

## Forum per Internet

Realizzare un'applicazione per la gestione di un forum nel Web in cui gli utenti possano leggere i messaggi lasciati dagli altri e possano aggiungerne di propri.

I forum sono servizi inseriti nei siti Web che risultano molto utili per creare ambienti collaborativi virtuali o per gestire dibattiti.

Si supponga che il database sia implementato in Access e registrato nel file *Messaggi.mdb*. Esso è associato tramite ODBC al nome *db\_msg*, specificato come nome della sorgente di dati, DSN (*Data Source Name*).

Il database contiene la tabella *Forum* con la seguente struttura:

Nome	Tipo	Dimensione
ID	Contatore	
Nome	Testo	30
Email	Testo	30
Messaggio	Memo	

L'applicazione è formata da due pagine JSP per le seguenti operazioni:

- inserimento di un nuovo messaggio
- visualizzazione dell'elenco dei messaggi.

### Inserimento

La prima pagina JSP deve consentire a un utente di scrivere un messaggio da aggiungere al forum. L'utente deve anche indicare il suo nome e il suo indirizzo di posta elettronica.

La pagina, memorizzata con il nome *Inserimento.jsp*, ha due funzionalità:

- visualizza il form con il quale inserire nuovi messaggi,
- esegue il comando SQL per inserire un nuovo messaggio nel database.

Il form e l'accesso al database sono contenuti nella stessa pagina JSP.

Per questo motivo all'inizio della pagina si assegna il valore nullo alla connessione per evitare di aprire una connessione già aperta:

```
<%@ page import="java.sql.*" %>
<%! Connection con = null; %>
```

Il form HTML per acquisire i dati dell'utente è il seguente:

```
<H1>Inserimento di un nuovo messaggio</H1>
<FORM ACTION="Inserimento.jsp" METHOD=POST>
Nome: <INPUT TYPE="text" SIZE=20 NAME="nome"><BR>
Email: <INPUT TYPE="text" SIZE=20 NAME="email"><BR>
Messaggio:<BR>
<TEXTAREA COLS="60" ROWS="4" NAME="messaggio"></TEXTAREA><BR>
<INPUT TYPE="submit" NAME="submit" VALUE="Aggiungi">
</FORM>
```

Per verificare che l'utente abbia premuto il pulsante *Aggiungi*, si controlla la presenza del parametro *submit*. Se questo parametro è presente, significa che l'utente ha inviato il form e vuole registrare un messaggio.

Il precedente controllo è gestito dalla seguente struttura di selezione:

```
if (request.getParameter("submit") != null)
{
    // esegue l'inserimento
}
```

L'inserimento dai dati nel database è preceduto dall'operazione di connessione al database:

```
if (con == null)
{
    // apre una connessione con il DB
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        con = DriverManager.getConnection("jdbc:odbc:db1", "", "");
    }
    catch(ClassNotFoundException ex)
    {
        out.println("Driver non trovato.");
    }
    catch(SQLException ex)
    {
        out.println("Connessione fallita.");
    }
}
```

L'inserimento è poi effettuato dal comando *Insert* del linguaggio SQL, con i parametri *nome*, *email* e *messaggio*.

```
PreparedStatement stmt;
String query = "INSERT INTO Forum (nome, email, messaggio)+"
              " VALUES (?, ?, ?)";
// legge i parametri
String str1 = request.getParameter("nome");
String str2 = request.getParameter("email");
String str3 = request.getParameter("messaggio");
try
{
    // crea il comando SQL
    stmt = con.prepareStatement(query);
    stmt.setString(1, str1);
    stmt.setString(2, str2);
    stmt.setString(3, str3);
    stmt.executeUpdate();
    stmt.close();
    out.println("Messaggio aggiunto con successo.");
}
catch(SQLException ex)
{
    out.println("Errore durante l'aggiunta del messaggio.");
}
}
```

## Visualizzazione

La seconda pagina JSP, memorizzata con il nome *VisualizzaMsg.jsp*, consente di visualizzare i messaggi registrati nel forum, specificando per ciascuno l'autore e il suo indirizzo di posta elettronica. La connessione al database è del tutto identica a quella della pagina di inserimento.

I messaggi sono recuperati dal database usando il comando *Select* del linguaggio SQL e sono visualizzati usando un ciclo *while*.

All'interno del ciclo i campi di ogni record sono formattati nel seguente modo:

- il nome è stampato in grassetto;
- l'indirizzo di posta elettronica è posizionato accanto al nome ed è un link attivabile per l'invio di un e-mail;
- il messaggio è stampato sulla riga successiva con lo stile corsivo.

```
<%
Statement stmt;
ResultSet rs;
String query = "SELECT * FROM Forum ORDER BY nome";
stmt = con.createStatement();
rs = stmt.executeQuery(query);
// ciclo per visualizzare i messaggi
while (rs.next())
{
String email = rs.getString("email");
%>
<P>
<B><%= rs.getString("nome") %></B>
(<A HREF="mailto:<%=email%>"><%=email%></A><BR>
<I><%= rs.getString("messaggio") %></I>
<%
} // fine ciclo while
stmt.close();
%>
```

L'applicazione può essere ampliata aggiungendo per ogni messaggio la data di inserimento e visualizzando poi i messaggi in ordine di data, in modo da far comparire per primi i messaggi più recenti.