

# ISAMEXPERT Hilfe-Index

[Isam-Record](#)

[Key-Proc](#)

[Index-Descriptor](#)

[Zielverzeichnis](#)

[Tabelle](#)

[Editor](#)

[Browser](#)

[Editor-Einstellungen](#)

[Browser-Optionen](#)

[DBase-Export](#)

[DBase-Import](#)

[TYP-Datei](#)

# Isam-Record

Geben Sie den zu bearbeitenden Record im üblichen Pascal-Format ein.

Beispiel:

TYPE

```
TESTRECORD = RECORD
  NAME: STRING[30];
  VORNAME: STRING[30];
  STRASSE: STRING[25];
  PLZ: STRING[5];
  ORT: STRING[25];
  ALTER: INTEGER;
END;
```

Sie können den Record auch aus einer ASCII-Datei einlesen, indem Sie den TYP-DATEI-Button drücken. Die ASCII-Datei sollte die Endung TYP haben,

Beispiel: TEST.TYP

## KEY-PROC

Die Schlüssel der ISAM-Datei legen Sie bitte mit folgender Eingabe fest:

```
F(NAME,20);
```

```
F(PLZ,5);
```

```
FORMDEZSTR(ALTER,10);
```

Diese Eingabe erzeugt eine Keyproc mit 3 Schlüsseln, den Schlüssel 1 mit den ersten 20 Zeichen aus dem Feld NAME und den Schlüssel 2 mit den ersten 5 Zeichen aus dem Feld PLZ und dem dritten Schlüssel der aus dem Feld Alter einen String mit der Länge 30 macht.

Die Felder werden auf jeden Fall mit F(...) auf die angegebene Länge mit Leerzeichen aufgefüllt.

Achten Sie darauf, dass diese Felder im Index-Descriptor genauer beschrieben werden müssen.

Auch aus mehreren Feldern zusammengesetzte Schlüssel sind möglich.

Beispiel:

F(NAME1,20)+F(NAME2,1) in der KeyProc und die entsprechenden Beschreibungen IID[1].KeyL:= 21; IID[1].AllowDupK:= FALSE; im Index-Descriptor erzeugen einen Schlüssel, der die Datei nach den ersten 20 Zeichen aus dem Feld NAME1 und dem ersten Zeichen aus dem Feld NAME2 sortiert.

# Index-Descriptor

Diese Eingabe beschreibt die Schlüssel der Datei genauer.

Beispiel:

```
IID[1].KeyL:= 20; IID[1].DupK:= FALSE
```

```
IID[2].KeyL:= 5; IID[2].DupK:= TRUE;
```

```
IID[3].KeyL:= 10; IID[3].DupK:= TRUE;
```

erzeugt den ersten Schlüssel als eindeutigen Schlüssel mit der Länge von 20 Zeichen und den zweiten Schlüssel als mehrdeutigen Schlüssel mit der Länge von 5 Zeichen und den dritten Schlüssel, ebenfalls mehrdeutig mit der Länge 10.

Auch aus mehreren Feldern zusammengesetzte Schlüssel sind möglich.

Beispiel:

F(NAME1,20)+F(NAME2,1) in der KeyProc und die entsprechenden Beschreibungen

```
IID[1].KeyL:= 21; IID[1].AllowDupK:= FALSE;
```

 im Index-Descriptor erzeugen einen

Schlüssel, der die Datei nach den ersten 20 Zeichen aus dem Feld NAME1 und dem ersten Zeichen aus dem Feld NAME2 sortiert.

## Zielverzeichnis

Hier legen Sie das Verzeichnis fest, in das der ISAM-EXPERTE die zu erzeugenden Files (\*.PAS und \*.DFM) schreibt, indem Sie den Schalter neben dem Eingabefeld betätigen und ein bereits bestehendes Verzeichnis auswählen.

Das Anlegen von neuen Verzeichnissen ist mit dem Experten nicht möglich. Nutzen Sie bitte hierzu den WINDOWS-EXPLORER (WIN95) oder den DATEIMANAGER (WIN 3.x) oder die entsprechenden MS-DOS Befehle.

# Tabelle

Hier geben Sie den Tabellennamen an, also z.B. TEST für die Tabelle TEST.DAT.

Der Extent (DAT) braucht nicht mit angegeben zu werden. Die Tabelle braucht noch nicht zu existieren. Sie kann später vom Programm erzeugt werden.

Der Befehl TABLE1.CREATETABLE im Sourcecode (vielleicht als Reaktion auf einen gedrückten Speedbutton oder einen Meneubefehl) erzeugt die Tabelle neu.

Sie muss aber vorher die üblichen Properties TABLENAME, ISAMRECORD, ISAMIID und ISAMKEYPROC erhalten haben. Sie sollten sich mit dem Experten ein Beispiel erzeugen, damit diese Properties bereits gefüllt sind. Dann können Sie in das Formular einen Speedbutton einfügen und seine Ereignis-Routine ONCLICK mit dem Befehl TABLE.CREATETABLE füllen.

## Editor

Hier wird der Name des Editor-Sourcecodes eingegeben, indem Sie den Schalter neben dem Eingabefeld betätigen und einen Dateinamen eingeben oder auswählen. Die Eingabe TEST1 erzeugt z.B. den Editor-Sourcecode in der Datei TEST1.PAS und das dazugehörige Formular in TEST1.DFM. Eventuell vorhandene Dateien mit demselben Namen werden überschrieben.

Sie sollten darauf achten, daß der Browser - Sourcecode auf jeden fall einen anderen Namen als der Editor erhält.

## Browser

Hier wird der Name des Browser-Sourcecodes eingegeben, indem Sie den Schalter neben dem Eingabefeld betätigen und einen Dateinamen eingeben oder auswählen. Die Eingabe TEST2 erzeugt z.B. den Browser-Sourcecode in der Datei TEST2.PAS und das dazugehörige Formular in TEST2.DFM. Eventuell vorhandene Dateien mit demselben Namen werden überschrieben.

Sie sollten darauf achten, daß der Editor - Sourcecode auf jeden fall einen anderen Namen als der Editor erhält.



## Editor-Einstellungen

Sie können auswählen, ob die Labels neben oder über den Eingabefeldern platziert werden. Außerdem können Sie festlegen, ob alle Eingabefelder die gleiche Länge erhalten oder ob die Länge der Eingabefelder von der Größe der Daten abhängig sein soll, also z.B. ein breiteres Eingabefeld für das Feld `NAME : STRING[20]` und ein kleineres Eingabefeld für das Feld `PLZ: STRING[5]`;

Die Position der Eingabefelder wird zunächst automatisch festgelegt. Sie können sie aber anschließend mit dem Delphi-Editor wie üblich verschieben.

## Browser-Optionen

Hier legen Sie fest, ob der Browser das Hauptfenster des Programms sein soll. Außerdem können Sie die Schriftart für den Browser festlegen. Die Eingabefelder INIDATEINAME und INIBROWSERNAME sind zur Zeit noch ohne Funktion, können also ignoriert werden.

## DBase-Export

Wenn Sie diese Option aktivieren, wird eine Unit erzeugt, in der die DBase-Struktur passend zum angegebenen Record erzeugt wird. Die Struktur-Datei wird automatisch in das Projekt eingebunden.

Sie sollten daran denken, daß Sie dann bei der Auslieferung des Programms die IDAPI-DLLs (angegeben in der Datei DEPLOY.TXT im Delphi-Verzeichnis) mit ausliefern müssen.

Achtung: Beim Typ WORD oder LONGINT wird normalerweise ein DBase-Feld vom Typ FldINT16 bzw. FldINT32 erzeugt. Die ISAM speichert jedoch auch Datumsfelder unter dem Typ LONGINT. Wenn der Experte bei einem Datumsfeld den Typ FldDATE erzeugen soll, müssen Sie im Record das Wort DATUM angeben.

Beispiel:

FIELD1: LONGINT	erzeugt ein DBase-Feld vom Typ FLDINT32
GEBDATUM: LONGINT DATUM	erzeugt ein DBase-Feld vom Typ FLDDATE, weil das Wort im Text enthalten ist.
FIELD2: LONGINT {DATUM} DATUM	erzeugt ebenfalls ein Feld vom Typ FLDDATE, weil das Wort in der Zeile steht.
FIELD3: WORD	erzeugt ein Feld vom Typ FLDINT16
FIELD4: WORD {DATUM}	erzeugt jedoch ein Feld vom Typ FLDDATE.

In der Browser-Unit wird ausserdem eine Prozedur xxxxDBASEEXPORTPROC erzeugt, die den Transfer der Daten übernimmt.

# DBase-Import

Wenn Sie diese Option aktivieren, wird eine Unit erzeugt, in der die DBase-Struktur passend zum angegebenen Record erzeugt wird. Die Struktur-Datei wird automatisch in das Projekt eingebunden.

Sie sollten daran denken, daß Sie dann bei der Auslieferung des Programms die IDAPI-DLLs (angegeben in der Datei DEPLOY.TXT im Delphi-Verzeichnis) mit ausliefern müssen.

Achtung: Beim Typ WORD oder LONGINT wird normalerweise ein DBase-Feld vom Typ FldINT16 bzw. FldINT32 erzeugt. Die ISAM speichert jedoch auch Datumsfelder unter dem Typ LONGINT. Wenn der Experte bei einem Datumsfeld den Typ FldDATE erzeugen soll, müssen Sie im Record das Wort DATUM angeben.

Beispiel:

FIELD1: LONGINT	erzeugt ein DBase-Feld vom Typ FLDINT32
GEBDATUM: LONGINT DATUM	erzeugt ein DBase-Feld vom Typ FLDDATE, weil das Wort im Text enthalten ist.
FIELD2: LONGINT {DATUM} DATUM	erzeugt ebenfalls ein Feld vom Typ FLDDATE, weil das Wort in der Zeile steht.
FIELD3: WORD	erzeugt ein Feld vom Typ FLDINT16
FIELD4: WORD {DATUM}	erzeugt jedoch ein Feld vom Typ FLDDATE.

In der Browser -Unit wird ausserdem eine Prozedur xxxxDBASEIMPORTPROC erzeugt, die den Transfer der Daten übernimmt.

# TYP-Datei

Die TYP-Datei ist eine reine ASCII-Datei, in der Sie zur Vereinfachung der Arbeit mit dem Experten die Isam-Record-Struktur speichern können.

Beispiel:

type

Kundenrecord = Record

```
Dummy      : LongInt;
Kundennummer : String[ 6];
Name        : String[32];
Strasse     : String[32];
Plz         : String[ 8];
Ort         : String[32];
LA          : ARRAY[1..5] of String[32];
GebDatum    : Word;
```

end;

```
(*{KEY BEGIN}
  {KEYNO=1} F(KUNDENNUMMER,6);  {KEYLEN=6}  {KEYNAME=KUNDENNUMM}
{DUPKEY=FALSE}
{KEY END}*)
```

Der RECORD wird in der üblichen PASCAL-Syntax angegeben. Die Zeilen zwischen KEY BEGIN und KEY END erzeugen den Index-Descriptor und die KeyProc für die ISAM-Datei.

