អ្នកទស្សនា មានការពិបាកក្នុងការមើលព័ត៌មាន ដែលជាហេតុធ្វើអោយ អ្នកទស្សនាបារម្ភ លែងចូលមក មើល គេហទំព័ររបស់យើងទៀត។

ក្នុងការប្រើ anchor យើងចាប់បង្កើតជាពីរដំណាក់កាល៖

ក. បង្កើតនូវចំនុច (anchor) ដើម្បីទុកសំរាប់ link មកកាន់

```
<a name= "anchor_name"> Label </a>
```

anchor_name : ជាឈ្មោះចំនុច

ឧទាហរណ៍

```
<a name="top"> Top Menu </a>
```

ខ. បង្កើត link ទៅកាន់ចំនុច (anchor)

```
<a href= "#anchor_name"> Label </a>
```

#anchor_name: កំនត់ឈ្មោះដែលបានបង្កើត anchor សំរាប់ link

ឧទាហរណ៍

```
<a href= "#top"> go to Top </a>
```

1.4 ការប្រើប្រាស់ Target Attribute

យើងប្រើ Target ដើម្បីកំនត់ទីកន្លែងដែល URL ត្រូវបង្ហាញ។

```
<a href= "url" target= "window_name">
```

target= "_self" បង្ហាញ URL នៅលើ Current window

target= "_blank" បង្ហាញ URL នៅលើ Window ថ្មី

1.5 ការប្រើប្រាស់ title Attribute

Title គឺជាឃ្លាដែលនឹងបង្ហាញនៅពេលដែលអ្នកទស្សនាយក mouse មកដាក់ចំណីឃ្លា ដែលត្រូវ link ។

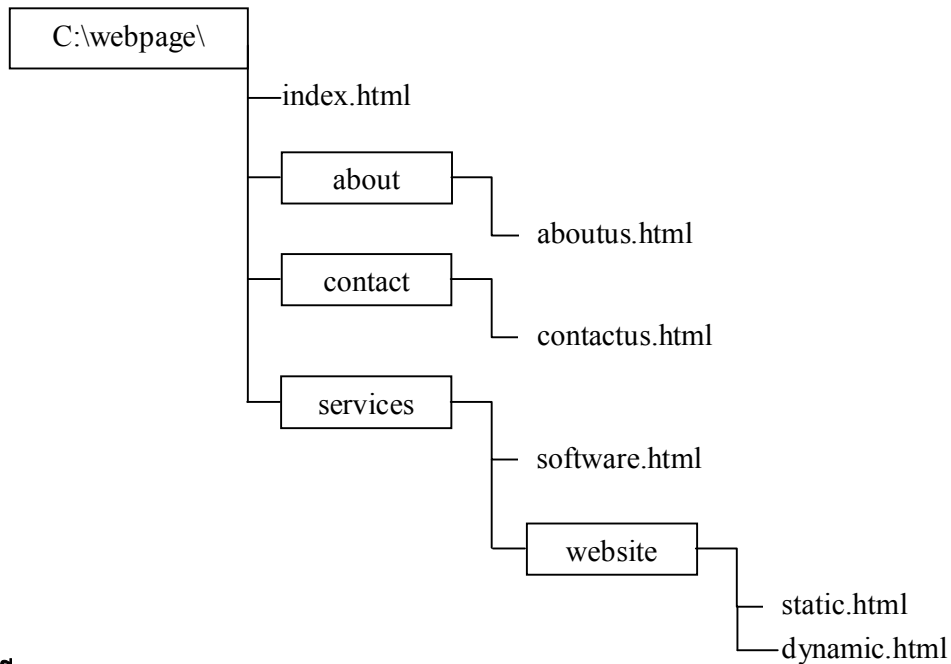
ឧទាហរណ៍

```
<a href="http://www.usea.edu.kh/" title= "Visit USEA Website"> USEA</a> <br/>
```

2. ការប្រើប្រាស់ Relative Link

យើងប្រើប្រាស់ Relative Link គឺងាយស្រួលណាស់ក្នុងការ link ដោយមិនចាប់ប្រើប្រាស់នូវ Path ពេញនោះទេយើងគ្រាន់តែប្រើនិមិត្តសញ្ញា “..” មានន័យថាថយមួយ Folder ។ នេះជាចំនុចសំខាន់បំផុតនៅពេលយក Files ទៅ Upload ឬ បើកនៅលើម៉ាស៊ីនកុំព្យូទ័រផ្សេង ទៀតព្រោះបើសិនជាយើងមិនប្រើវិធីនេះទេ ពេលនោះវាមិន ស្គាល់ Path ទេ នៅពេលយើង link ។

ឧទាហរណ៍: យើងមានរចនាសម្ព័ន្ធ ដែលមាន ទំរង់ដូចខាងក្រោម:



វិធី Link:

+ យើងនៅលើ Wed Page index.html ដើម្បី Link ទៅកាន់ aboutus.htm, contactus.html, software.html, static.html និង dynamic.html

```

<a href = “about/aboutus.html”> About Us </a>
<a href = “contact/contactus.html”> Contact Us </a>
<a href = “services/software.html”> Software </a>
<a href = “services/ website/static.html”> Static Page </a>
<a href = “services/website/dynamic.html”> Dynamic Page </a>
    
```

+ យើងនៅលើ Wed Page aboutus.html ដើម្បី Link ទៅកាន់ index.html, contactus.html, software.html, static.html និង dynamic.html

```

<a href = “../index.html”> Home </a>
<a href = “../contact/contactus.html”> Contact Us </a>
<a href = “../services/software.html”> Software </a>
<a href = “../services/ website/static.html”> Static Page </a>
<a href = “../services/website/dynamic.html”> Dynamic Page </a>
    
```

+ យើងនៅលើ Wed Page contactus.html ដើម្បី Link ទៅកាន់ index.html, aboutus.html, software.html, static.html និង dynamic.html

```

<a href = “../index.html”> Home </a>
<a href = “../about/aboutus.html”> About Us </a>
<a href = “../services/software.html”> Software </a>
<a href = “../services/ website/static.html”> Static Page </a>
<a href = “../services/website/dynamic.html”> Dynamic Page </a>
    
```

+ យើងនៅលើ Wed Page software.html ដើម្បី Link ទៅកាន់ index.html, aboutus.html, contactus.html, static.html និង dynamic.html

```

<a href = “../index.html”> Home </a>
<a href = “../about/aboutus.html”> About Us </a>
<a href = “../contact/contactus.html”> Contact Us </a>
<a href = “website/static.html”> Static Page </a>
    
```

```
<a href = “website/dynamic.html”> Dynamic Page </a>
```

+ យើងនៅលើ Wed Page static.html ដើម្បី Link ទៅកាន់ index.html, aboutus.html, contactus.html, software.html និង dynamic.html

```
<a href = “../index.html”> Home </a>
<a href = “../about/aboutus.html”> About Us </a>
<a href = “../contact/contactus.html”> Contact Us </a>
<a href = “../software.html”> Software </a>
<a href = “dynamic.html”> Dynamic Page </a>
```

+ យើងនៅលើ Wed Page dynamic.html ដើម្បី Link ទៅកាន់ index.html, aboutus.html, contactus.html, software.html និង static.html

```
<a href = “../index.html”> Home </a>
<a href = “../about/aboutus.html”> About Us </a>
<a href = “../contact/contactus.html”> Contact Us </a>
<a href = “../software.html”> Software </a>
<a href = “static.html”> Static Page </a>
```

3. ការបង្កើត Link ផ្សេងៗ

3.1 ការ link ទៅកាន់ FTP(File Transfer Protocol)

នៅពេលដែលយើងកំនត់យក FTP ជា protocol នៅក្នុង URL នៅ browser នឹងចាំផ្ដើមភ្ជាប់ ដោយស្វ័យប្រវត្តិ ទៅកាន់ទីតាំងដែលបានកំនត់ និងបញ្ជូន request file ។

ទំរង់ទូទៅ

```
ftp://hostname/directory_name/filename
```

ឧទាហរណ៍

```
<a href= “ftp://snsolution.net/link.html”> FTP Link </a>
```

3.2 ការ link ទៅកាន់ mail

យើងប្រើវាដើម្បីធ្វើការផ្ញើ email ទៅកាន់នរណាម្នាក់។ ពេលដែលយើងចុចលើ Label សម្រាប់ link ភ្លាមៗ នៅពេលនឹងធ្វើការហៅ កម្មវិធីសម្រាប់ផ្ញើសារភ្លាម។

ទំរង់ទូទៅ

```
mailto:username@hostname
```

ឧទាហរណ៍

```
<a href= “chanseriya@yahoo.com”> Mail to Seriya </a>
```

3.3 ការ link ទៅកាន់ Image File (Gif , JPEG, រឺ ភាពយន្ត)

គេអាច link ទៅកាន់រូបភាព ដោយប្រើ នៅពេលគេចុច វានឹងបង្ហាញ រូបភាព ។ លទ្ធផលនៃការ បង្ហាញ វាខុសគ្នាអាស្រ័យទៅ និង Browser ដែលប្រើប្រាស់។

ឧទាហរណ៍

```
<a href= “mypic.gif”> View my picture </a>
```

4. ការផ្លាស់ប្តូរ ពណ៌នៅពេល Link

ដើម្បីធ្វើការ ផ្លាស់ប្តូរ ពណ៌អក្សររបស់ link ដូចជា មុនពេល link, កំពុង link និង ក្រោយពេល link យើងប្រើប្រាស់ Attributes link, alink, vlink នៅក្នុង tag <body>។ សូមមើលទម្រង់ខាងក្រោម:

```
<body link = "rrggbb" alink = "rrggbb" vlink = "rrggbb">
.....
</body>
```

- link : ជាពណ៌ដំបូងរបស់ អត្ថបទ link នៅពេលដែលអ្នកទស្សនាមិនទាន់បើកមើល (Default ពណ៌ខៀវ)
- alink : ជាពណ៌ក្នុងខណៈពេលដែលអ្នកទស្សនាចុចទៅលើ អត្ថបទ link (Default ពណ៌ក្រហម)
- vlink : ជាពណ៌អត្ថបទ ក្រោយពីអ្នកទស្សនាបើកមើលរួច link ហើយ (Default ពណ៌ផ្កាឈូក)

Keyboard Shortcuts for Links

```
<a href = "url" accesskey="char" > Label </a>
```

យើងអាចប្រើប្រាស់ Character ជា Shortcut key សំរាប់ Link ដោយអ្នកប្រើប្រាស់គ្រាន់តែចុច ALT ជាមួយ Character ដែលយើងផ្ទេរតំលៃអោយ (ALT+char)។

```
<a href = "http://www.microsoft.com/" accesskey="m" > Microsoft site </a>
```

ជំពូក ៦ ការប្រើប្រាស់រូបភាព

រូបភាពជា ធាតុមួយដ៏ពិសេសដែលគួរឲ្យយកចិត្តទុកដាក់ប្រើនៅក្នុង ការបង្កើតគេហទំព័រដើម្បីតុបតែង អោយមានសោភ័ណភាពល្អ និង ទាក់ទាញអារម្មណ៍យ៉ាងខ្លាំងពីអ្នកទស្សនា។ រូបភាពដែល យើងប្រើក្នុងពេលចូលប្រើប្រាស់ប្រភេទណាស់ ចំណែកប្រភេទ នីមួយៗមាន លក្ខណៈ ពិសេសខុសគ្នាដែរ ដូចជា រូបភាពខ្លះមានចលនា ។ យើងអាចសំគាល់ប្រភេទ នីមួយៗ បានដោយសារ extension សូមមើលតារាងខាងក្រោម៖

Description	Extension
GIF image	.gif
JPEG image	.jpg or .jpeg
Portable Network Graphic	.png
Windows Bitmap Image	.bmp
TIFF image	.tiff
Encapsulate PostScript	.eps

ទំរង់ទូទៅ៖

```
<img src = "url"
border = "n"
alt = "Alternate Text"
width = "x"
height = "y"
align = "left | right"
hspace = "a"
vspace = "b" />
```

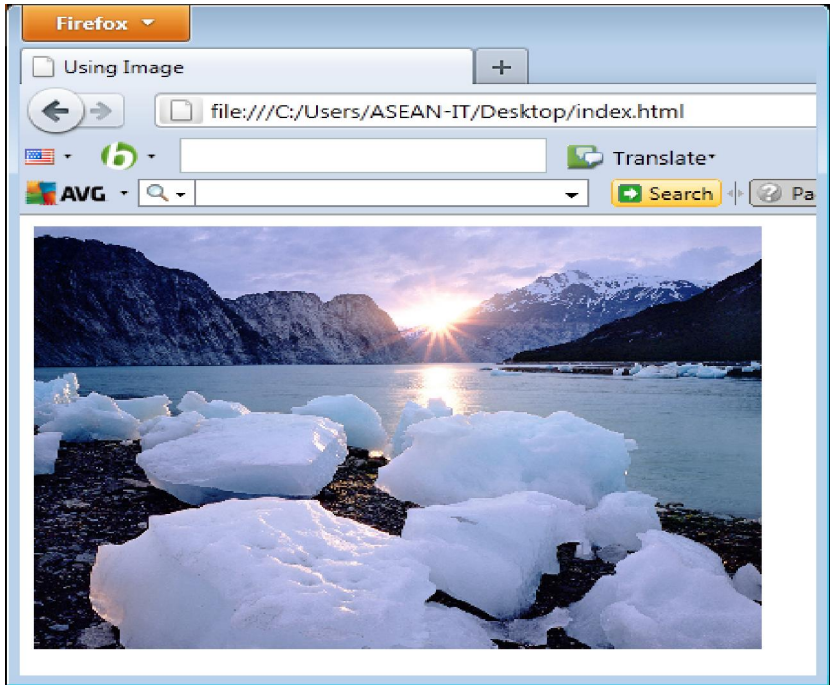
- src = "url" : កំនត់រូបភាពដែលចង់បង្ហាញ
- border = "n" : កំនត់កំរាស់រួនតំរូវ ិញរូបភាព គិតជា pixel
- alt = "Alternate Text" : កំនត់អត្ថបទដើម្បីរៀបរាប់ពីរូបភាពពេលដែលក mouse ដាក់លើវា
- width = "x" : កំនត់ប្រវែងបណ្តោយនៃរូបភាពគិតជា pixel
- height = "y" : កំនត់ប្រវែងទទឹងនៃរូបភាពគិតជា pixel
- align = "left | right" : កំនត់ទីតាំងអក្សរនៅជុំវិញរូបភាព
- hspace = "a" : កំនត់ចំងាយ horizontal space រវាងរូបភាព និង អត្ថបទ
- vspace = "b" : កំនត់ vertical space រវាងរូបភាព និង អត្ថបទ

ឧទាហរណ៍:

```

<html>
  <head><title>Using Image </title>
  </head>
  <body>
    
  </body>
</html>

```



1. ការកំណត់ទីតាំងអក្សរនៅជុំវិញរូបភាព

align = "top | middle | bottom | left | right"

- top: កំណត់ទីតាំងនៃអត្ថបទនៅពីលើរូបភាព
- middle: កំណត់ទីតាំងនៃអត្ថបទនៅកណ្តាលរូបភាព
- bottom: កំណត់ទីតាំងនៃអត្ថបទអោយស្មើទៅនឹងជួរចន្លោះរបស់រូបភាព (default)
- left: កំណត់អោយទីតាំងនៃរូបភាពនៅផ្នែកខាងឆ្វេងនៃផ្ទាំង Window
- right: កំណត់អោយទីតាំងនៃរូបភាពនៅផ្នែកខាងស្តាំនៃផ្ទាំង Window

សូមមើលការសរសេរដូចខាងក្រោម:

```

<html>
  <head><title>Using Image </title></head>
  <body>
    
    Google SEO Tips
    SEO is an acronym for "search engine optimization" or "search engine optimizer."
    Deciding to hire an SEO is a big decision that can potentially improve your site and save
    time, but you can also risk damage to your site and reputation.
    <br clear="all"/>
    <br/>
    
    Google SEO Tips
    SEO is an acronym for "search engine optimization" or "search engine optimizer."
    Deciding to hire an SEO is a big decision that can potentially improve your site and save
    time, but you can also risk damage to your site and reputation.
    <br clear="all"/>
    <br/>
    
    
    Google SEO Tips
    SEO is an acronym for "search engine optimization" or "search engine optimizer."
    Deciding to hire an SEO is a big decision that can potentially improve your site and save
    time, but you can also risk damage to your site and reputation.
  </body>
</html>

```



Google SEO Tips SEO is an acronym for "search engine optimization" or "search engine optimizer." Deciding to hire an SEO is a big decision that can potentially improve your site and save time, but you can also risk damage to your site and reputation.

Google SEO Tips SEO is an acronym for "search engine optimization" or "search engine optimizer." Deciding to hire an SEO is a big decision that can potentially improve your site and save time, but you can also risk damage to your site and reputation.



Google SEO Tips SEO is an acronym for "search engine optimization" or "search engine optimizer." Deciding to hire an SEO is a big decision that can potentially improve your site and save time, but you can also risk damage to your site and reputation.



2. ការកំណត់ទំហំរូបភាព

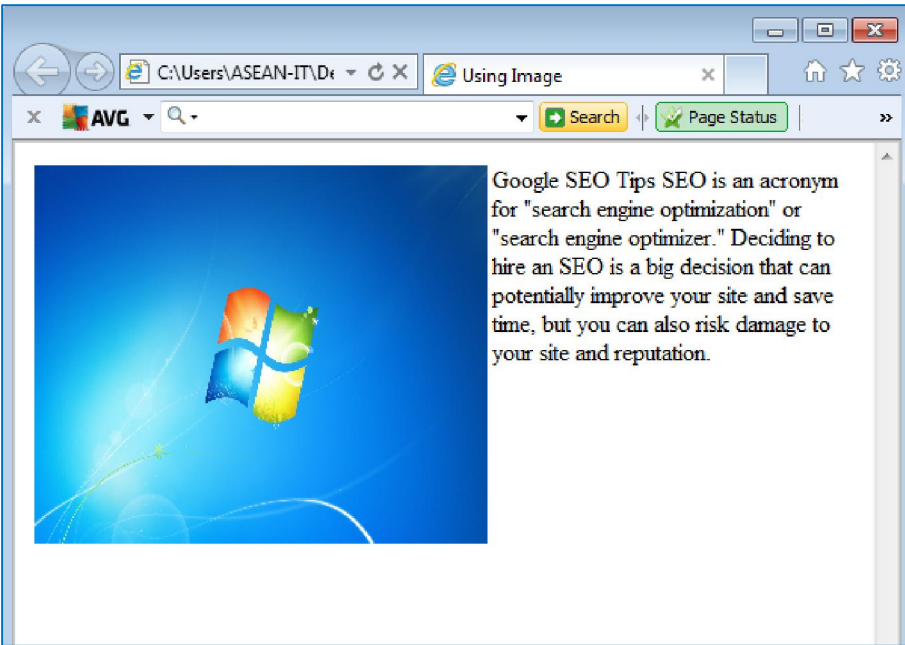
នៅពេលដែលយើងយករូបភាពទៅដាក់ក្នុង page ដោយមិនបានកំណត់ទំហំ ទទឹង ឬក៏បណ្តោយ របស់រូបភាពទេ នៅ browser នឹងបង្ហាញតាមទំហំដើមរបស់វា។ ទំហំដើមមានន័យថាជាទំហំ ពិតរបស់រូបភាព ក្នុងភាសា HTML យើងក៏អាចធ្វើការកែប្រែទំហំបានផងដែរដោយប្រើប្រាស់ attribute width និង height របស់ tag ។

ឧទាហរណ៍:

```

<html>
  <head><title>Using Image </title>
  </head>
  <body>
    
    Google SEO Tips
    SEO is an acronym for "search engine optimization" or "search engine
    optimizer." Deciding to hire an SEO is a big decision that can potentially improve
    your site and save time, but you can also risk damage to your site and reputation.
  </body>
</html>

```



3. ការកំណត់ Alternate Text

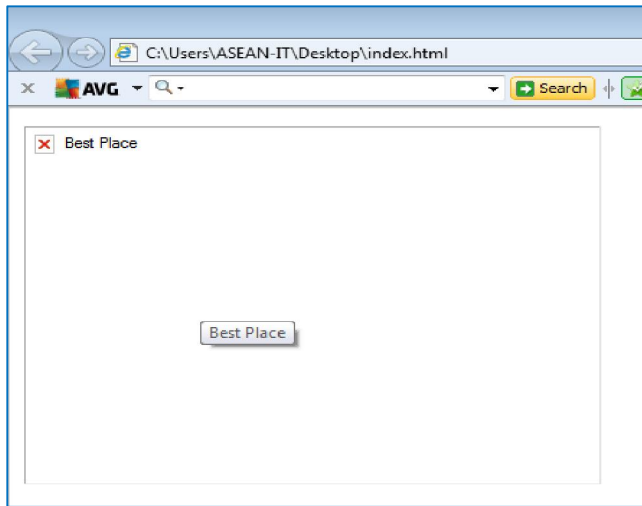
យើងប្រហែលជាធ្លាប់ជួបហើយនៅពេលដែលយើងចូលទៅកាន់ web site ណាមួយហើយយក mouse ទៅដាក់លើរូបភាពស្រាប់តែឃើញអក្សរលេចចេញមក អក្សរនោះយើងហៅថា Alternate Text ។ ដើម្បីបង្កើត Alternate Text យើងត្រូវប្រើប្រាស់ Attribute alt របស់ tag

```

```



Alternate Text ក៏បង្ហាញផងដែរ នៅពេលដែលរូបភាពមិនបានបង្ហាញ



4. ការកំណត់ចំងាយពីរូបភាពទៅអត្ថបទ

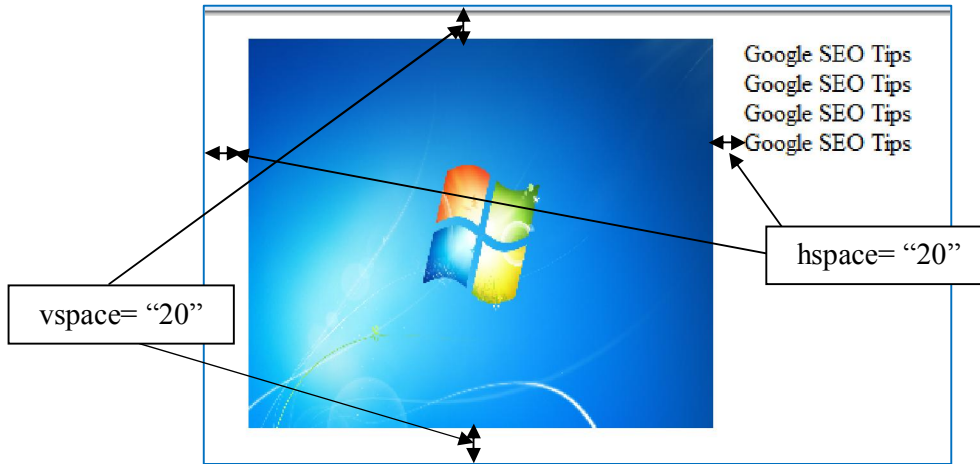
ការធ្វើអោយរូបភាព និងតួអក្សរមានគំលាតពីគ្នា អាចជួយអោយការរៀបចំអត្ថបទរបស់យើង នៅក្នុង page មានភាពស្អាតគួរអោយទាក់ទាញ។

- hspace = "a" : កំណត់ចំងាយ horizontal space រវាងរូបភាព និង អត្ថបទ
- vspace = "b" : កំណត់ vertical space រវាងរូបភាព និង អត្ថបទ

```

<html>
  <head><title>Using Image </title>
  </head>
  <body>
    
    Google SEO Tips<br/>Google SEO Tips<br/>Google SEO Tips<br/>Google
    SEO Tips<br/>
  </body>
</html>

```

5. Border and Image Links

ជាធម្មតា browser បកប្រែ Link ជាមួយ Text ដោយបង្កើតអោយមានពណ៌ផ្សេងៗទៅតាម events និងការគូសយន្តភ័ព៌ក្រោម ឥឡូវនេះយើងមានលទ្ធភាព Link តាមរយៈ image ។ ប្រសិនបើយើងបញ្ចូលរូបភាពនៅចន្លោះ Anchor Container វានឹងអាចជាវតួមួយអាច link ទៅកាន់ URL ។ នៅពេលដែលយើង link ដោយប្រើប្រាស់រូបភាព ជាធម្មតា browser នឹងបកប្រែដោយមាន Border នៅជុំវិញរូបភាពមានកំរាស់ 2pixels និងមានពណ៌ដូចគ្នាទៅនឹងពណ៌ Link text ។ ជាទូទៅ Default ពណ៌របស់ Border គឺ Blue ។

```

<html>
  <head><title>Using Image Link </title>
  </head>
  <body>
    <a href="http://www.microsoft.com/">
      
    </a>
    <a href="http://www.microsoft.com/">
      
    </a>
    <a href="http://www.microsoft.com/">
      
    </a>
  </body>
</html>

```



6. Zone Click Image

យើងប្រើ Zone Click Image ដើម្បីកំណត់តំបន់នៅលើរូបភាពដើម្បីភ្ជាប់ទៅកាន់ URL ណាមួយ។

Syntax

```
<map name="MAPNAME">
  <area shape="rect"
        coords="x0,y0,x1,y1"
        href="url"
        alt="Text">
  <area shape="circle"
        coords="x0,y0,r"
        href="url"
        alt="Text">
  <area shape="poly"
        coords="x0,y0,x1,y1,x2,y2.....,xn,yn"
        href="url"
        alt="Text">
</map>

```

ឧទាហរណ៍:

```
<html>
  <head><title>Image Map</title>
  </head>
  <body>
    <map name="microsoft">
      <area shape="rect"
            coords="36,31,156,128"
            href="http://www.microsoft.com/"
            alt="Visit Microsoft">
      <area shape="circle"
            coords="302,102,50"
            href="http://www.google.com/"
            alt="Visit Google">
    </map>
    
  </body>
</html>
```

ទំព័រ ៧

តារាង

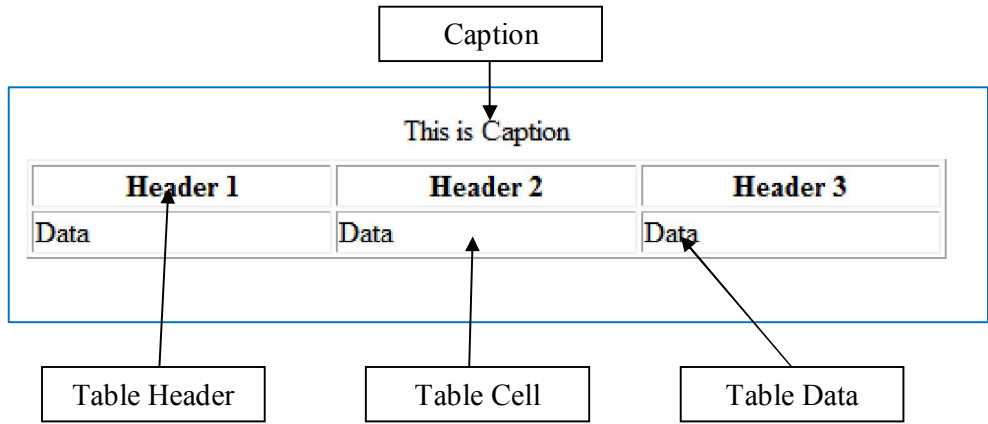
តារាង គឺជាធាតុមួយយ៉ាងសំខាន់ ហើយយើងប្រើប្រាស់ជាញឹកញាប់សំរាប់ បង្កើតគេហទំព័រ ម្យ៉ាង វិញទៀតគេអាចយក តារាងមកធ្វើជា ទំរង់ (layout) របស់ page ផងដែរព្រោះវាមានភាពងាយស្រួល ហើយពុំសូវមានភាពសុំញ៉ាំ ព្រមទាំងមានសោភ័ណភាពល្អទៀតផង។ យើងអាចជ្រើសរើសយក Table មកប្រើដើម្បីផ្ទុក ឬ រៀបចំទិន្នន័យផ្សេងៗទៅតាមតំរូវការ បានដោយងាយ។

តារាងកើតឡើងដោយការផ្សំគ្នារវាងជួរដេក (Row) និងជួរឈរ (Column) ។ យើងបំបែកទិន្នន័យ នៅក្នុងតារាងដោយប្រើប្រាស់ Tag <TH> (Table Header) រឺ <TD> (Table Data) ហើយបង្កើត ជួរដេកថ្មីដោយប្រើ Tag <TR> (Table Row) យើងត្រូវគិតថា <TR> Tag ប្រើសំរាប់បង្កើតជួរដេកថ្មី។ Table Header ជាទូទៅបង្ហាញអក្សរជិត (Bold) ហើយ Table Data បង្ហាញអក្សរដោយយកតំរូវ តាមអក្សរនៅក្នុង Body ។ Cell នៅក្នុង Table អាចប្រើសំរាប់ផ្ទុកធាតុដូចខាងក្រោម:

- Text
- List
- Hypertext
- Image
- Tag Element

ពាក្យមួយចំនួនដែលយើងប្រើប្រាស់ញឹកញាប់នៅក្នុងការបង្កើតតារាង

- Caption: ចំនងជើងតារាង
- Table Header : កំនត់អក្សរធំនៅដើមពាក្យ និង អក្សរជិត
- Table Data: ទិន្នន័យនៅក្នុងតារាង
- TableCell: ចំនុចប្រសព្វរវាង ជួរដេក និង ជួរឈរ



1. Basic HTML Table Element

ដើម្បីបង្កើតតារាងក្នុងHTML យើងប្រើ `<table>.....</table>` Tag ហើយមានធាតុមួយចំនួនទៀតនៅក្នុងវា ដូចជា `<tr>.....</tr>` tag ត្រូវបានប្រើសំរាប់បង្កើតជួរដេកនៅក្នុងតារាង។ នៅក្នុង `<tr>.....</tr>` យើងមាន tag មួយចំនួនទៀតដូចជា `<th>.....</th>` ប្រើសំរាប់បង្កើតជា Header ជាធម្មតាគេប្រើវានៅ Row ដំបូងរឺ Column ដំបូងនៃតារាង និងធាតុមួយទៀតគឺ `<td>.....</td>` ប្រើសំរាប់ជួរ កទិស្តីន័យ។

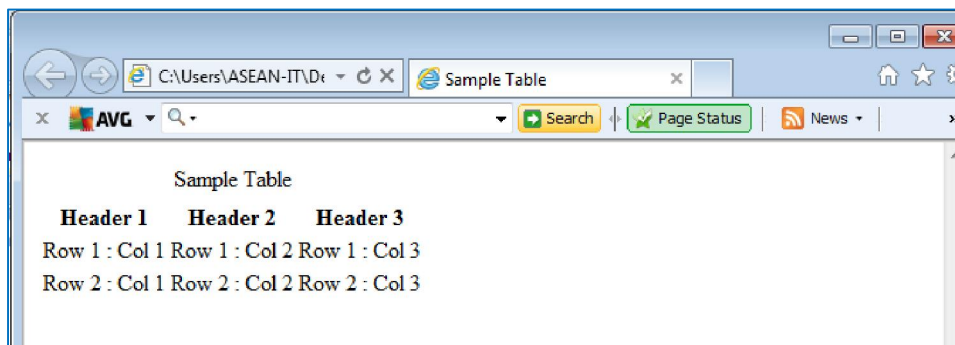
ឧទាហរណ៍ខាងក្រោមនេះបង្ហាញពីការបង្កើតតារាងធម្មតា:

Table.html

```

<html>
  <head><title>Sample Table</title>
  </head>
  <body>
    <table>
      <caption>Sample Table</caption>
      <tr>
        <th>Header 1</th>
        <th>Header 2</th>
        <th>Header 3</th>
      </tr>
      <tr>
        <td>Row 1 : Col 1</td>
        <td>Row 1 : Col 2</td>
        <td>Row 1 : Col 3</td>
      </tr>
      <tr>
        <td>Row 2 : Col 1</td>
        <td>Row 2 : Col 2</td>
        <td>Row 2 : Col 3</td>
      </tr>
    </table>
  </body>
</html>

```



➤ ទំរង់ទូទៅរបស់ Table

```

<table>
  <caption>Sample Table</caption>
  <tr>
    <th>.....</th>
    <th>.....</th>
  </tr>
  <tr>
    <td>.....</td>
    <td>.....</td>
  </tr>
</table>

```

2. ការប្រើប្រាស់ Attribute របស់ Table

❖ ការប្រើប្រាស់ Attribute background

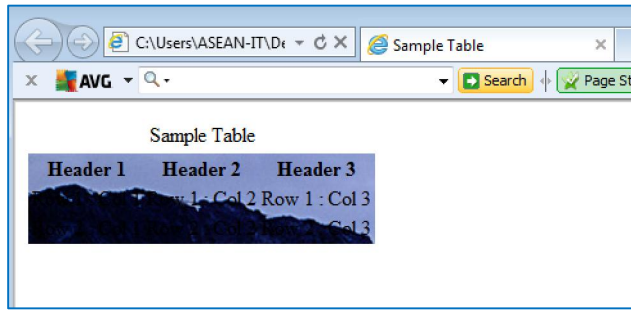
យើងប្រើប្រាស់ Attribute នេះដើម្បីធ្វើការកំណត់ផ្ទៃរបស់តារាងជារូបភាពដោយគ្រាន់តែកំណត់ឈ្មោះរបស់រូបភាពព្រមទាំងទីតាំង ប្រសិនបើទីតាំងរបស់រូបភាព និងទីតាំងរបស់ file ដែលយើងកំពុងបង្កើតមិននៅក្នុង folder តែមួយ។

ឧទាហរណ៍:

```

<html>
  <head><title>Sample Table</title>
  </head>
  <body>
    <table background="img8.jpg">
      <caption>Sample Table</caption>
      <tr>
        <th>Header 1</th>
        <th>Header 2</th>
        <th>Header 3</th>
      </tr>
      <tr>
        <td>Row 1 : Col 1</td>
        <td>Row 1 : Col 2</td>
        <td>Row 1 : Col 3</td>
      </tr>
      <tr>
        <td>Row 2 : Col 1</td>
        <td>Row 2 : Col 2</td>
        <td>Row 2 : Col 3</td>
      </tr>
    </table>
  </body>
</html>

```



❖ **ការប្រើប្រាស់ Attribute bgcolor**

យើងប្រើប្រាស់ Attribute នេះដើម្បីធ្វើការកំណត់ពណ៌ផ្ទៃរបស់តារាង។

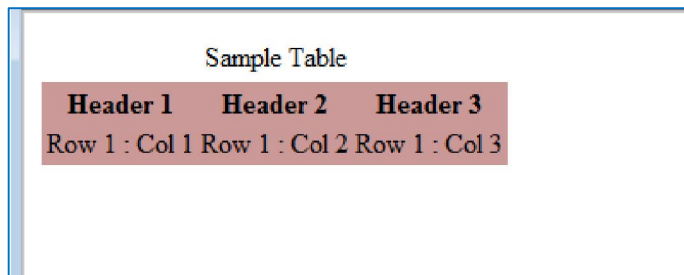
`<table bgcolor= "color_name" | "color_code" >`

.....

`</table>`

```

<html>
  <head><title>Sample Table</title>
  </head>
  <body>
    <table bgcolor="#CC9999">
      <caption>Sample Table</caption>
      <tr>
        <th>Header 1</th>
        <th>Header 2</th>
        <th>Header 3</th>
      </tr>
      <tr>
        <td>Row 1 : Col 1</td>
        <td>Row 1 : Col 2</td>
        <td>Row 1 : Col 3</td>
      </tr>
    </table>
  </body>
</html>
  
```



❖ **ការប្រើប្រាស់ Attribute Border**

នៅក្នុង `<table>` tag យើងអាចប្រើ Attribute Border ដើម្បីកំណត់ទំហំដែល ព័ទ្ធ ជុំវិញតារាង ឧទាហរណ៍:

Tableborder.html

```
<html>
  <head><title>Sample Table</title> </head>
  <body>
    <table border="1">
      <caption>Sample Table</caption>
      <tr>
        <th>Header 1</th>
        <th>Header 2</th>
        <th>Header 3</th>
      </tr>
      <tr>
        <td>Row 1 : Col 1</td>
        <td>Row 1 : Col 2</td>
        <td>Row 1 : Col 3</td>
      </tr>
    </table>
  </body></html>
```

Header 1	Header 2	Header 3
Row 1 : Col 1	Row 1 : Col 2	Row 1 : Col 3

❖ ការប្រើប្រាស់ Attribute bordercolor, bordercolorlight, bordercolordark

- bordercolor : សំរាប់កំណត់ពណ៌អោយស៊ុមរបស់តារាង
- bordercolorlight : សំរាប់កំណត់ពណ៌ស្រួចនៃតារាងអោយមានលក្ខណៈជា 3D
- bordercolordark : សំរាប់កំណត់ពណ៌ទៅអោយស៊ុមរបស់តារាងដែលមានភាពងងឹត និងមានលក្ខណៈជា 3D

```
<html>
  <head><title>Table</title></head>
  <body>
    <table width="500" border="5" align="center" cellspacing="2"
    bordercolor="#0099FF" bordercolorlight="blue" bordercolordark="red"
    height="150" cellpadding="2">
      <tr>
        <th>No</th>
        <th>Product</th>
        <th>Amount</th>
      </tr>
      <tr>
        <td>1</td>
        <td>Cocacola</td>
        <td>20</td>
      </tr>
    </table>
```

```

<tr>
  <td>2</td>
  <td>Pepsi</td>
  <td>10</td>
</tr>
<tr>
  <td>3</td>
  <td>Spy</td>
  <td>15</td>
</tr>
</table>
</body>
</html>

```

No	Product	Amount
1	Cocacola	20
2	Pepsi	10
3	Spy	15

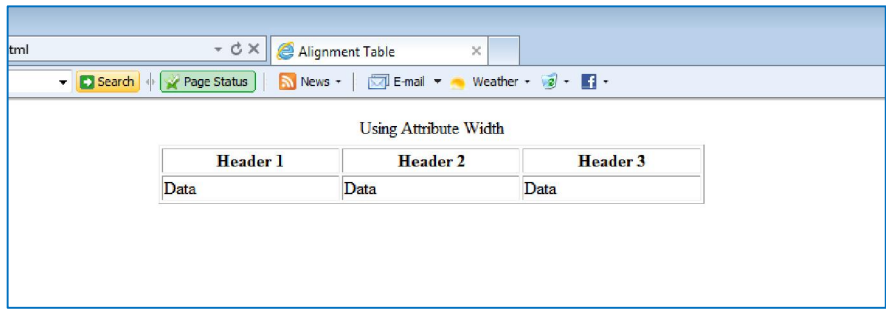
❖ ការប្រើប្រាស់ Attribute width

យើងអាចប្រើប្រាស់ Attribute នេះដើម្បីកំណត់ប្រវែងទទឹងរបស់តារាង ដែលកំណត់ដោយ

```

<html>
  <head><title>Alignment Table</title></head>
  <body>
    <table width="500" border="1" align="center">
      <caption>Using Attribute Width</caption>
      <tr>
        <th scope="col">Header 1</th>
        <th scope="col">Header 2</th>
        <th scope="col">Header 3</th>
      </tr>
      <tr>
        <td> Data </td>
        <td> Data </td>
        <td> Data </td>
      </tr>
    </table>
  </body></html>

```



❖ ទីតាំងនៃធាតុនៅក្នុងតារាង (Aligning Table Element)

ការកំណត់ទីតាំងអត្ថបទនៅក្នុង Table cell ប្រើដោយ Attribute Align និង Valign ។ Attribute ទាំងនេះអាចប្រើបាននៅទីកន្លែងជាច្រើនដូចជានៅក្នុង <caption> ,<tr>,<th> និង <td>

<caption> : ប្រើជាមួយ Attribute align សំរាប់កំណត់ទីតាំងនៃចំនងជើង Table ដើម្បីអោយនៅ លើតារាងយើងប្រើ Align= “top” និងនៅពីក្រោមតារាង Align= “bottom” ជាទូទៅ Default របស់វាគឺ top

<tr> : មាន Attribute ដូចខាងក្រោម

align: សំរាប់កំណត់ទីតាំងអោយមានតំលៃ left, right និង center (Default=left)

valign: សំរាប់កំណត់ទីតាំងអោយមានតំលៃ top, bottom និង middle (default=middle)

<th> : មាន Attribute ដូចខាងក្រោម

align: សំរាប់កំណត់ទីតាំងអោយមានតំលៃ left, right និង center (Default=left)

valign: សំរាប់កំណត់ទីតាំងអោយមានតំលៃ top, bottom និង middle (default=middle)

<td> : មាន Attribute ដូចខាងក្រោម

align: សំរាប់កំណត់ទីតាំងអោយមានតំលៃ left, right និង center (Default=left)

valign: សំរាប់កំណត់ទីតាំងអោយមានតំលៃ top, bottom និង middle (default=middle)

ឧទាហរណ៍

```
<html>
  <head><title>Alignment Table</title></head>
  <body>
    <table border="1" cellspacing="5" width="400">
      <caption align="bottom">Sample Table</caption>
      <tr>
        <th align="left">Left</th>
        <th align="center">Center</th>
        <th align="right">Right</th>
      </tr>
      <tr height="60">
        <td valign="top">Top</td>
        <td valign="middle" align="center">Middle</td>
        <td valign="bottom" align="right">Bottom</td>
      </tr>
    </table>
  </body></html>
```

Left	Center	Right
Top	Middle	Bottom

Sample Table

❖ Cell គ្មានធាតុ (Empty Cells)

បើសិនជាអ្នកចង់អោយ cell គ្មានអ្វីនៅក្នុងវា នោះជាការងាយស្រួលណាស់ដោយគ្រាន់តែនៅចន្លោះ <th> រឺ <td> យើងមិនបាច់ប្រើអ្វីទាំងអស់។

ឧទាហរណ៍:

```

<html>
  <head><title>Alignment Table</title>
  </head>
  <body>
    <table border="1" cellspacing="5" width="400">
      <tr>
        <th>AAAAAA</th>
        <th></th>
        <th>BBBBBB</th>
      </tr>
      <tr>
        <td></td>
        <td>CCCCCC</td>
        <td>DDDDDD</td>
      </tr>
    </table>
  </body>
</html>

```

AAAAAA		BBBBBB
	CCCCCC	DDDDDD

❖ ការបញ្ចូល Cell ច្រើនទៅជា Cell តែមួយ

យើងអាចធ្វើការបញ្ចូល Rows ច្រើន ឬ Columns ច្រើនអោយទៅជា Cell តែមួយបានដោយប្រើប្រាស់នូវ Attribute “colspan” or “rowspan”។

<td colspan= “n” rowspan= “n”></td>

សូមពិនិត្យមើលការសរសេរ code ដូចខាងក្រោម:

```

<html>
  <head><title>Table</title></head>
  <body>
    <table width="500" border="1" align="center">
      <tr>
        <th>Header 1</th>
        <th>Header 2</th>
        <th>Header 3</th>
      </tr>
      <tr>
        <td rowspan="3"> Data </td>
        <td colspan="2">Data</td>
      </tr>
      <tr>
        <td>Data</td>
        <td>Data</td>
      </tr>
      <tr>
        <td colspan="2">Data</td>
      </tr>
    </table>
  </body>
</html>

```

សូមពិនិត្យមើលលទ្ធផលខាងក្រោម៖

Header 1	Header 2	Header 3
Data	Data	
	Data	Data
	Data	

❖ ការកំណត់ចំងាយពី Cell ទៅ Cell និងពី ស៊ុមនៃ Cell ទៅធាតុរបស់វា

ដើម្បីធ្វើការកំណត់ចំងាយពី Cell មួយ ទៅ Cell មួយទៀតយើងប្រើប្រាស់ Attribute cellspacing ហើយចំពោះការកំណត់ចំងាយពី ស៊ុមនៃ Cell ទៅធាតុរបស់វាយើងប្រើប្រាស់ Attribute cellpadding នៅក្នុង <table> tag ។

```

<table cellspacing="n" cellpadding="n">

```

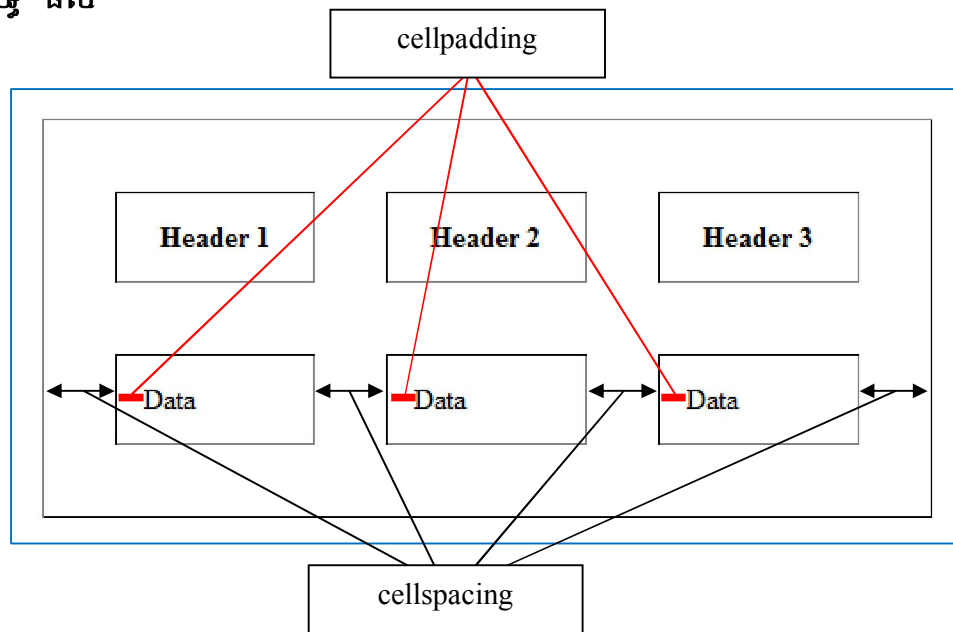
ឧទាហរណ៍

```

<html>
  <head><title>Table</title></head>
  <body>
    <br/><br/><br/>
    <table width="500" border="1" align="center" cellpadding="40"
cellpadding="15">
  <tr>
    <th>Header 1</th>
    <th>Header 2</th>
    <th>Header 3</th>
  </tr>
  <tr>
    <td>Data</td>
    <td>Data</td>
    <td>Data</td>
  </tr>
</table>
</body>
</html>

```

លទ្ធផល



❖ ការកំណត់ទំហំតារាងតាមជួរដេក

ខាងក្រោមនេះជា Syntax សំរាប់កំណត់ Attribute របស់តារាងតាមជួរដេក

<thead>.....</thead> សំរាប់កំណត់ទំហំអោយ Table Header

<tbody>.....</tbody> សំរាប់កំណត់ទំហំអោយ Table Body

<tfoot>.....</tfoot> សំរាប់កំណត់ទំហំអោយ Table Footer

ឧទាហរណ៍

```

<html>
  <head>
    <title>Table</title>
  </head>
  <body>
    <table width="500" border="1" align="center" cellspacing="2"
      height="200" cellpadding="2">
      <thead>
        <tr>
          <th>No</th>
          <th>Product</th>
          <th>Amount</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td>1</td>
          <td>Cocacola</td>
          <td>20</td>
        </tr>
        <tr>
          <td>2</td>
          <td>Pepsi</td>
          <td>10</td>
        </tr>
      </tbody>
      <tfoot>
        <tr bgcolor="#0099CC">
          <td colspan="2" align="right">Total</td>
          <td align="center">30</td>
        </tr>
      </tfoot>
    </table>
  </body>
</html>

```

លទ្ធផល

No	Product	Amount
1	Cocacola	20
2	Pepsi	10
Total		30

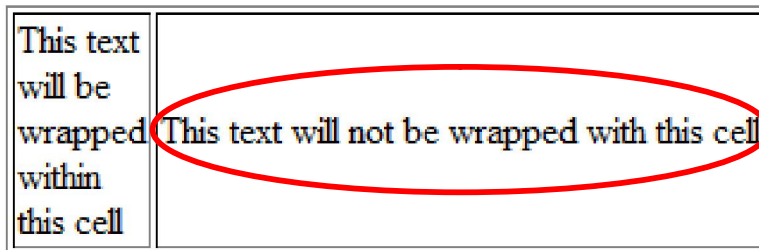
❖ កំណត់អត្ថបទមិនអោយចុះបន្ទាត់

នៅពេលដែល Browser ជួប Attribute “nowrap” ពេលនោះធ្វើការ បកប្រែអោយអត្ថបទ នៅ ក្នុង Cell មានតែមួយបន្ទាត់ប៉ុណ្ណោះ។

```
<td nowrap>.....</td>
<th nowrap>.....</th>
```

ឧទាហរណ៍

```
<html>
  <head><title>Table</title></head>
  <body>
    <br/><br/><br/>
    <table width="200" border="1" align="center">
      <tr>
        <td>This text will be wrapped within this cell</td>
        <td nowrap>This text will not be wrapped with this cell</td>
      </tr>
    </table>
  </body>
</html>
```



❖ ប្រភេទ Border

យើងអាចកំណត់ប្រភេទ Border របស់តារាងដោយប្រើ Attribute Frame និង Rules នៅក្នុង Table Tag ដូចខាងក្រោម:

```
<table border="n" frame="frameConstant" rules="area">
```

frame= frameConstant សំរាប់កំណត់ប្រភេទបន្ទាត់នៅផ្នែកខាងក្រៅតារាងដែលមានប្រភេទដូច

ខាងក្រោម:

1. void គ្មានបន្ទាត់ផ្នែកខាងក្រៅតារាង
2. above បន្ទាត់ជុំវិញតារាងមានតែផ្នែកខាងលើ
3. below បន្ទាត់ជុំវិញតារាងមានតែផ្នែកខាងក្រោម
4. hside Horizontal Sides បន្ទាត់ជុំវិញតារាងមានតែផ្នែកខាងលើនិងខាងក្រោម
5. vside Vertical Side បន្ទាត់ជុំវិញតារាងមានតែផ្នែកខាងឆ្វេង និងខាងស្តាំ
6. rhs Right hand side បន្ទាត់ជុំវិញតារាងមានតែផ្នែកខាងស្តាំ
7. lhs Left hand side បន្ទាត់ជុំវិញតារាងមានតែផ្នែកខាងឆ្វេង

8. box or border (Default) បន្ទាត់ជុំវិញតារាងគ្រប់ផ្នែកទាំងអស់

- Rules = area កំណត់ប្រភេទបន្ទាត់ដែលស្ថិតនៅផ្នែកខាងក្នុង ឯតារាងដែលមានប្រភេទ

ដូចខាងក្រោម:

1. none គ្មានបន្ទាត់ផ្នែកខាងក្នុង ឯតារាង
2. rows បន្ទាត់នៅផ្នែកខាងក្នុង ឯតារាងមានតែជួរដេក
3. cols បន្ទាត់នៅផ្នែកខាងក្នុង ឯតារាងមានតែជួរឈរ

🔗 បង្កើតទំព័រ Web Page ដោយប្រើប្រាស់ Tables

វិធីសាមញ្ញក្នុងការបង្កើតទំព័រ web page ដោយប្រើប្រាស់ HTML <table> tag ៗ ខាងក្រោម នេះជាឧទាហរណ៍ដោយប្រើ Table ដែលមាន 3 rows និង 2 columns

```

<html>
  <head>
    <title>Sample HTML Layout</title>
  </head>
  <body>
    <table width="500" border="0">
      <tr>
        <td colspan="2" style="background-color:#06F;">
          <h1>Header</h1>
        </td>
      </tr>
      <tr valign="top">
        <td style="background-color:#FC9;width:100px;text-align:top;">
          <b>MENU</b><br />
          Home<br />
          About Us<br />
          Contact Us<br />
          Location
        </td>
        <td style="background-color:#EEEEEE;height:200px;width:400px;text-align:top;">
          Content</td>
      </tr>
      <tr>
        <td colspan="2" style="background-color:#06F;text-align:center;">
          Copyright © 2011 USEA Web</td>
        </tr>
    </table>
  </body>
</html>

```

លទ្ធផល



🔗 បង្កើតទំព័រ Web Page ដោយប្រើប្រាស់ Div Elements

The div element is a block level element used for grouping HTML elements. ខាងក្រោមនេះ ជាឧទាហរណ៍ដោយប្រើប្រាស់ div element បង្កើតជាទំព័រ Website

```

<html>
<body>
  <div id="container" style="width:500px; margin:0px auto;">
    <div id="header" style="background-color:#03F;">
      <h1 style="margin-bottom:0;">Header</h1></div>
      <div id="menu" style="background-
color:#F99;height:200px;width:100px;float:left;">
        <h2>Menu</h2><br />
        Home<br />
        About Us<br />
        Contact Us<br />
        Location
      </div>
      <div id="content" style="background-
color:#EEEEEE;height:200px;width:400px;float:left;">
        Content</div>
      <div id="footer" style="background-color:#03F;clear:both;text-
align:center;">
        Copyright 2011 USEA Web</div>
    </div>
  </body>
</html>

```


លទ្ធផល

*HTML Layout - Useful Tips*

Tip: The biggest advantage of using CSS is that, if you place the CSS code in an external style sheet, your site becomes MUCH EASIER to maintain. You can change the layout of all your pages by editing one file.

ជំពូក ៨

FRAME

Frame អនុញ្ញាតអោយយើងធ្វើការចែកនូវផ្ទាំង window ជាច្រើនផ្នែក ដើម្បីបង្ហាញ pages ច្រើនខុសគ្នា ។ ឧទាហរណ៍នៅក្នុងផ្ទាំង window តែមួយយើងអាចធ្វើការបែងចែកជា ៤ ផ្នែកផ្សេងៗគ្នា ដោយ ផ្នែកខាងលើសំរាប់ដាក់នូវ logo ក្រុមហ៊ុន ផ្នែកខាងឆ្វេងសំរាប់ដាក់ជា menu សំរាប់ link ចំនែកផ្នែកខាងស្តាំ សំរាប់ផ្ទុកព័ត៌មានផ្សេងៗ និងផ្នែកខាងក្រោមសំរាប់ដាក់ copyright សូមមើល page ខាងក្រោម:



- Frame មានមុខងារដូចជា:
- Frame អនុញ្ញាតអោយអ្នកទស្សនាមើលឃើញ Web page ច្រើនក្នុងពេលតែមួយ
 - Frame នីមួយៗមាន Web page ផ្ទាល់ខ្លួន ហើយអាចបង្ហាញតាមផ្នែករបស់ Window

ដោយឯករាជ្យ

- គុណវិបត្តិរបស់ Frame
- Web developer ត្រូវរក្សាទុក HTML Document ច្រើន
- ពិបាក Print page ទាំងមូល

1. ការបង្កើត Frameset

- Frameset: ត្រូវបានគេគិតថាជា Window ដែលមានផ្នែកតូចៗរបស់វា សំរាប់ផ្ទុកព័ត៌មានផ្សេងគ្នា

- អ្នកត្រូវសំរេចចិត្តថាតើនៅក្នុង Window មួយបែងចែកជាប៉ុន្មានផ្នែក? ហើយ Border មានលក្ខណៈ :យ៉ាងដូចម្តេច? ហើយមានការប្រើប្រាស់ Scroll bar រឺទេ?

- នៅពេលដែលអ្នកបង្កើត Window អ្នកអាចផ្តល់ URL ទៅអោយផ្នែកនីមួយៗបន្ទាប់មកវាភ្ជាយទៅជា Frames ។

ទំរង់ទូទៅ

```
<frameset rows= "r1, r2 ,r3,....."
              cols= "c1, c2, c3 ,....." >
  <frameset rows= "r1, r2 ,r3,....."
              cols= "c1, c2, c3 ,....." >
    .....
  </frameset>
  <frameset rows= "r1, r2 ,r3,....."
              cols= "c1, c2, c3 ,....." >
    .....
  </frameset>
  <frame.....>
  .....
</Frameset>
```

<frameset> ជា tag ដែលប្រាប់ទៅ Browser ដើម្បីបែងចែក Window ទៅជា Rows រឺ Columns រឺបែងចែក Rows ផង និង Columns ផង ។ <frameset> tag មាន Attributes ដូចខាងក្រោម:

- rows = "r1, r2 ,r3,....." ជាតំលៃដែលខ្ញុំចូលដោយសញ្ញាកៀសហើយវាអាចប្រើជា "*" ជាចំនួន(គិតជា Pixels) រឺជាភាគរយដើម្បីកំណត់កំពស់នៃ Frame នីមួយៗ

Ex: rows = "10%, 40%,*" => *=50%

- cols= "c1, c2 , c3," ជាតំលៃដែលខ្ញុំចូលដោយសញ្ញាកៀសហើយវាអាចប្រើជា "*" ជាចំនួន(គិតជា Pixels) រឺជា ភាគរយដើម្បីកំណត់បណ្តាយនៃ Frame នីមួយៗ ។

Ex: cols= "40%, *, 20%" Browser នឹងបំបែក "*" ទៅជាតំលៃ 40%

ចំណាំ យើងអាចប្រើ "*" លើសពីមួយពេល Browser ជួបសញ្ញា "*" វានឹងបែកចែកតំលៃដែលនៅសល់ស្មើគ្នា។

ឧទាហរណ៍

header.html

left.htm

```
<html>
  <head><title>Header</title></head>
  <body
    bgcolor=red><h1><center>Frame
    1</center></h1></body>
</html>
```

```
<html>
  <head><title>Menu</title></head>
  <body leftmargin=0 topmargin=0
    bgcolor="#FF9966" text="white">
    <h2>
    Frame 2<br/>
    Left
    </h2>
</body></html>
```

right.html

footer.htm

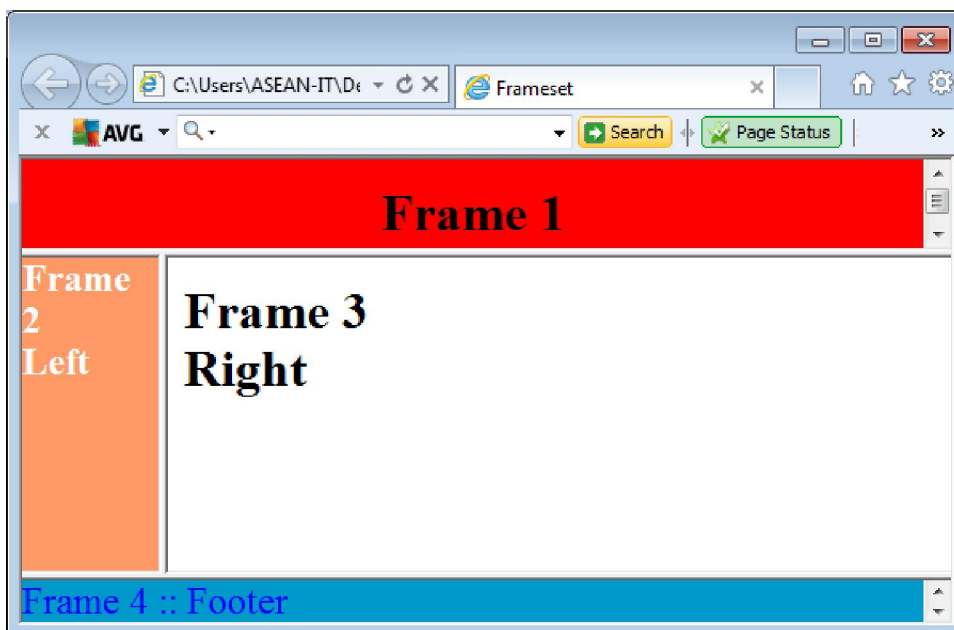
```
<html>
  <head><title>Home
page</title></head>
  <body>
    <h1>
      Frame 3<br/>
      Right
    </h1>
  </body>
</html>
```

```
<html>
  <head><title>Header</title></head>
  <body leftmargin=0 topmargin=0
bgcolor="#0099CC" text="white">
    <font color=blue size=5>
      Frame 4 :: Footer
    </font>
  </body>
</html>
```

index.html

```
<html>
  <head><title>Frameset</title></head>
  <frameset rows="20%,*,10%" >
    <frame src="header.html">
      <frameset cols="15%,*">
        <frame src="left.html">
        <frame src="right.html">
      </frameset>
    <frame src="footer.html">
  </frameset>
</html>
```

លទ្ធផល



ចំណាំ

- ប្រសិនបើ frame មានប្រភេទជុំវិញ នៅអ្នកទស្សនាអាចពង្រីក ឬបង្រួម មុំហំ frame បានដោយទាញ ប្រភេទដែលនៅជុំវិញ ដើម្បីការពារកុំអោយអ្នកទស្សនាទាញបានត្រូវបន្ថែម attribute noresize= "noresize" ទៅ អោយ <frame> tag ។

- បញ្ចូល <noframes> tag សំរាប់ browsers ដែលអត់ស្គាល់ frames

- យើងមិនអាចប្រើប្រាស់ <body></body> tags ជាមួយ <frameset></frameset> tags បានឡើយ។ តែប្រសិនបើយើងបញ្ចូល <noframes> tag ចូលដើម្បីដាក់អត្ថបទសំរាប់ browsers ដែលអត់ស្គាល់ frames នៅ យើងត្រូវតែបន្ថែម <body></body> tags ។

2. ការប្រើប្រាស់ Attribute របស់ Frameset

2.1. Attribute Border

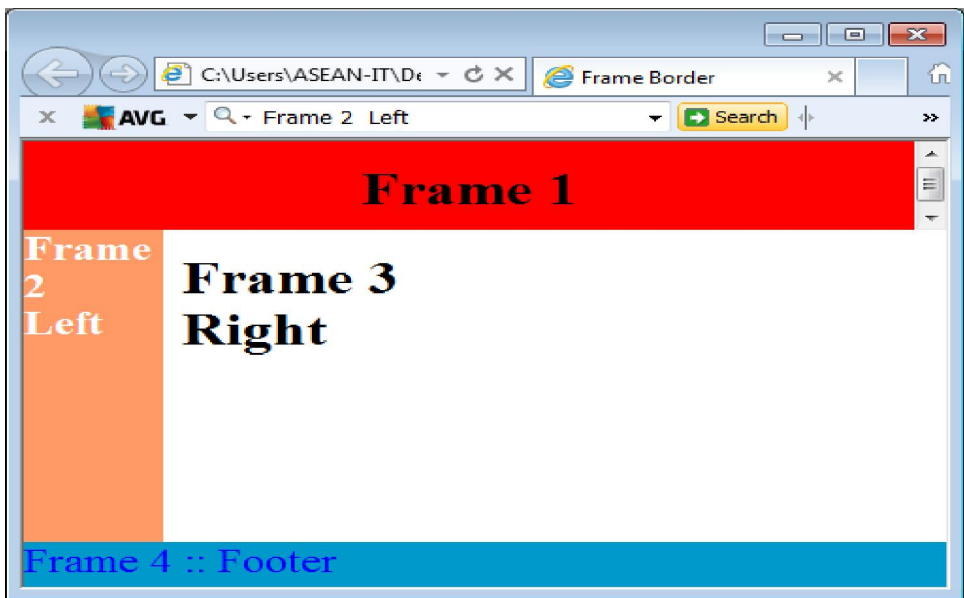
វាគឺជាប្រភេទដែលខ្លួនចែករវាង Frame និង Frame ។ ប្រសិនបើយើងមិនចង់អោយមាន border ពេលដែលប្រើប្រាស់ browser ទេយើងត្រូវកំណត់អោយតំលៃរបស់ border = 0 ។

```

<html>
  <head><title>Frame Border</title></head>

  <frameset rows="20%,*,10%" border="0">
    <frame src="header.html">
      <frameset cols="15%,*">
        <frame src="left.html">
          <frame src="right.html">
        </frameset>
    <frame src="footer.html">
  </frameset></noframes></noframes>
</html>

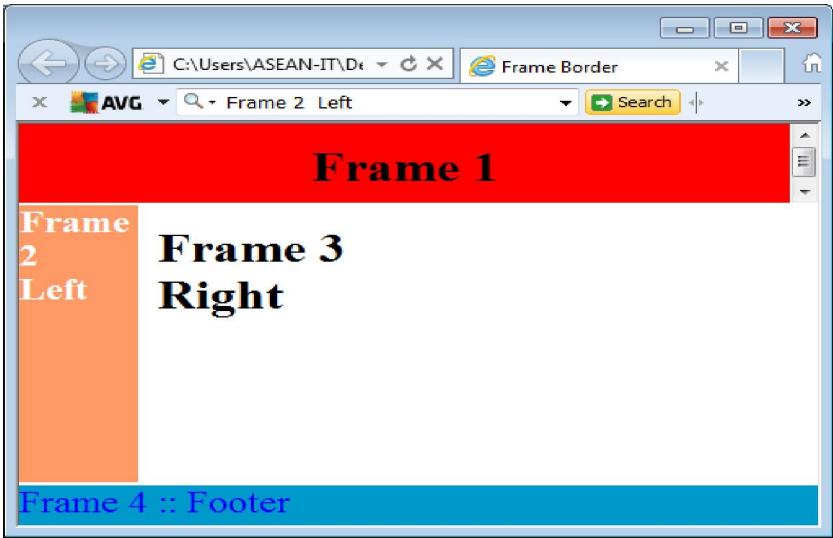
```



2.2. Attribute FrameBorder

វាខុសពី attribute border ត្រង់ថាវាមានតំលៃតែពីរប៉ុណ្ណោះគឺ 1 រឺ 0 និង yes រឺ no ប្រើសំរាប់កំណត់ថា Frameborder មាន រឺមិនមាន។

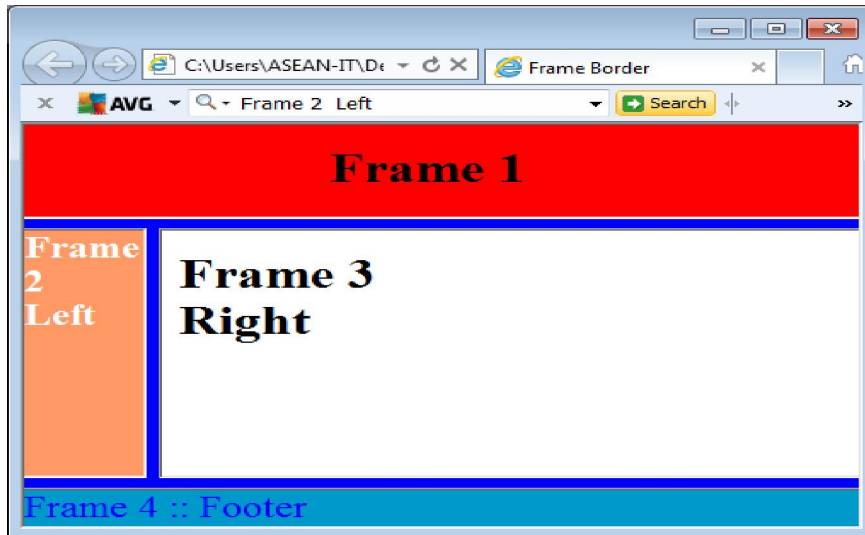
```
<html>
  <head><title>Frame Border</title></head>
    <frameset rows="20%,*,10%" frameborder="no">
      <frame src="header.html">
        <frameset cols="15%,*">
          <frame src="left.html">
            <frame src="right.html">
          </frameset>
        <frame src="footer.html">
      </frameset><noframes></noframes>
</html>
```



2.3. Attribute BorderColor

ដើម្បីកំណត់ពណ៌អោយ border របស់ frame យើងប្រើប្រាស់ attribute bordercolor <frameset bordercolor="color_name | color_code">

```
<html>
  <head><title>Frame Border</title></head>
    <frameset rows="20%,*,10%" bordercolor="#0000FF"
border="7">
      <frame src="header.html">
        <frameset cols="15%,*">
          <frame src="left.html">
            <frame src="right.html">
          </frameset>
        <frame src="footer.html">
      </frameset><noframes></noframes>
</html>
```



3. ការប្រើប្រាស់ Attribute របស់ Frame

<frame></frame> ជា tag មួយដែលយើងប្រើប្រាស់ជាមួយនឹង frameset ។ Frameset ប្រើសំរាប់បែងចែកផ្ទៃរបស់ Window ជាច្រើនផ្នែក ចំណែកឯ Frame tag ប្រើសំរាប់បង្ហាញ page ផ្សេងៗ ។

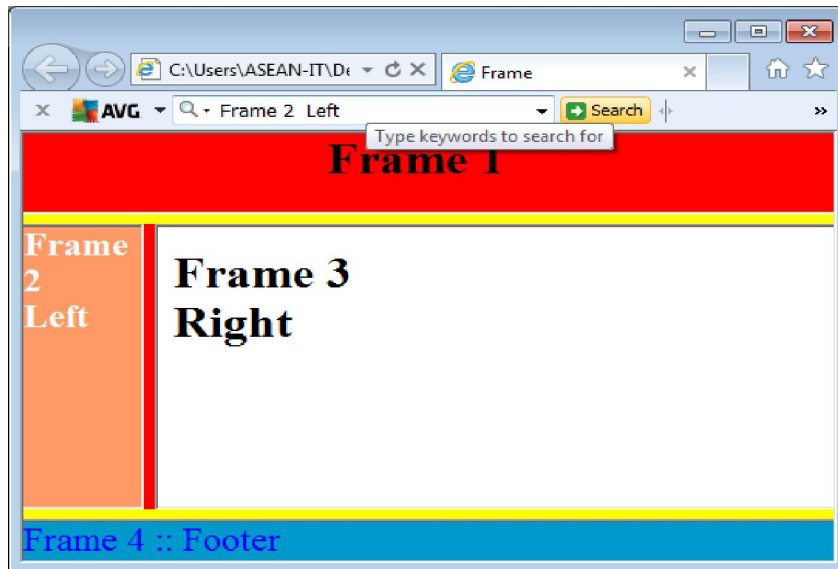
Attribute របស់ Frame មាន :

- src: កំណត់អាស័យដ្ឋានរបស់ឯកសារដែលត្រូវបង្ហាញ
- name: សំរាប់ដាក់ឈ្មោះអោយ frame ដែលឈ្មោះនេះយើងនឹងយកទៅប្រើនៅពេលដែលយើងហៅ Page ផ្សេងៗមកបង្ហាញនៅក្នុង frame
- bordercolor: សំរាប់កំណត់ពណ៌ border របស់ frame
- scrolling: ប្រើសំរាប់កំណត់នូវ Scrollbar របស់ frame
 - scrolling = "yes/no/auto"
 - yes : មាន Scrollbar
 - no : មិនមាន Scrollbar
 - auto : បង្ហាញ Scrollbar នៅពេលដែលអត្ថបទពេញផ្ទៃ frame (default)
- noresize: មិនអនុញ្ញាតិអោយអ្នកទស្សនាពង្រីក រឺបង្រួមទំហំរបស់ frame

```

<html>
  <head><title>Frame </title></head>
  <frameset rows="20%,*,10%" bordercolor="#0000FF" border="7">
    <frame src="header.html" marginwidth="30">
    <frameset cols="15%,*">
      <frame src="left.html" noresize bordercolor="#FF0000"
scrolling="auto">
      <frame src="right.html" name="right" scrolling="auto">
    </frameset>
    <frame src="footer.html" bordercolor="yellow">
  </frameset>
</html>

```



4. ការប្រើប្រាស់ Link ទៅកាន់ទីតាំងណាមួយនៅក្នុង Frame

នេះជាចំនុចមួយដ៏សំខាន់នៅក្នុង frame ដែលយើងត្រូវប្រើជាញឹកញាប់នៅពេលដែលយើងរចនាគេហទំព័រដោយប្រើ frame ។ វាអនុញ្ញាតិអោយយើងធ្វើការ link ដោយហៅ page មកដាក់ត្រង់ទីតាំងណាមួយជាក់លាក់ ។ នៅក្នុងផ្នែកនេះយើងកំនត់ Target attribute នៅក្នុង <A> tag ឆ្ពោះទៅកាន់ Browser ដើម្បីអោយវា Load ឯកសាររបស់ HREF Attribute ទៅកាន់ Window ឬ Frame ណាមួយយ៉ាងច្បាស់លាស់។

Ex: About Us

នៅពេល Load HTML Document ហៅ about.html អោយបង្ហាញនៅក្នុង Window ឬ Frame ដែលមានឈ្មោះ “right” នៅពេលអ្នកប្រើប្រាស់ចុច About Us Link។ ប្រសិនបើ Frame គ្មានឈ្មោះ “right” នៅទេ Browser នឹងបង្កើត Window ថ្មីហើយកំនត់ឈ្មោះវាថា “right” ។

<base target= “target_name”> សំរាប់បង្កើត Target គ្រប់ Links ទាំងអស់លើលំដាប់ណាដែលមានឈ្មោះ ។

Special Targets:

Browser support 4 Target ពិសេសដែលចាំបាច់ដោយសញ្ញា “_” ។ ការប្រើប្រាស់ “_” បានតែចំពោះ Target ទាំងនោះតែប៉ុណ្ណោះចំពោះ Target ផ្សេងទៀតអ្នកប្រើសញ្ញានេះពេល Browser ជួបវាមិនស្គាល់ទេ។

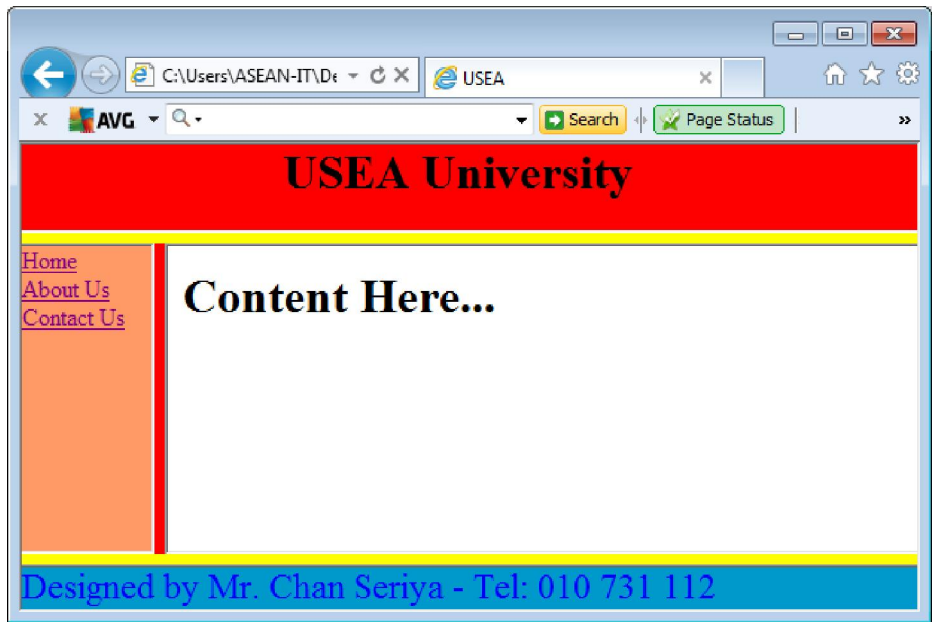
- _blank : ត្រូវបានគេប្រើប្រាស់ដើម្បីអោយ Browser ចាំបាច់បើក Window ថ្មី ។ វាជាទូទៅបានគេប្រើសំរាប់ Window ដែលគ្មានឈ្មោះ។
- _self : ប្រសិនបើ <A>tag អត់មាន target value វាក៏ជា Default target ។ Target នេះចង្អុលទៅកាន់ Current Frame or Window

- `_parent` : វានឹងបើកការ Link ទៅលើផ្ទាំង `window` ដដែលដោយបំបាត់ `frame` ដែលមានពីមុនមក។

- `_top` : វាមានលក្ខណៈស្រដៀងទៅនឹង `_parent` ដែរតែវាខុសនៅពេលដែលយើងប្រើប្រាស់វាជាមួយនឹង `nested frameset`

index.html

```
<html>
  <head><title>USEA</title></head>
  <frameset rows="20%,*,10%" bordercolor="#0000FF" border="7">
    <frame src="header.html" marginwidth="30">
    <frameset cols="15%,*">
      <frame src="left.html" noresize bordercolor="#FF0000"
scrolling="auto">
      <frame src="right.html" name="right" scrolling="auto">
    </frameset>
    <frame src="footer.html" bordercolor="yellow">
  </frameset>
</html>
```



left.html

```

<html>
  <head><title>Menu</title></head>
  <body leftmargin=0 topmargin=0 bgcolor="#FF9966" text="white">
    <a href="home.html" target="right">
      Home
    </a> <br/>
    <a href="aboutus.html" target="right">
      About Us
    </a> <br/>
    <a href="contact.html" target="right">Contact Us</a> <br/>
  </body>
</html>

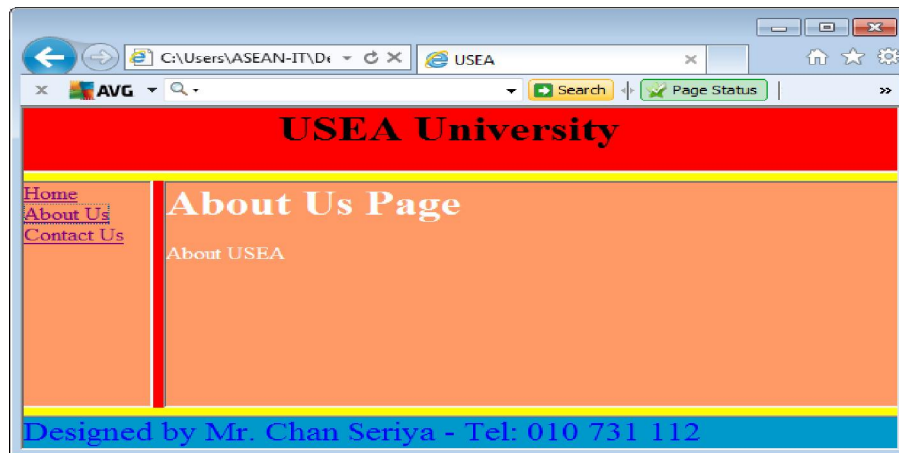
```

aboutus.html

```

<html>
  <head><title>About Us</title></head>
  <body leftmargin=0 topmargin=0 bgcolor="#FF9966" text="white">
    <h1> About Us Page</h1>
    <p>
      About USEA
    </p>
  </body>
</html>

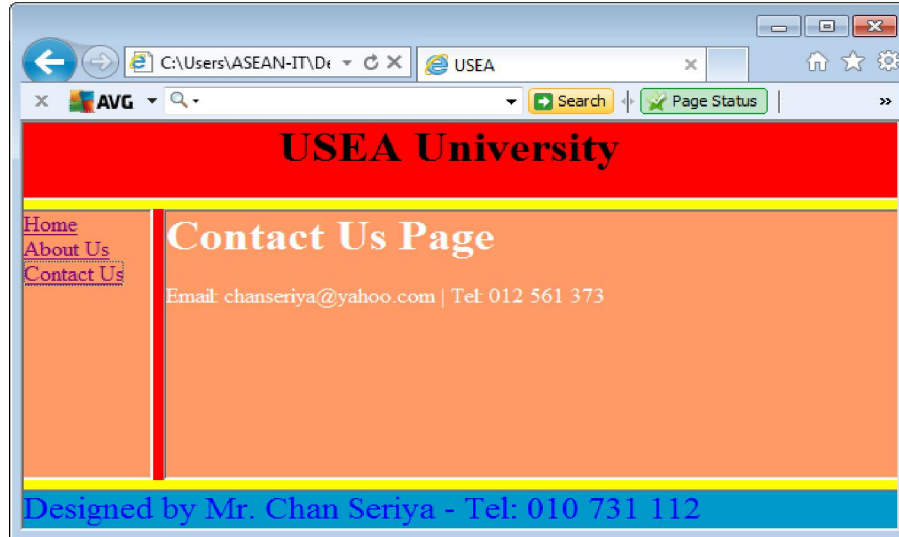
```

contact.html

```

<html>
  <head><title>Contact Us</title></head>
  <body leftmargin=0 topmargin=0 bgcolor="#FF9966" text="white">
    <h1> Contact Us Page</h1>
    <p>
      Email: chanseriya@yahoo.com | Tel: 012 561 373
    </p>
  </body></html>

```



Using iframe

An iframe is used to display a web page within a web page


Syntax for adding an iframe:

```
<iframe src="URL" width = "w" height = "h" frameborder= "n" ></iframe>
```

- URL points to the location of the separate page
- Width and height: set width and height to iframe in pixels by default, but they can also be in percent (like "60%")
- frameborder attribute specifies whether or not to display a border around the iframe

Example:

```
<html>
  <head><title>Using iframe</title></head>
  <body>
    <iframe src="http://www.fbis.edu.kh/" width="500"
    height="300"></iframe>
    <p>Some older browsers don't support iframes.</p>
    <p>If they don't, the iframe will not be visible.</p>
  </body>
</html>
```



The screenshot shows a website header with a navigation menu containing 'Home', 'About FBIS', 'Academic Programs', and 'Admissions'. Below the menu is a 'Home' section with a large heading 'Welcome to Future Bright International School'. Underneath the heading is a video player showing a group of men in suits sitting on a stage. The video player has a play button and a progress bar. Below the video player, there is a text block that reads: 'Some older browsers don't support iframes. If they don't, the iframe will not be visible.'



FORM

ជាធម្មតាយើងធ្វើការបង្កើតគេហទំព័រមិនមែនត្រឹមតែបង្ហាញ អ្នកប្រើប្រាស់ អោយយើងតែ ព័ត៌មាន ឬក៏ ផលិតផលផ្សេងៗនោះទេ គឺយើងអាចធ្វើការប្រមូលព័ត៌មាន ពីអ្នកប្រើប្រាស់ដោយ ផ្តល់ឈ្នួ ភាពអោយ អ្នកប្រើប្រាស់ចុះឈ្មោះ រឺក៏ទិញផលិតផលផ្សេងៗជាដើម។ លក្ខណៈទាំងអស់ នេះគឺអាចធ្វើទៅបានត្រូវពឹងផ្អែកលើ Form ។ Form គឺជាធាតុមួយ ដែលយើងប្រើសំរាប់ បង្កើតជាទម្រង់បញ្ជូនទិន្នន័យ ដើម្បីបញ្ជូនព័ត៌មានទៅកាន់ Server ដើម្បីធ្វើការ Process យើងអាច សរសេរធាតុផ្សេងៗរបស់ Form ដូចជា Textbox, Password, Button ... បាននៅចន្លោះរវាង <form> និង </form> ។

1. ទម្រង់ទូទៅសំរាប់បង្កើត Form

```
<form name= "name_of_form"
      action= "URL"
      method= "Post |Get"
      >
      .....Form Elements .....
      .....Form Elements .....
      .....Form Elements .....
</Form>
```

- name : កំនត់ឈ្មោះអោយ Form
- action = "URL" : កំនត់ទីតាំងដែលត្រូវយកទិន្នន័យទៅដាក់
- method = "Post | Get" : គឺជាវិធីសាស្ត្រ ក្នុងការបញ្ជូនទិន្នន័យទៅកាន់ Server (Default : Get)

2. ការបង្កើត Textbox

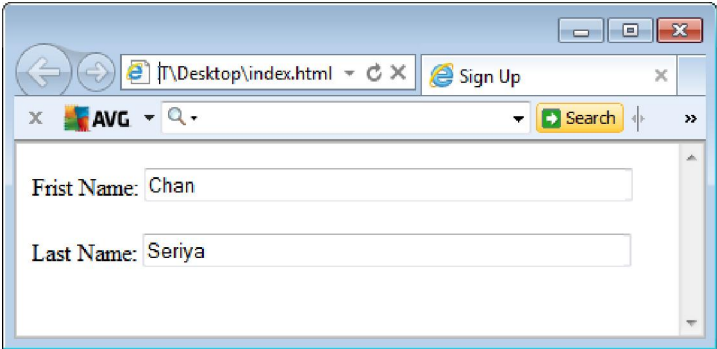
Textbox យើងប្រើសំរាប់ visitor ធ្វើការបញ្ជូនទិន្នន័យដែលមានតែមួយបន្ទាត់ប៉ុណ្ណោះ

```
<input type="text"
      name="Field_Name"
      value="Default Value"
      size="n"
      maxlength="n"
      readonly
      disabled >
```

- value= "Default" : កំនត់តំលៃ Default សំរាប់ Textbox
- size = "n" : កំនត់ទំហំនៃ Textbox ដែលត្រូវនឹងប្រវែង n Characters
- maxlength= "n" : កំនត់ចំនួនតួអក្សរច្រើនបំផុតដែលយើង លទ្ធភាពទៅក្នុង Textbox

readonly កំណត់មិនអោយអ្នកប្រើប្រាស់ធ្វើការកែប្រែតែអាចតែអានប៉ុណ្ណោះ
 disable កំណត់អោយមិនអាចចុចបាន ឬធ្វើសកម្មភាពផ្សេងៗបាន

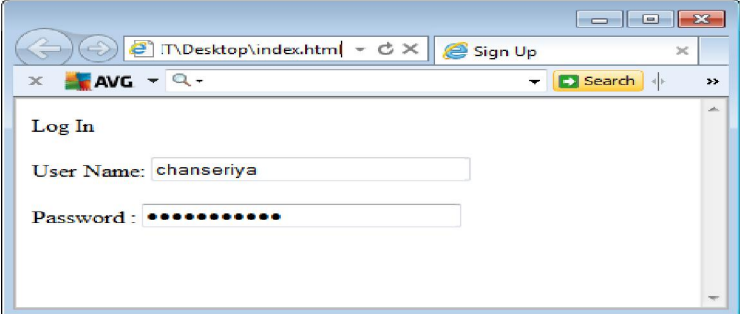
```
<html>
  <head><title>Sign Up</title> </head>
  <body>
    <form name="form1" method="post" action="">
      Frist Name: <input type="text" name="FirstNameTextbox"
size="50" maxlength="50" ><br/><br/>
      Last Name: <input type="text" name="LastNameTextbox"
size="50" maxlength="50" >
    </form>
  </body>
</html>
```



3. ការបង្កើត Password Boxes

Password Boxes មានលក្ខណៈដូចគ្នាទៅនឹង Textbox ដែរ តែខុសគ្នាថាតំលៃដែលយើងបញ្ចូលវាជាសញ្ញា Asterisks “*” រីជា bullets

```
<html>
  <head><title>Log In</title></head>
  <body>
    Log In
    <form name="form1" method="post" action="">
      User Name: <input type="text" name="username"
size="30"><br/><br/>
      Password : <input type="password" name="password" size="30">
    </form>
  </body>
</html>
```



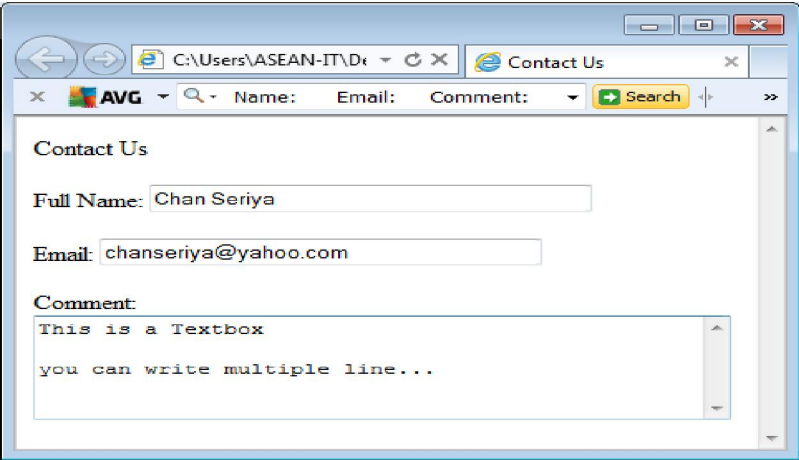
4. ការបង្កើត TextArea

TextArea ខុសពី Textbox ត្រង់ថាវាអាចផ្ទុកអត្ថបទមួយឬ ច្រើនបន្ទាត់ ហើយអាច ពង្រីកបាននៅ ពេលត្រូវការ

```
<textarea name="Field Name"
rows="n"
cols="n"
wrap="off | hard | soft">
..... Default Text .....
</textarea>
```

- rows = "n" ជាចំនួនជួរដេកនៅក្នុង Textarea
- cols = "n" ប្រវែង Textarea ដែលយើងដាក់តួអក្សរបាន
- wrap អនុញ្ញាតិអោយអត្ថបទអាចចុះបន្ទាត់បាននៅក្នុង ដែនកំណត់របស់ Textarea
 - off ចុះបន្ទាត់លុះត្រាតែយើងប្រើ Key Enter
 - hard | soft ចុះបន្ទាត់ដោយស្វ័យប្រវត្តិ

```
<html>
<head>
<title>Contact Us</title>
</head>
<body>
Contact Us
<form name="form1" method="post" action="">
Full Name: <input type="text" name="fullname" size="40">
<br/><br/>
Email: <input type="text" name="email" size="40"><br/><br/>
Comment: <textarea name="comment" rows="5" cols="50"
wrap="hard"></textarea>
</form>
</body>
</html>
```



5. ការបង្កើត Radio Button

Radio Button ប្រើសំរាប់អោយអ្នកទស្សនាអាចជ្រើសរើសតំលៃបានតែមួយក្នុងចំនោមតំលៃជាច្រើន។

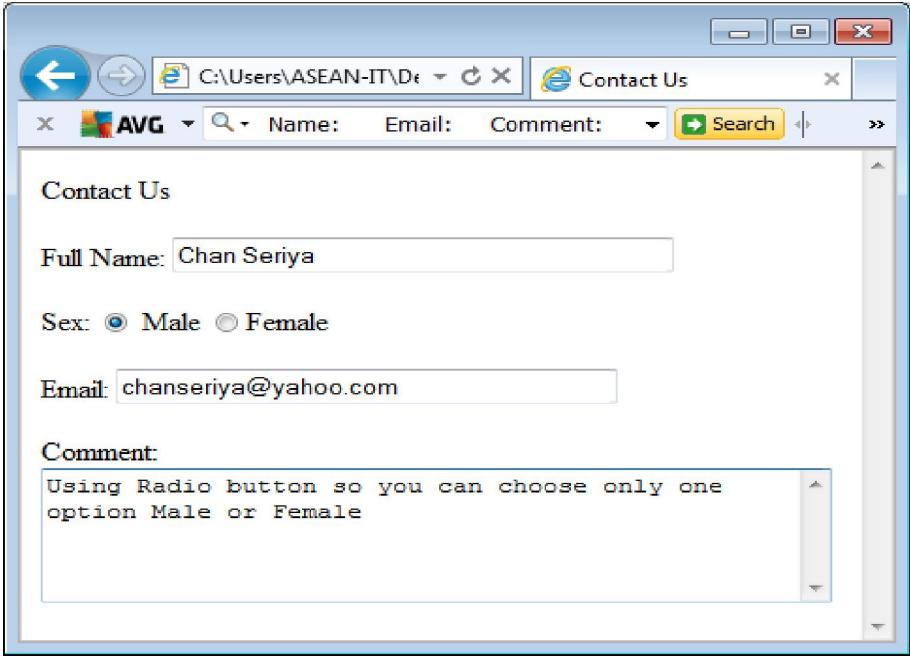
```
<input type="radio"
      name="Field Name"
      value="Field Value"
      checked >
```

name = “Field Name” : សំរាប់កំនត់ឈ្មោះអោយ Radio Button

value= “Field Value” : សំរាប់កំនត់តំលៃ

checked : សំរាប់កំនត់ថា Select រឺក៏មិន Select

```
<html>
  <head><title>Contact Us</title></head>
  <body>
    Contact Us
    <form name="form1" method="post" action="">
      Full Name: <input type="text" name="fullname" size="40">
    <br/><br/>
      Sex:<input type="radio" name="sex" value="Male" checked> Male
      <input type="radio" name="sex" value="Female">Female<br/><br/>
      Email: <input type="text" name="email" size="40"><br/><br/>
      Comment: <textarea name="comment" rows="5" cols="50"
wrap="hard"></textarea>
    </form>
  </body>
</html>
```



6. ការបង្កើត Checkboxes

អនុញ្ញាតិអោយអ្នកទស្សនាអាចមានជំរើសច្រើនក្នុងពេលតែមួយ

```

<input type="checkbox"
      name="Field Name"       : កំនត់ឈ្មោះរបស់ Checkbox
      value="Field Value"     : កំនត់តំលៃរបស់ Checkbox
      checked>                : កំនត់ Checkbox ថា Select រឺក៏មិន Select

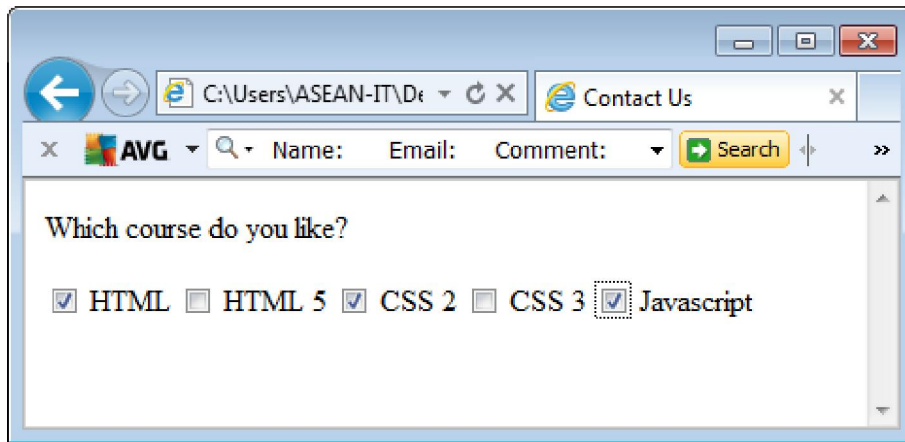
```

ឧទាហរណ៍

```

<html>
  <head><title>Checkbox Example</title></head>
  <body>
    <form name="form1" method="post" action="">
      Which course do you like?<br/><br/>
      <input type="checkbox" name="course" value="HTML" checked> HTML
      <input type="checkbox" name="course" value="HTML5"> HTML 5
      <input type="checkbox" name="course" value="css2"> CSS 2
      <input type="checkbox" name="course" value="css3"> CSS 3
      <input type="checkbox" name="course" value="js"> Javascript
    </form>
  </body>
</html>

```



7. ការបង្កើត Hidden Field

Hidden Fields មិនបានបង្ហាញអោយអ្នកទស្សនាមើលឃើញលើ browser window ទេ។

សារៈសំខាន់របស់ Hidden Fields គឺប្រើដើម្បីរក្សាទុកតំលៃផ្សេងៗសំរាប់បញ្ចូលនូវ Form ។

```

<input type="hidden"
      value="Field Value"
      name="Field Name" >

```

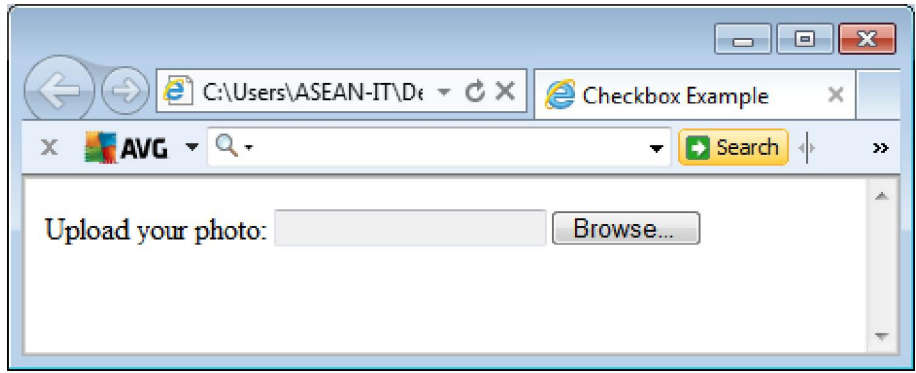
8. ការបង្កើត File

File អនុញ្ញាតអោយយើងបញ្ជូនឯកសារ (File) ពីក្នុងម៉ាស៊ីនរបស់យើង ទៅកាន់ម៉ាស៊ីន Server

```

<html>
  <head><title>Upload</title></head>
  <body>
    <form name="form1" method="post" action="" enctype="multipart/form-data">
      Upload your photo:
      <input type="file" name="upload" size="20">
    </form>
  </body>
</html>

```



9. ការបង្កើត Select

Select ប្រើដើម្បីបង្កើត Drop down menus ហើយ Select អនុញ្ញាតិអោយអ្នកទស្សនាអាចជ្រើសរើសមួយ រឺច្រើន Items ពី List ហើយអាចប្រើ Optgroup ប្រើដើម្បីបង្កើតក្រុមនៃ Items នៅក្នុងធាតុ Select ។

```

<select name="Field Name" size="n" multiple >
  <option value="value1"> Text 1</option>
  <option value="value2" selected>Text 2</option>
  .....
  .....
  <option value="valuen"> Text n </option>
</select>

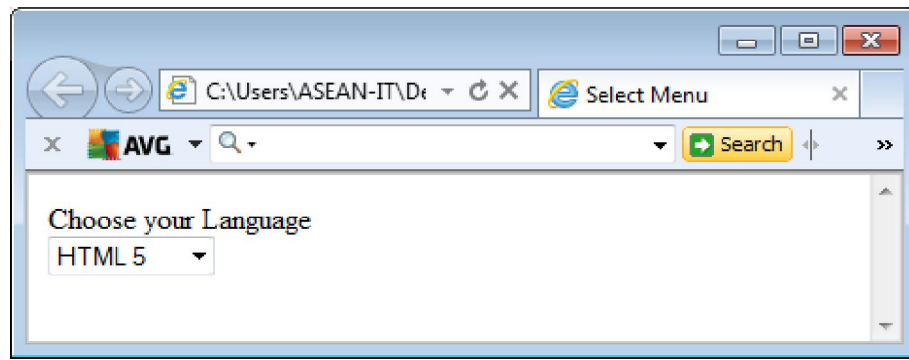
```

- multiple : អាចជ្រើសរើស Option បានចាំមួយឡើងទៅ
- size = "n" : កំនត់ចំនួន Items ដែលត្រូវបង្ហាញ ប្រសិនបើយើងមិនបានកំនត់ នៅចំនួន Item ដែលត្រូវបង្ហាញគឺមានតែមួយគត់
- selected : ជា Attribute ប្រើនៅក្នុង Option ដើម្បីជ្រើសរើស Option ណាមួយ

```

<html>
  <head><title>Select Menu</title></head>
  <body>
    <form name="form1" method="post" action="">
      Choose your Language<br/>
      <select name="language" size="1">
        <option value="html">HTML</option>
        <option value="html5" selected>HTML 5</option>
        <option value="css2">CSS 2</option>
        <option value="css3">CSS 3</option>
        <option value="js">Javascript</option>
      </select>
    </form>
  </body>
</html>

```



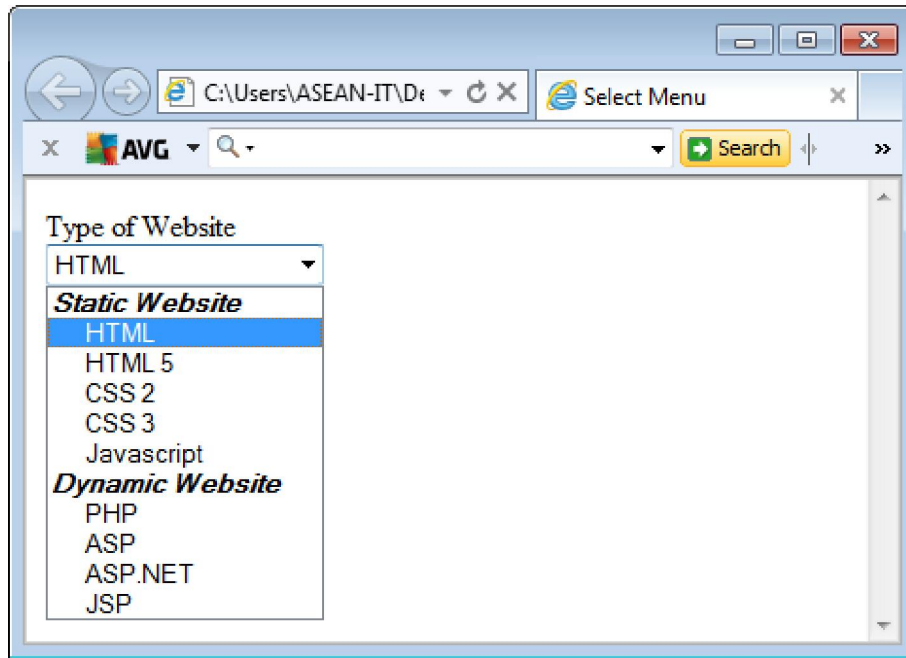
➤ ការប្រើជាមួយ Optgroup

Optgroup ប្រើដើម្បីបង្កើតជា ក្រុមនៃ items នៅក្នុង Select

```

<html>
  <head><title>Select Menu</title></head>
  <body>
    <form name="form1" method="post" action="">
      Type of Website<br/>
      <select name="language">
        <optgroup label="Static Website">
          <option value="html">HTML</option>
          <option value="html5" selected>HTML 5</option>
          <option value="css2">CSS 2</option>
          <option value="css3">CSS 3</option>
          <option value="js">Javascript</option>
        </optgroup>
        <optgroup label="Dynamic Website">
          <option value="php">PHP</option>
          <option value="asp">ASP</option>
          <option value="aspnet">ASP.NET</option>
          <option value="jsp">JSP</option>
        </optgroup>
      </select>
    </form>
  </body></html>

```



10. ការបង្កើត Button

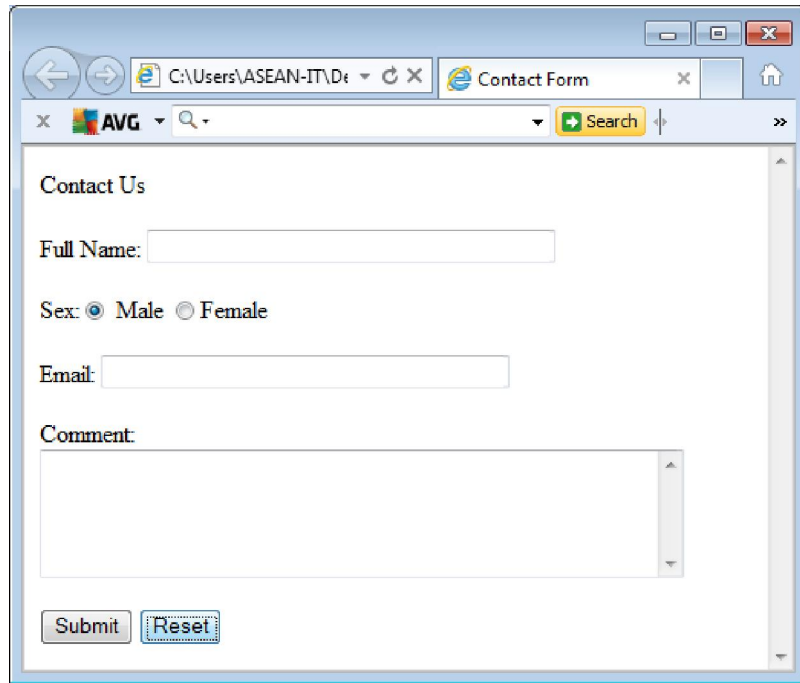
ក្នុងភាសា HTML យើងអាចបង្កើត Button បានចំនួន ៤ប្រភេទ:

- ❖ Submit button ប្រើដើម្បីធ្វើការបញ្ជូនរាល់ទិន្នន័យផ្សេងៗ ទៅកាន់ចំណុចដូចដែលបានកំណត់នៅក្នុង Action Attribute នៃ Form tag ។

```
<input type="submit" value="Field Value" name="Field Name" disabled >
```

Example:

```
<html>
  <head><title>Contact Form</title></head>
  <body>
    Contact Us
    <form name="form1" method="post" action="contact.php">
      Full Name: <input type="text" name="fullname" size="40">
    <br/><br/>
      Sex:<input type="radio" name="sex" value="Male" checked> Male
      <input type="radio" name="sex" value="Female">Female<br/><br/>
      Email: <input type="text" name="email" size="40"><br/><br/>
      Comment: <textarea name="comment" rows="5" cols="50"
      wrap="hard"></textarea>
      <br/><br/>
      <input type="submit" value="Submit" name="Submit">
    </form>
  </body>
</html>
```

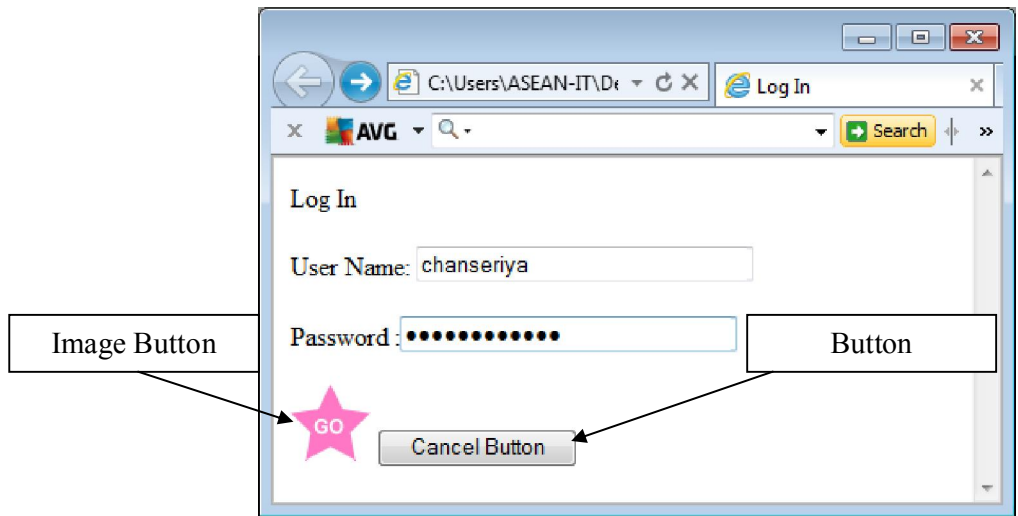
❖ **Button** វាមានលក្ខណៈខុសពី Submit និង Reset ព្រោះវាពុំមានការកំនត់ Function ច្បាស់លាស់ទេ។ យើងអាចដាក់ Script លើ Button ដើម្បីអោយវាដំណើរការនៅពេលមាន Events កើតឡើង ដូចជា Event on Click ជាដើម ។

```
<input type="button" value="Field Value" name="Field Name" disabled >
```

❖ **Image Button** ប្រសិនបើយើងមិនចូលចិត្តប្រើ button ដែលមានលក្ខណៈដូចជា Submit នោះទេយើង អាចប្រើប្រាស់ប្រភេទជារូបភាពជំនួសអោយ Button Submit ។ នៅពេលយើង ចុចលើរូបភាព ពេលនោះ វាដំណើរដូចគ្នាទៅនឹង button Submit ដែរ។

```
<input type="image"
      src="URL"
      name="Field Name"
      value="Field Value"
      alt="Description">
```

```
<html>
  <head><title>Log In</title></head>
  <body>
    Log In
    <form name="form1" method="post" action="login.php">
      User Name: <input type="text" name="username" size="30"> <br/><br/>
      Password :<input type="password" name="password" size="30"><br/><br/>
      <input type="image" src="go.gif" name="btngo" value="GO" alt="Click Go">
      <input type="button" value="Cancel Button" name="btnClick">
    </form>
  </body></html>
```



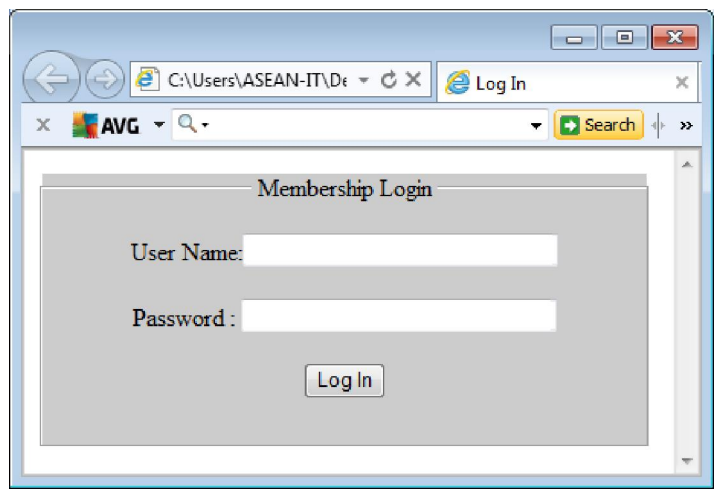
11. ការប្រើប្រាស់ Fieldset

The <fieldset> tag is used to logically group together elements in a form.

The <fieldset> tag draws a box around the related form elements.

The <legend> tag defines a caption for the fieldset element.

```
<html>
  <head><title>Log In</title></head>
  <body>
    <form name="form1" method="post" action="login.php">
      <fieldset style="width:400px; height:180px; text-align:center; background-color:#CCC">
        <legend align="center">Membership Login</legend><br/>
        User Name:<input type="text" name="username" size="30"> <br/><br/>
        Password : <input type="password" name="password" size="30"><br/><br/>
        <input type="submit" value="Log In" name="btnLogin">
      </fieldset>
    </form>
  </body>
</html>
```





Multimedia

សំឡេង និង រូបភាពជាផ្នែកមួយដែលឆ្លងអោយគេហទំព័ររបស់យើងទទួលបានប្រជាប្រិយភាពពីអ្នក ទស្សនា ជាពិសេសគឺបង្កើនសោភ័ណភាពដល់គេហទំព័រយើង។ សំរាប់ Browser ជំនាន់ថ្មីៗគឺ អនុញ្ញាតអោយយើង ធ្វើការជ្រើសរើសយកប្រភេទ file ជាច្រើនមកធ្វើការរចនាគេហទំព័រដូចជា file ដែលមាន extension: .wav, .aiff, .au, .mid, .mp3, .mp4, .swf, .ra, .ram, .avi..... ។ល។

1. Sound

Displaying audio in HTML is not easy!

You must add a lot of tricks to make sure you audio files will play in all browsers (Internet Explorer, Chrome, Firefox, Safari, Opera) and on all hardware (PC, Mac , iPad, iPhone).

Format	File
MIDI	.mid .midi
RealAudio	.rm .ram
Wave	.wav
WMA	.wma
MP3	.mp3 .mpga

★ What Format To Use?

The WAVE is the most popular **uncompressed** sound format on the Internet, and it is supported by all popular browsers. If you want uncompressed sound (music or speech) to be available to all your visitors, you should use the WAVE format.

The MP3 format is the newest format for **compressed** recorded music. The term MP3 has become synonymous with digital music. If your website is about recorded music, the MP3 format is the choice.

Using The <embed> Element

The purpose of the <embed> tag is to embed multimedia elements in HTML pages.

ទំរង់ទូទៅ

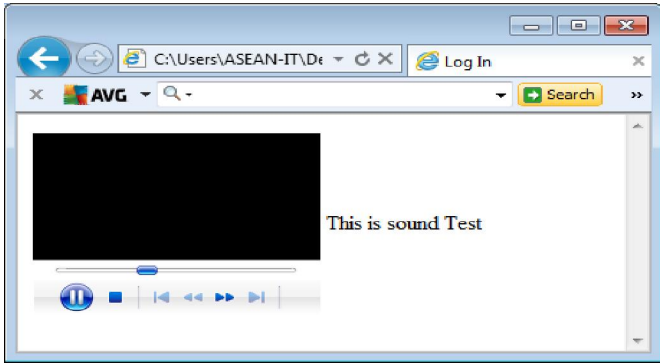
```

<embed      src="URL"           : កំនត់ទីតាំងរបស់ Sound file
           width="w"           : ទំហំបណ្តោយ
           height="h"          : ទំហំកំពស់
           autostart="true | false" : កំនត់ថាអោយវាចាប់ផ្តើមលេង រឺមិនលេង
           loop="true | false"  : កំនត់ចំនួនដំសំរាប់ Play
           align="left | right"> : សំរាប់តំរឹម ឆ្វេង ស្តាំ
</embed>
    
```

Example

```

<html>
  <head><title>Multimedia</title></head>
  <body>
    <embed src="02_Fanta_Calltune_chlong_chleuy.mp3" width="200"
    height="150" autostart="true" loop="true" align="absmiddle"></embed>
    This is sound Test
  </body>
</html>
    
```



Problems:

- The <embed> tag is unknown to HTML 4. Your page will not validate correctly.
- If your browser does not support the file format, your audio will not play.
- If you convert your file to another format, it will still not play in all browsers.

1.1 Using the <audio> Element

The <audio> element is an HTML 5 element, unknown to HTML 4, but it works in new browsers.

Example

```
<audio controls="controls">
  <source src="song.mp3" type="audio/mpeg" />
  <source src="song.ogg" type="audio/ogg" />
  Your browser does not support this audio
</audio>
```

The example above uses an Ogg file, to make it work in Firefox, Opera and Chrome. To make the audio work in Internet Explorer and Safari, a file of the type MP3 is added.

Currently, there are 3 main formats for the audio element:

Format	IE 9	Firefox 3.5	Opera 10.5	Chrome 3.0	Safari 3.0
Ogg Vorbis	No	Yes	Yes	Yes	No
MP3	Yes	No	No	Yes	Yes
Wav	No	Yes	Yes	Yes	Yes

Problems:

- You must convert your videos to many different formats.
- The <audio> element does not work in older browsers.
- The <audio> element does not validate in HTML 4 and XHTML.

The Best HTML Solution

Example

```
<audio controls="controls" height="50px width="100px">
  <source src="song.mp3" type="audio/mpeg" />
  <source src="song.ogg" type="audio/ogg" />
  <embed height="50px width="100px" src="song.mp3" />
</audio>
```

The example above uses 4 different audio formats. The HTML 5 <audio> element tries to play the video either as ogg or mp3. If this fails, the code "falls back" to try the <embed> element. If this also fails, it displays an error.

Problems:

- You must convert your videos to many different formats.
- The <audio> element does not validate in HTML 4 and XHTML.
- The <embed> element does not validate in HTML 4 and XHTML.

NOTE: Using <!DOCTYPE html> solves the validation problem.

1.2 Using the Yahoo Media Player

Using the Yahoo Media Player is a different approach. You simply let Yahoo do the job of playing your songs.

Example

```

<a href="song.mp3">Play Song</a>

<script type="text/javascript"
src="http://mediaplayer.yahoo.com/js"></script>

```

1.3 Using Google

Example

```

<a href="song.mp3">Play Song</a>

<embed type="application/x-shockwave-flash"
wmode="transparent"
src="http://www.google.com/reader/ui/3523697345-audio-
player.swf?audioUrl=song.mp3" height="27"
width="320"></embed>

```

2. Background Sound (Internet Explore Only)

Background sound មានន័យថាជាសំឡេងដែលនឹង លេងនៅពេលដែលយើងបើក webpage ជាពិសេសគឺវាមិនបង្ហាញនូវ form control ឡើយ។

```

<bgsound src="URL" : កំនត់ទីតាំង file sound
loop="n"> : កំនត់ចំនួនដងនៃសំឡេងដែលត្រូវលេង
ប្រសិនបើ n = -1 មានន័យថាវាផ្តើររហូត

```

Example

```

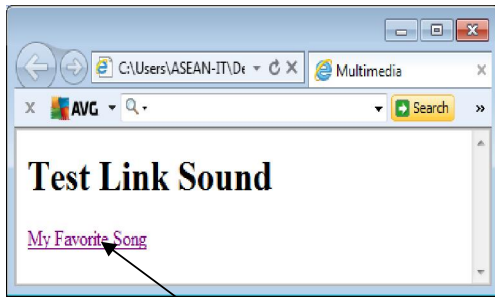
<html>
  <head><title>Multimedia</title></head>
  <body>
    <h1> Test Backgournd Sound</h1>
    <bgsound src="02_Fanta_Calltune chlong chleuy.mp3" loop="-1">
  </body>
</html>

```

3. Adding a link to a sound

យើងអាចធ្វើការ Link ទៅកាន់ sound file បានដោយគ្រាន់តែកំនត់ URL របស់ href attribute នៃ <a> tag ទៅកាន់ទីតាំង file sound អោយបានត្រឹមត្រូវ។

```
<html>
  <head><title>Multimedia</title>
</head>
<body>
  <h1> Test Link Sound</h1>
  <a href="02_Fanta_Calltune chlong chleuy.mp3">My Favorite Song</a>
</body>
</html>
```



នៅពេលចុចលើ Link វានឹងទៅបើកកម្មវិធីសំរាប់ចំរៀងរួចធ្វើការ play

4. ការបញ្ចូល Video File ទៅក្នុង Webpage

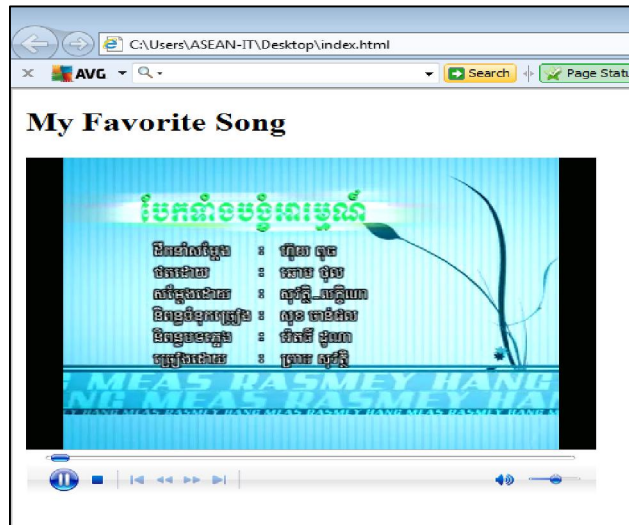
ឯកសារ Video file ជាប្រភេទឯកសារមានទាំងសំឡេងនិងរូបភាព យើងអាចប្រើប្រាស់ Video file មួយចំនួនដូចជា:

Description	Extension
Quick Time Movie	.mov or .qt
MPEG Movie	.mpeg or .mpg
Audio-Video Interleaved	.avi
Windows Media movies	.wmv
Flash	.swf
Mpeg-4	.mp4

4.1 The <embed> Tag

Example

```
<html>
  <head><title>Video</title></head>
  <body>
    <h2>My Favorite Song</h2>
    <embed src="Preap Sovath.avi" width="400" height="300"
  autostart="true" loop="true" align="absmiddle"></embed>
  </body>
</html>
```



Problems

- The <embed> tag is unknown to HTML 4. Your page will not validate correctly.
- If the browser does not support Flash, your video will not play.
- iPad and iPhone cannot display Flash videos.
- If you convert the video to another format, it will still not play in all browsers.

4.2 Using the <video> Tag

The <video> element is new in HTML 5.

The purpose of the <video> tag is to embed video elements in HTML pages.

The following HTML fragment displays a video in ogg, mp4, or webm format embedded in a web page:

```
<video width="320" height="240" controls="controls">
  <source src="movie.mp4" type="video/mp4" />
  <source src="movie.ogg" type="video/ogg" />
  <source src="movie.webm" type="video/webm" />
  Your browser does not support the video tag.
</video>
```

Problems:

- You must convert your videos to many different formats.
- The <video> element does not work in older browsers.
- The <video> element does not validate in HTML 4 and XHTML

The Best HTML Solution: HTML 5 + <object> + <embed>

```
<video width="320" height="240" controls="controls">
  <source src="movie.mp4" type="video/mp4" />
  <source src="movie.ogg" type="video/ogg" />
  <source src="movie.webm" type="video/webm" />
  <object data="movie.mp4" width="320" height="240">
  <embed src="movie.swf" width="320" height="240">
  Your browser does not support video
</object>
</video>
```

The example above uses 4 different video formats. The HTML 5 <video> element tries to play the video either in mp4, ogg, or webm formats. If this fails, the code "falls back" to try the <object> element. If this also fails, it "falls back" to the <embed> element.

Problems:

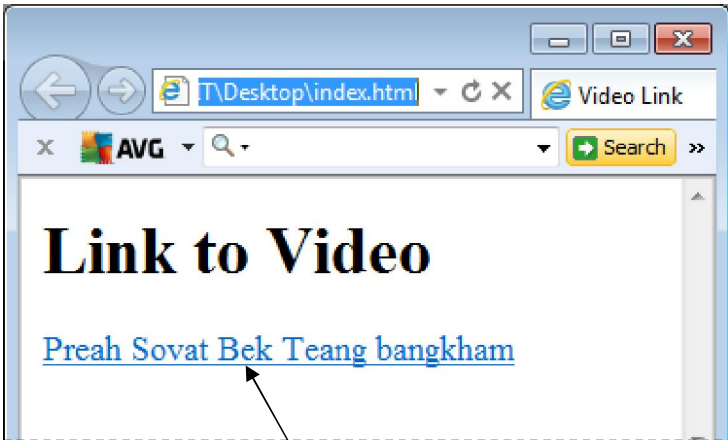
- You must convert your videos to many different formats.
- The <video> element does not validate in HTML 4 and XHTML.
- The <embed> element does not validate in HTML 4 and XHTML.

NOTE: Using <!DOCTYPE html> solves the validation problem.

5. Adding a Link to External Video

យើងអាចធ្វើការ Link ទៅកាន់ video file បានដោយគ្រាន់តែកំណត់ URL របស់ href attribute នៃ <a> tag ទៅកាន់ទីតាំង file video អោយបានត្រឹមត្រូវ។

```
<html>
  <head><title>Video Link</title>
</head>
<body>
  <h1> Link to Video</h1>
  <a href=" Preah Sovath.avi">Preah Sovat – Bek Teang bangkham</a>
</body>
</html>
```



នៅពេលចុចលើ Link វានឹងទៅបើកកម្មវិធីសំបាប់ចំរៀងរួចធ្វើការ play

6. Playing a YouTube Video in HTML

The easiest way to display videos (others or your own) in HTML is to use YouTube.If you want to display a video in a web page, you can upload the video to YouTube and insert HTML code to display the video in your web page.

Example

```
<html>
<body>
  <iframe width="420" height="315"
    src="http://www.youtube.com/embed/EXkfU1klbJU"
    frameborder="0" allowfullscreen></iframe>
</body>
</html>
```

7. Embedding Windows Media

```
<object width="100%" height="100%"
  type="video/x-ms-asf" url="3d.wmv" data="3d.wmv"
  classid="CLSID:6BF52A52-394A-11d3-B153-00C04F79FAA6">
  <param name="url" value="3d.wmv">
  <param name="filename" value="3d.wmv">
  <param name="autostart" value="1">
  <param name="uiMode" value="full" />
  <param name="autosize" value="1">
  <param name="playcount" value="1">
  <embed type="application/x-mplayer2" src="3d.wmv" width="100%"
  height="100%" autostart="true" showcontrols="true"
  pluginspage="http://www.microsoft.com/Windows/MediaPlayer/"></embed>
</object>
```

8. Embedding Flash

```
<object width="300" height="150" id="flash1" name="flash1"
  classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
  codebase="http://fpdownload.macromedia.com/pub/shockwave/ca
  bs/flash/swflash.cab#version=8,0,0,0">
  <param name="movie" value="test.swf">
  <param name="quality" value="high" >
  <embed src="test.swf" width="300" height="150" id="flash1"
  name="flash1" quality="autohigh" bgcolor="#ffffff" type="application/x-
  shockwave-flash" ></embed>
</object>
```

9. Creating a Marquee (for Internet Explorer Only)

The marquee element is unique to Internet Explorer. It displays HTML content in a scrolling region on the page.

```

<marquee behavior="scroll | slide | alternate"
          direction="left | right | up | down"
          loop="-1 | n"
          width="w"
          height="h"
          bgcolor="color_name | color_code"
          hspace="n"
          vspace="n"
          scrolldelay="n"
          scrollamount="n"
          >
    Scrolling Text
</marquee>

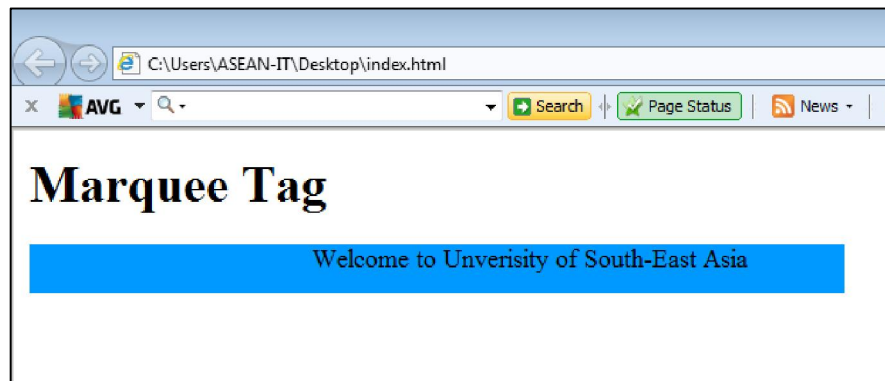
```

Example

```

<html>
  <head><title>Scrolling Text</title> </head>
  <body>
    <h1> Marquee Tag</h1>
    <marquee behavior="scroll"
            direction="right"
            loop="-1"
            width="500"
            height="30"
            bgcolor="#0099FF"
            scrollamount="10">
      Welcome to Unverisity of South-East Asia
    </marquee>
  </body></html>

```





Head Section

The head element is a container for all the head elements. Elements inside <head> can include scripts, instruct the browser where to find style sheets, provide meta information, and more. The following tags can be added to the head section: <title>, <base>, <link>, <meta>, <script>, and <style>.

1. title Element

The <title> </title> tag ប្រើសំរាប់កំណត់ចំនងជើងអោយ Webpage ។ title element is required in all HTML/XHTML documents ។ យើងប្រើប្រាស់ title element ដើម្បី:

- កំណត់ចំនងជើងក្នុង the browser toolbar
- ផ្តល់នូវចំនងជើងនៅពេលដែលយើង added to favorites
- បង្ហាញចំនងជើងសំរាប់ page in search-engine results

Example :

```
<html>
<head>
<title>Title of the document</title>
</head>

<body>
The content of the document.....
</body>
</html>
```

2. base Element

The <base> tag specifies a default address or a default target for all links on a page:

```
<head>
<base href="http://www.usea.edu.kh/images/" />
<base target="_blank" />
</head>
```

3. link Element

The <link> tag សំរាប់កំណត់ទំនាក់ទំនងរវាង HTML document និង resource ខាងក្រៅ។ The <link> tag គឺភាគច្រើនប្រើសំរាប់ធ្វើការភ្ជាប់ទៅកាន់ style sheets:

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css" />
</head>
```

4. style Element

The <style> tag គឺប្រើសំរាប់កំណត់ style information សំរាប់ an HTML document ។ នៅក្នុង style element you specify how HTML elements should render in a browser:

```
<head>
<style type="text/css">
body {background-color:yellow}
p {color:blue}
</style>
</head>
```

5. meta Element

The <meta> tag provides metadata about the HTML document. Metadata is information about data. Metadata will not be displayed on the page, but will be machine parsable. Meta elements are typically used to specify page description, keywords, author of the document, last modified, and other metadata. <meta> tag always goes inside the head element.

🔍 Keywords for Search Engines

Some search engines will use the name and content attributes of the meta element to index your pages. The following meta element defines a description of a page:

```
<meta name="description" content="Free Web tutorials on HTML, CSS, XML" />
```

The following meta element defines keywords for a page:

```
<meta name="keywords" content="HTML, CSS, XML" />
```

Note: A lot of webmasters have used <meta> tags for spamming, like repeating keywords (or using wrong keywords) for higher ranking. Therefore, most search engines have stopped using <meta> tags to index/rank pages.

🔗 Controlling other Information

You can also add information to your page about who wrote it, what program was used to generate the HTML code and if it is copyrighted. Note, however, that search engines do not currently use this information and browser do not display it.

```
<meta name= "author" content= "name">
```

```
<meta name= "generator" content= "program">
```

```
<meta name= "copyright" content= "copyright info">
```

🔗 Specifying Character Encoding with <meta>

The <meta> tag is also used to specify the character encoding for your documents

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

🔗 Automatic Document Refresh with <meta>

The <meta> tag can also be used to force a browser to automatically reload a document

```
<meta http-equiv="refresh" content="10 url=http://www.google.com/" />
```

browser will wait 10 seconds before reloading the current document.

6. script Element

The <script> tag is used to define a client-side script, such as a JavaScript. The script element either contains scripting statements or it points to an external script file through the src attribute. Common uses for JavaScript are image manipulation, form validation, and dynamic changes of content.

The script below writes Hello World! to the HTML output:

```
<script type="text/javascript">  
    document.write("Hello World!")  
</script>
```

🔗 noscript Element

The <noscript> tag is used to provide an alternate content for users that have disabled scripts in their browser or have a browser that doesn't support client-side scripting. The noscript element can contain all the elements that you can find inside the body element of a normal HTML page. The

content inside the noscript element will only be displayed if scripts are not supported, or are disabled in the user's browser:

```
<script type="text/javascript">
document.write("Hello World!")
</script>
<noscript>Sorry, your browser does not support
JavaScript!</noscript>
```

7. Submit Your Site to Search Engines

Web search engines (like Google) automatically add new web sites to their search index every time they crawl the web.

If your web site is new and unknown, it may take some time before your site is "discovered".

Luckily most search engines invites you to submit your site: *

Google: <http://www.google.com/addurl.html>

Yahoo: <http://search.yahoo.com/info/submit.html>

Bing: <http://www.bing.com/webmaster/SubmitSitePage.aspx>

Open Directory: <http://www.dmoz.org/help/submit.html>

Normally you will have to enter the full URL of your site including the http:// prefix.

Example: <http://www.usea.edu.kh/>

When you submit your site to a search engine, you only need to specify the top-level (home) page. You do not need to submit each page. The search engine will find the rest based on your links (keep your navigation menus tidy).

សេចក្តីបន្ថែម

HTML Entities

Reserved characters in HTML must be replaced with character entities. It is not possible to use the less than (<) or greater than (>) signs in your text, because the browser will mix them with tags. To actually display reserved characters, we must use character entities in the HTML source code.

A character entity looks like this:

```
&entity_name;

OR

&#entity_number;
```

- To display a less than sign we must write: **<** or **<**;
- the non-breaking space we must write ** **;

Tip: The advantage of using an entity name, instead of a number, is that the name is easier to remember. However, the disadvantage is that browsers may not support all entity names (the support for entity numbers is very good).

HTML Useful Character Entities

Note: Entity names are case sensitive!

Result	Description	Entity Name	Entity Number
	non-breaking space	 	
<	less than	<	<
>	greater than	>	>
&	ampersand	&	&
¢	cent	¢	¢
£	pound	£	£
¥	yen	¥	¥
€	euro	€	€
§	section	§	§
©	copyright	©	©
®	registered trademark	®	®
™	trademark	™	™

Symbols

Character	Entity Number	Entity Name	Description
	 	 	non-breaking space
¡	¡	¡	inverted exclamation mark
¢	¢	¢	cent
£	£	£	pound
¤	¤	¤	currency
¥	¥	¥	yen
¦	¦	¦	broken vertical bar
§	§	§	section

¨	¨	¨	spacing diaeresis
©	©	©	copyright
^a	ª	ª	feminine ordinal indicator
«	«	«	angle quotation mark (left)
¬	¬	¬	negation
–	­	­	soft hyphen
®	®	®	registered trademark
—	¯	¯	spacing macron
°	°	°	degree
±	±	±	plus-or-minus
²	²	²	superscript 2
³	³	³	superscript 3
´	´	´	spacing acute
μ	µ	µ	micro
¶	¶	¶	paragraph
·	·	·	middle dot
¸	¸	¸	spacing cedilla
¹	¹	¹	superscript 1
º	º	º	masculine ordinal indicator
»	»	»	angle quotation mark (right)
¼	¼	¼	fraction 1/4
½	½	½	fraction 1/2
¾	¾	¾	fraction 3/4
¿	¿	¿	inverted question mark
×	×	×	multiplication
÷	÷	÷	division

Characters

Character	Entity Number	Entity Name	Description
À	À	À	capital a, grave accent
Á	Á	Á	capital a, acute accent
Â	Â	Â	capital a, circumflex accent
Ã	Ã	Ã	capital a, tilde
Ä	Ä	Ä	capital a, umlaut mark
Å	Å	Å	capital a, ring
Æ	Æ	Æ	capital ae
Ç	Ç	Ç	capital c, cedilla
È	È	È	capital e, grave accent
É	É	É	capital e, acute accent
Ê	Ê	Ê	capital e, circumflex accent
Ë	Ë	Ë	capital e, umlaut mark
Ì	Ì	Ì	capital i, grave accent
Í	Í	Í	capital i, acute accent
Î	Î	Î	capital i, circumflex accent
Ï	Ï	Ï	capital i, umlaut mark
Ð	Ð	Ð	capital eth, Icelandic
Ñ	Ñ	Ñ	capital n, tilde
Ò	Ò	Ò	capital o, grave accent
Ó	Ó	Ó	capital o, acute accent

Ô	Ô	Ô	capital o, circumflex accent
Õ	Õ	Õ	capital o, tilde
Ö	Ö	Ö	capital o, umlaut mark
Ø	Ø	Ø	capital o, slash
Ù	Ù	Ù	capital u, grave accent
Ú	Ú	Ú	capital u, acute accent
Û	Û	Û	capital u, circumflex accent
Ü	Ü	Ü	capital u, umlaut mark
Ý	Ý	Ý	capital y, acute accent
Þ	Þ	Þ	capital THORN, Icelandic
ß	ß	ß	small sharp s, German
à	à	à	small a, grave accent
á	á	á	small a, acute accent
â	â	â	small a, circumflex accent
ã	ã	ã	small a, tilde
ä	ä	ä	small a, umlaut mark
å	å	å	small a, ring
æ	æ	æ	small ae
ç	ç	ç	small c, cedilla
è	è	è	small e, grave accent
é	é	é	small e, acute accent
ê	ê	ê	small e, circumflex accent
ë	ë	ë	small e, umlaut mark
ì	ì	ì	small i, grave accent
í	í	í	small i, acute accent
î	î	î	small i, circumflex accent
ï	ï	ï	small i, umlaut mark
ð	ð	ð	small eth, Icelandic
ñ	ñ	ñ	small n, tilde
ò	ò	ò	small o, grave accent
ó	ó	ó	small o, acute accent
ô	ô	ô	small o, circumflex accent
õ	õ	õ	small o, tilde
ö	ö	ö	small o, umlaut mark
ø	ø	ø	small o, slash
ù	ù	ù	small u, grave accent
ú	ú	ú	small u, acute accent
û	û	û	small u, circumflex accent
ü	ü	ü	small u, umlaut mark
ý	ý	ý	small y, acute accent
þ	þ	þ	small thorn, Icelandic
ÿ	ÿ	ÿ	small y, umlaut mark

အိန္ဒိယ ၅၆ Cascading Style Sheets

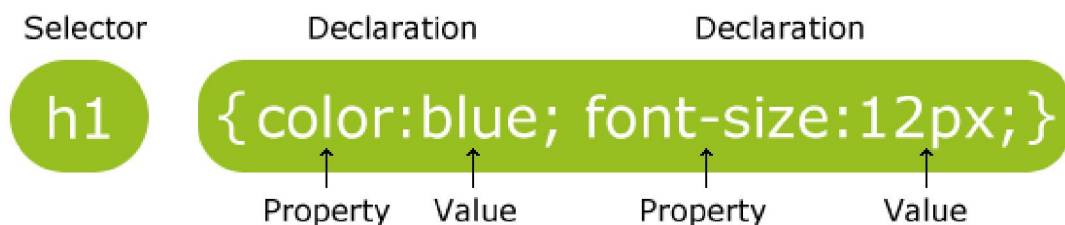
1. CSS Overview

Cascading Style Sheets were created to provide a powerful, yet flexible means for formatting HTML content. Style sheets allow authors to specify layout and design elements, such as fonts, colors, and text indentation. Style sheets give you precise control over how elements appear on a Web page. What's even better is that you can create a single style sheet for an entire Web site to ensure that the layout and display of your content is consistent from page to page. And for the last bit of icing on the cake, Web style sheets are easy to build and even easier to integrate into your Web pages.

Generally, style sheets give you more flexibility than markup ever will, and the HTML element collection won't grow to include more display-oriented tags. When you want tight control over the display of your Web pages, style sheets are the way to go.

2. CSS Syntax

A CSS rule has two main parts: a selector, and one or more declarations:



The selector is normally the HTML element you want to style. Each declaration consists of a property and a value. The property is the style attribute you want to change. Each property has a value.

3. CSS Comments

Comments are used to explain your code, and may help you when you edit the source code at a later date. Comments are ignored by browsers.

A CSS comment begins with "/*", and ends with "*/", like this:

```

/*This is a comment*/
p
{
text-align:center;
/*This is another comment*/
color:black;
font-family:arial;
}

```


4. CSS Id and Class

In addition to setting a style for a HTML element, CSS allows you to specify your own selectors called "id" and "class".

4.1 The id Selector

The id selector is used to specify a style for a single, unique element. The id selector uses the id attribute of the HTML element, and is defined with a "#". The style rule below will be applied to the element with id="test":

```
#test
{
    text-align:justify;
    color:black;
}
```

NOTE: Do NOT start an ID name with a number! It will not work in Mozilla/Firefox.

4.2 The class Selector

The class selector is used to specify a style for a group of elements. Unlike the id selector, the class selector is most often used on several elements. This allows you to set a particular style for any HTML elements with the same class. The class selector uses the HTML class attribute, and is defined with a "."

In the example below, all HTML elements with class="center" will be center-aligned:

```
.center {text-align:center;}
```

5. Applying CSS to an HTML document

There are three ways you can apply CSS to an HTML document. These methods are all outlined below. We recommend that you focus on the third method i.e. external.

5.1 Method 1: In-line (the attribute style)

One way to apply CSS to HTML is by using the HTML attribute style. Building on the above example with the red background color, it can be applied like this:

```
<html>
<head>
  <title>Example</title>
</head>
<body style="background-color: #FF0000;">
  <p>This is a red page</p>
</body>
</html>
```

5.2 Method 2: Internal (the tag style)

Another way is to include the CSS codes using the HTML tag `<style>`.

```
<html>
  <head>
    <title>Example</title>
    <style type="text/css">
      body {background-color: #FF0000;}
    </style>
  </head>
  <body>
    <p>This is a red page</p>
  </body>
</html>
```

5.3 Method 3: External (link to a style sheet)

The recommended method is to link to a so-called external style sheet. Throughout this tutorial we will use this method in all our examples. An external style sheet is simply a text file with the extension `.css`. Like any other file, you can place the style sheet on your web server or hard disk. For example, let's say that your style sheet is named `style.css` and is located in a folder named `style`. The situation can be illustrated like this:



The trick is to create a link from the HTML document (`default.htm`) to the style sheet (`style.css`). Such link can be created with one line of HTML code:

```
<link rel="stylesheet" type="text/css" href="style/style.css" />
```

Notice how the path to our style sheet is indicated using the attribute `href`. The line of code must be inserted in the header section of the HTML code i.e. between the `<head>` and `</head>` tags.

Like this:

```
<html>
  <head>
    <title>My document</title>
    <link rel="stylesheet" type="text/css" href="style/style.css" />
  </head>
  <body> .....
```

This link tells the browser that it should use the layout from the CSS file when displaying the HTML file. The really smart thing is that several HTML documents can be linked to the same style sheet. In other words, one CSS file can be used to control the layout of many HTML documents.

6. CSS Background

CSS background properties are used to define the background effects of an element.

CSS properties used for background effects:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

6.1 Background Color

The background-color property specifies the background color of an element. The background color of a page is defined in the body selector:

```
body {background-color:#b0c4de;}
h1 {background-color:#6495ed;}
p {background-color:#e0fff;}
div {background-color:#b0c4de;}
```

The background color can be specified by:

- name - a color name, like "red"
- RGB - an RGB value, like "rgb(255,0,0)"
- Hex - a hex value, like "#ff0000"

6.2 Background Image

The background-image property specifies an image to use as the background of an element. By default, the image is repeated so it covers the entire element. The background image for a page can be set like this:

```
body {background-image:url('mypic.gif');}
```

6.3 Repeat background image [background-repeat]

In the example above, did you notice that by default the image was repeated both horizontally and vertically to cover the entire screen? The property background-repeat controls this behaviour. The table below outlines the four different values for background-repeat.

Value	Description
background-repeat: repeat-x	The image is repeated horizontally
background-repeat: repeat-y	The image is repeated vertically
background-repeat: repeat	The image is repeated both horizontally and vertically
background-repeat: no-repeat	The image is not repeated

For example, to avoid repetition of a background image the code should look like this:

```
body {
    background-color: #FFCC66;
    background-image: url("butterfly.gif");
    background-repeat: no-repeat;
}
```

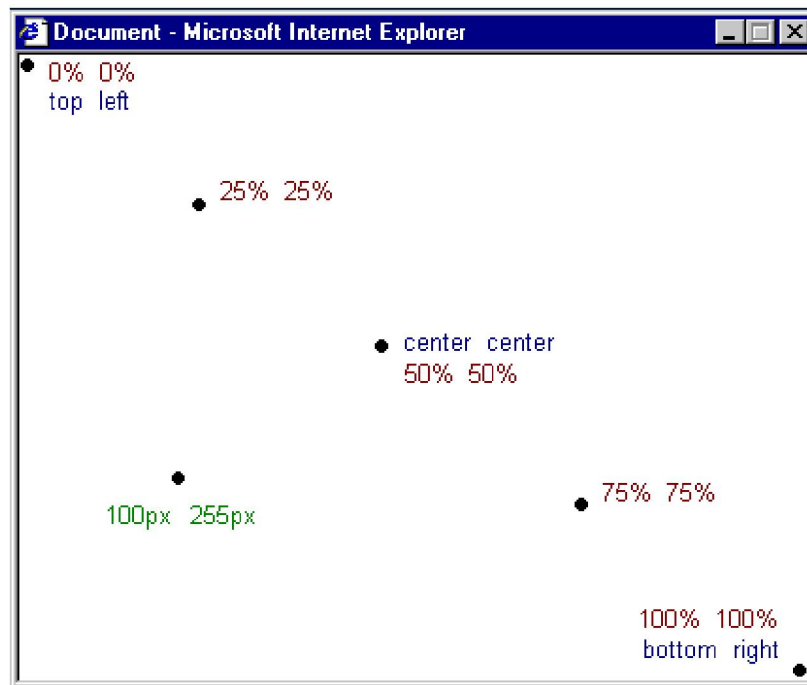
6.4 Lock background image [background-attachment]

The property background-attachment specifies whether a background picture is fixed or scrolls along with the containing element. A fixed background image will not move with the text when a reader is scrolling the page, whereas an unlocked background image will scroll along with the text of the web page. The table below outlines the two different values for background-attachment.

Value	Description
Background-attachment: scroll	The image scrolls with the page - unlocked
Background-attachment: fixed	The image is locked

6.5 Place background image [background-position]

By default, a background image will be positioned in the top left corner of the screen. The property background-position allows you to change this default and position the background image anywhere you like on the screen. There are numerous ways to set the values of background-position. However, all of them are formatted as a set of coordinates. For example, the value '100px 200px' positions the background image 100px from the left side and 200px from the top of the browser window. The coordinates can be indicated as percentages of the browser window, fixed units (pixels, centimetres, etc.) or you can use the words top, bottom, center, left and right. The model below illustrates the system:



The table below gives some examples.

Value	Description
background-position: 2cm 2cm	The image is positioned 2 cm from the left and 2 cm down the page
background-position: 50% 25%	The image is centrally positioned and one fourth down the page
background-position: top right	The image is positioned in the top-right corner of the page

6.6 Background - Shorthand property

As you can see from the examples above, there are many properties to consider when dealing with backgrounds. To shorten the code, it is also possible to specify all the properties in one single property. This is called a shorthand property. The shorthand property for background is simply "background":

```
body {background:#ffffff url('img_tree.png') no-repeat right top;}
```

When using the shorthand property the order of the property values are:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

7. CSS Font

We will also look at how to work around the issue that specific fonts chosen for a website can only be seen if the font is installed on the PC used to access the website. The following CSS properties will be described:

- font-family
- font-style
- font-variant
- font-weight
- font-size
- font

7.1 Font family [font-family]

The property font-family is used to set a prioritized list of fonts to be used to display a given element or web page. If the first font on the list is not installed on the computer used to access the site, the next font on the list will be tried until a suitable font is found.

There are two types of names used to categorize fonts: family-names and generic families. The two terms are explained below.

Family-name

Examples of a family-name (often known as "font") can e.g. be "Arial", "Times New Roman" or "Tahoma".

Generic family

Generic families can best be described as groups of family-names with uniformed appearances. The difference can also be illustrated like this:

Times New Roman
Garamond
Georgia

These three font-families belong to the generic family **serif**. They are characterized by all having "feet".

Trebuchet
Arial
Verdana

These three font-families belong to the generic family **sans-serif**. They are all characterized by not having "feet".

Courier
Courier New
Andale Mono

These three font-families belong to the generic family **monospace**. They are all characterized by all characters having a fixed width

When you list fonts for your web site, you naturally start with the most preferred font followed by some alternative fonts. It is recommended to complete the list with a generic font family. That way at least the page will be shown using a font of the same family if none of the specified fonts are available.

7.2 Font style [font-style]

The property font-style defines the chosen font either in **normal**, **italic** or **oblique**. In the example below, all headlines marked with <h2> will be shown in italics.

```
h1 {font-family: arial, verdana, sans-serif;}
h2 {font-family: "Times New Roman", serif; font-style: italic;}
```

7.3 Font variant [font-variant]

The property font-variant is used to choose between **normal** or **small-caps** variants of a font. A **small-caps** font is a font that uses smaller sized capitalized letters (upper case) instead of lower case letters. Confused? Take a look at these examples:

Sans Book SC	Sans Bold SC	Serif Book SC	Serif Bold SC
ABCABC	ABCABC	ABCABC	ABCABC

If font-variant is set to **small-caps** and no small-caps font is available the browser will most likely show the text in uppercase instead.

```
h1 {font-variant: small-caps;}
h2 {font-variant: normal;}
```

7.4 Font weight [font-weight]

The property font-weight describes how bold or "heavy" a font should be presented. A font can either be **normal** or **bold**. Some browsers even support the use of numbers between 100-900 (in hundreds) to describe the weight of a font.

```
p {font-family: arial, verdana, sans-serif;}
td {font-family: arial, verdana, sans-serif; font-weight: bold;}
```

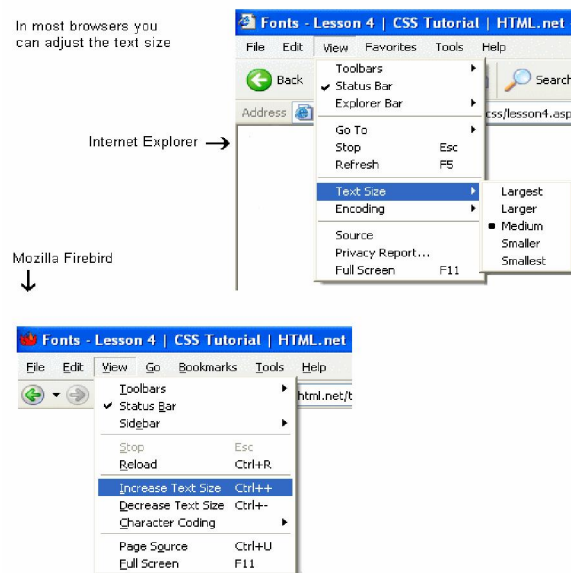
7.5 Font size [font-size]

The size of a font is set by the property font-size. There are many different units (e.g. pixels and percentages) to choose from to describe font sizes. In this tutorial we will focus on the most common and appropriate units. Examples include:

```
h2 {font-size: 12pt;}
h3 {font-size: 120%;}
p {font-size: 1em;}
```

There is one key difference between the four units above. The units 'px' and 'pt' make the font size absolute, while '%' and 'em' allow the user to adjust the font size as he/she see fit. Many users are disabled, elderly or simply suffer from poor vision or a monitor of bad quality. **To make your website accessible** for everybody, you should use adjustable units such as '%' or 'em'.

Below you can see an illustration of how to adjust the text size in Mozilla Firefox and Internet Explorer. Try it yourself - neat feature, don't you think?



7.6 Compiling [font]

Using the font short hand property it is possible to cover all the different font properties in one single property. For example, imagine these four lines of code used to describe font-properties for <p>:

```
p {
    font-style: italic;
    font-weight: bold;
    font-size: 30px;
    font-family: arial, sans-serif;
}
```

Using the short hand property, the code can be simplified:

```
p { font: italic bold 30px arial, sans-serif; }
```


The order of values for font is:

font-style | font-variant | font-weight | font-size | font-family

8. CSS Text

Formatting and adding style to text is a key issue for any web designer. In this lesson you will be introduced to the amazing opportunities CSS gives you to add layout to text. The following properties will be described:

- text-indent
- text-align
- text-decoration
- letter-spacing
- text-transform

8.1 Text indentation [text-indent]

The property text-indent allows you to add an elegant touch to text paragraphs by applying an indent to the first line of the paragraph. In the example below a **30px** is applied to all text paragraphs marked with <p>:

```
p { text-indent: 30px; }
```

8.2 Text alignment [text-align]

The CSS property text-align corresponds to the attribute align used in old versions of HTML. Text can either be aligned to the **left**, to the **right** or **centred**. In addition to this, the value **justify** will stretch each line so that both the right and left margins are straight. You know this layout from for example newspapers and magazines.

In the example below the text in table headings <th> is aligned to the right while the table data <td> are center. In addition, normal text paragraphs are justified:

```
th {  
    text-align: right;  
}  
td {  
    text-align: center;  
}  
p {  
    text-align: justify;  
}
```

8.3 Text decoration [text-decoration]

The property text-decoration makes it is possible to add different "decorations" or "effects" to text. For example, you can underline the text, have a line through or above the text, etc. In the following example, <h1> are underlined headlines, <h2> are headlines with a line above the text and <h3> are headlines with a line though the text.

```
h1 { text-decoration: underline; }
h2 { text-decoration: overline; }
h3 { text-decoration: line-through; }
```

8.4 Letter space [letter-spacing]

The spacing between text characters can be specified using the property letter-spacing. The value of the property is simply the desired width. For example, if you want a spacing of **3px** between the letters in a text paragraph <p> and **6px** between letters in headlines <h1> the code below could be used.

```
h1 { letter-spacing: 6px; }
p { letter-spacing: 3px; }
```

8.5 Text transformation [text-transform]

The text-transform property controls the capitalization of a text. You can choose to **capitalize**, use **uppercase** or **lowercase** regardless of how the original text is looks in the HTML code. An example could be the word "headline" which can be presented to the user as "HEADLINE" or "Headline". There are four possible values for text-transform:

capitalize

Capitalizes the first letter of each word. For example: "the sun" will be "The Sun".

uppercase

Converts all letters to uppercase. For example: " the sun " will be "THE SUN".

lowercase

Converts all letters to lowercase. For example: "JOHN DOE" will be "john doe".

none

No transformations - the text is presented as it appears in the HTML code.

Let's say that we want names to be capitalized and headlines to be presented in uppercase letters. Try to take a look at the HTML code for this example and you will see that the text actually is in lowercase.

```
h1 { text-transform: uppercase; }  
li { text-transform: capitalize; }
```

9. CSS Links

You can apply what you already learned in the previous lessons to links (i.e. change colors, fonts, underline, etc). The new thing is that CSS allows you to define these properties differently depending on whether the link is unvisited, visited, active, or whether the cursor is on the link. This makes it possible to add fancy and useful effects to your website. To control these effects you use so-called pseudo-classes.

What is a pseudo-class?

A pseudo-class allows you to take into account different conditions or events when defining a property for an HTML tag. Let's look at an example. As you know, links are specified in HTML with <a> tags. We can therefore use a as a selector in CSS:

```
a { color: blue; }
```

A link can have different states. For example, it can be visited or not visited. You can use pseudo-classes to assign different styles to visited and unvisited links.

```
a:link { color: blue; }  
  
a:visited { color: red; }
```

Use a:link and a:visited for unvisited and visited links respectively. Links that are active have the pseudo-class a:active and a:hover is when the cursor is on the link. We will now go through each of the four pseudo-classes with examples and further explanation.

Pseudo-class: link

The pseudo-class :link is used for links leading to pages that the user has not visited. In the code example below, unvisited links will be light blue.

```
a:link { color: #6699CC; }
```

Pseudo-class: visited

The pseudo-class `:visited` is used for links leading to pages that the user has visited. For example, the code below would make all visited links dark purple:

```
a:visited { color: #660099; }
```

Pseudo-class: active

The pseudo-class `:active` is used for links that are active. This example gives active links a yellow background color:

```
a:active { background-color: #FFFF00; }
```

Pseudo-class: hover

The pseudo-class `:hover` is used when the mouse pointer hovers over a link. This can be used to create interesting effects. For example, if we want our links to be orange and be italicized when the cursor is pointed at them, our CSS should look like this:

```
a:hover { color: orange; font-style: italic; }
```

10. CSS Lists

The CSS list properties allow you to:

- Set different list item markers for ordered lists
- Set different list item markers for unordered lists
- Set an image as the list item marker

In HTML, there are two types of lists:

- unordered lists - the list items are marked with bullets
- ordered lists - the list items are marked with numbers or letters

With CSS, lists can be styled further, and images can be used as the list item marker. The type of list item marker is specified with the `list-style-type` property:

```
ul.a {list-style-type: circle;}  
ul.b {list-style-type: square;}  
ol.c {list-style-type: upper-roman;}  
ol.d {list-style-type: lower-alpha;}
```

An Image as The List Item Marker

To specify an image as the list item marker, use the list-style-image property

```
ul
{
list-style-image: url('sqpurple.gif');
}
```

The example above does not display equally in all browsers. IE and Opera will display the image-marker a little bit higher than Firefox, Chrome, and Safari. If you want the image-marker to be placed equally in all browsers, a crossbrowser solution is explained below.

Crossbrowser Solution

The following example displays the image-marker equally in all browsers:

```
ul
{
list-style-type: none;
padding: 0px;
margin: 0px;
}
li
{
background-image: url(sqpurple.gif);
background-repeat: no-repeat;
background-position: 0px 5px;
padding-left: 14px;
}
```

List - Shorthand property

It is also possible to specify all the list properties in one, single property. This is called a shorthand property. The shorthand property used for lists, is the list-style property:

```
ul
{
list-style: square url("sqpurple.gif");
}
```

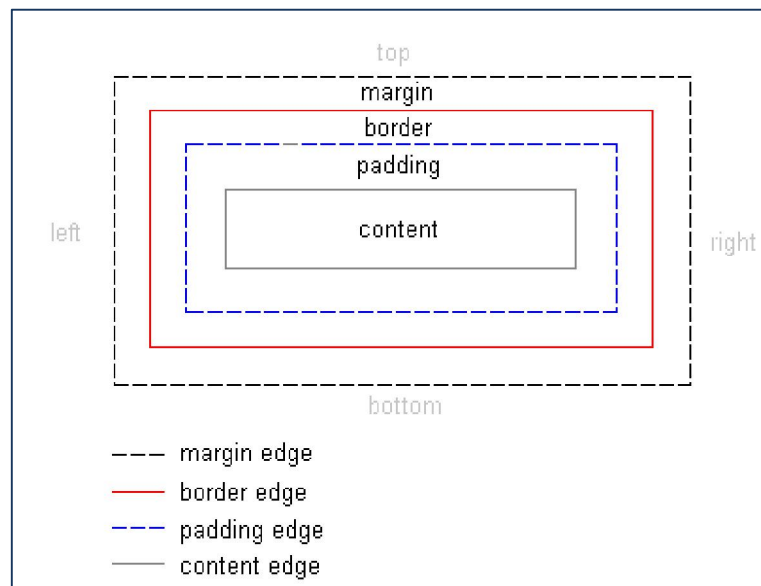
When using the shorthand property, the order of the values are:

- list-style-type
- list-style-position (for a description, see the CSS properties table below)
- list-style-image

11. CSS Box Model

All HTML elements can be considered as boxes. In CSS, the term "box model" is used when talking about design and layout. The CSS box model is essentially a box that wraps around HTML elements, and it consists of: margins, borders, padding, and the actual content. The box model allows

us to place a border around elements and space elements in relation to other elements. The image below illustrates the box model:



Explanation of the different parts:

- **Margin** - Clears an area around the border. The margin does not have a background color, and it is completely transparent
- **Border** - A border that lies around the padding and content. The border is affected by the background color of the box
- **Padding** - Clears an area around the content. The padding is affected by the background color of the box
- **Content** - The content of the box, where text and images appear

In order to set the width and height of an element correctly in all browsers, you need to know how the box model works.

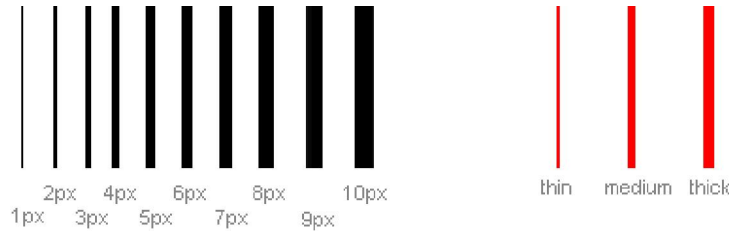
12. CSS Border

Borders can be used for many things, for example as a decorative element or to underline a separation of two things. CSS gives you endless options when using borders in your pages.

- border-width
- border-color
- border-style
- border

12.1 The width of borders [border-width]

The width of borders is defined by the property border-width, which can obtain the values thin, medium, and thick, or a numeric value, indicated in pixels. The figure below illustrates the system:



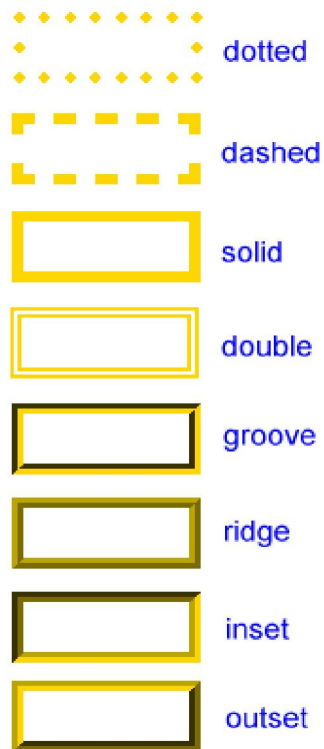
12.2 The color of borders [border-color]



The property border-color defines which color the border has. The values are the normal color-values for example "#123456", "rgb(123,123,123)" or "yellow".

12.3 Types of borders [border-style]

There are different types of borders to choose from. Below are shown 8 different types of borders as Internet Explorer 5.5 interprets them. All examples are shown with the color "gold" and the thickness "thick" but can naturally be shown in other colors and thicknesses. The values none or hidden can be used if you do not want any border.



Border - Individual sides

In CSS it is possible to specify different borders for different sides:

```
p {
  border-top-style:dotted;
  border-right-style:solid;
  border-bottom-style:dotted;
  border-left-style:solid;
}
```

Border - Shorthand property

The shorthand property for the border properties is "border":

```
border:5px solid red;
```

When using the border property, the order of the values are:

- border-width
- border-style
- border-color

It does not matter if one of the values above are missing (although, border-style is required), as long as the rest are in the specified order.

13. Height and width

Up until now, we have not cared much about the dimensions of the elements we have worked with. In this lesson, we will take a look at how you easily can define the height and width of an element.

- width
- height

13.1 Setting the width [width]

With the width-property, you can define a certain width of an element. The simple example below provides us with a box wherein text can be typed:

```
div.box {  
    width: 200px;  
    border: 1px solid black;  
    background: orange;  
}
```

13.2 Setting the height [height]

In the example above notice how the height of the box is set by the content of the box. You can affect the height of an element with the property height. As an example let us try to make the box in the example 500px high:

```
div.box {  
    height: 500px;  
    width: 200px;  
    border: 1px solid black;  
    background: orange;  
}
```


14. CSS Margin

The margin clears an area around an element (outside the border). The margin does not have a background color, and is completely transparent. The top, right, bottom, and left margin can be changed independently using separate properties. A shorthand margin property can also be used, to change all margins at once.

```
margin-top:10px;
```

Margin - Individual sides

In CSS, it is possible to specify different margins for different sides:

```
margin-top:100px;  
margin-bottom:100px;  
margin-right:50px;  
margin-left:50px;
```

The margin property can have from one to four values.

margin:25px 50px 75px 100px;

- top margin is 25px
- right margin is 50px
- bottom margin is 75px
- left margin is 100px

margin:25px 50px 75px;

- top margin is 25px
- right and left margins are 50px
- bottom margin is 75px

margin:25px 50px;

- top and bottom margins are 25px
- right and left margins are 50px

margin:25px;

- all four margins are 25px

15. CSS Padding

The padding clears an area around the content (inside the border) of an element. The padding is affected by the background color of the element. The top, right, bottom, and left padding can be changed independently using separate properties. A shorthand padding property can also be used, to change all paddings at once. Padding - Individual sides

```
padding-top:25px;  
padding-bottom:25px;  
padding-right:50px;  
padding-left:50px;
```

Padding - Shorthand property

To shorten the code, it is possible to specify all the padding properties in one property. This is called a shorthand property. The shorthand property for all the padding properties is "padding":

```
padding:25px 50px;
```

The padding property can have from one to four values.

padding:25px 50px 75px 100px;

- top padding is 25px
- right padding is 50px
- bottom padding is 75px
- left padding is 100px

padding:25px 50px 75px;

- top padding is 25px
- right and left paddings are 50px
- bottom padding is 75px

padding:25px 50px;

- top and bottom paddings are 25px
- right and left paddings are 50px

padding:25px;

- all four paddings are 25px

16. CSS Positioning

The CSS positioning properties allow you to position an element. It can also place an element behind another, and specify what should happen when an element's content is too big. Elements can be positioned using the top, bottom, left, and right properties. However, these properties will not work unless the position property is set first. They also work differently depending on the positioning method.

There are four different positioning methods:

16.1 Static Positioning

HTML elements are positioned static by default. A static positioned element is always positioned according to the normal flow of the page. Static positioned elements are not affected by the top, bottom, left, and right properties.

16.2 Fixed Positioning

An element with fixed position is positioned relative to the browser window. It will not move even if the window is scrolled:

```
p.fixed {  
    position:fixed;  
    top:50px;  
    right:15px; }  
}
```

16.3 Relative Positioning

A relative positioned element is positioned relative to its normal position

```
h2.left  
{  
    position:relative;  
    left:-20px;  
}  
h2.pos_right  
{  
    position:relative;  
    left:20px;  
}
```

The content of a relatively positioned elements can be moved and overlap other elements, but the reserved space for the element is still preserved in the normal flow. Relatively positioned element are often used as container blocks for absolutely positioned elements.

16.4 Absolute Positioning

An absolute position element is positioned relative to the first parent element that has a position other than static. If no such element is found, the containing block is <html>:

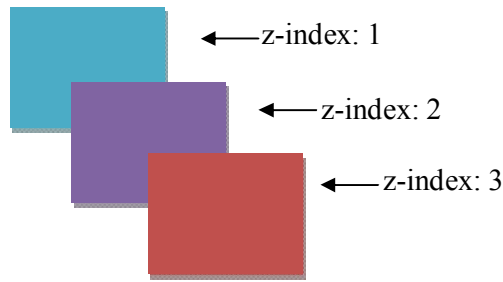
```
h2  
{  
    position:absolute;  
    left:100px;  
    top:150px;  
}
```

Absolutely positioned elements are removed from the normal flow. The document and other elements behave like the absolutely positioned element does not exist. Absolutely positioned elements can overlap other elements.

17. Layer on layer with z-index

we will learn how to let different elements become layers. In short, this means the order of which the elements overlap one another. For that purpose, you can assign each element a number (z-index). The system is that an element with a higher number overlaps an element with a lower number.

Let us say we are playing poker and have a royal flush. Our hand can be presented in a way where each card has got a z-index:



In this case, the numbers follow on another (1-3) but the same result can be obtained by using 3 different numbers. The important thing is the chronological sequence of the numbers (the order).

The code in the card example could look like this:

```
#div1 {  
    position: absolute;  
    left: 100px;  
    top: 100px;  
    z-index: 1; }  
  
#div2 {  
    position: absolute;  
    left: 115px;  
    top: 115px;  
    z-index: 2; }  
  
#div3 {  
    position: absolute;  
    left: 130px;  
    top: 130px;  
    z-index: 3; }
```

18. CSS Float

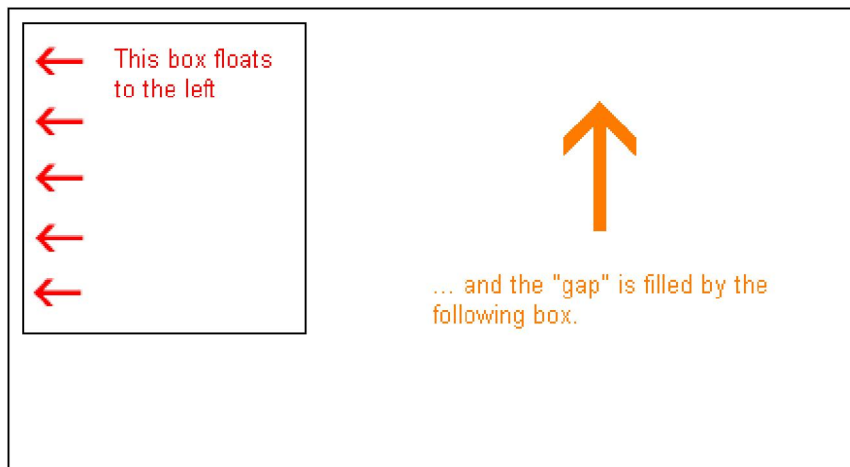
With CSS float, an element can be pushed to the left or right, allowing other elements to wrap around it. Float is very often used for images, but it is also useful when working with layouts. How Elements Float Elements are floated horizontally, this means that an element can only be floated left or right, not up or down. A floated element will move as far to the left or right as it can. Usually this means all the way to the left or right of the containing element. The elements after the floating element will flow around it. The elements before the floating element will not be affected. If an image is floated to the right, a following text flows around it, to the left:

```
img { float:right; }
```

Floating Elements Next to Each Other. If you place several floating elements after each other, they will float next to each other if there is room.

Here we have made an image gallery using the float property:

```
.thumbnail
{
float:left;
width:110px;
height:90px;
margin:5px;
}
```



Turning off Float - Using Clear

Elements after the floating element will flow around it. To avoid this, use the clear property. The clear property specifies which sides of an element other floating elements are not allowed. Add a text line into the image gallery, using the clear property:

```
.cleartext
{
clear:both;
}
```

19. CSS Image Opacity / Transparency

Creating transparent images with CSS is easy.

Note: This is not yet a CSS standard. However, it works in all modern browsers, and is a part of the W3C CSS 3 recommendation.

```

```

Firefox uses the property **opacity:x** for transparency, while IE uses **filter:alpha(opacity=x)**.

Tip: The CSS3 syntax for transparency is **opacity:x**.

In Firefox (opacity:x) x can be a value from 0.0 - 1.0. A lower value makes the element more transparent. In IE (filter:alpha(opacity=x)) x can be a value from 0 - 100. A lower value makes the element more transparent.

20. CSS Media Types

Media Types allow you to specify how documents will be presented in different media. The document can be displayed differently on the screen, on the paper, with an aural browser, etc.

Some CSS properties are only designed for a certain media. For example the "voice-family" property is designed for aural user agents. Some other properties can be used for different media types. For example, the "font-size" property can be used for both screen and print media, but perhaps with different values. A document usually needs a larger font-size on a screen than on paper, and sans-serif fonts are easier to read on the screen, while serif fonts are easier to read on paper.

The @media Rule

The @media rule allows different style rules for different media in the same style sheet. The style in the example below tells the browser to display a 14 pixels Verdana font on the screen. But if the page is printed, it will be in a 10 pixels Times font. Notice that the font-weight is set to bold, both on screen and on paper:

```
<html>
<head>
<style>
@media screen
{
p.test {font-family:verdana,sans-serif;font-size:14px;}
}
@media print
{
p.test {font-family:times,serif;font-size:10px;}
}
@media screen,print
{
p.test {font-weight:bold;}
}
</style>
</head>
<body>
....
</body>
</html>
```

📖 Web-standards and validation

W3C is the [World Wide Web Consortium](http://www.w3.org/), which is an independent organization that manages code standards on the web (e.g. HTML, CSS, XML and others). Microsoft, The Mozilla Foundation and many others are a part of W3C and agree upon the future developments of the standards. If you have been working just a bit with web design, you probably know that there can be a big differences in how a webpage is presented across different browsers. It can be very frustrating and time-consuming to create a webpage which can be viewed in Mozilla, Internet Explorer, Opera and all the rest of the existing browsers. The idea of having standards is to agree upon a common denominator on how to use web technologies. This means that by observing the standards, a webdeveloper has a certainty that what he or she does will work in a more appropriate manner across different platforms. **We therefore recommend that you back up the work carried out by the W3C and validate your CSS in order to observe the standard.**

The validator can also be found at this link: <http://jigsaw.w3.org/css-validator/>

CSS Sample



HTML File(index.html)

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Untitled Document</title>
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
  <div id="wrapper">
    <div id="header">
      <div class="sitelogo">
        <h1>Logo</h1>
      </div>
      <div class="bannertop">
        <h1>Banner Top Advertising</h1>
      </div>
    </div>
    <div id="menu">
      <ul>
        <li><a href="#home">Home</a></li>
        <li id="sample_attach_menu_parent" class="sample_attach"><a
href="#news">News</a></li>
        <li><a href="#contact">Contact</a></li>
        <li><a href="#about">About</a></li>
      </ul>
    </div>
    <div id="search">
      Search
    </div>
    <div id="breadcrumb">
      site map
    </div>
    <div id="ads_top">
      Advertising Company
    </div>
    <div id="googleadstop">
      Google Adsense
    </div>
    <div id="content">
      <div id="left">
        <ul>
          <li><a href="#home">Home</a></li>
          <li><a href="#news">News</a></li>
          <li><a href="#contact">Contact</a></li>
          <li><a href="#about">About</a></li>
        </ul>
        <div class="leftads">
          <h2>Advertising</h2>
        </div>
      </div>
    </div>

```



```

    <div id="center">
      <div class="centerads">
        Advertising Center
      </div>
      Content...
    </div>
    <div id="right">
      <div class="login">
        <h2>LogIn</h2>
      </div>
      <div class="rightads">
        <h2>Our Sponser</h2>
        
      </div>
    </div>
  </div>
  <div id="facebook">Facebook Comment</div>
  <div id="footer">Footer</div>
</div>
</body>
</html>

```

CSS File(style.css)

```

*{
  margin:0;
  padding:0;}
body
{
  font-family:Verdana, Geneva, sans-serif;
  font-size:100%;
  background-color:#FFF;
  line-height:24px;
  word-spacing:1px;
  color:black;
}
h1{
  font-size:2.5em;
}
h2{
  font-size:1.875em;
}
p{
  font-size:0.875em;
}
#wrapper{
  width:990px;
  height:auto;
  background-color:#9CF;
  margin:0 auto;
}

```

```

#header{
    width:990px;
    height:120px;
    background-color:#36F;
    color:yellow;
}
#header .sitelogo{
    float:left;
    width:300px;
    height:90px;
    padding-top:30px;
    text-align:center;
}
#header .bannertop{
    float:left;
    width:690px;
    height:90px;
    padding-top:30px;
    background-color:#C93;
    text-align:center;
}
#menu{
    clear:left;
    width:990px;
    height:40px;
    background-color:#98bf21;
    color:red;
}
#menu ul{
list-style-type:none;
margin:0;
padding:0;
}
#menu ul li{
display:inline;
}
#menu a:link,#menu a:visited{
    float:left;
    line-height:40px;
    /*width:150px;*/
    padding:0 25px;
    border-right:1px inset #39C;
    font-weight:bold;
    color:#FFFFFF;
    background-color:#98bf21;
    text-align:center;
    text-decoration:none;
    text-transform:uppercase;
    -moz-border-radius:15px 15px 0
0px;
    border-radius:15px 15px 0 0px;
}

```

```

#menu a:hover,#menu a:active{
background-color:#7A991A;}
#search{
    width:990px;
    height:60px;
    background-color:#FC9;}
#breadcrumb{
    width:980px;
    height:auto;
    margin:5px 0px;
    padding:0px 5px;
    text-align:left;
    font-size:14px;
    background-color:#CCC;}
#ads_top{
    width:980px;
    height:auto;
    text-align:center;
    border:0px;
    outline:none;
    margin:5px;
    background-color:#fff;}
#googleadstop{
    width:980px;
    height:80px;
    text-align:center;
    border:0px;
    outline:none;
    margin:5px;
    background-color:#fff;}
#content{
    width:990px;
    height:auto;
    background-color:#CCC;
    color:#00F;
    overflow:auto;
    margin-bottom:10px;}
#content #left{
    float:left;
    width:200px;
    height:auto;
    background-color:#066;
    color:#00F;}
#content #left ul{
list-style-type:none;
margin:0;
padding:0;
}

```

```

#content #left a:link,#content #left a:visited{
display:block;
font-weight:bold;
color:#FFFFFF;
background-color:#06F;
width:192px;
text-align:center;
padding:4px;
line-height:30px;
text-decoration:none;
text-transform:uppercase;
border-bottom:1px inset yellow;
}
#content #left a:hover,#content #left a:active{
background-color:#C96;
}
#content #left .leftads{
width:190px;
height:200px;
border:1px solid blue;
margin:4px;
text-align:center;
background-color:#FCF;
margin-top:5px;
}
#content #left .leftads h2{
background-color:#66C;
font-size:20px;
height:35px;
padding-top:5px;
}
#content #left .leftads h2:hover{
background-color:#993;
}
#content #center{
float:left;
width:580px;
height:auto;
color:#00F;
margin-left:5px;
margin-right:5px;
}
#content #center .centerads{
width:580px;
height:200px;
background-color:white;
}

```

```

#content #right{
float:left;
width:200px;
height:auto;
background-color:#9F3;
color:#00F;
}
#content #right .login{
width:190px;
height:200px;
border:1px solid blue;
margin:4px;
text-align:center;
background-color:#FCF;
margin-top:5px;
}
#content #right .login h2{
background-color:#66C;
font-size:20px;
height:35px;
padding-top:5px;
}
#content #right .login h2:hover{
background-color:#993;
}
#content #right .rightads{
width:190px;
height:auto;
border:1px solid blue;
margin:4px;
text-align:center;
background-color:#FCF;
margin-top:5px;
}
#content #right .rightads h2{
background-color:#66C;
font-size:20px;
height:35px;
padding-top:5px;
}
#content #right .rightads h2:hover{
background-color:#993;
}
#content #right .rightads img{
width:184px;
padding:3px;
}
#content #right .rightads img:hover{
opacity:0.4;
filter:alpha(opacity=40);
}

```

```
#facebook{
  clear:left;
  width:990px;
  height:100px;
  background-color:#FC9;
  color:#000;
  text-align:center;
  margin-top:10px;
}
#footer{
  margin-top:10px;
  width:990px;
  height:70px;
  background-color:#CF6;
  color:#000;
}
```