

MySQL Database Management System

II parte

Progetto gestione booking prenotazioni on-line

1. Progettazione e creazione struttura database che chiameremo **booking**.
2. Progettazione e creazione di un form per l'inserzione dei record in un file di nome **insertion.php**.
3. Progettazione e creazione di una pagina **index.php** che dovrà contenere i pacchetti inseriti suddivisi automaticamente in più pagine.

4. Progettazione e creazione di una pagina **view.php** che visualizzi il pacchetto completo selezionato in precedenza nella pagina **index.php**.
5. Progettazione e creazione di un motore di ricerca attraverso una pagina **search.php** che, in base ad alcuni campi inseriti dall'utente, cercherà i pacchetti che li contengono.
6. Progettazione e creazione di una pagina **result.php** che avrà il compito di elaborare la richiesta inserita dall'utente.

7. Progettazione e creazione di una pagina **booking.php** per l'inserzione del record di prenotazione per il pacchetto selezionato in precedenza nella pagina **view.php** o **result.php**.
8. Progettazione e creazione di una pagina **estratto_conto.php** che dovrà contenere le prenotazioni inserite divise per gruppi suddivisi automaticamente in più pagine.
9. Progettazione e creazione di una pagina **lista_passeggeri.php** che dovrà contenere le prenotazioni inserite divise per pacchetti e per date partenza suddivisi automaticamente in più pagine.

DATABASE booking

Il database dovrà contenere 6 tabelle:

1. Destinazioni (mare, montagna, crociere, ecc....)
2. Nazioni
3. Regioni (contiene un campo in relazione a Nazioni)
4. Pacchetti
 - Descrizione
 - Destinazione
 - Regione
 - Data partenza
 - Disponibilità
 - prezzo
 - Cognome

5. Prenotazioni

- Gruppo
- Pacchetto
- Totale

6. Partecipanti

- Gruppo
- Nome

Dovrà essere popolata tramite la pagina insertion.php soltanto la tabella **pacchetti** mentre le tabelle **destinazione**, **nazioni**, **regioni** saranno preparate direttamente nella creazione del database. In base ai record in esse contenuti potremo scegliere le specifiche per quelli da inserire in **pacchetti**.

SELECT

Questo genere di query servirà per ricavare i dati precedentemente inseriti in una tabella:

```
SELECT colonne FROM tbl_name WHERE condizioni opzioni
```

- *colonne* indica tutte le colonne di cui vogliamo il valore.
- *condizioni* sono le condizioni che devono verificarsi, perché una riga sia visualizzata. Queste sono dello stesso tipo di quelle viste nei casi UPDATE e DELETE.
- *opzioni* ci serviranno per esempio per ordinare le righe.

Per fare un esempio

- ```
SELECT id, cognome, nome, count(*) AS tot
FROM my_table WHERE cognome='Rossi'
ORDER BY nome, id DESC
```
- ```
SELECT cognome FROM my_table GROUP BY  
cognome
```
- ```
SELECT * FROM my_table WHERE cognome
LIKE 'Ro%' OR nome LIKE '%ri%'
```

Divideremo ora il nostro archivio di notizie su due tabelle spezzando la tabella *news* in due parti: prenderemo le colonne autore e mail e ne faremo una tabella nuova che chiameremo *autori*. Nella parte rimanente della tabella *news* aggiungeremo una colonna *autore* che mi farà corrispondere ogni riga della tabella a una sola riga della tabella *autori*. A questo punto non ci resta che vedere come visualizzare le informazioni in modo tale da specificare che quando un articolo è scritto da una certa persona, vogliamo che sia visualizzato proprio il suo indirizzo mail.

Il metodo più semplice è quello di selezionare le singole colonne:

```
$query = "SELECT news.titolo, news.testo, news.data,
news.nome AS autore, autori.mail FROM news, autori
WHERE news.nome = autori.nome AND id='$id';"
```

Normalmente per ottenere l'unione di due tabelle si usa JOIN:

```
$query "SELECT titolo, testo, data, news.nome AS autore,
mail FROM news LEFT JOIN autori ON news.nome =
autori.nome WHERE id='$id';"
```

Dopo aver popolato il database con un certo numero di pacchetti possiamo occuparci di visualizzarne il contenuto. Creiamo la pagina che dovrà contenere le descrizioni inserite e la chiamiamo **index.php**. Inviando quindi la query al database e ne ricaviamo un identificatore nella variabile \$result:

```
$result = mysql_query($query, $db);
```

Attraverso \$result, potremo ora ottenere le singole righe che ci vengono restituite dal database utilizzando la funzione **mysql\_fetch\_array**:

```
while ($row = mysql_fetch_array($result))
```

La funzione restituisce una sola riga del database in base alla selezione della query identificata da \$result. Una volta ottenuta la prima riga, sarà possibile ricavare la riga seguente richiamando nuovamente la funzione mysql\_fetch\_array. Quando le righe saranno finite, la funzione restituirà FALSE.

In seguito dovremo specificare in un blocco cosa vogliamo fare con la riga selezionata, prima di passare alla successiva:

```
{ echo "" . date("j/n/y",
$row[data]) . " - $row[descrizione]
"; }
```

- `<a href=\"view.php?id=$row[id]\">` crea il link che permette all'utente di cliccare sulla descrizione per poter leggere tutto il contenuto e porta alla pagina *view.php* passandogli come parametro l'id del pacchetto che stampiamo sotto forma di *\$row[id]*.
- `date("j/n/y", $row[data])` è una funzione Php che in base a una data in timestamp , *\$row[data]*, crea la data nel formato g/m/aa.
- `- $row[descrizione]</a><br>` dopo aver inserito un delimitatore , “-”, visualizza la descrizione, chiude il tag del link e va a capo.

Vediamo come possiamo accedere al contenuto di un pacchetto a partire dal solo id. Creiamo una nuova pagina che chiamiamo **view.php** che ha come unico scopo quello di visualizzare il contenuto completo di un pacchetto selezionato in precedenza nella pagina *index.php*. La query da aggiungere sarà:

```
$query = "SELECT * FROM pacchetti WHERE id='$id'";
$result = mysql_query($query, $db);
$row = mysql_fetch_array($result);
$data = date("j/n/y", $row[data]);
echo "$row[descrizione]

";
.....
```

In fondo al pacchetto inseriamo due link: il primo porterà nuovamente alla prima pagina (index.php), il secondo aprirà una nuova pagina (booking.php) in cui potremo inserire una prenotazione.

```
echo "
Torna alla pagina
iniziale
";
echo "Prenota
";
```

Vediamo quindi come visualizzare i pacchetti suddivisi automaticamente in più pagine.

```
$query = "SELECT id,data,descrizione FROM pacchetti
ORDER BY data DESC LIMIT $start,$step";

$result = mysql_query($query, $db);
while ($row = mysql_fetch_array($result))
{ echo "" . date("j/n/y",
$row[data]) . " - $row[descrizione]
"; }
```

Prima di inserire il link alla pagina successiva, creiamo una sorta di indice con tutte le pagine elencate e numerate.

```
$query = "SELECT count(*) AS tot FROM pacchetti";
$result = mysql_query($query, $db);
$row = mysql_fetch_array($result);
$pages = intval(($row[tot]-1) / $step)+1;
```

A questo punto possiamo creare un link per ogni pagina:

```
for ($i=0; $i<$pages AND $i<20; $i++)
{ $start_page = $i * $step;
echo "" . ($i+1) .
" "};
```

Infine possiamo inserire il link alla pagina successiva.

```
if ($start + $step < $row[tot])
{ $start_next = $start + $step;
echo "successivi";
}
```

Fatto questo inseriamo un link al motore di ricerca:

```
echo "Cerca negli articoli";
```

Inseriremo un motore di ricerca creando la pagina **search.php** che, in base ad alcuni campi inseriti dall'utente, cercherà i pacchetti che li contengono. Dovremo creare un form che contenga campi di testo per immettere i campi da cercare:

```
<form method=post action=result.php>
<input type=text name=chiave1>
<input type=text name=chiave2>
.....
<input type=submit value=cerca>

</form>
```

Veniamo alla pagina **result.php** che avrà il compito di elaborare la stringa inserita dall'utente.

```
$query = " SELECT id, descrizione, data FROM pacchetti
WHERE ";
$chiave1 = trim($chiave1);
if ($chiave1!= "")
$query .= "destinazione = $chiave1 OR ";
$chiave2 = trim($chiave2);
if ($chiave2!= "")
$query .= "prezzo < $chiave2 OR ";
.....$query .= "0";

$result = mysql_query($query, $db);
while ($row = mysql_fetch_array($result))
{ echo "" . date("j/n/y",
$row[data]) . " - $row[descrizione]
"; }
```

## **mysql\_affected\_rows()**

Dopo aver inviato una query di modifica al database (INSERT, UPDATE, o DELETE) possiamo utilizzare questa funzione per ottenere il numero di righe su cui la modifica ha avuto effetto.

```
$query = "UPDATE my_table SET nome='Mario'
WHERE cognome='Rossi'";
mysql_query($query);
$num = mysql_affected_rows();
echo "Modificate $num righe";
```

## **mysql\_num\_rows(\$result)**

Restituisce il numero di righe di cui è composto il risultato di una query. Questa funzione è valida solo per query del tipo SELECT.

```
$query = "SELECT * FROM my_table WHERE
cognome='Rossi'";
$result = mysql_query($query);
$num = mysql_num_rows($result);
echo "Trovati $num Rossi";
```

## **mysql\_insert\_id()**

Dopo un inserimento, restituisce il valore utilizzato per la colonna di tipo AUTO\_INCREMENT.

```
$query = "INSERT INTO my_table (id, nome,
cognome) VALUES ('', 'Luigi', 'Bianchi')";
mysql_query($query);
$num = mysql_insert_id();
echo "Luigi Rossi ha id: $num";
```

## **mysql\_list\_tables(\$db\_name)**

Restituisce una lista contenente i nomi delle tabelle nel database specificato.

```
$result = mysql_list_tables($my_db);
while (list($table) = mysql_fetch_array($tables))
echo "$table ";
```

## **mysql\_escape\_string(\$stringa)**

Tramite questa funzione vengono inseriti i caratteri di escape nella stringa.

```
$nome = mysql_escape_string($nome);
$pass = mysql_escape_string($pass);
$query = "SELECT * FROM utenti WHERE
nome='$nome' AND password='$pass'";
```

## **mysql\_errno()**

Otteniamo il numero dell'errore dell'ultima operazione eseguita da MySQL. Se non ci sono stati errori la funzione restituisce 0.

```
$query = "INSERT INTO my_table (id, nome,
cognome) VALUES (13, 'Luigi', 'Bianchi')";
if (!(@mysql_query($query)))
echo "Errore nr ". mysql_errno() ." nell'inserimento";
```

## **mysql\_error()**

Come la funzione precedente, ma invece di restituire il numero dell'errore, ne restituisce una descrizione.

```
$query = "INSERT INTO my_table (id, nome,
cognome) VALUES (13, 'Luigi', 'Bianchi')";
if (!(@mysql_query($query)))
echo "Errore nell'inserimento: ". mysql_error();
```