

ODBC mit 2 Tabellen B

1. Datenbank (Fussball.mdb)

Tabelle: Mannschaften

ID	Name
1	FC Bayern
2	TSV 1860
3	SpVgg Unterhaching

Beziehungen

Tabelle: Spieler

ID	Mannschaft	Name	Tore
1	1	Oliver Kahn	0
2	1	Mehmet Scholl	1
3	1	Roque Santa Cruz	3
4	1	Michael Ballack	5
5	2	Michael Hofmann	0
6	2	Roman Tyce	0
7	2	Rodrigo Costa	1
8	2	Harald Cerny	2
9	3	Philipp Heerwagen	0
10	3	Christoph Teinert	6
11	3	Necat Aygün	5
12	3	Carsten Sträßer	1

2. ODBC-Datenquelle (Fussball)

3. SDI-Projekt (Fussball)

Datenquelle: Fussball, Tabelle: nur Mannschaften

CFussballSet

Membervariablen

```
long m_ID; // MannschaftsID, Primaerschluesel
CString m_Name; // Mannschaftsname
```

CFussballView

Membervariablen für Steuerelemente

```
IDC_EDIT_ID_MANNCHAFT: m_pSet->m_ID
IDC_EDIT_NAME_MANNCHAFT: m_pSet->m_Name
IDC_LIST_SPIELER: m_cSpieler
```

DoDataExchange

```
...
if (!pDX->m_bSaveAndValidate)
    SpielerlisteFuellen();
```

SpielerlisteFuellen

```
// aktuelle Mannschaftsnummer bestimmen
int nIDMannschaft=m_pSet->m_ID;

// Alle Spieler der akutellen Mannschaft
CString sSQL;
sSQL.Format("SELECT * FROM [Spieler]
WHERE [Mannschaft]=%d ORDER BY [ID]",nIDMannschaft);

// Recordset oeffnen
CSpielerSet Spieler;
Spieler.Open(CRecordset::snapshot, sSQL);

// Liste leeren
m_cSpieler.ResetContent();

// Liste fuellen, falls mind. 1 Datensatz gefunden
if (!(Spieler.IsEOF() && Spieler.IsBOF())){
    Spieler.MoveFirst();
    while(!Spieler.IsEOF()){
        m_cSpieler.AddString(Spieler.m_Name);
        Spieler.MoveNext();
    }
}

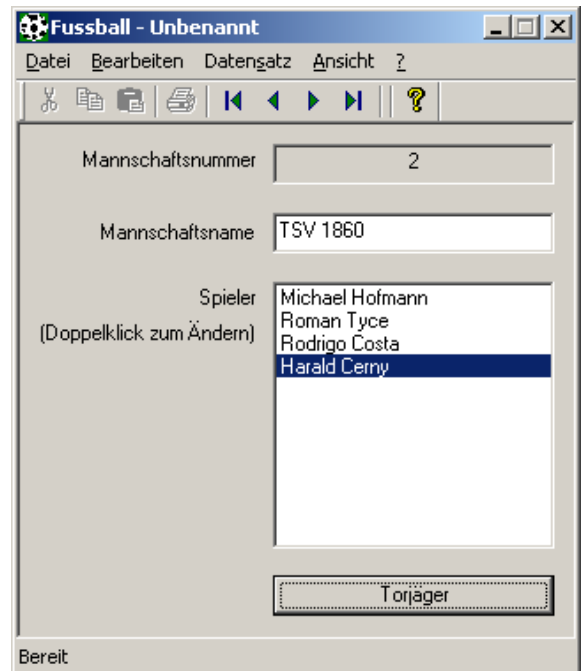
// Recordset schliessen
Spieler.Close();
```

CSpielerSet

Neue MFC-Klasse, Basisklasse: CRecordset, Datenquelle: Fussball
Tabelle: nur Spieler

Membervariablen

```
long m_ID; // SpielerID
long m_Mannschaft; // Fremdschluesel
CString m_Name; // Spielername
long m_Tore; // Tore
```



OnDbclickListSpieler

```

// Nummer des ausgewaehlten Spielers innerhalb der Liste ermitteln
int i = m_cSpieler.GetCurSel();

// Alle Spieler der aktuellen Mannschaft
CString sSQL;
sSQL.Format("SELECT * FROM [Spieler] WHERE [Mannschaft]=%d", m_pSet->m_ID);

// Recordset oeffnen
CSpielerSet Spieler;
Spieler.Open(CRecordset::dynaset,sSQL);

// zum entsprechenden Spieler springen
Spieler.SetAbsolutePosition(i+1);

// Falls in der Zwischenzeit die Daten veraendert wurden,
// wird eventuell ein falscher Datensatz angezeigt!

// Spielerfenster
CSpielerDlg dlg;

// alte Daten uebergeben
dlg.m_sName = Spieler.m_Name;
dlg.m_sTore.Format("%d",Spieler.m_Tore);

// Spielerfenster anzeigen
if(dlg.DoModal()==IDOK) {

    // neue Spielerdaten speichern
    Spieler.Edit();
    Spieler.m_Name = dlg.m_sName;
    Spieler.m_Tore = atoi(dlg.m_sTore);
    Spieler.Update();

    // Liste leeren
    m_cSpieler.ResetContent();

    // Liste fuellen
    Spieler.MoveFirst();
    while(!Spieler.IsEOF()){
        m_cSpieler.AddString(Spieler.m_Name);
        Spieler.MoveNext();
    }

    // aktuellen Eintrag wieder markieren
    m_cSpieler.SetCurSel(i);
}

// Recordset schliessen
Spieler.Close();

```

OnButtonTorjaeger

```

// Torjaeger-Dialog anzeigen
CTorjaegerDlg dlg;
dlg.DoModal();

```

CTorjaegerDlg**OnInitDialog**

```

// Alle Spieler sortiert nach Toren
CString sSQL;
sSQL="SELECT * FROM [Mannschaften],[Spieler]
WHERE [Mannschaften].[ID]=[Spieler].[Mannschaft]
ORDER BY [Spieler].[Tore] DESC";

// Recordset oeffnen
CGesamtSet Torjaeger;
Torjaeger.Open(CRecordset::snapshot,sSQL);

// Liste fuellen, falls mindestens 1 Datensatz vorhanden
if(!(Torjaeger.IsBOF() && Torjaeger.IsEOF())){
    Torjaeger.MoveFirst();
    while(!Torjaeger.IsEOF()){
        CString sTorjaeger;
        sTorjaeger.Format("%d  %s  (%s)",
            Torjaeger.m_Tore, Torjaeger.m_Name2, Torjaeger.m_Name);
        m_cTorjaeger.AddString(sTorjaeger);
        Torjaeger.MoveNext();
    }
}

// Recordset schliessen
Torjaeger.Close();

```

