

Universidad de Chile
Facultad de Ciencias Físicas y Matemáticas

Guía Básica de HTML

Índice

| | |
|--|----|
| 1 Comandos Básicos en Linux..... | 3 |
| 2 Servidor para Tareas | 4 |
| 3 Creando una Página WEB | 4 |
| 4 Usando SFTP | 5 |
| 4.1 Conectarse al Servidor | 5 |
| 4.2 Descripción del Programa..... | 6 |
| 4.3 Cambiar permisos de Directorios y Archivos | 6 |
| 5 HTML | 7 |
| 5.1 Cuerpo de una Página | 7 |
| 5.2 Texto | 8 |
| 5.3 Links | 8 |
| 5.4 Tablas..... | 9 |
| 5.5 Listas | 10 |
| 5.6 Imágenes | 10 |
| 5.7 Frames..... | 10 |
| 5.8 Formularios | 12 |
| 6 Hojas de Estilo (CSS) | 13 |

1 Comandos Básicos en Linux

cd <directorio>

Cambia de directorio. Si usan cd sin parámetros vuelven a la raíz de su cuenta. Para devolverse de una carpeta se usa “cd ..” .

ls <directorio>

Lista el contenido de un directorio. Opciones:

- l: información extra (como los permisos de los archivos).
- a: muestra además los archivos ocultos (que parten con punto).

mkdir <directorio>

Crea un directorio.

rmdir <directorio>

Borra un directorio.

rm <archivo>

Borra un archivo.

cat <archivo>

Muestra el contenido de un archivo en forma de texto.

chmod <usuarios><modo><permisos> <archivo>

Cambia los permisos de un archivo o directorio.

Los usuarios pueden ser:

- | | | |
|---|---|--|
| u | → | user (el dueño del archivo) |
| g | → | group (el grupo al que pertenece el usuario) |
| o | → | el resto |
| a | → | todos los anteriores |

Los modos pueden ser:

- | | | |
|---|---|-----------------|
| + | → | Dar permisos |
| - | → | Quitar permisos |

Los permisos pueden ser:

- | | | |
|---|---|---------|
| r | → | Read |
| w | → | Write |
| x | → | Execute |

Entonces, para darle permisos a un archivo sería:

chmod a+rw hola.txt ← todos pueden leer y modificar el archivo hola.txt

2 Servidor para Tareas

Para hacer las tareas de este curso se les activará una cuenta en un servidor del ADI. Los datos de conexión a este servidor son:

IP: 200.9.100.61

PUERTO SSH: 22

URL: `http://200.9.100.61/~<username>/`

Una vez que tengan su username y contraseña deben conectarse vía ssh (usando un cliente como putty o ssh tectia client) y ejecutar lo siguiente:

```
> cd
> chmod o+x .
> mkdir public_html
> chmod o+x public_html
> cd public_html
> echo "Hola Mundo" >index.html
```

y listo! Ahora para ver tú página recién creada tienes que abrir un navegador y cargar la siguiente URL:

`http://200.9.100.61/~<username>/index.html`

3 Creando una Página WEB

Para crear páginas existen muchos programas gratis o shareware disponibles en Internet del estilo WYSIWYG (What You See Is What You Get), donde es posible abstraerse del lenguaje HTML y simplemente dibujar el contenido que se quiere desplegar.

Ejemplos de editores de páginas web son:

- Netscape Composer (viene con el navegador Netscape)
- NVU (<http://nave.escomposlinux.org/productos/nvu/>)
- Macromedia Dreamweaver (no es gratis)
- Mozilla Suite (<http://www.mozilla.org>)

4 Usando SFTP

Una vez que hayas creado tus páginas en tu computador, estas deberán ser subidas a la carpeta `public_html` que creamos en el punto 2, para que así sean visibles desde Internet por el resto del mundo.

Un programa muy conocido para subir archivos a sus cuentas es el SSH Tectia File Transfer, el cual debería estar instalado en todos los computadores de los laboratorios. Sino pueden descargarlo desde la siguiente URL:

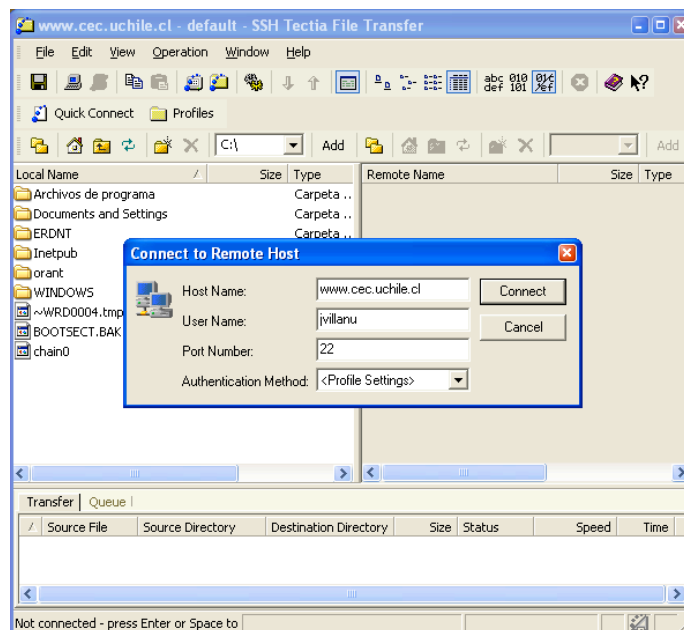
<ftp://ftp.cec.uchile.cl/pub/windows/SFTP/>

4.1 Conectarse al Servidor

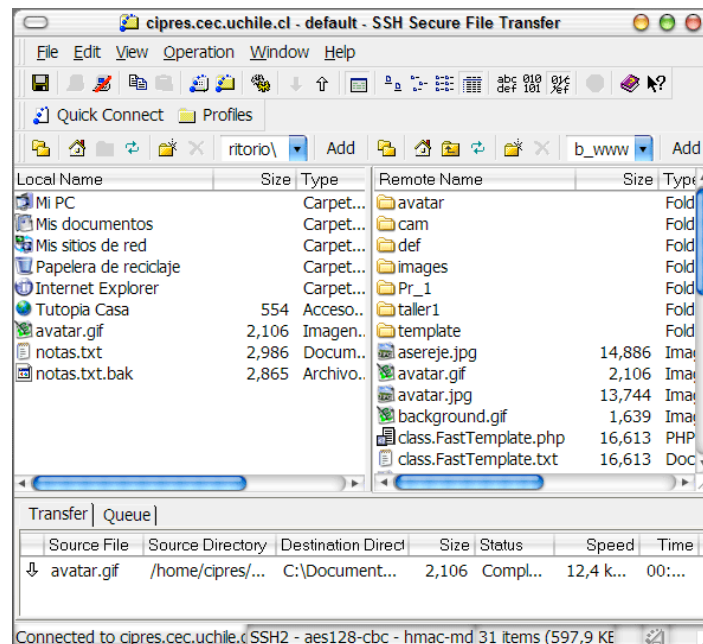
Host Name: 200.9.100.61

User Name: <tu nombre de usuario>

Port Number: 22



4.2 Descripción del Programa

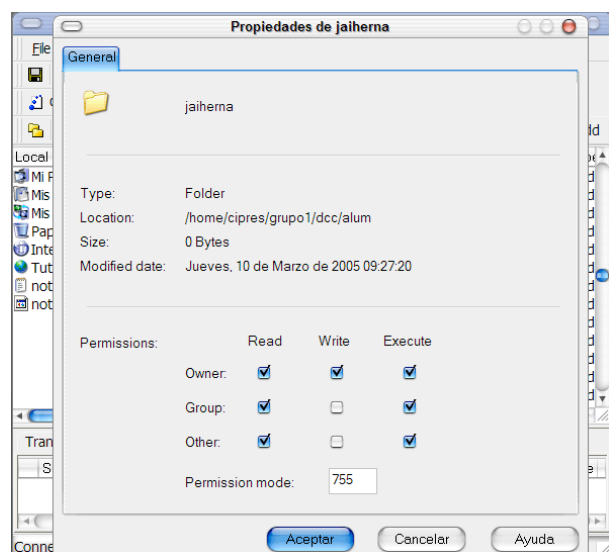
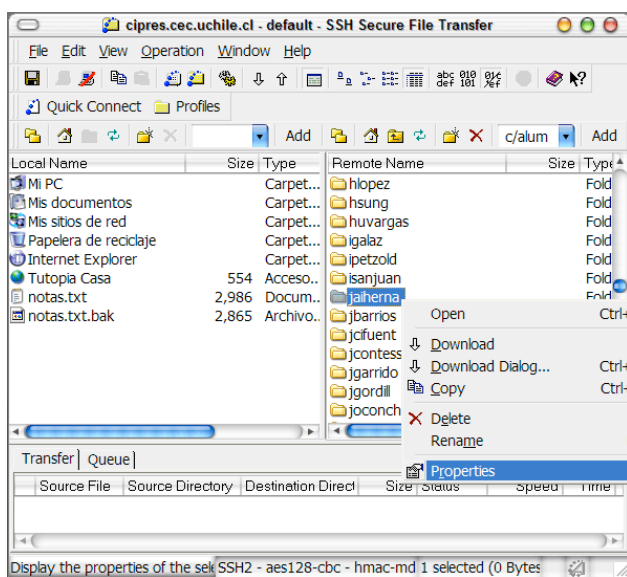


En la ventana izquierda están los archivos del PC local.

En la ventana derecha están los archivos de tu cuenta.

Para transferir archivos desde un lado hacia el otro basta con arrastrarlos.

4.3 Cambiar permisos de Directorios y Archivos



5 HTML

HTML es un lenguaje mediante el cual se le especifica a los navegadores “como” deben mostrar el contenido. Se basa en el uso de “tags” los cuales se componen de un nombre, atributos y contenido:

```
<tag atributo1="valor1" atributo2="valor2">
contenido
...
</tag>
```

Los tags, una vez abiertos, deben ser cerrados usando / antes de su nombre .

Existen tags que por definición nunca deben tener contenido. Estos deben escribirse de la siguiente forma:

```
<tag />
```

Los tags pueden contener otros tags, es decir, pueden anidarse. Ej:

```
<p><b>Esto es un texto en negritas</b></p>
```

Existen muchas etiquetas, las cuales sirven para definir, entre muchas otras cosas, elementos, formatos, parámetros, etc.

5.1 Cuerpo de una Página

Todo documento HTML debe comenzar con el tag <HTML> y debe terminar cerrando dicho tag:

```
<html>
<head>
  <title> Titulo de la Pagina </title>
  <meta name="keywords" content="Ejemplo, Pagina WEB" />
</head>
<body>
...
</body>
</html>
```

El tag <head> corresponde al encabezado de nuestra página, donde podremos definir el título de ésta (que se ve en la parte superior del navegador), además de otros tags que sirven para que los buscadores sepan que va a contener nuestro sitio.

Dentro del tag <body> deberá ir todo el “contenido visible” de nuestra página web.

5.2 Texto

Para escribir texto en HTML basta que éste se encuentre dentro del tag <body>:

```
<body>
Hola como estas
</body>
```

Para escribir texto en párrafos se debe encerrar en el tag <p>. Ej:

```
<p>Esto es el parrafo 1.</p>
<p>Esto es el parrafo 2.</p>
```

Para agregar los títulos y subtítulos se utilizan los tags de encabezados <h1>, <h2>, <h3>, etc. Ej:

```
<h1>Curso de HTML</h1>
<h2>Introducción</h2>
<p>Esto es una introducción</p>
<h2>Apendice</h2>
<p>Esto es un apendice</p>
```

Para darle formato al texto existen los siguientes tags:

Ahora, para darle un formato en especial se pueden usar otros tags:

| | |
|----------|--|
| | ← el texto aparece en negritas |
| <ins> | ← el texto aparece <u>subrayado</u> |
| | ← el texto aparece tachado |
| | ← el texto aparece en <i>itálico</i> |
| <sup> | ← el texto aparece como ^{superíndice} |
| <sub> | ← el texto aparece como _{subíndice} |

5.3 Links

Una de las grandes cosas que destaca a Internet es la capacidad de apuntar a otros sitios y contenidos. Para esto se usa el tag <a>. Ej:

```
<a href="http://www.google.cl">Link a Google</a>
```

Para que se abra en una ventana nueva, podemos agregar el atributo target="_blank".

Además de apuntar a otras páginas, también podemos apuntar a zonas dentro de nuestra página. Para esto se usa una combinación de los tags <a>. Ej:


```
<a name="pregunta7">¿Que hace el tag dl?</a>
```

y para cuando hagan click sobre nuestro link se vaya a la pregunta 7 agregamos lo siguiente:

```
<a href="#pregunta7">ir a la pregunta 7</a>
```

5.4 Tablas

Para crear una tabla se ocupan los siguientes tags:

| | |
|----------------------------|-------------------------------------|
| <code><table></code> | ← da el inicio a una tabla |
| <code><tr></code> | ← indica que comienza una fila |
| <code><td></code> | ← indica que comienza una columna |
| <code><th></code> | ← indica que comienza un encabezado |

Ej:

```
<table>
<tr>
  <th>Nombre</th>
  <th>Apellido</th>
</tr>
<tr>
  <td>Javier</td>
  <td>Villanueva</td>
</tr>
<tr>
  <td>Paula</td>
  <td>Silva</td>
</tr>
</table>
```

Si se quiere estirar una celda se ocupan los atributos:

| | |
|---------------------------------------|-------------------------|
| <code>colspan="n° de columnas"</code> | ← se estira n° columnas |
| <code>rowspan="n° de filas"</code> | ← se estira n° filas |

5.5 Listas

Para listar elementos existen 2 tags:

- `` ← Unordered List
- `` ← Ordered List (van con números)

Los elementos que serán listados deberán ir encerrador en el tag `` (List Item). Ej:

```
<ol>
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3</li>
  <ul>
    <li>Subitem A</li>
    <li>SubItem B</li>
  </ul>
</ol>
```

5.6 Imágenes

Para agregar imágenes a una página basta con agregar el tag ``:

```

```

El atributo alt sirve para aquellos navegadores que no son capaces de mostrar imagines, y además para navegadores para personas con discapacidad visual.

5.7 Frames

Para dividir el navegador en varias páginas, es usual usar los llamados FRAMES.

```
<html>
<head></head>
<frameset>
  <frame/>
</frameset>
</html>
```

- `<frameset>` ← da inicio a una serie de frames, los cuales se despliegan en forma de filas o columnas dependiendo del atributo que contenga (row o col)
- `<frame>` ← a través del atributo src dice que página se mostrará en ese frame.

Ej:

```
<frameset rows="30%, *">  
  <frame src="pagina1.html" />  
  <frame src="pagina2.html" />  
</frameset>
```

o también:

```
<frameset cols="*, 100, 500">  
  <frame src="pagina1.html" />  
  <frame src="pagina2.html" />  
  <frame src="pagina3.html" />  
</frameset>
```

El tag frame puede llevar los siguientes atributos especiales:

| | |
|-----------------------------|--|
| marginheight="valor" | ← altura del frame |
| marginwidth="valor" | ← ancho del frame. |
| name="nombre-ventana" | ← nombre del frame. |
| noresize="true" | ← impide cambiar el tamaño del frame. |
| scrolling="yes no auto" | ← indica si se muestra la barra de desplazamiento. |

La idea de usar name es que un link de un frame puede cargarse en otro frame si al tag de link (<a>) se le agrega el atributo TARGET. Ej:

```
<a ref="www.google.cl" target="nombre">Google</a>
```

NOTA: Un frameset puede contener otros frameset en su interior.

5.8 Formularios

Los formularios sirven para que el usuario ingrese información hacia la página que esta viendo. Para crear un formulario se debe utilizar el tag `<form>`:

`<form action="pagina_siguiente.html">` ← action corresponde a la página donde llegarán los datos

Dentro de este tag pueden ir una serie de elementos propios de un formulario:

| | |
|---|--|
| <code><input type="text" /></code> | ← campo de texto |
| <code><input type="image" src="" /></code> | ← botón de estilo imagen |
| <code><input type="button" value="entrar" /></code> | ← botón normal |
| <code><input type="submit" value="entrar" /></code> | ← botón que además hace que la información viaje a la siguiente página |
| <code><input type="checkbox" /></code> | ← opción en forma de ticket |
| <code><input type="radio" /></code> | ← opción en forma de círculo |
| <code><input type="reset" /></code> | ← borra la información de formulario. |
| <code><input type="password" /></code> | ← para ingresar passwords |
| <code><textarea rows="4" cols="40">...</textarea></code> | ← cuadrado para ingresar texto. |
| <code><select></code> <code><option>opcion 1</option></code> <code><option>opcion 2</option></code> <code></select></code> | ← lista de opciones (si se define el atributo multiple, se puede elegir más de 1 a la vez) |

Ej:

```
<form method="post" action="fin.html">
<table>
<tr>
  <td>Nombre:</td>
  <td><input type="text" name="nombre" value="nombre..." /></td>
</tr>
<tr>
  <td>Password:</td>
  <td><input type="password" name="clave" maxlength="10" /></td>
</tr>
<tr>
  <td>Comentario</td>
  <td><texarea name="comentario">un comentario</textarea></td>
</tr>
<tr>
  <td colspan="2">
    <input type="submit" value="Entrar" />
    <input type="reset" value="Limpiar" />
  </td>
</tr>
</table>
</form>
```

6 Hojas de Estilo (CSS)

CSS es un lenguaje para definir el “como” deben mostrarse los contenidos de nuestras páginas web. Para darle forma a algun elemento de nuestra página se utiliza el atributo “style”, cuyo contenido le da el formato al tag que lo contenga. Ej:

<p style="color: red; font-weight: bold; font-size: 16px">Esto es un parrafo</p>

Algunos ejemplos de lo que puede ir dentro del atributo style son:

color: #FF0000
font-weight: normal | bold | ...
text-decoration: none | underline | strike | ...
font-size: 20px | 10pt
text-align: left | center | right | justify
border: 1px solid red
border-(left | right | top | bottom): 2px dashed blue
margin: 10px 5px 20px 50px (top right bottom left)
padding: 5px 10px (es equivalente a 5px 10px 5px 10px)
background: #00FF00 url('imagen.jpg') scroll repeat-x top left

Las cosas que pueden ir dentro del style dependerán del tag que se esté utilizando.
Para ver la referencia completa de CSS visiten:

http://www.w3schools.com/CSS/CSS_reference.asp

La forma limpia de separar el “como” se ve nuestra página es usando una hoja de estilo, la cual corresponde a un archivo de extensión .css el cual debe ser incluido en el encabezado de nuestra página web:

```
<head>  
  <link rel="stylesheet" media="all" href="estilos.css" />  
</head>
```

La gracia de esta separación es que simplemente cambiando este archivo podemos hacer que nuestra página se vea completamente diferente.

Dentro del archivo estilos.css podemos definir “clases” que tendran cierta forma de mostrarse. Aquellos elementos que sean de esa clase se verán tal como lo describe la hoja de estilos.
Ej:

<p class="claseX">Esto es un parrafo</p> ← se referencia como p.claseX
<p id="parrafo1">Esto es un parrafo con id</p> ← se referencia como p#parrafo1

A continuación veamos un ejemplo más completo:

```
<html>
<head>
    <link rel="stylesheet" media="all" href="estilos.css" />
</head>
<body>
<h1>Esto es un titulo</h1>

<h2 class=" subrayado">Esto es un subtítulo</h2>
<p>Esto es un párrafo</p>

<h2 class="subtitulo_rojo">Esto es otrosubtítulo</h2>
<p class="importante">
Esto es otro párrafo con un
<a href="http://www.google.cl">link a google</a>.
</p>

</body>
</html>
```

y nuestro archivo estilos.css contendrá:

```
body {
    background-color: gray;
    background-image: url('http://www.google.cl/imagen.png' );
}

h1{
    color: red;
    font-size: 20px;
    text-align: center;
    border-bottom: 5px dashed red;
}

h2{
    color: green;
    text-align: left;
    padding-left: 50px;
}

.subrayado{                                ← cualquier tag que tenga class="subrayado"
    text-decoration: underline;
}

p{                                           ← para todos los <p>
    font-size: 12px;
    font-family: Verdana;
}
p.importante{
    font-weight: bold;
}
p.importante a{                            ← los <a> dentro los <p class="importante">
    color: yellow;
}
p.importante a:hover{                      ← cuando el mouse pasa por arriba del link
    color: red;
}
```