



Indice

1. Informazioni sulla versione PHP installata	2
2. Generatore di numeri random	2
3. Contatore di visitatori	2
4. Programma di inserimento di un nuovo record in un file	4
scrivi_record.htm	5
scrivi_record.php	6
5. Dati i conoscenti presenti in una rubrica telefonica,	8
conf.php.....	11
crea_db.php	12
crea_tabella.php	13
aggiungi1.php	14
aggiungi2.php	16
Soluzione n.1: seleziona.php	19
Soluzione n.2: seleziona.php	21
modifica1.php.....	24
modifica2.php.....	26
cancella.php	27
Promemoria PHP per la gestione di un db MySql	29

Esercizi ed appunti PHP



1. Informazioni sulla versione PHP installata

```
<?php
    phpinfo();
?>
```



2. Generatore di numeri random

```
<?php
    $max = trim($_POST['max']);
?>
:
:
<form name="form1" method="post" action="<? echo $_SERVER['PHP_self']; ?>">
    <input name="max" type="text" value="<?php echo $max; ?>" size="10" maxlength="4">
    Numero casuale generato <?php echo rand(1,$max); ?>
    <input type="submit" name="Submit" value="Invia">
</form>:
```



3. Contatore di visitatori

Questa pagina ha avuto 8 visitatori

Esercizi ed appunti PHP

```
<?php
include('config_contatore.php');

if (!file_exists($Contatore)):

    $scontonumerico= 0;
    $fp = fopen($Contatore,"w");
    flock($fp, 2);
    fwrite ($fp,$scontonumerico);
    flock($fp, 3);
    fclose($fp);

else:

    $fp = fopen($Contatore,"r+");
    $scontonumerico= fread($fp,filesize($Contatore));
    fclose($fp);

endif;

$scontonumerico++;
$fp = fopen($Contatore,"w");
flock($fp, 2);
fwrite ($fp,$scontonumerico);
flock($fp, 3);
fclose($fp);
?>

<html>
<head>
```

```
<?php
$Contatore =
"public/../../visite_php.dat";
?>
```

Per aumentare la portabilità dello script conviene mantenerlo esterno nome e path del file da usare.

Se il file non esiste viene:

aperto in scrittura (w) (=creato)
bloccato in scrittura (2)
scritto un valore nullo per il contatore
sbloccato (3)
chiuso

Se il file esiste viene:

aperto in lettura e scrittura (r+)
letto il contatore
chiuso

In ogni caso viene:
incrementato il contatore
bloccato in scrittura (2)
scritto il nuovo valore per il contatore
sbloccato (3)
chiuso

Nella pagina HTML viene visualizzato il valore del contatore appena calcolato

Esercizi ed appunti PHP

```
<title>Contatore</title>
</head>
<body>
    Questa pagina ha avuto
```

```
<? echo $contonumerico; ?>
```

```
visitatori
</body>
</html>
```

(\$contonumerico).

La **funzione** (o meglio costruito linguistico) **echo()** serve per scrivere o stampare l'output inviato al browser del visitatore che accede al sito. L'output a video prodotto da echo() deve essere in formato html.



4. Programma di inserimento di un nuovo record in un file

Nome	<input type="text"/>
Cognome	<input type="text"/>
<input type="button" value="Inserisci dati"/> <input type="button" value="Ripulisci"/>	

scrivi_record.htm

```
<html>
<head>
  <title>Scrivi e leggi record</title>
</head>
<body>

  <form action="scrivi_record.php" method="post">
    <table border="0" align="center" cellpadding="10"
cellspacing="10">
      <tr>
        <td>Nome</td>
        <td><input name="nome" type="text"
size="30" maxlength="30"></td>
      </tr>
      <tr>
        <td>Cognome</td>
        <td><input name="cognome" type="text"
size="30" maxlength="30"></td>
      </tr>
      <tr align="center" valign="middle">
        <td colspan="2">
          <input name="Invia" type="submit"
value="Inserisci dati">
          <input name="Ripristina" type="reset"
value="Ripulisci">
        </td>
      </tr>
    </table>
  </form>
```

Nelle applicazioni web, l'input dell'utente si acquisisce tramite i moduli (o **FORM**) HTML. **Un qualsiasi elemento di un form sarà automaticamente disponibile per uno script PHP.**

Un semplice form in HTML che non presenta nessun tipo di tags particolari. Quando l'utente riempie questo form e preme il pulsante submit, viene richiamata la pagina **scrivi_record.php**: Quando si definisce un **form** è opportuno: utilizzare sempre un **"NAME"** per ogni elemento di dati (INPUT, SELECT, TEXTAREA etc.). **Questi attributi NAME diventeranno nomi delle variabili PHP**: non si potrà accedere ai valori se non si utilizza un attributo NAME per ciascuno. Il NAME di un campo di modulo non deve necessariamente essere uguale al nome di campo di database corrispondente, ma spesso è una buona idea. E' possibile (spesso indispensabile) specificare un **"VALUE"** invece che

Esercizi ed appunti PHP

```
</body>
</html>
```

lasciare che HTML spedisca il valore predefinito. VALUE può essere **impostato sui dati che si desidera visualizzare nel modulo.**

scrivi_record.php

```
<?php
include ('studenti_configurazione.php');

if (isset($_POST['cognome']) || isset($_POST['nome'])):

    // Scrivi record
    $fp = fopen($file,"a+");
    flock($fp, 2);

    fwrite ($fp,$_POST['cognome'] . ", " .
$_POST['nome'] . "\n");

    flock($fp, 3);
    fclose($fp);
endif;
?>
```

```
<?php
$file="public/../../studenti.dat";
?>
Per aumentare la portabilità dello script
conviene mantenerlo esterno nome e path del
file da usare.

Viene verificato l'avvenuto passaggio dei
dati in input (provengono dalla pagina
scrivi_record.htm)

$fp = fopen($file,"a+"): Apre in lettura e
scrittura; posiziona il puntatore alla fine
del file. Se il file non esiste, tenta di
crearlo
flock($fp, 2): blocca il file in scrittura
(2)

fwrite...: scrivi record

flock($fp, 3): rilascio del file (3)
fclose($fp): chiuso
```

Esercizi ed appunti PHP

```
<html>
<head>
  <title>Scrivi e leggi record</title>
</head>
<body bgcolor="#FFFFFF" title="File">
  <table width="100%" border="1" cellspacing="0"
cellpadding="10">
    <tr>
      <td>Cognome</td>
      <td>Nome</td>
    </tr>

<?
$line = "";
// Apre in lettura e scrittura e posiziona il puntatore
all'inizio del file.
$fp = @fopen($file, "r+") or die("Impossibile trovare
il file");

    while (!feof($fp)):

        $line = fgets($fp, 4096);

        if ($line != ""):
            $vet=explode(",",$line);

            echo "<tr>";
            echo "<td>" . $vet[0] . "</td>";
            echo "<td>" . $vet[1] . "</td>";
            echo "</tr>";
        endif;
    endwhile;
```

`$fp = fopen($file, "r+")`: il file viene aperto in lettura e scrittura (r+) letto il contatore chiuso

`fgets / fread`: la funzione legge un file riga per riga.
`fgets` è un alias di `fread`

Trasferimento `$line` in un vettore

Visualizzazione dei dati letti

La **funzione echo()** (o meglio costruito linguistico) serve per scrivere o stampare l'output inviato al browser del visitatore che accede al sito. L'output a video prodotto da `echo()` deve essere in formato

Esercizi ed appunti PHP

```
fclose($fp);  
?>
```

```
</table>  
<a href="scrivi_record.htm" target="_self">Torna  
alla pagina precedente</a>  
</body>  
</html>
```

html.

In ogni caso viene:
incrementato il contatore
bloccato in scrittura (2)
scritto il nuovo valore per il contatore
sbloccato (3)
chiuso



5. Dati i conoscenti presenti in una rubrica telefonica, di cui sono noti i seguenti dati:

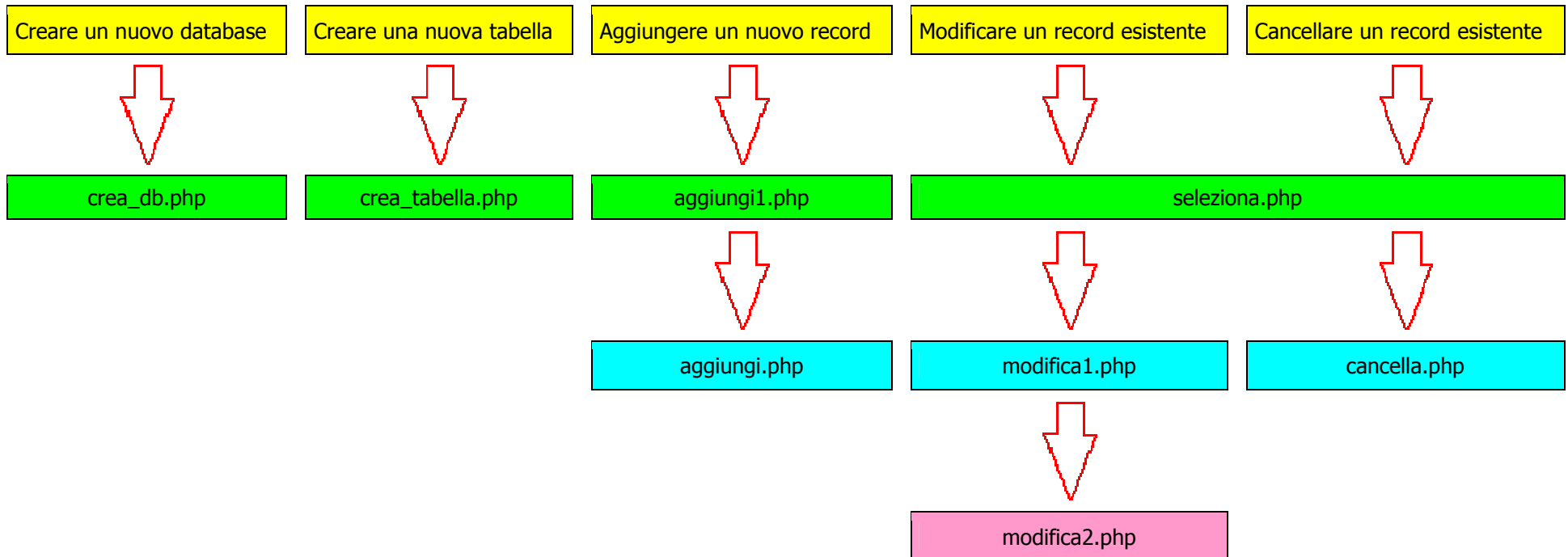
<u>id</u>	nome	cognome	prefisso	telefono	data
-----------	------	---------	----------	----------	------

con id corrispondente alla chiave primaria,
realizzare i seguenti programmi

- creazione di un database
- creazione della tabella rubrica
- inserimento di un nuovo record
- modifica di un record
- cancellazione di un record

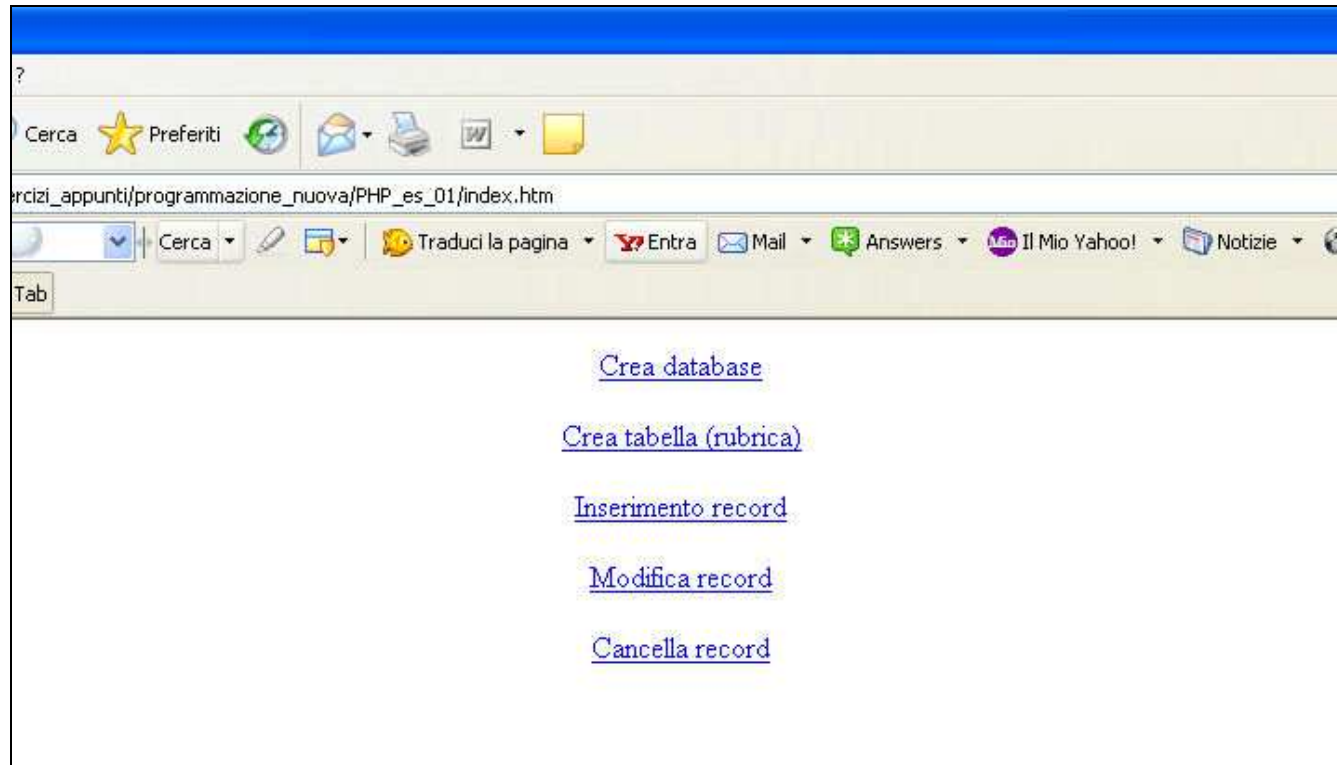
Esercizi ed appunti PHP

Gestione database



Esercizi ed appunti PHP

index.htm (solo html): programma "accessorio" necessario per organizzare la procedura di gestione della tabella



```
<html>
<head>
  <title>Esercizio PHP: gestione database</title>
</head>
<body>
  <p align="center"><a href="crea_db.php" target="_self">Crea database</a></p>
  <p align="center"><a href="crea_tabella.php" target="_self">Crea tabella (rubrica) </a></p>
```

Esercizi ed appunti PHP

```
<p align="center"><a href="aggiungli.php" target="_self">Inserimento record</a></p>
<p align="center"><a href="seleziona.php" target="_self">Modifica record </a> </p>
<p align="center"><a href="seleziona.php" target="_self">Cancella record </a> </p>
</body>
</html>
```

conf.php

(solo php): programma "accessorio" per la definizione dei dati necessari per la connessione al database e l'apertura della tabella

```
<?php
$dbhost = "localhost";

$dbname = "rubrica";
$dbuser = " root";
$dbpass = "";
$connessione =
@mysql_connect($dbhost,$dbuser,$dbpass) or
die("Impossibile stabilire una connessione");
```

Il nome dell'host (hostname) su cui si trova mysql

Le variabili **\$dbhost**, **\$dbuser** e **\$dbpass**

contengono i parametri di connessione specificati per l'utente che deve connettersi al server mysql.

E' bene tenere queste variabili in un **file separato, di estensione .php**, per evitare di dover riscrivere, in tutti i file che utilizzano una connessione al db, i valori di user e password di connessione:

- nome database
- nome utente (username)
- password

Questa funziona accetta tre argomenti e restituisce un identificativo di connessione (o **handle**), che deve essere utilizzato per le successive operazioni fornendolo come argomento alle altre funzioni che seguono.

Il **simbolo @** davanti ad una funzione PHP disabilita eventuali messaggi di **warning** che possono apparire all'esecuzione dello script.

Esercizi ed appunti PHP

```
$db = @mysql_select_db($dbname, $connessione) or  
die("Impossibile selezionare il database");
```

```
?>
```

La **funzione die** è un alias della funzione **exit()** e, come tale, visualizza un messaggio e termina lo script corrente.

Serve per **selezionare (aprire) il database su cui lavorare**. Come argomento accetta il nome del database e quello della connessione.

Attenzione

Si ricordi di aggiungere l'estensione php a tutti i file da includere, per evitare che un utente smalizzato visualizzi il codice sul browser semplicemente collegandosi all'indirizzo dovuto.

Provate a vedere con Google quanti siti hanno un file db.inc che può essere visualizzato in chiaro e che conterrà probabilmente il nome utente e la password per connettersi al database.

crea_db.php

(php + mysql): creazione di un nuovo database

```
<?php  
$dbhost="localhost";  
$dbname="rubrica";  
$dbuser="root";  
$dbpass="";  
$connessione =  
@mysql_connect($dbhost,$dbuser,$dbpass) or  
die("Impossibile collegarsi al server @mysql.");  
$db = @mysql_create_db($dbname,$connessione) or  
die("Impossibile creare il database rubrica");
```

Apertura della connessione

Crea un nuovo database mysql sull'host associato al link più vicino aperto

Esercizi ed appunti PHP

```
@mysql_close($connessione);  
echo "Il database è stato creato con successo";  
?>
```

Chiusura del database

crea_tabella.php

(php + mysql): creazione di una nuova tabella

```
<?php  
include("conf.php");
```

```
$sql = "CREATE TABLE rubrica(Id int not null  
AUTO_INCREMENT PRIMARY KEY, Nome varchar(40),  
Cognome varchar(40), Prefisso varchar(5), Telefono  
varchar(20), Data date)";  
$risultato=@mysql_query($sql, $connessione) or die  
("Impossibile creare la tabella!");
```

La funzione "**include**" (simile alla require) accetta come unico parametro il percorso del file da includere e può restituire un valore (il file incluso è in realtà una funzione che ha una istruzione di return alla fine) Genera la tabella con i vari campi e ed il campo chiave opportunamente definiti

La funzione **mysql_query()** si utilizza per eseguire una query qualsiasi al database selezionato.

Tale funzione accetta come argomenti la query da eseguire, di tipo stringa, e l'identificativo di connessione precedentemente ottenuto con le funzioni connect.

la funzione restituisce **FALSE** se la query non è stata eseguita correttamente

query di selezione: la funzione restituisce un recordset, nel caso la query sia di tipo **SELECT**, che deve essere successivamente scandito per prelevare i singoli record, **query di comando:** è possibile sapere su quanti record ha avuto effetto la query

Esercizi ed appunti PHP

```
@mysql_close($connessione);
```

```
echo "La tabella è stata creata con successo";  
?>
```

Dopo aver effettuato tutte le operazioni sul database, quali selezioni, cancellazioni, update etc., bisogna chiudere la connessione tramite la funzione `mysql_close()`, che accetta come argomento l'handle di connessione

aggiungi1.php

(html + php): programma di inserimento di un nuovo record e segnalazione dell'avvenuto salvataggio di un record nuovo

```
<html>  
<head>  
    <title>Inserimento record</title>  
</head>  
<body>  
    <h1 align="center">Inserimento</h1>  
    <p align="center">  
        <? echo $_REQUEST['messaggio'] ?>  
  
    </p>  
    <form name="form1" method="post"  
action="aggiungi.php">  
        <table width="350">  
            <tr>  
                <td width="59">Nome</td>  
                <td width="240">  
                    <input name="nome" type="text"  
size="25" maxlength="40">  
                </td>
```

`$_REQUEST` contiene le variabili inviate come parametro.

Al posto della funzione `$_REQUEST` può essere usata le funzione `$_GET` e `&_POST`

Esercizi ed appunti PHP

```
</tr>
<tr>
  <td>Cognome</td>
  <td>
    <input name="cognome" type="text"
size="25" maxlength="40">
  </td>
</tr>
<tr>
  <td>Prefisso</td>
  <td>
    <input name="prefisso"
type="text" size="5" maxlength="5">
  </td>
</tr>
<tr>
  <td>Telefono</td>
  <td>
    <input name="telefono"
type="text" size="20" maxlength="20">
  </td>
</tr>
<tr>
  <td>Data</td>
  <td>
    <input name="data" type="text"
size="12" maxlength="12">
  </td>
</tr>
<tr align="center">
  <td height="23" colspan="2">
    <input name="Submit"
type="submit" value="Inserisci">
  </td>
</tr>
```

Esercizi ed appunti PHP

```
        <input name="Reset" type="reset"
id="Reset" value="Annulla">
        </td>
    </tr>
</table>
</form>
<h2 align="center"><a href="index.htm">Torna
al men&ugrave;</a></h2>
</body>
</html>
```

aggiungi2.php

(php + mysql): programma di salvataggio di un nuovo record e preparazione del messaggio che segnala dell'avvenuto salvataggio di un record nuovo

Esercizi ed appunti PHP

The screenshot shows a Microsoft Internet Explorer window with the title 'Inserimento record - Microsoft Internet Explorer'. The address bar displays the URL: `http://localhost/S_I_T_I/SITO_Grazia/esercizi_appunti/programmazione_nuova/PHP_es_01/db_aggiungi_record.htm`. The browser's menu bar includes 'File', 'Modifica', 'Visualizza', 'Preferiti', and 'Strumenti'. The toolbar contains various icons for navigation and actions. Below the address bar, there are search and translation tools. The main content area displays a form with the following fields and labels: 'Nome', 'Cognome', 'Prefisso', 'Telefono', and 'Data'. Each label is followed by a text input field. At the bottom of the form, there are two buttons: 'Inserisci' and 'Annulla'. In the bottom right corner of the browser window, there is a link that says 'Torna al menù'.

```
<?php
$nome = $_POST['nome'];
$cognome = $_POST['cognome'];
$prefisso = $_POST['prefisso'];
$telefono = $_POST['telefono'];
$data = $_POST['data'];
include("conf.php");

$sql = "Insert into rubrica
(nome, cognome, prefisso, telefono, data)
values
('$nome', '$cognome', '$prefisso', '$telefono',
```

\$_POST contiene le variabili inviate con il metodo POST
\$_REQUEST contiene le variabili inviate con il metodo POST o con il metodo GET

La funzione "**include**" include il file per l'apertura della connessione e del database
Preparazione query SQL

Esercizi ed appunti PHP

```
'$data');"
$risultato = @mysql_query($sql, $connessione) or
die("Impossibile eseguire l'interrogazione");

$messaggio = "Record n. ".
mysql_insert_id($connessione) . " inserito con
successo!!";

mysql_close($connessione);
header("Location: aggiungil.php?messaggio=" .
$messaggio);

exit;

?>
```

La funzione `mysql_query()` si utilizza per eseguire una query qualsiasi al database selezionato.

Quando la tabella è dotata di un **campo autoincrementale**, tramite la funzione `mysql_insert_id(...)`, eseguita immediatamente dopo una query di inserimento nel database, è possibile **ottenere il valore dell'unico campo autoincrementale** della tabella.

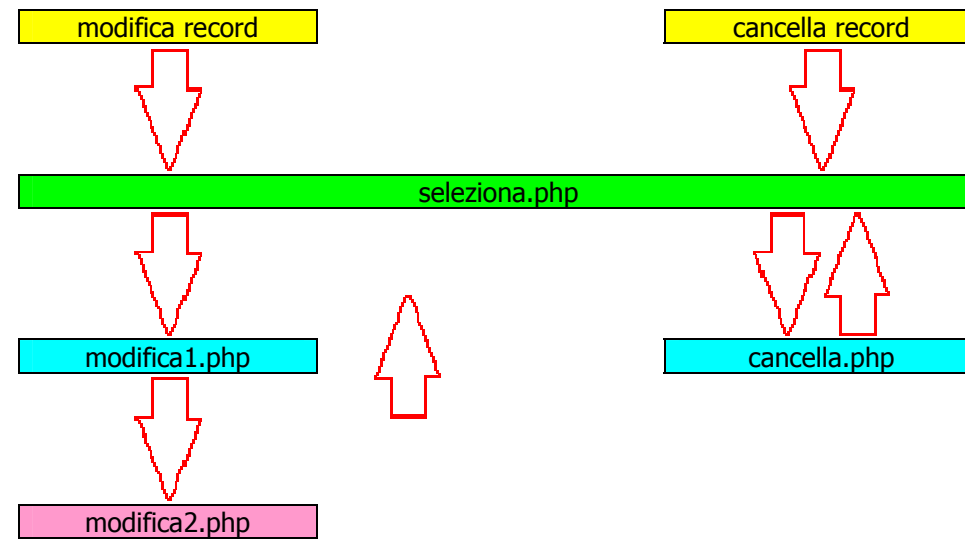
Chiusura connessione

Ritorno al programma di inserimento:

Spedisce un header HTTP. Utilizzando **"Location"** si trasmette al browser, oltre un header, anche un **REDIRECT** con codice di stato (302) il nome indicato **dopo il simbolo ?** (messaggio, in questo caso) rappresenta **il parametro da passare** al programma chiamato (aggiungil.php, in questo caso). Tutto quello che troviamo dopo il simbolo = è il valore che verrà passato al programma chiamato. Più parametri possono essere separati con il **simbolo &**

`exit` assicura che il codice sottostante non sia eseguito dopo il redirectionamento

Esercizi ed appunti PHP



Soluzione n.1: seleziona.php

(php + html + mysql): ricerca, in base all'Id o a Cognome e Nome, del record da modificare/cancellare

Esercizi ed appunti PHP

Scelta record dalla rubrica - Microsoft Internet Explorer

Indirizzo: http://localhost/S_I_T_I/SITO_Grazia/esercizi_appunti/programmazione_nuova/PHP_es_01/db_estrai_modifica.htm

Scelta record dalla rubrica

Id:

Cognome:

Nome:

Visualizza Annulla

[Torna al menù](#)

```
<html>
<head>
<title>Esercizio PHP: gestione database</title>
</head>
<body>
<form id="form1" name="form1" method="post" action=".....php">
  <table width="100%" border="0">
    <tr>
      <td width="9%">Id</td>
      <td width="91%"><input name="id" type="text" id="id" size="10" />
    </td>
    </tr>
    <tr>
      <td>Cognome</td>
      <td><input name="nome" type="text" id="nome" size="50" /></td>
    </tr>
  </table>
</form>
</body>
</html>
```

Esercizi ed appunti PHP

```
<tr>
  <td>Nome</td>
  <td><input name="cognome" type="text" id="cognome" size="50" /></td>
</tr>
<tr>
  <td>&nbsp;</td>
  <td><input type="submit" name="Invio" id="Invio" value="Visualizza" />
    <input type="reset" name="ripristina" id="ripristina" value="Annulla" />
  </td>
</tr>
</table>
</form>
<h1 align="right"><a href="index.htm">Torna al menù</a></h1>
</body>
</html>
```

Soluzione n.2: seleziona.php

(php + html + mysql): visualizzazione di tutti i record ricercati e possibilità di scelta di un record da modificare/cancellare

```
<html>
<head>
  <title>Seleziona i record della rubrica</title>
</head>
<body>
  <p align="center">
    <?php echo $_REQUEST['messaggio'] ?></p>

    <TABLE border="1" align="center"
cellpadding="10" cellspacing="0">

      <TR>
        <TH>Modifica</TH>
```

\$_REQUEST contiene le variabili inviate come parametro. Al posto della funzione **\$_REQUEST** può essere usata la funzione **\$_GET**

Intestazione tabulato

Esercizi ed appunti PHP

```
<TH>Cancellazione</TH>
<TH>Id</TH>
<TH>Cognome</TH>
<TH>Nome</TH>
<TH>Prefisso</TH>
<TH>Telefono</TH>
<TH>Data</TH>
</TR>
<?php
include("conf.php");
$sql = "SELECT * FROM rubrica order by Cognome";
$resultato = @mysql_query($sql,$connessione) or
die( "Errore: " . mysql_error());

$num=@mysql_num_rows($resultato);
```

Interrogazione della rubrica
Esecuzione comando SQL o messaggio di errore
mysql_query(): restituisce un **recordset** che deve essere successivamente scandito per prelevare i singoli record o di **FALSE** in caso di esito negativo
Per determinare il **numero di righe risultante da una query di selezione**, si utilizza la funzione **@mysql_num_rows()**, che accetta come argomento il recordset prelevato e **restituisce il numero di record risultante**.

```
for ($i=1;$i<=$num;$i++) {
    :
}
```

Un **ciclo definito** viene utilizzato quando sappiamo esattamente quante volte bisogna eseguire un blocco di istruzioni.
\$i++:
l'operatore incrementale **\$i++** equivale a **\$i=\$i+1**
while(\$row=@mysql_fetch_assoc(\$resultato)):
:
endwhile;

Esercizi ed appunti PHP

```
while($row=@mysql_fetch_assoc($risultato)):
?>
```

```
<TR>
  <TD align='center'>
    <form name="form1" method="post"
action="modifica.php">
      <input name='id' type='hidden'
value='<?php echo $row['id'] ?>'>
      <input type="submit" name="Invio"
value="Modifica utente">
    </form>
  </TD>
  <TD align='center'>
    <form name="form1" method="post"
action="cancella.php">
```

Un **ciclo indefinito** viene utilizzato quando non sappiamo esattamente quante volte bisogna eseguire un blocco di istruzioni. Per prelevare le informazioni da un **recordset** ed effettuare operazioni sui singoli record quali la visualizzazione sul browser: `$row=@mysql_fetch_assoc($risultato)`: è equivalente alla chiamata di `mysql_fetch_array()`. Questa restituisce solo un **array associativo** oppure **FALSE** se non ci sono più righe.

`$row=@mysql_fetch_row($risultato)`: Restituisce un array che corrisponde al **recordset** caricato oppure **FALSE** se non ci sono più righe. **Ogni colonna del risultato è memorizzata in un indice dell'array**, partendo dall'indice 0. E' possibile accedere ai singoli campi semplicemente ricordando la sequenza dei campi prelevati tramite l'istruzione **SELECT**

Inizio riga di dettaglio

type="hidden": Per trasferire un valore al programma chiamato senza che l'utente ne abbia conoscenza. Il valore trasmesso proviene dalla lettura di un recordset.

type="hidden": Per trasferire un valore al programma chiamato senza che l'utente ne

Esercizi ed appunti PHP

```
<input name='id' type='hidden'
value='<?php echo $row['id'] ?>'>
<input type="submit" name="Invio"
value="Elimina utente">
</form>
</TD>
<TD><?php echo $row['id'] ?></TD>
<TD><?php echo trim($row['cognome'])
?></TD>
<TD><?php echo trim($row['nome']) ?></TD>
<TD><?php echo $row['prefisso'] ?></TD>
<TD><?php echo $row['telefono'] ?></TD>
<TD><?php echo $row['data'] ?></TD>
</TR>
<?php
endwhile;
@mysql_close($connessione);
?>
</TABLE>
<p align="center">Numero record trovati: <?php
echo $num?></p>
</body>
</html>
```

abbia conoscenza. Il valore trasmesso proviene dalla lettura di un recordset.

Fine riga di dettaglio

Chiusura ciclo
Chiusura database

modifica1.php

(php + html + mysql): programma di modifica del record scelto

```
<?
$id = trim($_POST['id']);
$nome = trim($_POST['nome']);
$cognome = trim($_POST['cognome']);
include("conf.php");
```

Da **seleziona.php** provengono a questa pagina:

- id
- cognome e nome

Esercizi ed appunti PHP

```
if ($id > 0):
    $sql = "SELECT * FROM rubrica WHERE Id=$id";
else:
    $sql = "SELECT * FROM rubrica WHERE Nome =
'$nome' AND Cognome = '$cognome' ORDER BY Cognome";
endif;
$risultato = @mysql_query($sql,$connessione) or
die( "Errore: " . mysql_error());

$row=@mysql_fetch_assoc($risultato) or die("Dato
mancante");

mysql_close($connessione);
?>
<html>
<head>
    <title>Modifica di un record nella
rubrica</title>
</head>
<body>
<form name="form1" method="post"
action="modifica2.php">
Id<?php echo $id; ?>

    <input name="id" type="hidden" value="<?php echo
$row['id']; ?>">

Nome
    <input name="nome" type="text" value="<?php echo
$row['nome']; ?>" size="25" maxlength="40">
Cognome
```

Query di lettura del record da modificare

Query di lettura del record da modificare

Esecuzione comando SQL o messaggio di errore
mysql_query(): restituisce un recordset che
deve essere successivamente scandito per
prelevare i singoli record
Estrazione del risultato

Chiusura connessione

type="hidden": Per trasferire un valore al
programma chiamato senza che l'utente ne
abbia conoscenza. Il valore trasmesso
proviene dalla lettura di un recordset.

Esercizi ed appunti PHP

```
<input name="cognome" type="text" value="<?php
echo $row['cognome']; ?>" size="25" maxlength="40">
Prefisso
<input name="prefisso" type="text" value="<?php
echo $row['prefisso']; ?>" size="5" maxlength="5">
Telefono
<input name="telefono" type="text" value="<?php
echo $row['telefono']; ?>" size="25"
maxlength="20">
Data
<input name="data" type="text" value="<?php echo
$row['data']; ?>" size="12" maxlength="12">
<input name="invio" type="submit" id="invio"
value="Modifica record">
</form>
</body>
</html>
```

modifica2.php

(php + mysql): programma di salvataggio del record modificato e preparazione del messaggio che segnala dell'avvenuto salvataggio del record modificato

```
<?
$id = $_POST['id'];
$nome = $_POST['nome'];
$cognome = $_POST['cognome'];
$prefisso = $_POST['prefisso'];
$telefono = $_POST['telefono'];
$data = $_POST['data'];
include("conf.php");
$sql = "Update rubrica
        Set nome='$nome',
```

Esercizi ed appunti PHP

```
cognome='$cognome',
prefisso='$prefisso',
telefono='$telefono',
data='$data'
WHERE id='$id';
$risultato = @mysql_query($sql, $connessione) or
die("Impossibile eseguire l'interrogazione");
@mysql_close($connessione);
$messaggio = "Record n. " . $id . " modificato con
successo!!";

header("Location: seleziona.php?messaggio=" .
$messaggio);
exit;
?>
```

Esecuzione query

Chiusura connessione
Messaggio

Ritorno al programma di modifica

cancella.php

(php + mysql): programma di cancellazione del record scelto e preparazione del messaggio che segnala dell'avvenuta cancellazione del record

```
<?php
include("conf.php");
$id = trim($_POST['id']);
$nome = trim($_POST['nome']);
$cognome = trim($_POST['cognome']);
if ($id > 0):
    $sql = "SELECT * FROM rubrica WHERE Id=$id";
else:
    $sql = "SELECT * FROM rubrica WHERE Nome =
'$nome' AND Cognome = '$cognome' ORDER BY Cognome";
endif;
$sql = "SELECT * FROM rubrica WHERE id=" . $id;
```

Da **seleziona.php** provengono a questa pagina:

- id
- cognome e nome

Query di lettura del record da cancellare

Query di lettura del record da cancellare

Esercizi ed appunti PHP

```
$risultato = @mysql_query($sql,$connessione) or  
die( "Errore: " . mysql_error());
```

```
$row=@mysql_fetch_row($risultato) or die("Dato  
mancante");
```

```
$id=$row[0];  
$sql = "DELETE FROM rubrica WHERE id=" . $id;  
$risultato = @mysql_query($sql, $connessione) or  
die("Impossibile eseguire l'interrogazione");  
@mysql_close($connessione);  
$messaggio = "Record n. " . $id . " cancellato con  
successo!!";  
header("Location: seleziona.php?messaggio=" .  
$messaggio);  
exit;  
?>
```

Esecuzione comando SQL o messaggio di errore
mysql_query():

- restituisce un recordset che deve essere successivamente scandito per prelevare i singoli record

Per prelevare le informazioni da un recordset ed effettuare operazioni sui singoli record quali la visualizzazione sul browser:

- **\$row=@mysql_fetch_row(\$risultato):**
Restituisce un array che corrisponde al recordset caricato oppure **FALSE** se non ci sono più righe. Ogni colonna del risultato è memorizzata in un indice dell'array, partendo dall'indice 0. E' possibile accedere ai singoli campi semplicemente ricordando la sequenza dei campi prelevati tramite l'istruzione SELECT

Eliminazione record scelto

Chiusura della connessione
Messaggio

Ritorno al programma di cancellazione

Promemoria PHP per la gestione di un db MySql

<code>\$connessione =@mysql_connect(nomehost, nomeutente, nomepassword) or die...</code>	Serve per stabilire la connessione con il database
<code>\$db =@mysql_create_db(nomeutente, nomeconnessione)</code>	Serve per creare il database
<code>\$db =@mysql_select_db(nomedb, nomeconnessione)</code>	Apri il database
<code>\$risultato=@mysql_query(nomequery, nomeconnessione)</code>	Restituisce il recordset che potrà successivamente essere letto (SELECT...) per prelevare i singoli record.nome query può contenere una qualunque istruzione sql (create table...,insert into...,select...,update...,delete from...).
<code>\$numero=@mysql_num_rows(nomerecordset)</code>	Estrae dal recordset il numero di righe che lo compongono
<code>\$riga=@mysql_fetch_row(nomerecordset)</code>	Legge una riga del recordset(tabella virtuale e/o query)
<code>\$riga=@mysql_fetch_assoc(nomerecordset)</code>	Legge una riga del recordset(tabella virtuale e/o query)in modalità associativa
<code>\$nomecampo= \$riga[nomecampo]</code>	Consente di assegnare alla variabile \$nomecampo il contenuto del campo scelto presente nella riga che sta leggendo
<code>@mysql_insert_id(nomeconnessione)</code>	Quando la tabella è dotata di un campo contatore/autoincrementale, la funzione, eseguita immediatamente dopo una query di inserimento nel database fornisce il valore dell'unico campo contatore della tabella.
<code>@mysql_close(nomeconnessione)</code>	Chiude la connessione con il server