

Basi di dati e Web (Moduli: Laboratorio e Siti Web centrati sui Dati)

Prova scritta del 14 luglio 2008

Avvertenze: e' severamente vietato consultare libri e appunti; chiunque verrà trovato in possesso di materiale attinente al corso vedrà annullata la propria prova. E' obbligatorio rispondere alle domande 1, 2 e 5.

Durata 1h 40m (1h per solo Laboratorio – 40m per solo Siti web centrati sui dati)

Cognome e nome: _____ Matricola: _____

Data la base di dati "CanaliTV" su SQLServer, contenente le seguenti tabelle:

CANALE (Codice, Nome, Frequenza)

MESSA_IN_ONDA (NomeProgramma, CodCanale, Data, Ora, Note)

PROGRAMMA (Nome, Categoria, Descrizione)

Vincoli di integrità referenziale:

MESSA_IN_ONDA.NomeProgramma → PROGRAMMA,,

MESSA_IN_ONDA.CodCanale → CANALE

MODULO SITI WEB CENTRATI SUI DATI – rispondere sul foglio protocollo (ATTENZIONE:

è necessario rispondere in modo sufficiente alle domande di questo modulo per avere la sufficienza sull'intera prova)

1. Progettare, secondo la metodologia basata sulla specifica di page-schema, lo schema logico di un sito web che presenti le informazioni contenute nella base di dati "CanaliTV". In particolare:
 - a. nella **homePage** si presenti l'elenco dei canali dell'emittente televisiva (secondo l'ordine alfabetico del loro nome) riportando: il codice e il nome del canale. Il nome del canale è un link verso lo schema di pagina **canalePage**.
 - b. nello schema di pagina **canalePage** si presentino tutti i dati di un canale: nome, codice e frequenza. Inoltre si mostri il numero di programmi diversi messi in onda sul canale fino a oggi. Si aggiunga l'elenco dei programmi messi in onda oggi indicando: il nome del programma, la categoria e l'ora di trasmissione. Il nome del programma è un link verso lo schema di pagina **programmaPage**.
 - c. nello schema di pagina **programmaPage** si presentino tutte le informazioni di uno specifico programma: nome, categoria, descrizione, data, ora della messa in onda e note. Si mostri inoltre il numero (quantità) di messe in onda del programma previste nel futuro (da oggi in poi).

Lo studente produca sia gli schemi di pagina che le interrogazioni SQL (DB to page schema) che li alimentano. Si supponga presente sul DBMS relazionale, che esegue le interrogazioni SQL, una funzione Today () che restituisce la data di oggi.

2. Illustrare le caratteristiche principali della tecnologia JSP e specificare il ruolo svolto dalle JSP nell'architettura software di un applicazione web basata sull'approccio Model View Controller (MVC) servlet-centric.

MODULO LABORATORIO – rispondere sul testo e usare eventualmente il foglio protocollo come brutta copia (ATTENZIONE: è necessario rispondere in modo **sufficiente** alle domande di questo modulo per avere la sufficienza sull'intera prova)

1. Scrivere il comando SQL per la creazione della tabella MESSA_IN_ONDA supponendo già create le tabelle CANALE e PROGRAMMA. Si richiede di precisare la chiave primaria e i vincoli di integrità referenziale specificando la politica cascade in caso di aggiornamento delle tabelle "master".

2. Partendo dallo schema logico progettato sopra, si completi la servlet Main.java mostrata nei fogli successivi con il codice necessario all'attivazione della JSP che implementa lo schema di pagina canalePage. Si supponga presente un parametro "ps" che indica la JSP da attivare. Se il parametro è assente viene invocata la JSP che implementa la **homePage**. Vanno scelti e inclusi nella richiesta HTTP i parametri aggiuntivi necessari per le interrogazioni che alimentano gli schemi di pagina.
3. Si completi la JSP canalePage.jsp che implementa lo schema di pagina **canalePage** nel seguente modo:
 - a. Indicare come titolo di primo livello il nome del canale.
 - b. Riportare dopo il titolo su righe separate: codice, frequenza di trasmissione e il numero di programmi messi in onda fino ad oggi.
 - c. Riportare poi in una tabella l'elenco dei programmi messi in onda oggi indicando: il nome del programma, la categoria e l'ora di trasmissione.
4. Si completi la classe java DBMS.java inserendo le interrogazioni da gestire, il metodo makeCanaleBean e il metodo getCanale che produce i dati di un canale che alimentano lo schema di pagina **canalePage**. Si suppongano presenti il metodo makeProgrammaBean e il metodo getProgrammi che estrae i dati relativi ai programmi di un canale messi in onda nella data corrente. Questi ultimi due metodi non vanno implementati.
5. Si scriva il Java Data Bean (CanaleBean.java) che serve per rappresentare le tuple estratte dalla interrogazione che genera i dati di un canale per canalePage.jsp specificando solo il costruttore e i metodi get. Si supponga presente il JDB ProgrammaBean.java con le proprietà: nome, categoria, ora, data, descrizione e note.

Cognome e nome: _____

Matricola: _____

main.java

```
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import beanTV.*;

public class main extends HttpServlet {
    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws IOException, ServletException {
        //Definizione e recupero dell'eventuale parametro ps della servlet
        String ps = "";
        RequestDispatcher rd = null;

        if (request.getParameter("ps") != null) {
            ps =
        }
        try {
            // Oggetto per l'interazione con il Database

            if (ps.equals("")) {
                // Parametro ps assente o vuoto, viene attivata la
                // home page del sito. Non implementare
            } else if
                // Implementare quanto è necessario per l'attivazione
                // della jsp canalePage.jsp

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Cognome e nome: _____

Matricola: _____

canalePage.jsp

```
<%@page import="java.io.*"%>
<%@page import="java.util.*"%>
<%@page import="beanTV.*"%>
<html>
<%
//Dichiaro un bean che conterrà i dati
CanaleBean canale;
//Recupero il bean passato come attributo contenente i dati del canale

//Recupero il vector passato come attributo contenente l'elenco dei programmi
//di oggi del canale
Vector lista =

%>
<!--Inizio la pagina HTML-->
<head>
<title>Pagina della singolo canale</title>
</head>
<body>

</body>
</html>
```

Cognome e nome:_____

Matricola:_____

package beanTV;

public class CanaleBean {

 //Definizione dei campi del Bean (variabili private: accessibili solo
 tramite i metodi get e set)

 //Definizione del costruttore del Bean

 //Metodi get

 //NON IMPLEMENTARE i Metodi set

} // end classe CanaleBean

Cognome e nome: _____

Matricola: _____

```
package beanTV;
```

```
DBMS.java
```

```
import java.sql.*;  
import java.util.*;
```

```
public class DBMS {  
    // Dati di identificazione dell'utente  
  
    // URL per la connessione alla base di dati  
  
    // Driver da utilizzare per la connessione JDBC  
  
    // Definizione delle interrogazioni SQL
```

```
    //Costruttore della classe DBMS
```

```
    //Metodi per la creazione di un bean a partire dal record attuale  
    //del ResultSet dato come parametro
```

```
    private CanaleBean makeCanaleBean (
```

```
    }
```

Cognome e nome: _____

Matricola: _____

```
public Vector getCanale(
    // Dichiarazione delle variabili necessarie
    Connection con = null;
    PreparedStatement pstmt = null;
    ResultSet rs = null;
    Vector result = new Vector();
    try {
```

```
}
```

```
private ProgrammaBean makeProgrammaBean (ResultSet rs) {...} NON IMPLEMENTARE
```

```
public Vector getProgrammi (String codCanale) {...} NON IMPLEMENTARE
```

```
} // end classe DBMS.java
```