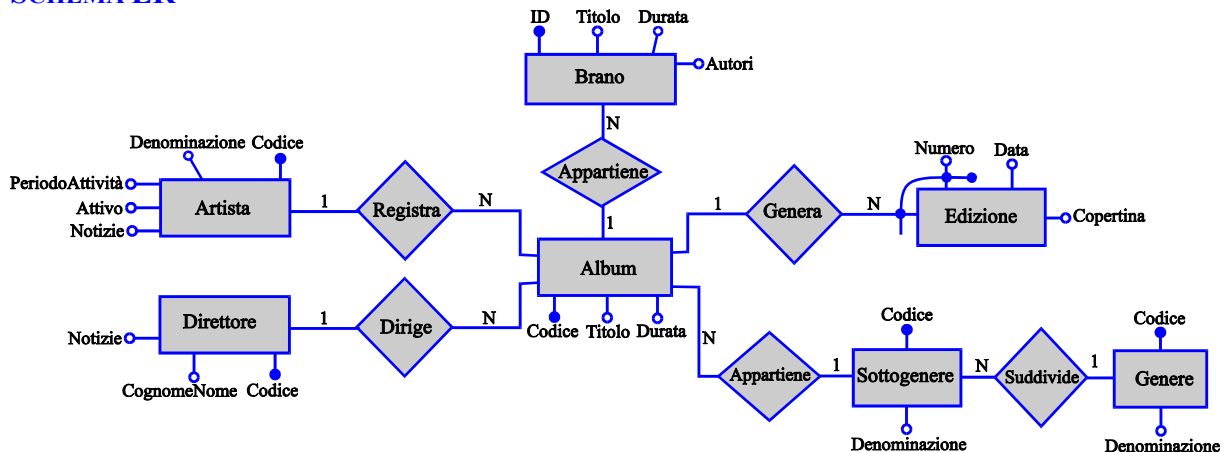


Esame di Stato Istituto Tecnico Industriale
Proposta di soluzione della seconda prova
 Indirizzi: Informatica - CORSO DI ORDINAMENTO e Progetto "ABACUS"
 Anno Scolastico: 2004-2005

1. Schema concettuale della base di dati

■■■ **PROGETTO CONCETTUALE**

SCHEMA ER

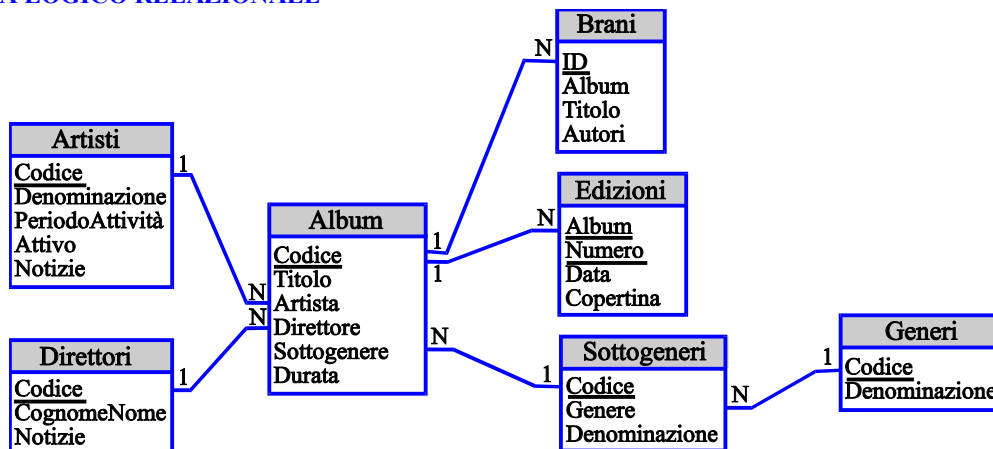


Lo schema ER deve seguire queste **regole**.

- R1.** L'attributo *Codice* di un *Artista* segue il formato "GA-XXXX" con X cifra decimale (adatto per 10000 gruppi o artisti con valori da "GA-0000" a "GA-9999").
- R2.** L'attributo *Attivo* di un *Artista* memorizza valori booleani (un valore *TRUE* rappresenta un artista o un gruppo in attività).
- R3.** L'attributo *PeriodoAttività* di un *Artista* memorizza un valore intero che rappresenta il numero di anni di attività di un gruppo o un artista.
- R4.** L'attributo *Codice* di un *Direttore* segue il formato "D-XXXX" con X cifra decimale (adatto per 10000 Direttori con valori da "D-0000" a "D-9999").
- R5.** L'attributo *Codice* di un *Genere* segue il formato "G-XXX" con X cifra decimale (adatto per 1000 generi con valori da "G-000" a "G-999").
- R6.** L'attributo *Codice* di un *Sottogenere* segue il formato "SG-XXX" con X cifra decimale (adatto per 1000 sottogeneri con valori da "SG-000" a "SG-999").
- R7.** L'attributo *Codice* di un *Album* segue il formato "A-XXXX" con X cifra decimale (adatto per 10000 album con valori da "A-0000" a "A-9999").
- R8.** L'attributo *Numero* di una *Edizione* è un numero intero che rappresenta l'edizione di un album.
- R9.** L'attributo *ID* di un *Brano* è un contatore (numero intero che si incrementa automaticamente).
- R10.** L'attributo *Copertina* di un *Brano* memorizza il percorso e il nome del file o un URL con l'immagine della copertina dell'album.

2. Schema logico della base di dati

■ ■ ■ REALIZZAZIONE SCHEMA LOGICO RELAZIONALE



Lo schema logico relazionale **rispetta le forme normali** ed è soggetto a questi **vincoli di integrità referenziale**.

- V1. La chiave esterna *Artista* della tabella *Album* è in relazione con la tabella *Artisti* mediante la chiave primaria *Codice*.
- V2. La chiave esterna *Direttore* della tabella *Album* è in relazione con la tabella *Direttori* mediante la chiave primaria *Codice*.
- V3. La chiave esterna *Sottogenere* della tabella *Album* è in relazione con la tabella *Sottogeneri* mediante la chiave primaria *Codice*.
- V4. La chiave esterna *Genere* della tabella *Sottogeneri* è in relazione con la tabella *Generi* mediante la chiave primaria *Codice*.
- V5. La chiave esterna *Album* della tabella *Edizioni* è in relazione con la tabella *Album* mediante la chiave primaria *Codice*.
- V6. La chiave esterna *ID* della tabella *Brani* è in relazione con la tabella *Album* mediante la chiave primaria *Codice*.

3. Definizione delle relazioni della base di dati in linguaggio SQL

CODIFICA

Per tradurre nel DBMS lo **schema logico del database** si deve eseguire (una sola volta) questo codice sorgente SQL.

```
CREATE DATABASE DBCasaDiscografica

CREATE TABLE Artisti
(
  Codice          CHAR(7) CHECK(Codice LIKE 'GA-____'),
  Denominazione   CHAR(50) NOT NULL,
  PeriodoAttività INTEGER NOT NULL,
  Attivo          BIT,
  Notizie         CHAR(255),
  PRIMARY KEY(Codice)
)
```

```
CREATE TABLE Direttori
(
  Codice          CHAR(6) CHECK(Codice LIKE 'D-____'),
  CognomeNome    CHAR(50) NOT NULL,
  Notizie        CHAR(255),
  PRIMARY KEY (Codice)
)

CREATE TABLE Generi
(
  Codice          CHAR(5) CHECK(Codice LIKE 'G-____'),
  Denominazione  CHAR(20) NOT NULL,
  PRIMARY KEY(Codice)
)

CREATE TABLE SottoGeneri
(
  Codice          CHAR(6) CHECK(Codice LIKE 'SG-____'),
  Genere          CHAR(5),
  Denominazione  CHAR(20) NOT NULL,
  PRIMARY KEY(Codice),
  FOREIGN KEY(Genere) REFERENCES Generi(Codice)
)

CREATE TABLE Album
(
  Codice          CHAR(6) CHECK(Codice LIKE 'A-____'),
  Titolo          CHAR(100) NOT NULL,
  Artista         CHAR(7),
  Direttore       CHAR(6),
  SottoGenere     CHAR(6),
  Durata         CHAR(10) NOT NULL,
  PRIMARY KEY(Codice),
  FOREIGN KEY(Artista) REFERENCES Artisti(Codice),
  FOREIGN KEY(Direttore) REFERENCES Direttori(Codice),
  FOREIGN KEY(SottoGenere) REFERENCES Sottogeneri(Codice)
)

CREATE TABLE Edizioni
(
  Album          CHAR(6),
  Numero         INTEGER,
  Data           DATE NOT NULL,
  Copertina     CHAR(20) NOT NULL,
  PRIMARY KEY(Album, Edizione),
  FOREIGN KEY(Album) REFERENCES Album(Codice)
)

CREATE TABLE Brani
(
  ID             INTEGER IDENTITY(1,1),
  Album          CHAR(6),
  Titolo        CHAR(100) NOT NULL,
  Durata        CHAR(10) NOT NULL,
  Autori        CHAR(100) NOT NULL,
  PRIMARY KEY (ID),
  FOREIGN KEY(Album) REFERENCES Album(Codice)
)
```

4. Implementazione dell'operazione in linguaggio SQL

Le **interrogazioni sul database** sono realizzate mediante questi codici sorgente SQL che devono essere eseguiti in un DBMS.

Interrogazione: informazioni album

Dato il titolo di un brano, quali sono le date di pubblicazione, la durata, l'artista e/o il gruppo musicale e/o l'orchestra con il relativo periodo di attività nonché gli autori e l'album di appartenenza.

```
SELECT Album.Titolo AS 'Titolo album',
       Edizioni.Numero AS 'Edizione album',
       Edizioni.Data AS 'Data edizione',
       Brani.Durata AS 'Durata brano',
       Artisti.Denominazione AS 'Artista, gruppo, orchestra',
       Brani.Autori AS 'Autori brano',
       Artisti.PeriodoAttività AS 'Periodo attività'
FROM   Artisti, Album, Brani, Edizioni
WHERE  Artisti.Codice = Album.Artista AND
       Album.Codice = Brani.Album AND
       Album.Codice = Edizioni.Album AND
       Brani.Titolo = [Digita titolo del brano]
```

Nel codice precedente, la frase racchiusa tra parentesi quadre *[Digita titolo del brano]* rappresenta il parametro di ingresso il cui valore viene richiesto dal DBMS prima dell'esecuzione dell'operazione.

Esame di Stato Istituto Tecnico Industriale

Proposta di soluzione della parte facoltativa della seconda prova

Indirizzi: Informatica - CORSO DI ORDINAMENTO e Progetto "ABACUS"
Anno Scolastico: 2004-2005

La soluzione può essere organizzata mediante una serie di quattro pagine ASP, richiamate una dopo l'altra per eseguire le operazioni del testo con il seguente ordine:

generi.asp → *sottogeneri.asp* → *nomi_artisti.asp* → *album.asp*

Le pagine precedenti sono memorizzate nella directory *infoautori* pubblicata nel sito Web della casa discografica (cartella Web *casadiscografica*).

La base di dati (ad esempio il file di Access *DBCasaDiscografica.mdb*) è memorizzata nella cartella *fpdb* del sito Web; la **connessione al database** può essere dichiarata come variabile globale in modo che sia visibile durante una sessione HTTP di un utente all'applicazione Web. Per questo obiettivo è sufficiente creare un file di testo di nome *global.asa*, nella radice del sito Web, e dichiarare la variabile *StringaConnessione* nel corpo dell'evento *Session_OnStart()* con la seguente sintassi.

```
global.asa
<SCRIPT LANGUAGE="VBScript" RUNAT="SERVER">
Sub Session_OnStart()
  Dim StringaConnessione
  Session("StringaConnessione") = "DRIVER={Microsoft Access Driver (*.mdb)};" &_
    "DBQ=" & Server.MapPath("fpdb/DBCasaDiscografica.mdb")
End Sub
</SCRIPT>
```

Per il richiamo automatico della pagina *generi.asp* possiamo inserire nella home folder del sito Web la seguente home page esemplificativa (utile per il collaudo delle operazioni realizzate). La pagina *default.asp*, oltre all'avvio delle operazioni richieste, ha anche la funzione di definire (tramite *global.asa*) la stringa di connessione al database.

```
default.asp
<%@ Language=VBScript %>
<% Response.Redirect("infoautori/generi.asp") %>
```

Le operazioni richieste dal testo possono essere realizzate, nell'ordine, nel seguente modo.

1. Una pagina con l'elenco dei generi musicali.

La prima pagina *generi.asp* nella directory Web *infoautori* esegue la seguente query presentando in una casella di riepilogo (creata dinamicamente nello script della pagina) la lista dei generi musicali disponibili.

```
SELECT Denominazione
FROM Generi
```

La pagina può essere organizzata nel seguente modo.



La casella di riepilogo è inserita in un form che richiama, quando l'utente fa clic sul pulsante, la pagina *sottogeneri.asp*. Il codice sorgente della pagina ASP *generi.asp* è il seguente.

```

infoautori/generi.asp
<%@ Language=VBScript %>
<% Response.Buffer=True %>

<HTML>
<HEAD><TITLE>Elenco dei generi</TITLE></HEAD>
<BODY>
<H3>Elenco dei generi</H3>
<% Dim ConnessioneDB, rsTabella, Query
   Set ConnessioneDB = Server.CreateObject("ADODB.Connection")
   ConnessioneDB.Open Session("StringaConnessione")

   Query = "SELECT Denominazione " &_
           " FROM Generi "

   Set rsTabella = ConnessioneDB.Execute(Query)%>
<% If rsTabella.BOF=True And rsTabella.EOF=True Then %>
   <H4>Nessun genere disponibile</H4>
<% Else %>
   <% Dim riga, denominazione %>
   <FORM ACTION="sottogeneri.asp" METHOD="POST">
     <SELECT MULTIPLE NAME="lstGeneri">
       <% Do until rsTabella.EOF=True
          denominazione = rsTabella.Fields("Denominazione")
          riga = "<OPTION VALUE='" & rsTabella.Fields("Denominazione") & "'> " &_
                denominazione & "</OPTION>"
          Response.Write riga
          rsTabella.MoveNext
        Loop
       %>
     </SELECT>
     <BR><INPUT TYPE="submit" NAME="cmdInvia" VALUE="Elenco sottogeneri">
   </FORM>
<% End If %>

<% rsTabella.Close : Set rsTabella = Nothing
   ConnessioneDB.Close : Set ConnessioneDB = Nothing
%>
</BODY>
</HTML>

```

2. Con un clic sul nome di un genere musicale, una pagina con l'elenco dei nomi dei sottogeneri musicali.

La seconda pagina *sottogeneri.asp* esegue la seguente query presentando in una casella di riepilogo (creata dinamicamente nello script della pagina) la lista dei sottogeneri musicali che corrispondono al genere scelto nella casella di riepilogo di *lstGeneri* inserita nella pagina *generi.asp*.

```

SELECT Sottogeneri.*
FROM Sottogeneri INNER JOIN Generi
ON Sottogeneri.Genere = Generi.Codice
WHERE Generi.Denominazione = [Genere]

```

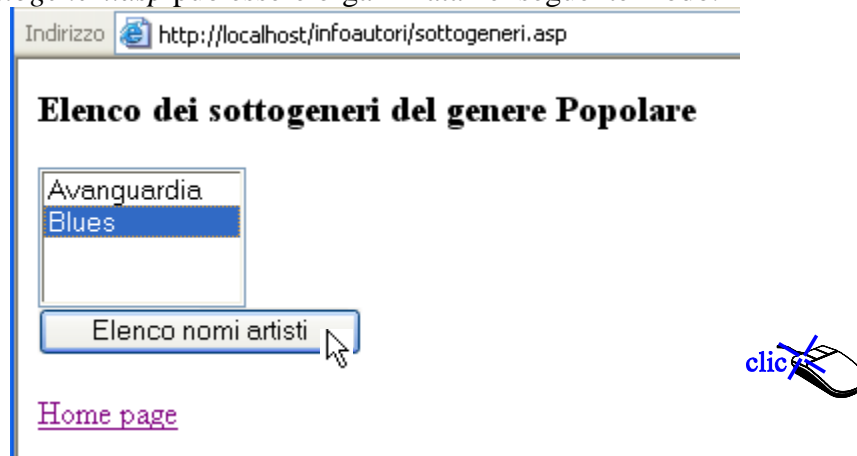
Nella pagina *sottogeneri.asp*, il genere è ottenuto mediante il parametro *Genere*, il cui valore deriva dalla scelta (con un clic) dell'utente nella casella di riepilogo *lstGeneri* inserita nella pagina *generi.asp* (che richiama appunto *sottogeneri.asp*).

```

<% Genere = Request.Form("lstGeneri") %>

```

La pagina *sottogeneri.asp* può essere organizzata nel seguente modo.



La casella di riepilogo è inserita in un form che richiama, quando l'utente fa clic sul pulsante, la pagina *nomi_artisti.asp*. Il codice sorgente della pagina ASP *sottogeneri.asp* è il seguente.

```

infoautori/sottogeneri.asp
<%@ Language=VBScript %>
<% Response.Buffer=True %>

<HTML>
<HEAD><TITLE> Elenco sottogeneri </TITLE></HEAD>
<BODY>
<% Dim ConnessioneDB, rsTabella, Query, Genere
Set ConnessioneDB = Server.CreateObject("ADODB.Connection")
ConnessioneDB.Open Session("StringaConnessione")

Genere = Request.Form("lstGeneri") %>
<H3>Elenco dei sottogeneri del genere <%=Genere%></H3>

<% Query="SELECT Sottogeneri.* " &_
        " FROM Sottogeneri INNER JOIN Generi " &_
        " ON Sottogeneri.Genere = Generi.Codice " &_
        " WHERE Generi.Denominazione = '" & Genere & "'" %>
Set rsTabella = ConnessioneDB.Execute(Query)%>
<% If rsTabella.BOF=True And rsTabella.EOF=True Then %>
  <H4>Nessun sottogenere esistente </H4>
<% Else %>
  <% Dim riga %>
  <FORM ACTION="nomi_artisti.asp" METHOD="POST">
    <SELECT MULTIPLE NAME="lstSottoGeneri">
      <% Do until rsTabella.EOF=True
        denominazione = rsTabella.Fields("Denominazione")
        riga = "<OPTION VALUE='" & rsTabella.Fields("Denominazione") & "'> " &_
              denominazione & "</OPTION>"
        Response.Write riga
        rsTabella.MoveNext
      Loop
    %>
    </SELECT>
    <BR><INPUT TYPE="submit" NAME="cmdInvia" VALUE="Elenco nomi artisti">
  </FORM>
<% End If %>

<% rsTabella.Close : Set rsTabella = Nothing
ConnessioneDB.Close : Set ConnessioneDB = Nothing
%>
<P><A HREF="generi.asp">Home page</A></P>
</BODY>
</HTML>

```

3. Con un clic sul nome di un sottogenere musicale, una pagina con l'elenco dei nomi degli artisti.

La terza pagina *nomi_artisti.asp* esegue la seguente query presentando in una casella di riepilogo (creata dinamicamente nello script della pagina) la lista degli artisti che hanno scritto album appartenenti al sottogenere scelto nella casella di riepilogo di *lstSottoGeneri* inserita nella pagina *sottogeneri.asp*.

```
SELECT DISTINCT Artisti.Denominazione
FROM Artisti, Album, Sottogeneri
WHERE Artisti.Codice = Album.Artista AND
      Album.Sottogenere = Sottogeneri.Codice AND
      Sottogeneri.Denominazione = [Sottogenere]
```

Nella pagina *nomi_artisti.asp*, il sottogenere è ottenuto mediante il parametro *SottoGenere*, il cui valore deriva dalla scelta (con un clic) dell'utente nella casella di riepilogo *lstSottoGeneri* inserita nella pagina *sottogeneri.asp* (che richiama appunto *nomi_artisti.asp*).

```
<% SottoGenere = Request.Form("lstSottoGeneri") %>
```

La pagina *nomi_artisti.asp* può essere organizzata nel seguente modo.



La casella di riepilogo è inserita in un form che richiama, quando l'utente fa clic sul pulsante, la pagina *album.asp*. Il codice sorgente della pagina ASP *nomi_artisti.asp* è il seguente.

```
infoautori/nomi_artisti.asp
<%@ Language=VBScript %>
<% Response.Buffer=True %>

<HTML>
<HEAD><TITLE>Elenco nomi artisti</TITLE></HEAD>
<BODY>
<% Dim ConnessioneDB, rsTabella, Query, SottoGenere
   Set ConnessioneDB = Server.CreateObject("ADODB.Connection")
   ConnessioneDB.Open Session("StringaConnessione")
   SottoGenere = Request.Form("lstSottoGeneri") %>

   <H3>Elenco dei nomi degli artisti del sottogenere <%=SottoGnenere%> </H3>

<% Query = "SELECT DISTINCT Artisti.Denominazione " &_
           " FROM Artisti, Album, Sottogeneri " &_
           " WHERE Artisti.Codice = Album.Artista AND " &_
           "         Album.Sottogenere = Sottogeneri.Codice AND " &_
           "         Sottogeneri.Denominazione = '" & SottoGenere & "';"

   Set rsTabella = ConnessioneDB.Execute(Query)%>
<% If rsTabella.BOF=True And rsTabella.EOF=True Then %>
   <H4>Nessun artista disponibile</H4>
<% Else %>
   <% Dim riga, denominazione %>
```



```

<FORM ACTION="album.asp" METHOD="POST">
  <SELECT MULTIPLE NAME="lstArtisti">
    <% Do until rsTabella.EOF=True
      denominazione = rsTabella.Fields("Denominazione")
      riga = "<OPTION VALUE='" & rsTabella.Fields("Denominazione") & "'> " &_
        denominazione & "</OPTION>"
      Response.Write riga
      rsTabella.MoveNext
    Loop %>
  </SELECT>
  <BR><INPUT TYPE="submit" NAME="cmdInvia" VALUE="Elenco album">
</FORM>
<% End If %>

<% rsTabella.Close : Set rsTabella = Nothing
  ConnessioneDB.Close : Set ConnessioneDB = Nothing
%>
<P><A HREF="generi.asp">Home page</A></P>
</BODY>
</HTML>

```

4. Con un clic sul nome di un artista, l'elenco degli album di quell'artista con le immagini di copertina, la durata, l'anno o gli anni di edizione, le notizie bibliografiche sull'artista ed il periodo di attività.

La quarta pagina *album.asp* esegue la seguente query presentando una tabella con le informazioni richieste relative all'artista scelto nella casella di riepilogo di *lstArtisti* inserita nella pagina *nomi_artisti.asp*.

```

SELECT Album.Titolo, Edizioni.Copertina, Album.Durata,
       Edizioni.Data, Artisti.Notizie, Artisti.PeriodoAttività
FROM Artisti, Album, Edizioni
WHERE Artisti.Codice = Album.Artista AND
       Album.Codice = Edizioni.Album AND
       Artisti.Denominazione= [NomeArtista]

```

Nella pagina *album.asp*, il nome dell'artista è ottenuto mediante il parametro *NomeArtista*, il cui valore deriva dalla scelta (con un clic) dell'utente nella casella di riepilogo *lstArtisti* inserita nella pagina *nomi_artisti.asp* (che richiama appunto *album.asp*).

```

<% NomeArtista = Request.Form("lstArtisti") %>

```

La pagina *album.asp* presenta le informazioni richieste in una tabella organizzata nel seguente modo.

Indirizzo <http://localhost/infoautori/album.asp> Vai Collegamenti >>

Elenco degli album dell'artista GRUPPO A

Titolo album	Copertina	Durata album	Data edizione	Notizie bibliografiche artista	Periodo attività
Titolo album A	Copertina A	60 m 25 s	02/02/2002	Notizie GRUPPO A	10
Titolo album A	Copertina B	60 m 25 s	03/04/2004	Notizie GRUPPO A	10
Titolo album B	Copertina C	56 m 4 s	06/07/2004	Notizie GRUPPO A	10

[Home page](#)

Il codice sorgente della pagina ASP *album.asp* è il seguente.

```

infoautori/album.asp
<%@ Language=VBScript %>
<% Response.Buffer=True %>

<HTML>
<HEAD><TITLE>Album di un artista</TITLE></HEAD>
<BODY>
<% Dim ConnessioneDB, rsTabella, Query, NomeArtista
Set ConnessioneDB = Server.CreateObject("ADODB.Connection")
ConnessioneDB.Open Session("StringaConnessione")
NomeArtista = Request.Form("lstArtisti") %>

<H3>Elenco degli album dell'artista <%=NomeArtista%></H3>

<% Query = "SELECT Album.Titolo, Edizioni.Copertina, Album.Durata, " &_
" Edizioni.Data, Artisti.Notizie, Artisti.PeriodoAttività " &_
" FROM Artisti, Album, Edizioni " &_
" WHERE Artisti.Codice = Album.Artista AND " &_
" Album.Codice = Edizioni.Album AND " &_
" Artisti.Denominazione = '" & NomeArtista & "'" %>

Set rsTabella = ConnessioneDB.Execute(Query)%>
<% If rsTabella.EOF=True And rsTabella.EOF=True Then %>
<H4>Nessun album disponibile</H4>
<% Else %>
<TABLE BORDER="1">
<TR>
<TH>Titolo album</TH><TH>Copertina</TH><TH>Durata album</TH>
<TH>Data edizione</TH><TH>Notizie bibliografiche artista</TH>
<TH>Periodo attività</TH>
</TR>
<% Dim riga, URLcopertina
Do until rsTabella.EOF=True %>
<TR>
<TD ALIGN="CENTER"><%=rsTabella.Fields("Titolo")%></TD>
<% URLcopertina = "images/" & rsTabella.Fields("copertina")
riga = "<TD ALIGN='CENTER'><IMG SRC=" & URLcopertina & "></TD>"
Response.Write riga
%>
<TD ALIGN="CENTER"><%=rsTabella.Fields("Durata")%></TD>
<TD ALIGN="CENTER"><%=rsTabella.Fields("Data")%></TD>
<TD ALIGN="CENTER"><%=rsTabella.Fields("Notizie")%></TD>
<TD ALIGN="CENTER"><%=rsTabella.Fields("PeriodoAttività")%></TD>
</TR>
<% rsTabella.MoveNext %>
<% Loop %>
</TABLE>
<% End If %>

<% rsTabella.Close : Set rsTabella = Nothing
ConnessioneDB.Close : Set ConnessioneDB = Nothing
%>
<P><A HREF="generi.asp">Home page</A></P>
</BODY>
</HTML>

```