

Controlli IBO database

Un tipo specifico di controlli aggiuntici per manipolare database mysql, sono gli IBO, che hanno come finalità la manipolazione di dati per la creazione di un report. Poniamo per esempio di avere il seguente form (FIGURA 1) e i seguenti controlli DB (FIGURA 2).

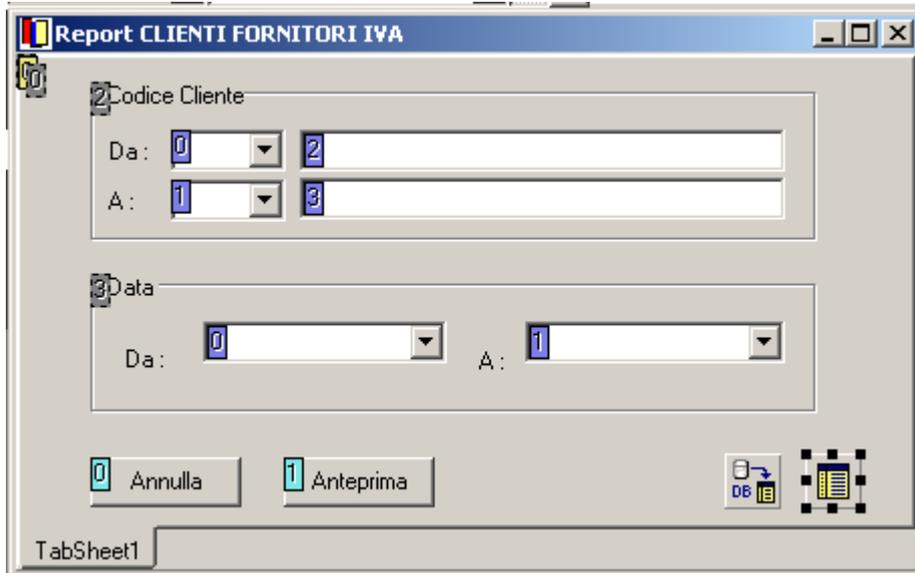


FIGURA 1

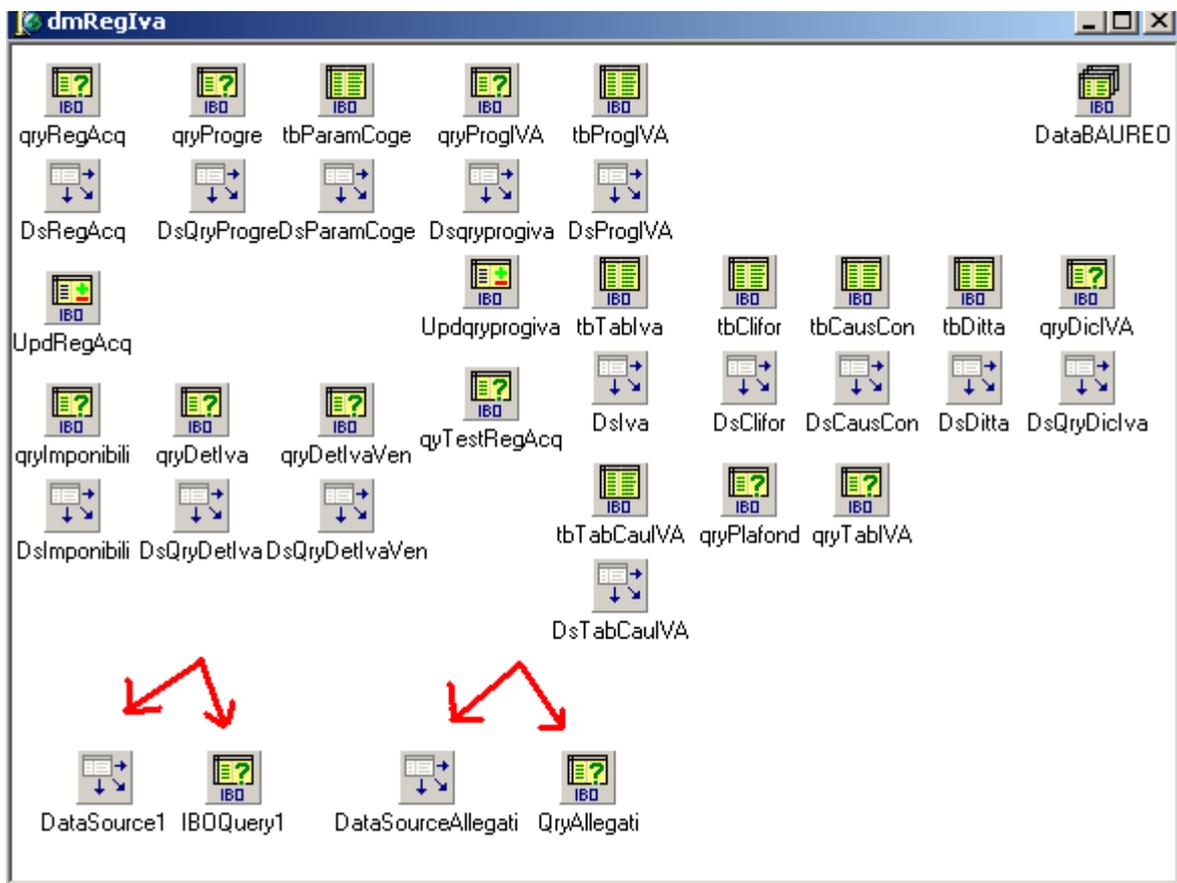


FIGURA 2

DataSource1 e IBOQuery1 li userò per riempire le caselle con i corrispettivi nomi delle aziende in base al loro numero identificativo, mentre DataSourceAllegati e QryAllegati li userò per fare la ricerca all'interno del database, facendo apparire il report con il risultato della medesima.

In figura 1 vi sono 2 controlli PPdbPipeline1 e PPReport1 che si trovano alla voce RBUILDER. Questi sono i controlli per creare il report. Il primo nella form create riceve il datasource (`ppDBPipeline1.DataSource:= dmReglva.datasourceallegati;`) mentre il secondo risponde alla procedura che crea il report :

```
procedure TfmAllegati.BtnPreviewClick(Sender: TObject);
begin
if (apriqry >0) then
  begin
  ppReport1.Template.FileName := GlbOption.PathReport + 'RptAllegati.rtm' ;
  ppReport1.Template.LoadFromFile;
  ppReport1.Device := dvScreen;
  try
    Frmpreview := TFrmpreview.Create(Application);
    Frmpreview.PreviewReport(ppReport1);
  finally
    Frmpreview.Free;
  end;
  end
else
begin
  showmessage('Anteprima non disponibile');
end
end;
```

Nel secondo inoltre bisogna specificare il nome del datapipeline a cui è collegato, in questo caso ppDBPipeline1.

In FIGURA 2 invece vi sono i controlli DataSource e DataSourceAllegati che sono dei normali Data Source di Data Access e poi i controlli IBOQuery1 e Queryallegati che invece sono IBOTdataset.

Per estrarre il nome della ditta dal suo numero di codice utilizzerò routine del genere :

```
procedure TfmAllegati.CBdaChange(Sender: TObject);
begin
// cambiamento
if (CBda.ItemIndex = -1) then
begin
ShowMessage('Devi selezionare qualcosa');
exit;
end
else
begin
cambiotesto1 := True;
//showmessage('ct1');
```

```

dmregiva.IBOQuery1.Close;
dmregiva.IBOQuery1.sql.clear;
dmregiva.IBOQuery1.sql.Add('select COD_CLI,RAG_SOCIALE from CLIFOR where
COD_CLI = ' + CBda.Items.Strings[CBda.ItemIndex] + ''');
dmregiva.IBOQuery1.Open;
if (dmregiva.IBOQuery1.RecordCount > 0 ) then
begin
dmregiva.IBOQuery1.FindFirst;
Edit1.Text := dmregiva.IBOQuery1.FieldByName('RAG_SOCIALE').asString;
end;
end;
end;

```

Routine che sono collegate a DataSource1 e IBOquery1 (FIGURA 2).
Per ottenere invece il report utilizzerò il seguente codice:

```

function TfmAllegati.Apriqry: Integer;
begin
    {
    0
    1 fornitore- fornitore
    2          anno - anno
    3 fornitore - fornitore - anno - anno
    4 fornitore - fornitore - anno
    5 fornitore - fornitore -   - anno
    6 fornitore -   - anno -
    7 fornitore -   -   - anno
    8 fornitore -   - anno - anno
    9   - fornitore - anno - anno
    10   - fornitore - anno -
    11   - fornitore -   - anno
    12   - anno -
    13   - anno
    14 fornitore -
    15   - fornitore -
    }

    dmregiva.qryallegati.sql.clear;
    dmregiva.QryAllegati.Close;

    // 3 fornitore - fornitore - anno - anno
    if ((cambiotesto1 = true) and (cambiotesto2 = true) and (cambiodata1 = true) and
(cambiodata2 = true)) then
    begin
        dmregiva.qryallegati.sql.Add('select C.cod_cli, max(C.rag_sociale),sum(R.imponibile)
IMPO, sum(R.iva) IVA from CLIFOR C left outer join REGIVA R on (R.COD_CLI =
C.COD_CLI) ');
        dmregiva.qryallegati.sql.Add('where (C.COD_CLI >=:c1) and (C.COD_CLI<=:c2) and
(R.data >=:c3) and (R.data <=:c4) group by c.COD_CLI order by c.COD_CLI');
    end;

```

```

dmregiva.QryAllegati.ParamByName('c1').value := cbda.Items[cbda.ItemIndex]; //
attraverso i fornitori
dmregiva.QryAllegati.ParamByName('c2').value := cba.Items[cba.ItemIndex];
dmregiva.QryAllegati.ParamByName('c3').value := Data1.text; // attraverso le date
dmregiva.QryAllegati.ParamByName('c4').value := Data2.text;
Memo1.Text := dmregiva.QryAllegati.sql.text;
dmregiva.QryAllegati.Open;

result:= dmregiva.QryAllegati.RecordCount;
exit;
end;

```


Di cui vi fò vedere solo questa prima parte e tralascio il resto che più o meno si costruisce nella stessa maniera. Da notare che la routine APRIQUERY è richiamata dalla pressione del pulsante Anteprema (FIGURA 1).

QueryAllegati deve avere 4 parametri per potergli passare i 2 codici aziendali e le due date, quindi di default la sua SQL dovrà essere :

```

select C.cod_cli, max(C.rag_sociale),sum(R.imponibile) IMPO, sum(R.iva) IVA from
CLIFOR C left outer join REGIVA R on (R.COD_CLI = C.COD_CLI

```

e dovrà presentare I seguenti parametri :

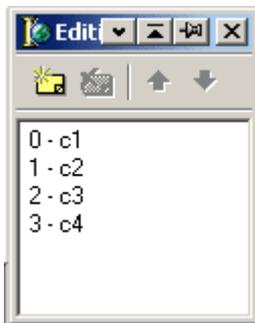


FIGURA 3