

Flat DataBase

Se dovesse essere necessario gestire piccoli database (alcune centinaia di oggetti), seguite questo consiglio del tutto personale, magari criticabile, ma provato e collaudato sul campo!
Lasciate perdere MySQL e simili, non sono la cosa più immediata da utilizzare, e sono consigliabili solo se il gioco ne vale la candela. Provate il metodo seguente, chiamato anche del flat file.

Se voglio, ad esempio, un database consultabile via web che tratti di automobili di cui desidero rendere selezionabili le seguenti caratteristiche, che costituiscono i campi del database:

Marca, Modello, Cilindrata, Alimentazione, Cambio, Carrozzeria, Velocità, Giri

creo allora un file `mts_flatauto.txt` che descrive un'auto su ogni riga:

```
Marca | Modello | Cilindrata | Alimentazione | Cambio | Carrozzeria | Velocità | Giri
fiat | punto | 1200 | benzina | manuale | cabriolet | 160 | 5000
audi | audi 4 | 1500 | benzina | manuale | berlina | 180 | 5500
bmw | 525 d | 2500 | diesel | automatico | berlina | 210 | 6000
```

e così via... aggiungendo una riga per ogni altra auto, (da notare l'uso del separatore |).

Poi per usare tutti questi dati, creo un form di interrogazione `mts_flat.html`

```
<HTML>
<HEAD><TITLE>Flat DataBase</TITLE>
</HEAD>
<BODY>
<FORM name="form1" method="post" action="lista.php">
<TABLE bgcolor="#E0E0E0" cellpadding="5" cellspacing="0" border="0">
<TR><TD>Alimentazione</TD>
    <TD align="right">Benzina <INPUT type="radio" name="ali" value='b' checked></TD>
    <TD align="right">Diesel <INPUT type="radio" name="ali" value='d'></TD>
    <TD align="right">&nbsp;</TD></TR>
<TR><TD>Carrozzeria</TD>
    <TD align="right">Berlina <INPUT type="radio" name="car" value='b' checked></TD>
    <TD align="right">Familiare <INPUT type="radio" name="car" value='f'></TD>
    <TD align="right">Cabriolet <INPUT type="radio" name="car" value='c'></TD></TR>
<TR><TD>Cambio</TD>
    <TD align="right">Manuale <INPUT type="radio" name="cam" value='m' checked></TD>
    <TD align="right">Automatico <INPUT type="radio" name="cam" value='a'></TD>
    <TD align="right">&nbsp;</TD></TR>
<TR><TD colspan="4">...</TD></TR>
<TR><TD colspan="2" align="center"><INPUT type="submit" value="Cerca"></TD>
    <TD colspan="2" align="center"><INPUT type="reset" value="Reset"></TD></TR>
</TABLE>
</FORM>
</BODY></HTML>
```

che farà apparire all'utente una schermata simile alla seguente:

Alimentazione	Benzina <input checked="" type="radio"/>	Diesel <input type="radio"/>	
Carrozzeria	Berlina <input checked="" type="radio"/>	Familiare <input type="radio"/>	Cabriolet <input type="radio"/>
Cambio	Manuale <input checked="" type="radio"/>	Automatico <input type="radio"/>	
...			
<input type="submit" value="Cerca"/>		<input type="reset" value="Reset"/>	

con la quale opererà la scelta, modificando i valori preimpostati, e confermando con un click sul pulsante **Cerca**. Il pulsante **Reset**, invece, serve per riproporre i valori preimpostati.

Tramite un programma in PHP [mts_flatlista.php](#) si raccoglieranno le richieste e si effettueranno due tipi di lavori: l'analisi delle richieste dell'utente e la costruzione di un report contenente il risultato della ricerca.

Nella prima parte si confrontano le caratteristiche delle auto richieste tramite il form, con quelle del database che verrà caricato su un array:

```
$vettore=file('auto.txt'); //(riempie l'array $vettore con le righe del file)
```

poi si farà scorrere l'array `$vettore` con un ciclo for, usando per ogni riga (corrispondente ad una determinata auto) la funzione **explode**, che creerà un altro vettore (`$campo`) grazie al carattere separatore, nel nostro caso '|'.

```
$campo=explode('|',$vettore);
```

Ora, per ogni riga del file (quindi per ogni macchina), si avrà:

<code>\$campo[0] = marca</code>	<code>\$campo[1] = modello</code>	<code>\$campo[2] = cilindrata</code>	<code>\$campo[3] = alimentazione</code>
<code>\$campo[4] = cambio</code>	<code>\$campo[5] = carrozzeria</code>	<code>\$campo[6] = velocità</code>	<code>\$campo[7] = giri</code>

Quindi se le variabili del form valgono: **\$ali = 'b'** (benzina), **\$cam = 'n'** (normale), **\$car = 'f'** (familiare), significa che l'utente chiede l'elenco di tutte le auto a benzina, con cambio normale e carrozzeria familiare. Con una routine come questa posso ottenere facilmente e velocemente l'elenco delle vetture che soddisfano i criteri impostati:

```
<?
// RICERCA
$alim = $_POST['ali'];           // Lettura Valori Form
$scarr = $_POST['car'];
$scamb = $_POST['cam'];

$vettore=file('mts_flatauto.txt');
$trovate=array();              // Array contenente gli indici delle auto trovate
$nauto=count($vettore);       // Numero delle auto del database
for ($i=0; $i<$nauto; $i++) {
    $campo=explode('|',$vettore[$i]); // Array con i valori dell'auto di indice $i
    for ($k=0; $k<count($campo); $k++) $campo[$k]=trim($campo[$k]);
    if($campo[3][0]==$alim && $campo[4][0]==$scamb && $campo[5][0]==$scarr) $trovate[]=$i;
}
// FINE RICERCA
```

Nella seconda parte del programma PHP si analizza il vettore `$trovate` dove ci saranno gli indici delle auto che soddisfano le richieste (alimentazione, carrozzeria, cambio). Potrete preparare facilmente un report mostrando il contenuto di `$trovate` mettendo i suoi indici in `$vettore` ed 'esplodendo' le singole righe. Se invece il vettore `$trovate` risulta vuoto, significa che nessuna auto soddisfa le caratteristiche richieste e verrà emesso un semplice messaggio di avvertimento.

```
// REPORT
if(count($trovate)==0)
    echo "Nessun risultato";
else {
    echo "<TABLE cellspacing=0 cellpadding=4 border=1><TR bgcolor=#E0E0E0>";
    $campo = explode("|",$vettore[0]);
    for($j=0; $j<count($campo); $j++) echo "<TD><b>" . $campo[$j] . "</b></TD>";
    echo "</TR>";

    for($i=0; $i<count($trovate); $i++) {
        $campoOK=explode("|",$vettore[$trovate[$i]]);
        echo "<TR>";
        for($k=0; $k<count($campoOK); $k++)
            echo "<TD>" . $campoOK[$k] . "</TD>";
        echo "</TR>";
    }
    echo "</TABLE>";
}
// FINE REPORT
?>
```