



REALIZZARE UNA TAG BOARD IN PHP

Di seguito verrà spiegato come realizzare una tag sia con supporto di un database MySQL (Anteprima, ) e sia con l'uso di file di testo .txt (senza database quindi; Anteprima, 

Creare una Tag con supporto MySQL

Vediamo innanzitutto il database necessario per l'uso della Tag; chiamiamo il database "tag" e sarà costituito da una sola tabella "dati":

```
CREATE TABLE dati (  
  id int(11) NOT NULL auto_increment,  
  nome varchar(100) NOT NULL default "",  
  email varchar(150) NOT NULL default "",  
  message longtext NOT NULL default "",  
  data varchar(100) NOT NULL default "",  
  www varchar(150) NOT NULL default "",  
  PRIMARY KEY (id)  
)
```

La Tag verrà realizzata su quattro files: smilies.php , tag.php , fmysql.php , tagboard.js . Il primo file contiene semplicemente l'elenco dei percorsi delle smilies, mentre il file javascript serve semplicemente a costruire la pop-up per postare le smilies. Gli altri due files php servono per il funzionamento della tag stessa.

Il file js lo trovate nel file zip disponibile per il download, mentre di seguito è riportato il contenuto del file *smilies.php*, contenente un'array \$smile_array, le cui chiavi sono le stringhe corrispondenti ai codici delle smilies, mentre i valori sono i tag IMG corrispondenti.

```
<?php  
$smile_array[:blink'] = "<IMG SRC=smilies/blink.gif width=20 height=20 border=0>";  
$smile_array[:byesigh'] = "<IMG SRC=smilies/bye2.gif width=26 height=18 border=0>";  
$smile_array[:clap'] = "<IMG SRC=smilies/clap.gif width=31 height=25 border=0>";  
$smile_array[:cool'] = "<IMG SRC=smilies/cool.gif width=20 height=20 border=0>";  
$smile_array[:crybaby'] = "<IMG SRC=smilies/crybaby.gif width=40 height=18 border=0>";  
$smile_array[:dry'] = "<IMG SRC=smilies/dry.gif width=20 height=20 border=0>";  
$smile_array[:blushing'] = "<IMG SRC=smilies/blushing.gif width=20 height=20 border=0>";  
$smile_array[:gratta?] = "<IMG SRC=smilies/g.gif width=25 height=23 border=0>";  
$smile_array[:hands'] = "<IMG SRC=smilies/hands.gif width=46 height=22 border=0>";  
$smile_array[:Happy'] = "<IMG SRC=smilies/happy.gif width=20 height=20 border=0>";  
$smile_array[:Huh2'] = "<IMG SRC=smilies/huh2.gif width=20 height=20 border=0>";  
$smile_array[:icon3'] = "<IMG SRC=smilies/icon3.gif width=19 height=19 border=0>";  
$smile_array[:icon9'] = "<IMG SRC=smilies/icon9.gif width=19 height=19 border=0>";  
$smile_array[:laughing1'] = "<IMG SRC=smilies/laughing1.gif width=35 height=30 border=0>";  
$smile_array[:kicking'] = "<IMG SRC=smilies/kicking.gif width=28 height=23 border=0>";  
$smile_array[:Mad'] = "<IMG SRC=smilies/mad.gif width=20 height=20 border=0>";  
$smile_array[:No'] = "<IMG SRC=smilies/no.gif width=20 height=20 border=0>";  
$smile_array[:ohmy'] = "<IMG SRC=smilies/ohmy.gif width=20 height=20 border=0>";  
$smile_array[:Sad'] = "<IMG SRC=smilies/sad.gif width=20 height=20 border=0>";  
$smile_array[:scratch'] = "<IMG SRC=smilies/scratch.gif width=30 height=26 border=0>";  
$smile_array[:shiny'] = "<IMG SRC=smilies/shiny.gif width=18 height=18 border=0>";  
$smile_array[::)] = "<IMG SRC=smilies/icon6.gif width=19 height=19 border=0>";  
$smile_array[::)] = "<IMG SRC=smilies/smile.gif width=20 height=20 border=0>";  
$smile_array[:P'] = "<IMG SRC=smilies/tongue.gif width=20 height=20 border=0>";  
$smile_array[:unsure'] = "<IMG SRC=smilies/unsure.gif width=20 height=20 border=0>";  
$smile_array[:wacko'] = "<IMG SRC=smilies/wacko.gif width=20 height=20 border=0>";  
$smile_array[:weeping'] = "<IMG SRC=smilies/weeping.gif width=18 height=18 border=0>";  
$smile_array[:yawn'] = "<IMG SRC=smilies/yawn.gif width=18 height=18 border=0>";  
$smile_array[:Yeah2'] = "<IMG SRC=smilies/yeah2.gif width=28 height=28 border=0>";  
$smile_array[:bye'] = "<IMG SRC=smilies/bye1.gif width=26 height=18 border=0>";  
?>
```

Di seguito è invece riportato il codice del file tag.php, file principale dell'applicazione:

```

<?php
define("TAG_DIR", realpath("."));

// sezione PHP in cui viene eseguito il controllo dei campi inseriti, il controllo di mess già presente nel db e l'inserimento

if ($_POST["nome"]!="NOME" && $_POST["message"]!="" && $_POST["nome"]!="nome")
{
    include_once(TAG_DIR."/fmysql.php");
    include_once(TAG_DIR."/smilies.php");
    $conn = connessione("tag");
    foreach($smile_array as $smile_face=>$smile_image)
    {
        $_POST["message"] = str_replace($smile_face, $smile_image, $_POST["message"]);
    }
    if(!esiste($_POST,"dati",$conn))
    {
        $today=date("j-m-Y H:i");
        $query = mysql_insert($_POST,$today,"dati",$conn);
    }
    closeConnect($conn);
}
?>

<html>
<head>
// Codice JS per l'indicazione dei caratteri rimanenti
<LINK HREF="stile.css" type="text/css" rel="stylesheet" >
<SCRIPT language=JavaScript>
<!--

function initcharsleft()
{
    charsleft(document.forms["tagform"].text);
}

function charsleft(feld)
{
    var anz = feld.value.length;
    if (anz> 150 )
    {
        feld.value = feld.value.substr(0,150);
        frei = 0;
    }
    else
    {
        frei = 150 -anz;
    }
    document.forms["tagform"].num.value = frei;
}
//-->
</SCRIPT>
</head>
<body>
// sezione HTML e PHP per la rappresentazione della Tag e messaggi
<TABLE border=0 cellpadding=0 cellspacing=0 width="20%" class=tableborder height="70%">
<TR>
<TD class=row4>
<TABLE border=0 cellpadding=0 cellspacing=4 width="100%">
<TR>
<TD>
<TABLE border=0 cellpadding=0 cellspacing=0 width="100%" class=tableborder>
<TR>
<TD valign=top class=row2><div class=div_tag>
<?php
include_once(TAG_DIR."/fmysql.php");
$conn = connessione("tag");
$query = mysql_list_all("dati",$conn);
if($query!=null)
{
    lista($query);
}
closeConnect($conn);
?>
</div></TD>
</TR>
</TABLE>
</TD>
</TR>
<TR height=10>
<TD> </TD>

```

```

</TR>
<TR>
  <TD valign=top><FORM METHOD=POST name=tagform ACTION="tag.php">
    <SCRIPT language="Javascript" src="tagboard.js" type="text/javascript"></SCRIPT>
    <CENTER>
      <TABLE border=0 cellpadding=0 cellspacing=1 width="100%">
        <TR height=25>
          <TD class=row3 colspan=2 align=middle><INPUT TYPE="text" NAME="nome" MAXLENGTH="20" size=20
value="NOME"></TD>
        </TR>
        <TR height=25>
          <TD class=row3 colspan=2 align=middle><INPUT TYPE="text" NAME="www" MAXLENGTH="150" size=20
value="http:// " ></TD>
        </TR>
        <TR height=25>
          <TD class=row3 colspan=2 align=middle><INPUT TYPE="text" NAME="email" MAXLENGTH="30" size=20
value="EMAIL"></TD>
        </TR>
        <TR>
          <TD class=row3 valign=middle align=middle><input size=3 value=150 name="num"><BR><A onclick="return
pop_up_smilies();" href="smilies.htm" target=_blank class=genmed> <l>smilies</l></A></TD>
          <TD class=row3 valign=middle align=middle><TEXTAREA name="message" ROWS="3" COLS="10"
maxlength="150" onKeyPress=charsleft(this); onKeyDown=charsleft(this); onBlur=charsleft(this); onKeyUp=charsleft(this);
onFocus=charsleft(this); wrap=VIRTUAL onChange=charsleft(this);></TEXTAREA></TD>
        </TR>
        <TR height=30>
          <TD colspan=2 class=row3 valign=middle><CENTER><INPUT TYPE="submit" value=" Invia " name="ok"
class=mainoption></CENTER></TD>
        </TR>
      </TABLE></CENTER></FORM></TD>
    </TR>
  </TABLE>
</TD>
</TR>
</TABLE><BR>
</body>
</html>

```

Il file tag.php è costituito da una prima porzione di codice php che viene eseguita solo nel caso in cui \$_POST["message"] e \$_POST["nome"] siano diversi dai valori di default; questo garantisce un controllo sui dati immessi nei due campi NOME e MESSAGE. In questa sezione, viene stabilita la connessione con il database, vengono sostituiti i codici abbreviati delle smiles con i tag html IMG corrispondenti, che sono stati inseriti nel file smilies.php ; per far ciò si usa la funzione di stringhe str_replace. Viene poi eseguito un controllo su db, per verificare che il messaggio non sia stato già inserito, tramite la funzione esiste, e nel caso in cui essa ritorna false, viene memorizzato il messaggio su db. Quest'ultimo controllo impedisce che il messaggio venga reinserito nel database quando si effettua un refresh di pagina subito dopo un primo inserimento.

La seconda parte del codice è puramente costituita dall'html necessario per creare la tag e un codice js necessario per far funzionare il contatore di caratteri rimanenti nella textarea del messaggio.

Successivamente seguono delle righe di codice in php necessarie a ricostruire il layer DIV contenente i vari messaggi, attraverso l'uso della funzione lista, anche essa definita in fmysql.php .

Segue poi tutto il rimanente codice html necessario a costruire il form centrale della tag. Notate bene la posizione in cui è stato inserito il tagboard.js necessario a far comparire la popup delle smiles.

Di seguito è invece riportato il codice del file fmysql.php :

```

<?php
function closeConnect($conn) // Chiude la connessione al db
{
mysql_close($conn);
}

function connessione($database) // Apre la connessione al db
{
$conn=mysql_connect("localhost","root","")
or die("Connessione non riuscita: ".mysql_error());
mysql_select_db($database) or die("Selezione del database non riuscita");
return $conn;
}

function mysql_list_all($table,$con) // Seleziona tutti i mess presenti
{

```

```

$result=mysql_query("SELECT * FROM $table order by id desc;",$con);
while($row=mysql_fetch_array($result,MYSQL_ASSOC))
{
$ret[]=$row;
}
return $ret;
}

function lista($value) // Printa la tabella con tutti i mess
{
print("<TABLE border=0 cellpadding=0 cellspacing=4 width=100%>");
foreach($value as $key=>$val)
{
print("<TR height=10><TD align=right valign=middle class=row4><SPAN class=genmed><IMG SRC='icon.gif' WIDTH='12'
HEIGHT='9' BORDER=0 ALT='Lasciato il ".$val[\"data\"]."> </SPAN></TD></TR>");
if($val[\"www\"]!='http://' && $val[\"www\"]!='')
{
print("<TR><TD class=row1><SPAN class=genmed> <A HREF='".$val[\"www\"]."' target=void class=genmed><IMG
SRC='icon_home.gif' WIDTH=13 HEIGHT=13 BORDER=0 alt='Il mio sito'></A>");
}
else print("<TR><TD class=row1><SPAN class=genmed>");
if($val[\"email\"]!='' && $val[\"email\"]!='EMAIL' && $val[\"email\"]!='email')
{
print(" <A HREF='mailto: ".$val[\"email\"]."' class=genmed><IMG SRC='mail.jpg' WIDTH='10' HEIGHT='8' BORDER=0 ALT='Manda
email'></A><B> Nome:</B> ".$val[\"nome\"]."</SPAN></TD></TR>");
}
else
{
print(" <IMG SRC='nomail.jpg' WIDTH='10' HEIGHT='8' BORDER=0><B> Nome:</B> ".$val[\"nome\"]."</SPAN></TD></TR>");
}
print("<TR><TD valign=top class=row3><SPAN class=genmed><B> Mex:</B> ".$val[\"message\"]."</SPAN></TD></TR>");
}
print("</TABLE>");
}

function mysql_insert($value,$data,$table,$conn) // Inserisce nuovi mess nel db
{
$query=(\"INSERT INTO $table (nome,email,message,data,www) VALUES
('".$value[\"nome\"].\"',\"".$value[\"email\"].\"',\"".$value[\"message\"].\"',\"".$data.\"',\"".$value[\"www\"].\"'\");
$result=mysql_query($query);
}

function esiste($arr,$table,$conn) // Controlla l'esistenza del nuovo mess nel db
{
$query = \"SELECT * FROM ".$table.\" WHERE nome='\".$arr[\"nome\"].\"' AND email='\".$arr[\"email\"].\"' AND
message='\".$arr[\"message\"].\"' AND www='\".$arr[\"www\"].\"'\";
$res = mysql_query($query, $conn);
while($row = mysql_fetch_array($res, MYSQL_ASSOC))
$ret = $row;
if(is_array($ret))
return true;
else
return false;
}
?>

```

Qui sono presenti rispettivamente le funzioni per chiudere la connessione, aprire la connessione al db, per ottenere tutti i records della tabella dati (`mysql_list_all`), per printare tutti i records su di una tabella (`lista`), per inserire i nuovi messaggi, e la funzione `esiste` che controlla se è stato già inserito un mess simile a quello in questione. Da evidenziare è la funzione `lista`, in cui vengono effettuati i controlli sull'indirizzo email e sull'url del sito personale di riferimento.

Tag-board realizzata su file

La stessa tag può essere realizzata tramite supporto di file di testo e quindi senza MySQL; per cui l'applicazione comprenderà gli stessi files `smilies.php` e `tagboard.js` dell'applicazione precedente, poi un file di testo su cui verranno memorizzati i vari messaggi, una `index.php` e il file `fFile.php`. Per vedere in anteprima l'applicazione, cliccate [qui](#).

Vediamo ora la `index.php` :

```

<HTML>
<HEAD>
<LINK HREF="stile.css" type="text/css" rel="stylesheet" >

// Codice JS per il conteggio dei caratteri rimanenti
<SCRIPT language=JavaScript>

```

```

<!--

function initcharsleft()
{
charsleft(document.forms["tagform"].text);
}

function charsleft(feld)
{
var anz = feld.value.length;
if (anz > 150 )
{
feld.value = feld.value.substring(0,225);
frei = 0;
}
else
{
frei = 150 -anz;
}
document.forms["tagform"].num.value = frei;
}
//--> -->
</SCRIPT>

// Codice HTML per la struttura della Tag
</HEAD>
<BODY>
<TABLE border=0 cellpadding=0 cellspacing=0 width="20%" class=tableborder height="70%">
<TR>
<TD class=row4>
<TABLE border=0 cellpadding=0 cellspacing=4 width="100%">
<TR>
<TD>
<TABLE border=0 cellpadding=0 cellspacing=0 width="100%" class=tableborder>
<TR>
<TD valign=top class=row2><div class=div_tag>
<?php // Codice PHP per il recupero dei mess da file e relativo print
if ($_GET["act"]=="printa")
{
if(file_exists("tag.txt"))
{
include_once("fFile.php");
lista("tag.txt");
}
}
if ($_GET["act"]=="exit" || !isset($_GET["act"]) || !isset($_POST["ok"]))
{
print("<CENTER><BR><A HREF='index.php?act=printa'>Vedi Tag</A></CENTER><BR><BR>");
}
?>
</div></TD>
</TR>
</TABLE>
</TD>
</TR>
<TR height=10>
<TD> </TD>
</TR>
<TR>
<TD valign="top"><FORM METHOD="POST" name="tagform" ACTION="index.php?act=insert">
<SCRIPT language="Javascript" src="tagboard.js" type="text/javascript"></SCRIPT>
<CENTER><TABLE border=0 cellpadding=0 cellspacing=1 width="100%">
<TR height=25>
<TD class=row3 colspan=2 align=middle><INPUT TYPE="text" NAME="nome" MAXLENGTH="20" size=20 value="NOME"></TD>
</TR>
<TR height=25>
<TD class=row3 colspan=2 align=middle><INPUT TYPE="text" NAME="email" MAXLENGTH="30" size=20 value="EMAIL"></TD>
</TR>
<TR>
<TD class=row3 valign=middle align=middle><input size=3 value=150 name="num"><BR><A onclick="return
pop_up_smilies();" href="smilies.htm" target=_blank class=genmed> <!--smilies--></A></TD>
<TD class=row3 valign=middle align=middle><TEXTAREA name="message" ROWS="3" COLS="10" maxlength="150"
onKeyPress=charsleft(this); onKeyDown=charsleft(this); onBlur=charsleft(this); onKeyUp=charsleft(this);
onFocus=charsleft(this); wrap=VIRTUAL onChange=charsleft(this);></TEXTAREA></TD>
</TR>
<TR height=30> <TD colspa
n=2 class=row3 valign=middle><CENTER><INPUT TYPE="submit" value=" Invia " name="ok"
class=mainoption></CENTER></TD>
</TR>
</TABLE></CENTER></FORM></TD>
</TR>
</TABLE>

```

```

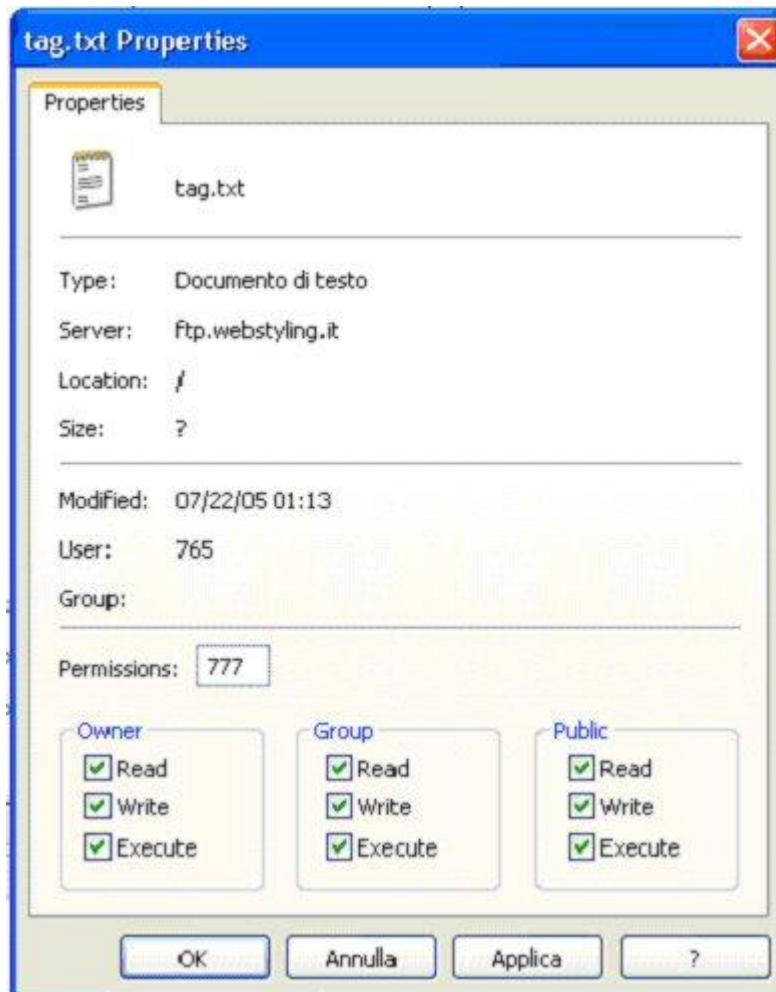
</TD>
</TR>
</TABLE>
</BODY>
</HTML>
<?php // Codice PHP per la memorizzazione del nuovo mess su file
if ($_GET["act"]=="insert")
{
if ($_POST["nome"]!="NOME" && $_POST["message"]!="" && $_POST["nome"]!="nome" && $_POST["nome"]!="")
{
if(file_exists("tag.txt"))
{
include_once("fFile.php");
include_once("smilies.php");
foreach($smile_array as $smile_face=>$smile_image) // Sostituisco il codice SMILIE con quello HTML
{
$_POST["message"] = str_replace($smile_face, $smile_image, $_POST["message"]);
}
$appo=make_string($_POST);
insert($appo,"tag.txt");
}
}
}
?>

```

Le porzioni di codice html sono le stesse, ma ovviamente cambiano quelle php; la prima parte di codice php serve per printare i messaggi, verificando prima l'esistenza del file di testo. La seconda parte di codice php, invece, effettua dei controlli sui valori di MESSAGE e NOME, sostituisce il codice delle smilies con quello html dei tag IMG, viene creata poi una stringa *\$appo* attraverso la funzione `make_string`, e successivamente viene eseguito il suo inserimento su file di testo, attraverso la funzione `insert`, anche essa presente sul file `fFile.php`.

Vediamo poi in dettaglio il file `fFile.php` che contiene tutte le funzioni necessarie per gestire il file di testo "tag.txt".

N.B.: se dovete usare questa versione della Tag su un server Linux, dovete impostare il CHMOD del file di testo "tag.txt" a 777, cambiando i suoi permessi di accesso, esattamente come mostra la figura di sotto, altrimenti non potrete salvare i messaggi sul file stesso.



```

<?php
function lista($file_name) // Stampa la tabella con tutti i mess
{
print("<TABLE border=0 cellpadding=0 cellspacing=4 width=100%>");
print("<TR height=10><TD class=row4> </TD></TR>");
$fp=fopen($file_name,"r");
while(!feof($fp))
{
$riga=fgets($fp,4096);
if($riga!="")
{
$appo=make_record($riga);
echo($appo);
}
}
print("</TABLE>");
print("<br><br><CENTER><A HREF='index.php?act=exit'>Exit</A></CENTER><BR><BR>");
fclose($fp);
}

function make_record($val) // Ritorna una riga HTML (TR) contenente il mess passatogli come stringa
{
$arr=explode("|",$val);
$stringa="<TR><TD class=row1><SPAN class=genmed><B> Nome:</B> ".$arr[0]."<BR><B> Email:</B>
".$arr[1]."</span></TD></TR><TR><TD valign=top class=row3><SPAN class=genmed><B> Mex:</B>
".$arr[2]."</span></TD></TR><TR height=10><TD class=row4> </TD></TR>";
return $stringa;
}

function make_string($arr) // Crea una stringa a partire dall'array contenente il mess
{
$stringa=$arr["nome"]."|".$arr["email"]."|".$arr["message"]."|";
return $stringa;
}

function insert($value,$file_name) // Salva il mess su file di testo

```

```

{
$fp=fopen($file_name,"r");
$old_data=fread($fp,filesize($file_name));
fclose($fp);
$totale=$value."\n".$old_data;
$fp=fopen($file_name,"w");
fwrite($fp,$totale,80000);
fclose($fp);
}

?>

```

In ordine quindi sono riportate le funzioni `lista`, che apre in lettura il file di testo, e fino a quando non viene raggiunta la sua fine (condizione `!feof($fp)` del ciclo `while`), legge una riga per volta e la memorizza su di una stringa, `$riga`, e poi, attraverso la funzione `make_record`, printa la tabella dei messaggi.

Bisogna ora precisare come vengono memorizzati i dati nel file di testo; per questo analizziamo la funzione `insert`, che apre il file dapprima in lettura per leggere tutti i messaggi precedentemente inseriti (attraverso la funzione natia `fread`, viene riportato tutto il contenuto del file su di una stringa chiamata `$old_data`), chiude il file per poi riaprirlo in scrittura e scriverci una stringa complessiva di vecchi e nuovi dati , `$totale` appunto.

Quindi tutti i dati relativi allo stesso messaggio vengono memorizzati su di una stessa riga; è quindi necessario accodare ad una stringa, `$stringa` appunto, il nome, l'email ed il messaggio separati da un carattere jolly, "|", che a noi serve per riconoscere la fine e quindi l'inizio dei tre successivi valori memorizzati. Ad esempio una riga salvata su file potrebbe essere la seguente:

Andrea|andruska1@fastwebnet.it|ciao a tutti!! |

e nel caso in cui non venga inserita l'email, unico campo non necessario per l'inserimento del mess, la riga sarebbe:

Andrea||Ciao a tutti!! |

Tutto ciò è realizzato dalla funzione `make_string`.

La funzione `make_record`, invece, ricevendo come input una stringa corrispondente ad una riga, non fa altro che separare i tre campi e printarli nella tabella html.