**FUNZIONI PER QUERY**

Public E\_Inf As Integer, E\_Sup As Integer

Public Sub Prova\_Query(E\_I As Integer, E\_S As Integer)

 E\_Inf = E\_I

 E\_Sup = E\_S

 TempVars.RemoveAll

 TempVars.Add "Estremo\_Sup", E\_S

 TempVars.Add "ID", E\_I

 Debug.Print TempVars("Estremo\_Sup")

 Debug.Print TempVars("ID")

 Application.DoCmd.OpenQuery ("AAAA5")

End Sub

Public Function Get\_EI() As Integer

 Get\_EI = E\_Inf

End Function

Public Function Get\_ES() As Integer

 Get\_ES = E\_Sup

End Function

Public Function Concat(Nome As String, Cognome As String) As String

 Concat = UCase(Left(Nome, 1)) & "." & UCase(Left(Cognome, 1)) & LCase(Mid(Cognome, 2))

End Function

Public Function Age(D As Date) As Integer

 Age = DateDiff("yyyy", D, Date)

 If Month(D) > Month(Date) Or \_

 Month(D) = Month(Date) And (Day(D) > Day(Date)) Then Age = Age - 1

End Function

**LISTA QUERY**

**AAA1**

SELECT ID, Concat(Nome, Cognome) AS Full\_Name, Age([Data Di Nascita]) AS Age

FROM Utenti;

**AAA2**

SELECT ID, Concat(Nome, Cognome) AS Full\_Name, Age([Data Di Nascita]) AS Age

FROM Utenti

WHERE Age([Data Di Nascita]) IN (20, 40);

WHERE Age([Data Di Nascita]) Between 20 And 40

**AAA3 – AAA4 – AAA5 – AAA6**

SELECT ID, Concat(Nome, Cognome) AS Full\_Name, Age([Data Di Nascita]) AS Age

FROM Utenti

WHERE Age([Data Di Nascita]) <= [Estremo Superiore] AND Age([Data Di Nascita]) >=[Estremo Inferiore];

SELECT ID, Concat(Nome, Cognome) AS Full\_Name, Age([Data Di Nascita]) AS Age

FROM Utenti

WHERE Age([Data Di Nascita]) <= Get\_ES() AND Age([Data Di Nascita]) >=Get\_EI();

SELECT ID, Concat(Nome, Cognome) AS Full\_Name, Age([Data Di Nascita]) AS Age

FROM Utenti

WHERE Age([Data Di Nascita]) <= TempVars!ES AND Age([Data Di Nascita]) >=TempVars!EI;

SELECT ID, Concat(Nome, Cognome) AS Full\_Name, Age([Data Di Nascita]) AS Age

FROM Utenti

WHERE Age([Data Di Nascita]) <= Maschere!AAAA.Txt\_LS.Value AND Age([Data Di Nascita]) >=Maschere!AAAA.Txt\_LI.Value;

**Banalissimo Codice del Form**

Option Compare Database

Private Sub Update\_Vars()

 E\_Inf = Me.Txt\_LI

 E\_Sup = Me.Txt\_LS

 TempVars.RemoveAll

 TempVars.Add "ES", Me.Txt\_LS.Value

 TempVars.Add "EI", Me.Txt\_LI.Value

End Sub

Private Sub Comando0\_Click()

 Call Update\_Vars

 Application.DoCmd.OpenQuery ("AAAA4")

End Sub

Private Sub Comando1\_Click()

 Call Update\_Vars

 Application.DoCmd.OpenQuery ("AAAA5")

End Sub

Private Sub Comando2\_Click()

 Call Update\_Vars

 Application.DoCmd.OpenQuery ("AAAA6")

End Sub

**AAA7**

SELECT Count(Nome) AS Numero

FROM (Select Distinct Nome From Utenti)

**AAA8**

SELECT Nome, Count(Nome) As Numero

FROM Utenti

GROUP BY Nome;

**AAA9**

SELECT Age([Data Di Nascita]) AS Age, Count(ID)

FROM Utenti

GROUP BY Age([Data Di Nascita]);

**AAAB1**

SELECT Count(ID)

FROM Utenti

WHERE Age([Data Di Nascita]) > (Select Avg(Age([Data Di Nascita])) From Utenti);

**AAAB2**

SELECT Cognome, Count(Corsi.ID)

FROM CORSI INNER JOIN (ISTRUTTORI INNER JOIN [CORSI-ISTRUTTORI] ON ISTRUTTORI.ID = [CORSI-ISTRUTTORI].[ID Istruttore]) ON CORSI.ID = [CORSI-ISTRUTTORI].[ID Corso]

GROUP BY Cognome

HAVING Count(Corsi.ID) >=2;

**AAAB3**

SELECT AAAB3\_2.Cognome, AAAB3\_2.Nome

FROM AAAB3\_1 INNER JOIN AAAB3\_2 ON AAAB3\_1.ID = AAAB3\_2.ID;

SELECT Istruttori.ID

FROM CORSI INNER JOIN (ISTRUTTORI INNER JOIN [CORSI-ISTRUTTORI] ON ISTRUTTORI.ID = [CORSI-ISTRUTTORI].[ID Istruttore]) ON CORSI.ID = [CORSI-ISTRUTTORI].[ID Corso]

GROUP BY Istruttori.ID

HAVING Count(Corsi.ID) >= 2;

SELECT Istruttori.ID, Cognome, Corsi.Nome

FROM CORSI INNER JOIN (ISTRUTTORI INNER JOIN [CORSI-ISTRUTTORI] ON ISTRUTTORI.ID = [CORSI-ISTRUTTORI].[ID Istruttore]) ON CORSI.ID = [CORSI-ISTRUTTORI].[ID Corso];

**AAAB4**

SELECT ISTRUTTORI.ID, ISTRUTTORI.COGNOME, CORSI.NOME

FROM ISTRUTTORI INNER JOIN (CORSI INNER JOIN [CORSI-ISTRUTTORI] ON CORSI.ID = [CORSI-ISTRUTTORI].[ID Corso]) ON ISTRUTTORI.ID = [CORSI-ISTRUTTORI].[ID Istruttore]

WHERE ISTRUTTORI.ID IN (SELECT Istruttori.ID

FROM CORSI INNER JOIN (ISTRUTTORI INNER JOIN [CORSI-ISTRUTTORI] ON ISTRUTTORI.ID = [CORSI-ISTRUTTORI].[ID Istruttore]) ON CORSI.ID = [CORSI-ISTRUTTORI].[ID Corso]

GROUP BY Istruttori.ID

HAVING Count(Corsi.ID) >= 2);

SELECT UTENTI.COGNOME, UTENTI\_1.COGNOME, UTENTI.USERNAME

FROM UTENTI INNER JOIN UTENTI AS UTENTI\_1 ON UTENTI.USERNAME = UTENTI\_1.USERNAME

WHERE Utenti.Id < Utenti\_1.Id

**AAAB5**

SELECT UTENTI.COGNOME, UTENTI\_1.COGNOME, UTENTI.USERNAME

FROM UTENTI INNER JOIN UTENTI AS UTENTI\_1 ON UTENTI.USERNAME = UTENTI\_1.USERNAME

WHERE Utenti.Id < Utenti\_1.Id;

**AAAB6**

SELECT UTENTI.COGNOME, Count(Ingressi.ID) AS Numero\_Ingressi

FROM UTENTI INNER JOIN INGRESSI ON UTENTI.ID = INGRESSI.[ID Utente]

GROUP BY COgnome;

**AAAB7**

SELECT UTENTI.COGNOME, Count(INGRESSI.ID) AS Numero\_Ingressi

FROM UTENTI LEFT JOIN INGRESSI ON UTENTI.ID = INGRESSI.[ID Utente]

GROUP BY UTENTI.COGNOME

ORDER BY Count(INGRESSI.ID);

**Tempo Medio IN**

SELECT Format(Avg(Minuti),"Standard") AS [Tempo medio]

FROM (SELECT Datediff('n', [ORA IN], [ORA OUT]) AS Minuti FROM INGRESSI) AS Minuti;