

BHT Berliner Hochschule für Technik

Wirtschaftsinformatik 2

LE 07 – Recordsets als Datenbankschnittstelle

Prof. Dr. Thomas Off
<http://www.ThomasOff.de/lehre/beuth/wi2>

BHT

Hinweis für Online-Lehrveranstaltung

Studierende bitten,

- Übungsdatenbank herunterzuladen (Link in Chat posten) und auszupacken
- Übungsfolien zu öffnen (Link in Chat posten)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 2

BHT

Ziel

- Aufgaben der Datenbankschnittstelle kennenlernen
- Verstehen der Konzeptes eines Recordsets
- Arbeiten mit Recordsets und Anwendung in MS Access
 - Lesen von Daten mit Recordsets
 - Navigation in Recordsets
 - Suchen und Finden von Daten in Recordsets
 - Ändern und Hinzufügen von Daten mit Recordsets

Wirtschaftsinformatik 2 - LE 07 - Recordsets 3

BHT

Einordnung

Wirtschaftsinformatik 2 - LE 07 - Recordsets 4

Inhalt BÄT

Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle
Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen
Fallstricke
Zusammenfassung und Ausblick

Wirtschaftsinformatik 2 - LE 07 - Recordsets 5

Inhalt BÄT

Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle
Recordsets


- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen
Fallstricke
Zusammenfassung und Ausblick

Wirtschaftsinformatik 2 - LE 07 - Recordsets 6

Rückblick  BÄT

Wirtschaftsinformatik 2 - LE 07 - Recordsets 7

Rückblick  BÄT


Datensätze auswählen mit SQL

```
SELECT vorname, name FROM kunden WHERE  
ort = 'Berlin';
```

Ausgangsrelation

Kunden	KndNr	Name	Vorname	Ort
	123	Albers	Willi	Aachen
	234	Boehrs	Vera	Berlin
	345	Dinkel	Ulrike	Berlin
	456	Dinkels	Thomas	Berlin
	567	Esser	Thomas	Dessau

Wirtschaftsinformatik 2 - LE 07 - Recordsets 8

Rückblick  B&T

Datensätze auswählen mit SQL

```
SELECT vorname, name FROM kunden WHERE
ort='Berlin';
```


Ausgangsrelation

Kunden	KndNr	Name	Vorname	Ort
	123	Albers	Willi	Aachen
	234	Boehrs	Vera	Berlin
	345	Dinkel	Ulrike	Berlin
	456	Dinkels	Thomas	Berlin
	567	Esser	Thomas	Dessau

↑ Projektion

← Restriktion/ Selektion

Wirtschaftsinformatik 2 - LE 07 - Recordsets 9

Rückblick  B&T

Datensätze auswählen mit SQL


```
SELECT vorname, name FROM kunden WHERE
ort='Berlin';
```

Ausgangsrelation

Kunden	KndNr	Name	Vorname	Ort
	123	Albers	Willi	Aachen
	234	Boehrs	Vera	Berlin
	345	Dinkel	Ulrike	Berlin
	456	Dinkels	Thomas	Berlin
	567	Esser	Thomas	Dessau

↓ Ergebnis

Wirtschaftsinformatik 2 - LE 07 - Recordsets 10

Rückblick  B&T

Datensätze auswählen mit SQL

```
SELECT vorname, name FROM kunden WHERE
ort='Berlin';
```

Ausgangsrelation


Kunden	KndNr	Name	Vorname	Ort
	123	Albers	Willi	Aachen
	234	Boehrs	Vera	Berlin
	345	Dinkel	Ulrike	Berlin
	456	Dinkels	Thomas	Berlin
	567	Esser	Thomas	Dessau

↓ Ergebnis

Vorname	Name
Vera	Boehrs
Ulrike	Dinkel
Thomas	Dinkels

Reihenfolge der Spalten, wie in SELECT-Anweisung angegeben, d.h. erst Vorname, dann Name!

Wirtschaftsinformatik 2 - LE 07 - Recordsets 11

Rückblick  B&T

Datensätze einfügen mit SQL


```
INSERT INTO kunden(Vorname, Name, KndNr)
VALUES ('Simon', 'Jakob', 345);
```

Ausgangsrelation

Kunden	KndNr	Name	Vorname
	123	Albers	Willi
	234	Boehrs	Vera

↓ Ergebnisrelation

Wirtschaftsinformatik 2 - LE 07 - Recordsets 12

Rückblick  **BÄT**

Datensätze einfügen mit SQL

```
INSERT INTO kunden(Vorname, Name, KndNr)
VALUES ('Simon', 'Jakob', 345);
```

Ausgangsrelation


Kunden	KndNr	Name	Vorname
	123	Albers	Willi
	234	Boehrs	Vera

`INSERT INTO kunden(Vorname, Name, KndNr) VALUES ('Simon', 'Jakob', 345);`

Ergebnisrelation

Kunden	KndNr	Name	Vorname
	123	Albers	Willi
	234	Boehrs	Vera
	345	Jakob	Simon

Wirtschaftsinformatik 2 - LE 07 - Recordsets 13

Rückblick  **BÄT**

Ändern von Datensätzen mit SQL

```
UPDATE kunden SET name='Albers'
WHERE kndnr=234;
```


Ausgangsrelation

Kunden	KndNr	Name	Vorname
	123	Albers	Willi
	234	Boehrs	Vera
	345	Muster	Michael


`UPDATE kunden SET name='Albers' WHERE kndnr=234;`

Ergebnisrelation

Kunden	KndNr	Name	Vorname
	123	Albers	Willi
	234	Albers	Vera
	345	Muster	Michael



Wirtschaftsinformatik 2 - LE 07 - Recordsets 14

Rückblick  **BÄT**

Ändern von Datensätzen mit SQL

```
UPDATE kunden SET name='Albers'
WHERE kndnr=234;
```

Ausgangsrelation


Kunden	KndNr	Name	Vorname
	123	Albers	Willi
	234	Boehrs	Vera
	345	Muster	Michael

`UPDATE kunden SET name='Albers' WHERE kndnr=234;`

Ergebnisrelation

Kunden	KndNr	Name	Vorname
	123	Albers	Willi
	234	Albers	Vera
	345	Muster	Michael

Wirtschaftsinformatik 2 - LE 07 - Recordsets 15

Rückblick  **BÄT**

Löschen von Datensätzen mit SQL

```
DELETE FROM kunden WHERE name='Albers';
```


Ausgangsrelation

Kunden	KndNr	Name	Vorname
	123	Albers	Willi
	234	Albers	Vera
	345	Muster	Michael


`DELETE FROM kunden WHERE name='Albers';`

Ergebnisrelation

Kunden	KndNr	Name	Vorname
	345	Muster	Michael



Wirtschaftsinformatik 2 - LE 07 - Recordsets 16

Rückblick  BÄT

Löschen von Datensätzen mit SQL

```
DELETE FROM kunden WHERE name='Albers';
```

Ausgangsrelation


Kunden	KndNr	Name	Vorname
	123	Albers	Willi
	234	Albers	Vera
	345	Muster	Michael

`DELETE FROM kunden WHERE name='Albers';`

Ergebnisrelation

Kunden	KndNr	Name	Vorname
	345	Muster	Michael

Wirtschaftsinformatik 2 - LE 07 - Recordsets 17

Rückblick  BÄT

Datensätze aus Tabellen mit INNER JOIN verbinden

```
SELECT produkte.pid, produkte.name,
       lieferanten.id, lieferaten.firma
FROM produkte INNER JOIN lieferaten
ON produkte.lid = lieferaten.id;
```


Ausgangsrelationen

Produkte	PID	Name	LID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	

`SELECT produkte.pid, produkte.name, lieferanten.id, lieferaten.firma FROM produkte INNER JOIN lieferaten ON produkte.lid = lieferaten.id;`

Lieferanten	ID	Firma	Ort
	987	Müller AG	Berlin
	876	Meier GmbH	Potsdam
	765	Bach&Sohn	Cottbus

Wirtschaftsinformatik 2 - LE 07 - Recordsets 18

Rückblick  BÄT

Datensätze aus Tabellen mit INNER JOIN verbinden

```
SELECT produkte.pid, produkte.name,
       lieferanten.id, lieferaten.firma
FROM produkte INNER JOIN lieferaten
ON produkte.lid = lieferaten.id;
```

Ausgangsrelationen


Produkte	PID	Name	LID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	

`SELECT produkte.pid, produkte.name, lieferanten.id, lieferaten.firma FROM produkte INNER JOIN lieferaten ON produkte.lid = lieferaten.id;`

Ergebnis

PID	Name	ID	Firma
123	Multi AB	987	Müller AG
234	Flexi 123	987	Müller AG
345	Mega+	876	Meier GmbH

Wirtschaftsinformatik 2 - LE 07 - Recordsets 19

Rückblick  BÄT

Datensätze aus Tabellen mit LEFT OUTER JOIN verbinden

```
SELECT produkte.pid, produkte.name,
       lieferanten.id, lieferaten.firma
FROM produkte LEFT JOIN lieferaten
ON produkte.lid = lieferaten.id;
```


Ausgangsrelationen

Produkte	PID	Name	LID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	

`SELECT produkte.pid, produkte.name, lieferanten.id, lieferaten.firma FROM produkte LEFT JOIN lieferaten ON produkte.lid = lieferaten.id;`

Lieferanten	ID	Firma	Ort
	987	Müller AG	Berlin
	876	Meier GmbH	Potsdam
	765	Bach&Sohn	Cottbus

Wirtschaftsinformatik 2 - LE 07 - Recordsets 20

Rückblick  **BÄT**

Datensätze aus Tabellen mit LEFT OUTER JOIN verbinden

```
SELECT produkte.pid, produkte.name,
       lieferanten.id, lieferaten.firma
FROM produkte LEFT JOIN lieferaten
ON produkte.lid = lieferanten.id;
```

Ausgangsrelationen

Produkte	PID	Name	IID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	


Lieferanten	ID	Firma	Ort
	987	Müller AG	Berlin
	876	Meier GmbH	Potsdam
	765	Bach&Sohn	Cottbus

```
SELECT produkte.pid, produkte.name,
       lieferanten.id, lieferaten.firma
FROM produkte LEFT JOIN lieferaten
ON produkte.lid = lieferanten.id;
```

Ergebnis

PID	Name	ID	Firma
123	Multi AB	987	Müller AG
234	Flexi 123	987	Müller AG
345	Mega+	876	Meier GmbH
456	Super XL		

Wirtschaftsinformatik 2 - LE 07 - Recordsets 21

Rückblick  **BÄT**

Datensätze aus Tabellen mit RIGHT OUTER JOIN verbinden

```
SELECT produkte.pid, produkte.name,
       lieferanten.id, lieferaten.firma
FROM produkte RIGHT JOIN lieferaten
ON produkte.lid = lieferanten.id;
```


Ausgangsrelationen

Produkte	PID	Name	IID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	

Lieferanten	ID	Firma	Ort
	987	Müller AG	Berlin
	876	Meier GmbH	Potsdam
	765	Bach&Sohn	Cottbus

```
SELECT produkte.pid, produkte.name,
       lieferanten.id, lieferaten.firma
FROM produkte RIGHT JOIN lieferaten
ON produkte.lid = lieferanten.id;
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 22

Rückblick  **BÄT**

Datensätze aus Tabellen mit RIGHT OUTER JOIN verbinden

```
SELECT produkte.pid, produkte.name,
       lieferanten.id, lieferaten.firma
FROM produkte RIGHT JOIN lieferaten
ON produkte.lid = lieferanten.id;
```

Ausgangsrelationen

Produkte	PID	Name	IID
	123	Multi AB	987
	234	Flexi 123	987
	345	Mega+	876
	456	Super XL	


Lieferanten	ID	Firma	Ort
	987	Müller AG	Berlin
	876	Meier GmbH	Potsdam
	765	Bach&Sohn	Cottbus

```
SELECT produkte.pid, produkte.name,
       lieferanten.id, lieferaten.firma
FROM produkte RIGHT JOIN lieferaten
ON produkte.lid = lieferanten.id;
```

Ergebnis

PID	Name	ID	Firma
123	Multi AB	987	Müller AG
234	Flexi 123	987	Müller AG
345	Mega+	876	Meier GmbH
		765	Bach&Sohn

Wirtschaftsinformatik 2 - LE 07 - Recordsets 23

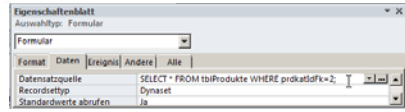
Rückblick  **BÄT**

SQL in MS Access

- SQL-Abfragen können als Datenquelle für Formulare verwendet werden
- Einfügen, Ändern und Löschen mit SQL mit vordefinierter VBA-Funktionen

```
CurrentDB.Execute ("<SQL-Anweisung> ")
```

- Grafische Abfragen und SQL
 - werden von MS Access in SQL übersetzt
 - SQL-Ansicht zeigt das generierte SQL
 - Ausführung von SQL direkt über SQL-Ansicht eines leeren Abfrageentwurfs oder Abfrage eines entsprechenden Abfragetyps möglich




Wirtschaftsinformatik 2 - LE 07 - Recordsets 24

Inhalt BÄT

Ziel und Einordnung
Rückblick
 Ausgangssituation
Datenbankschnittstelle
Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen
Fallstricke
Zusammenfassung und Ausblick



Wirtschaftsinformatik 2 - LE 07 - Recordsets 25

Inhalt BÄT

Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle
Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen
Fallstricke
Zusammenfassung und Ausblick

Wirtschaftsinformatik 2 - LE 07 - Recordsets 26

Ausgangssituation BÄT

Benutzeroberfläche der Anwendung

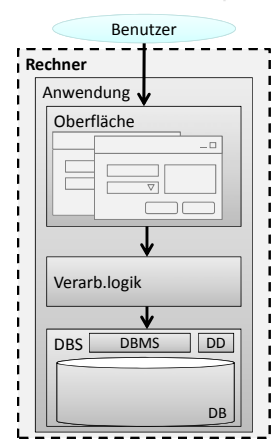
- Formulare mit Feldern, Schaltfläche usw.
- Nutzung der Verarbeitungslogik

Verarbeitungslogik

- Verarbeitung der Daten in der Oberfläche vor ihrer Speicherung
- Verarbeitung aus der DB geladener Daten vor ihrer Anzeige

Datenbank

- mit Tabellen (Relationen)
- und Daten (Tupeln)



Wirtschaftsinformatik 2 - LE 07 - Recordsets 27

Ausgangssituation BÄT

Benutzeroberfläche der Anwendung

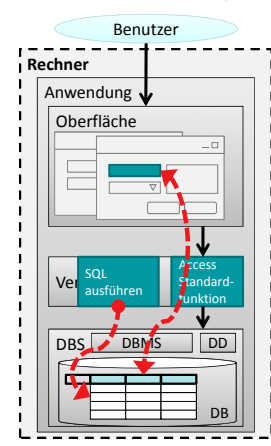
- Formulare mit Feldern, Schaltfläche usw.
- Nutzung der Verarbeitungslogik

Verarbeitungslogik

- bisher lediglich durch Nutzung der Standard-Access-Funktionen
 - zur Anzeige in Oberfläche
 - zum Ausführen von Abfragen/SQL

Datenbank

- mit Tabellen (Relationen)
- und Daten (Tupeln)



Wirtschaftsinformatik 2 - LE 07 - Recordsets 28

Toll! BÄT

SQL: SELECT,
UPDATE, INSERT,
DELETE

Wirtschaftsinformatik 2 - LE 07 - Recordsets 29

Toll! BÄT

Aber...

- Geburtsdatum des Kunden lesen, sein Alter errechnen und in einer Variable speichern
- Geschlecht des Kunden lesen, passende Anrede "Frau" bzw. "Herr" mit dem Namen verbinden und in Messagebox ausgeben
- ...

Überhaupt irgendwas aus der Datenbank lesen, um es in einer Variable zu speichern.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 30

Ausgangssituation BÄT

Benutzeroberfläche der Anwendung

- Formulare mit Feldern, Schaltfläche usw.
- Nutzung der Verarbeitungslogik

Verarbeitungslogik

- bisher lediglich durch Nutzung der Standard-Access-Funktionen
 - zur Anzeige in Oberfläche
 - zum Ausführen von Abfragen/SQL

Datenbank

- mit Tabellen (Relationen)
- und Daten (Tupeln)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 31

Ausgangssituation BÄT

Benutzeroberfläche der Anwendung

- Formulare mit Feldern, Schaltfläche usw.
- Nutzung der Verarbeitungslogik

Verarbeitungslogik

- jetzt Nutzung einer Möglichkeit direkt auf die Daten der zugreifen und
- Daten zu verarbeiten, ohne eine Verbindung mit der Oberfläche herstellen zu müssen

Datenbank

- mit Tabellen (Relationen)
- und Daten (Tupeln)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 32

Ausgangssituation

Benutzeroberfläche der Anwendung

- Formulare mit Feldern, Schaltfläche usw.
- Nutzung der Verarbeitungslogik

Verarbeitungslogik

- jetzt Nutzung einer Möglichkeit direkt auf die Daten der zuzugreifen und
- Daten zu verarbeiten, ohne eine Verbindung mit der Oberfläche herstellen zu müssen

Datenbank

- mit Tabellen (Relationen)
- und Daten (Tupeln)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 33

Inhalt

Ziel und Einordnung

Rückblick

Ausgangssituation

Datenbankschnittstelle

Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen

Fallstricke

Zusammenfassung und Ausblick

Wirtschaftsinformatik 2 - LE 07 - Recordsets 34

Inhalt

Ziel und Einordnung

Rückblick

Ausgangssituation

Datenbankschnittstelle

Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen

Fallstricke

Zusammenfassung und Ausblick

Wirtschaftsinformatik 2 - LE 07 - Recordsets 35

Programmierschnittstelle zur Datenbank

Programmiersprache

- dient der Programmentwicklung
- prozedurale Sprache, z.B. mit Variablen die einen Bereich des Arbeitsspeichers referenzieren und einen bestimmten Datentyp haben

```

Programmiersprache
Dim strName _
  As String
Let strName = "...
' ...
    
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 36

Programmierschnittstelle zur Datenbank BÄT

Datenbanksprache

- dient u.a. der Abfrage, Manipulation von Daten
- Konzepte relationaler Datenbanken: Tabellen, Datensätze, Spalten, Schlüssel

Programmiersprache

```
Dim strName _
  As String
Let strName = ".."
'...
```

Datenbanksprache

```
SELECT * FROM
Kunden
WHERE Stadt='Köln'
```


Wirtschaftsinformatik 2 - LE 07 - Recordsets37

Programmierschnittstelle zur Datenbank BÄT

Um innerhalb einer Programmiersprache auf die Datenbank zugreifen zu können, ...

Programmiersprache

```
Dim strName _
  As String
Let strName = ".."
'...
```



Datenbanksprache

```
SELECT * FROM
Kunden
WHERE Stadt='Köln'
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets38

Programmierschnittstelle zur Datenbank BÄT

Um innerhalb einer Programmiersprache auf die Datenbank zugreifen zu können, muss eine Verbindung zwischen Sprachelementen der Programmiersprache und Elementen der Datenbanksprache existieren:

Programmiersprache

```
Dim strName _
  As String
Let strName = ".."
'...
```

↔

Datenbanksprache

```
SELECT * FROM
Kunden
WHERE Stadt='Köln'
```

Datenbankschnittstelle

Wirtschaftsinformatik 2 - LE 07 - Recordsets39

Funktionen einer Datenbankschnittstelle BÄT

Funktionen einer Datenbankschnittstelle

- Übersetzen der Befehle einer Programmiersprache in Befehle der Datenbanksprache
- Übersetzen des Ergebnisses einer Datenbankabfrage in Elemente einer Programmiersprache
- Bereitstellen des Funktionsumfangs für
 - Abfragen von Datensätzen
 - Auswählen nach bestimmten Kriterien
 - Navigation über die gefundenen Datensätze
 - Einfügen von neuen Datensätzen
 - Ändern vorhandener Datensätze
 - Löschen vorhandener Datensätze

→ **Wir lernen diese Grundkonzepte anhand von VBA-Recordsets kennen**

Wirtschaftsinformatik 2 - LE 07 - Recordsets40

Inhalt

Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle
Recordsets
– Was sind Recordsets
– Mit Recordsets Daten lesen
– Navigation im Recordset per Zeiger
– Suchen und Finden im Recordset
– Ändern und Hinzufügen im Recordset
– Kombination mit grafischen Abfragen und SQL
Weitere VBA-Funktionen
Fallstricke
Zusammenfassung und Ausblick



BÄT

Wirtschaftsinformatik 2 - LE 07 - Recordsets 41

Was sind Recordsets?

Recordset

BÄT

Datenbanken - LE 07 - Recordsets 42

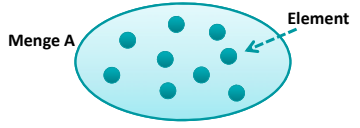
Was sind Recordsets?

Recordset

Set = Menge

Menge

- fasst Elemente zusammen
- Sonderform „leere Menge“ und unendliche Menge



BÄT

Datenbanken - LE 07 - Recordsets 43

Was sind Recordsets?

Recordset

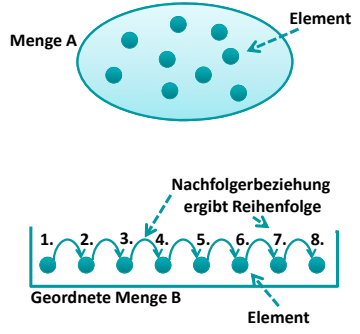
Set = Menge

Menge

- fasst Elemente zusammen
- Sonderform „leere Menge“ und unendliche Menge

Geordnete Menge

- fasst Elemente zusammen
- definiert eine „Reihenfolge“ zwischen Elementen (z.B. durch Nachfolgerbeziehung)
- entspricht mathematischer Ordnungsrelation



BÄT

Datenbanken - LE 07 - Recordsets 44

Was sind Recordsets? BÄT

Recordset

Record = Datensatz

Datensatz

- repräsentiert etwas, z.B. eine Person, ein Produkt mit Eigenschaften
- fasst einzelne Datenfelder zusammen, z.B. Name, Vorname, Geburtsdatum
- wird als Zeile in Tabellen

Knd234	Müller	Mike	Köln
Knd345	Müller	Sabine	Bonn
Knd456	Schmidt	Susi	Berlin

Prd01	Multi XY	150 €
Prd02	Flexi AC	75 €

Datenbanken - LE 07 - Recordsets 45

Was sind Recordsets? BÄT

Recordset

Recordset = geordnete Menge gleichartiger Datensätze

Beispiele:

- Recordset von Personen

Knd234	Müller	Mike	Köln
Knd345	Müller	Sabine	Bonn
Knd456	Schmidt	Susi	Berlin
Knd678	Schmidt	Hans	Bremen
Knd789	Yilmaz	Ali	Berlin

Datenbanken - LE 07 - Recordsets 46

Was sind Recordsets? BÄT

Recordset

Recordset = geordnete Menge gleichartiger Datensätze

Beispiele:

- Recordset von Personen
- Recordset von Produkten

Prd01	Multi XY	150 €
Prd02	Flexi AC	75 €
Prd03	Hyper R	175 €
Prd04	C3PO	80 €

Datenbanken - LE 07 - Recordsets 47

Was sind Recordsets? BÄT

Recordset

Recordset = geordnete Menge gleichartiger Datensätze

Beispiele:

- Recordset von Personen
- Recordset von Produkten
- Recordset von Kunde und Produkt

Müller	Mike	Multi XY	150 €
Yilmaz	Ali	Flexi AC	75 €
Müller	Sabine	Hyper R	175 €
Schmidt	Susi	C3PO	80 €

Datenbanken - LE 07 - Recordsets 48

Was sind Recordsets? BÄT

Recordset

Recordset = geordnete Menge gleichartiger Datensätze

Kein Beispiel

- 2 Kunden und 2 Produkte
- Nicht gleichartig

Kein Recordset!

Müller	Mike	Köln
Yilmaz	Ali	Berlin
Multiplex	150 €	
Flexi AC	75 €	

FALSCH

Datenbank

Datenbanken - LE 07 - Recordsets 49

Was sind Recordsets? BÄT

Eigenschaften Recordset

Recordset

Datenbank

Datenbanken - LE 07 - Recordsets 50

Was sind Recordsets? BÄT

Eigenschaften Recordset

- Geordnete Reihenfolge durch Nachfolger-Beziehung zw. Datensätzen

Recordset

Nachfolgerbeziehung ergibt Reihenfolge der Datensätze

Datenbank

Datenbanken - LE 07 - Recordsets 51

Was sind Recordsets? BÄT

Eigenschaften Recordset

- Geordnete Reihenfolge durch Nachfolger-Beziehung zw. Datensätzen
- Beginn: vor dem ersten Datensatz
- Ende: hinter dem letzten Datensatz

Recordset

Beginn

Ende

Nachfolgerbeziehung ergibt Reihenfolge der Datensätze

Datenbank

Datenbanken - LE 07 - Recordsets 52

Was sind Recordsets?

Eigenschaften

- Geordnete Reihenfolge durch Nachfolger-Beziehung zw. Datensätzen
- Beginn: vor dem ersten Datensatz
- Ende: hinter dem letzten Datensatz
- Zeiger
 - der auf das aktuelle Element, den Beginn oder das Ende zeigt
 - zum nächsten Datensatz bewegt werden kann

Datenbanken - LE 07 - Recordsets 53

Was sind Recordsets?

Eigenschaften

- Geordnete Reihenfolge durch Nachfolger-Beziehung zw. Datensätzen
- Beginn: vor dem ersten Datensatz
- Ende: hinter dem letzten Datensatz
- Zeiger
 - der auf das aktuelle Element, den Beginn oder das Ende zeigt
 - zum nächsten Datensatz bewegt werden kann
- Beziehung zur Datenbank, aus der die Datensätze stammen

Datenbanken - LE 07 - Recordsets 54

Was sind Recordsets?

Eigenschaften

- Geordnete Reihenfolge durch Nachfolger-Beziehung zw. Datensätzen
- Beginn: vor dem ersten Datensatz
- Ende: hinter dem letzten Datensatz
- Zeiger
 - der auf das aktuelle Element, den Beginn oder das Ende zeigt
 - zum nächsten Datensatz bewegt werden kann
- Beziehung zur Datenbank, aus der die Datensätze stammen

Datenbanken - LE 07 - Recordsets 55

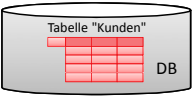
Inhalt

- Ziel und Einordnung
- Rückblick
- Ausgangssituation
- Datenbankschnittstelle
- Recordsets**
 - Was sind Recordsets
 - Mit Recordsets Daten lesen
 - Navigation im Recordset per Zeiger
 - Suchen und Finden im Recordset
 - Ändern und Hinzufügen im Recordset
 - Kombination mit grafischen Abfragen und SQL
- Weitere VBA-Funktionen
- Fallstricke
- Zusammenfassung und Ausblick

Wirtschaftsinformatik 2 - LE 07 - Recordsets 56

Mit Recordsets Daten lesen

Ziel: Aus Kunden den Vornamen und Nachnamen des ersten Kunden lesen



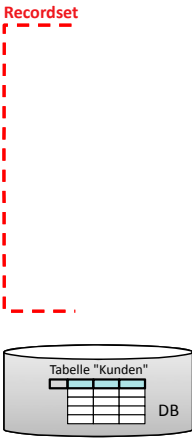
Wirtschaftsinformatik 2 - LE 07 - Recordsets 57

Mit Recordsets Daten lesen

Ziel: Aus Kunden den Vornamen und Nachnamen des ersten Kunden lesen

Ansatz

- Recordset deklarieren



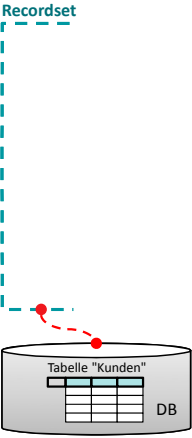
Wirtschaftsinformatik 2 - LE 07 - Recordsets 58

Mit Recordsets Daten lesen

Ziel: Aus Kunden den Vornamen und Nachnamen des ersten Kunden lesen

Ansatz

- Recordset deklarieren
- Verbindung zur Datenbank vorbereiten



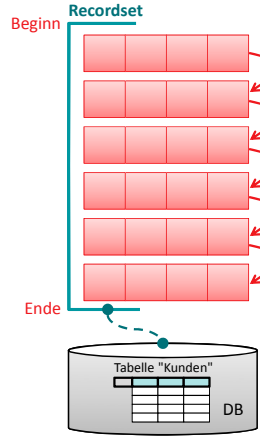
Wirtschaftsinformatik 2 - LE 07 - Recordsets 59

Mit Recordsets Daten lesen

Ziel: Aus Kunden den Vornamen und Nachnamen des ersten Kunden lesen

Ansatz

- Recordset deklarieren
- Verbindung zur Datenbank vorbereiten
- Recordset initialisieren und dadurch füllen



Wirtschaftsinformatik 2 - LE 07 - Recordsets 60

Mit Recordsets Daten lesen

BÄT

Ziel: Aus Kunden den Vornamen und Nachnamen des ersten Kunden lesen

Ansatz

- Recordset deklarieren
- Verbindung zur Datenbank vorbereiten
- Recordset initialisieren und dadurch füllen
- per Zeiger (steht auf erstem Element) auf Felder zugreifen

The diagram illustrates a Recordset as a vertical stack of five rows, each containing four columns. A red arrow labeled 'Zeiger' points to the first row, which is highlighted in red. The top of the stack is labeled 'Beginn' and the bottom 'Ende'. A dashed line connects the 'Ende' of the Recordset to a database icon labeled 'Tabelle "Kunden" DB'.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 61

Mit Recordsets Daten lesen

BÄT

Ziel: Aus Kunden den Vornamen und Nachnamen des ersten Kunden lesen

Ansatz

- Recordset deklarieren
- Verbindung zur Datenbank vorbereiten
- Recordset initialisieren und dadurch füllen
- per Zeiger (steht auf erstem Element) auf Felder zugreifen

The diagram is identical to slide 61, showing a Recordset structure with a pointer at the beginning and a connection to a database table 'Kunden'.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 62

Mit Recordsets Daten lesen

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()

End Sub
```

The diagram shows a database icon labeled 'Tabelle "Kunden" DB' with a red highlight on the first row of the data table.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 63

Mit Recordsets Daten lesen

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()

' Recordset deklarieren

' Verbindung zur DB vorbereiten

' Recordset initialisieren

' auf Felder zugreifen/ausgeben

End Sub
```

The diagram is identical to slide 63, showing a database icon labeled 'Tabelle "Kunden" DB' with a red highlight on the first row of the data table.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 64

Mit Recordsets Daten lesen

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()
' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur DB vorbereiten

' Recordset initialisieren

' auf Felder zugreifen/ausgeben

End Sub
    
```

Recordset

Tabelle "Kunden"
DB

65

Wirtschaftsinformatik 2 - LE 07 - Recordsets

Mit Recordsets Daten lesen

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()
' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren

' auf Felder zugreifen/ausgeben

End Sub
    
```

Recordset

Tabelle "Kunden"
DB

66

Wirtschaftsinformatik 2 - LE 07 - Recordsets

Mit Recordsets Daten lesen

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()
' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

' auf Felder zugreifen/ausgeben

End Sub
    
```

Recordset

Beginn

Ende

Tabelle "Kunden"
DB

67

Wirtschaftsinformatik 2 - LE 07 - Recordsets

Mit Recordsets Daten lesen

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()
' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

' auf Felder zugreifen/ausgeben
Debug.Print rcsKnd.Fields("kndName")
Debug.Print rcsKnd.Fields("kndVorname")

End Sub
    
```

Recordset

Beginn

Zeiger

Ende

Tabelle "Kunden"
DB

68

Wirtschaftsinformatik 2 - LE 07 - Recordsets

Mit Recordsets Daten lesen

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()

' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

' auf Felder zugreifen/ausgeben
Debug.Print rcsKnd.Fields("kndName")
Debug.Print rcsKnd.Fields("kndVorname")

End Sub
```

69

Mit Recordsets Daten lesen

BÄT

Modul

```
(Allgemein)
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()

' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset("tblKunden")

' per Zeiger (steht auf erstem Element) auf
' Felder zugreifen und ausgeben
Debug.Print rcsKnd.Fields("kndName")
Debug.Print rcsKnd.Fields("kndVorname")

End Sub
```

70

Mit Recordsets Daten lesen

BÄT

Modul

```
(Allgemein)
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()

' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset("tblKunden")

' per Zeiger (steht auf erstem Element) auf
' Felder zugreifen und ausgeben
Debug.Print rcsKnd.Fields("kndName")
Debug.Print rcsKnd.Fields("kndVorname")

End Sub
```

tblKunden	kndIdPk	kndName	kndVorname
1	Meier	Thomas	
2	Müller	Sven	
3	Meier	Mike	
4	Müller	Silke	

71

Mit Recordsets Daten lesen

BÄT

Modul

```
(Allgemein)
Option Compare Database
Option Explicit

Sub ausgebenKundeNameVorname()

' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset("tblKunden")

' per Zeiger (steht auf erstem Element) auf
' Felder zugreifen und ausgeben
Debug.Print rcsKnd.Fields("kndName")
Debug.Print rcsKnd.Fields("kndVorname")

End Sub
```

tblKunden	kndIdPk	kndName	kndVorname
1	Meier	Thomas	
2	Müller	Sven	
3	Meier	Mike	
4	Müller	Silke	

72

Mit Recordsets Daten lesen

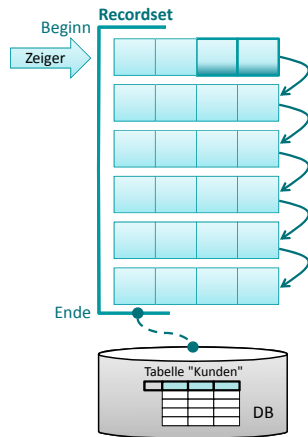
BÄT

Vorgehensweise

- Recordset deklarieren
- Verbindung zur Datenbank vorbereiten
- Recordset initialisieren und dadurch füllen
- per Zeiger (steht auf erstem Element) auf Felder zugreifen

Zugriff auf Felder im Recordset

`<einRecordset>.Fields(<name>)`



Wirtschaftsinformatik 2 - LE 07 - Recordsets

73

Hinweis für Online-Lehrveranstaltung

BÄT

Hier die Übung Ü7.1 mit ca. 20 min Bearbeitungszeit einplanen

Wirtschaftsinformatik 2 - LE 07 - Recordsets

74

Inhalt

BÄT

Ziel und Einordnung

Rückblick

Ausgangssituation

Datenbankschnittstelle

Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen

Fallstricke

Zusammenfassung und Ausblick



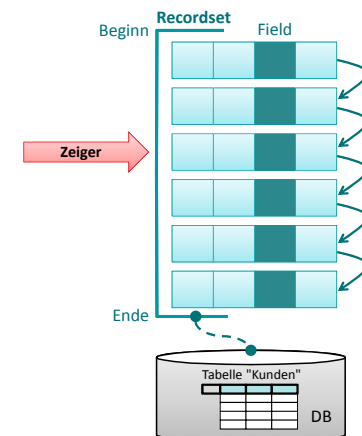
Wirtschaftsinformatik 2 - LE 07 - Recordsets

75

Navigation im Recordset per Zeiger

BÄT

Navigationsmöglichkeiten



Wirtschaftsinformatik 2 - LE 07 - Recordsets

76

Navigation im Recordset per Zeiger

Navigationmöglichkeiten

- zum nächsten Datensatz verschieben
`<einRecordset>.MoveNext`
- zum vorherigen Datensatz verschieben
`<einRcrdst>.MovePrevious`

Wirtschaftsinformatik 2 - LE 07 - Recordsets 77

Navigation im Recordset per Zeiger

Navigationmöglichkeiten

- zum nächsten Datensatz verschieben
`<einRecordset>.MoveNext`
- zum vorherigen Datensatz verschieben
`<einRcrdst>.MovePrevious`
- zum letzten Datensatz verschieben
`<einRecordset>.MoveLast`
- zum ersten Datensatz verschieben
`<einRecordset>.MoveFirst`

Wirtschaftsinformatik 2 - LE 07 - Recordsets 78

Navigation im Recordset per Zeiger

Navigationmöglichkeiten

- zum nächsten Datensatz verschieben
`<einRecordset>.MoveNext`
- zum vorherigen Datensatz verschieben
`<einRcrdst>.MovePrevious`
- zum letzten Datensatz verschieben
`<einRecordset>.MoveLast`
- zum ersten Datensatz verschieben
`<einRecordset>.MoveFirst`

Positionsbestimmung, ob Zeiger am Beginn oder am Ende steht (End of File)

- `<einRecordset>.BOF`
- `<einRecordset>.EOF`

Wirtschaftsinformatik 2 - LE 07 - Recordsets 79

Navigation im Recordset per Zeiger

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende

Wirtschaftsinformatik 2 - LE 07 - Recordsets 80

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- **Prüfen, ob Ende des Recordsets erreicht**

Wirtschaftsinformatik 2 - LE 07 - Recordsets 81

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht
- **Wenn nicht Ende, dann Felder des Datensatzes ausgeben**

Wirtschaftsinformatik 2 - LE 07 - Recordsets 82

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht
- Wenn nicht Ende, dann Felder des Datensatzes ausgeben
- **Zeiger zum nächsten Element verschieben**

Wirtschaftsinformatik 2 - LE 07 - Recordsets 83

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht
- Wenn nicht Ende, dann Felder des Datensatzes ausgeben
- Zeiger zum nächsten Element verschieben
- **Solange fortfahren, bis Ende erreicht**

Wirtschaftsinformatik 2 - LE 07 - Recordsets 84

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht**
- Wenn nicht Ende, dann Felder des Datensatzes ausgeben
- Zeiger zum nächsten Element verschieben
- Solange fortfahren, bis Ende erreicht

Wirtschaftsinformatik 2 - LE 07 - Recordsets 85

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht
- Wenn nicht Ende, dann Felder des Datensatzes ausgeben**
- Zeiger zum nächsten Element verschieben
- Solange fortfahren, bis Ende erreicht

Wirtschaftsinformatik 2 - LE 07 - Recordsets 86

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht
- Wenn nicht Ende, dann Felder des Datensatzes ausgeben
- Zeiger zum nächsten Element verschieben**
- Solange fortfahren, bis Ende erreicht

Wirtschaftsinformatik 2 - LE 07 - Recordsets 87

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht**
- Wenn nicht Ende, dann Felder des Datensatzes ausgeben
- Zeiger zum nächsten Element verschieben
- Solange fortfahren, bis Ende erreicht

Wirtschaftsinformatik 2 - LE 07 - Recordsets 88

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht
- **Wenn nicht Ende, dann Felder des Datensatzes ausgeben**
- Zeiger zum nächsten Element verschieben
- Solange fortfahren, bis Ende erreicht

Wirtschaftsinformatik 2 - LE 07 - Recordsets 89

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht
- Wenn nicht Ende, dann Felder des Datensatzes ausgeben
- **Zeiger zum nächsten Element verschieben**
- Solange fortfahren, bis Ende erreicht

Wirtschaftsinformatik 2 - LE 07 - Recordsets 90

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- **Prüfen, ob Ende des Recordsets erreicht**
- Wenn nicht Ende, dann Felder des Datensatzes ausgeben
- Zeiger zum nächsten Element verschieben
- Solange fortfahren, bis Ende erreicht

Wirtschaftsinformatik 2 - LE 07 - Recordsets 91

Navigation im Recordset per Zeiger

BÄT

Ziel: Recordset aus Tabelle "tblKunden" laden und Name, Vorname aller Kunden im Direktbereich ausgeben

Ansatz

- Recordset befüllen und Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Prüfen, ob Ende des Recordsets erreicht
- Wenn nicht Ende, dann Felder des Datensatzes ausgeben
- Zeiger zum nächsten Element verschieben
- Solange fortfahren, bis Ende erreicht

Wirtschaftsinformatik 2 - LE 07 - Recordsets 92

Navigation im Recordset per Zeiger

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

End Sub
    
```

The diagram illustrates a Recordset as a grid of records. A blue arrow labeled 'Zeiger' points to the first record, labeled 'Beginn'. A dashed line connects the 'Ende' of the Recordset to a database cylinder labeled 'Tabelle "Kunden" DB'.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 93

Navigation im Recordset per Zeiger

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF

Loop

End Sub
    
```

The diagram is similar to slide 93, but with a red arrow labeled 'Zeiger' pointing to the first record, and a red text label 'EOF? Nein!' next to it, indicating the end of the recordset.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 94

Navigation im Recordset per Zeiger

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")

Loop

End Sub
    
```

The diagram shows the Recordset with a red arrow labeled 'Zeiger' pointing to the first record, and a red text label 'Fields' next to it, indicating the current record's fields.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 95

Navigation im Recordset per Zeiger

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Zeiger weiterrücken
rcsKnd.MoveNext

Loop

End Sub
    
```

The diagram shows the Recordset with a red arrow labeled 'Zeiger' pointing to the first record, and a red text label 'MoveNext' next to it, indicating the next record.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 96

Navigation im Recordset per Zeiger

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Zeiger weiterrücken
rcsKnd.MoveNext
' Solange fortfahren, bis Ende erreicht
Loop
End Sub
```

Diagram: A Recordset structure with three rows. A blue arrow labeled "Zeiger" points to the first row. A dashed line labeled "Ende" points to the bottom of the Recordset. A red arrow labeled "EOF? Nein!" points to the first row. Below the Recordset is a database icon labeled "Tabelle 'Kunden'" and "DB".

Wirtschaftsinformatik 2 - LE 07 - Recordsets 97

Navigation im Recordset per Zeiger

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Zeiger weiterrücken
rcsKnd.MoveNext
' Solange fortfahren, bis Ende erreicht
Loop
End Sub
```

Diagram: A Recordset structure with three rows. A blue arrow labeled "Zeiger" points to the first row. A dashed line labeled "Ende" points to the bottom of the Recordset. A red arrow labeled "EOF? Nein!" points to the first row. The second row is highlighted in red. Below the Recordset is a database icon labeled "Tabelle 'Kunden'" and "DB".

Wirtschaftsinformatik 2 - LE 07 - Recordsets 98

Navigation im Recordset per Zeiger

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Zeiger weiterrücken
rcsKnd.MoveNext
' Solange fortfahren, bis Ende erreicht
Loop
End Sub
```

Diagram: A Recordset structure with three rows. A blue arrow labeled "Zeiger" points to the first row. A red arrow labeled "MoveNext" points down to the second row. A dashed line labeled "Ende" points to the bottom of the Recordset. Below the Recordset is a database icon labeled "Tabelle 'Kunden'" and "DB".

Wirtschaftsinformatik 2 - LE 07 - Recordsets 99

Navigation im Recordset per Zeiger

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Zeiger weiterrücken
rcsKnd.MoveNext
' Solange fortfahren, bis Ende erreicht
Loop
End Sub
```

Diagram: A Recordset structure with three rows. A blue arrow labeled "Zeiger" points to the first row. A red arrow labeled "EOF? Nein!" points to the first row. A dashed line labeled "Ende" points to the bottom of the Recordset. Below the Recordset is a database icon labeled "Tabelle 'Kunden'" and "DB".

Wirtschaftsinformatik 2 - LE 07 - Recordsets 100

Navigation im Recordset per Zeiger

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Zeiger weiterrücken
rcsKnd.MoveNext
' Solange fortfahren, bis Ende erreicht
Loop

End Sub
```

101

Navigation im Recordset per Zeiger

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Zeiger weiterrücken
rcsKnd.MoveNext
' Solange fortfahren, bis Ende erreicht
Loop

End Sub
```

102

Navigation im Recordset per Zeiger

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Zeiger weiterrücken
rcsKnd.MoveNext
' Solange fortfahren, bis Ende erreicht
Loop

End Sub
```

103

Navigation im Recordset per Zeiger

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren
Set rcsKnd=db.OpenRecordset("tblKunden")

'Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' Name, Vorname des aktuellen
' Elements ausgeben (Zeiger zeigt drauf)
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Zeiger weiterrücken
rcsKnd.MoveNext
' Solange fortfahren, bis Ende erreicht
Loop

End Sub
```

104

Navigation im Recordset per Zeiger

Modul

```

[Allgemein]
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
    ' Recordset deklarieren
    Dim rcsKnd As Recordset

    ' Verbindung zur Datenbank vorbereiten
    Dim db As Database
    Set db = CurrentDb

    ' Recordset initialisieren und dadurch füllen
    Set rcsKnd = db.OpenRecordset("tblKunden")

    'Schleife bis zum Ende des Recordset
    Do Until rcsKnd.EOF

        ' Name und Vorname des aktuellen Elements
        ' (auf das der Zeiger zeigt) ausgeben
        Debug.Print rcsKnd.Fields("kndName") & _
            ", " & rcsKnd.Fields("kndVorname")

        ' Zeiger weiterrücken
        rcsKnd.MoveNext

    ' Solange fortfahren, bis Ende erreicht
    Loop

End Sub
    
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 105

Navigation im Recordset per Zeiger

Modul

```

[Allgemein]
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
    ' Recordset deklarieren
    Dim rcsKnd As Recordset

    ' Verbindung zur Datenbank vorbereiten
    Dim db As Database
    Set db = CurrentDb

    ' Recordset initialisieren und dadurch füllen
    Set rcsKnd = db.OpenRecordset("tblKunden")

    'Schleife bis zum Ende des Recordset
    Do Until rcsKnd.EOF

        ' Name und Vorname des aktuellen Elements
        ' (auf das der Zeiger zeigt) ausgeben
        Debug.Print rcsKnd.Fields("kndName") & _
            ", " & rcsKnd.Fields("kndVorname")

        ' Zeiger weiterrücken
        rcsKnd.MoveNext

    ' Solange fortfahren, bis Ende erreicht
    Loop

End Sub
    
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 106

Navigation im Recordset per Zeiger

Modul

```

[Allgemein]
Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
    ' Recordset deklarieren
    Dim rcsKnd As Recordset

    ' Verbindung zur Datenbank vorbereiten
    Dim db As Database
    Set db = CurrentDb

    ' Recordset initialisieren und dadurch füllen
    Set rcsKnd = db.OpenRecordset("tblKunden")

    'Schleife bis zum Ende des Recordset
    Do Until rcsKnd.EOF

        ' Name und Vorname des aktuellen Elements
        ' (auf das der Zeiger zeigt) ausgeben
        Debug.Print rcsKnd.Fields("kndName") & _
            ", " & rcsKnd.Fields("kndVorname")

        ' Zeiger weiterrücken
        rcsKnd.MoveNext

    ' Solange fortfahren, bis Ende erreicht
    Loop

End Sub
    
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 107

Navigation im Recordset per Zeiger

Navigationmöglichkeiten

- zum nächsten Datensatz verschieben
`<einRecordset>.MoveNext`
- zum vorherigen Datensatz verschieben
`<einRcrdst>.MovePrevious`
- zum letzten Datensatz verschieben
`<einRecordset>.MoveLast`
- zum ersten Datensatz verschieben
`<einRecordset>.MoveFirst`

Positionsbestimmung, ob Zeiger am Beginn oder am Ende steht (End of File)

```

<einRecordset>.BOF
<einRecordset>.EOF
    
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 108

Ändern und Hinzufügen: Demo 07.01

BÄT

D07.01: Verwenden Sie ein Recordset, um

- die Bezeichnung und den Preis aller Produkte im Direktbereich auszugeben
- Ausgangspunkt Tabelle "tblProdukte" mit Spalte "prdBezeichnung" und "prdPreis"

tblProdukte	tblProduktKategorie	prdIdPk	prdBezeichnung	prdBeschreibung	prdPreis
		1	Blumenzwiebeln Sorte XY	Rote Blumen, sehr schön	0,59 €
		2	Blumensamen, Sorte ABC	Gelbe Blumen, klein	0,99 €
		3	Sonnenblumensamen, Sorte XXL	Große gelbe Blüten, sehr schön	1,99 €
		4	Rasendünger Maxi	5kg, Lanzeitdünger	5,99 €
		5	Rasendünger Mega	10kg, Sofortwirkung	9,99 €
		6	Blumendünger "Sabine"	500 ml, Grün- und Zimmerpflanzen	0,99 €
		7	Blumendünger "Dörte"	500 ml, Bio-Dünger	2,99 €
		8	Rindenmulch fein	6 kg, feiner Mulch	2,49 €
		9	Rindenmulch grob	6 kg, grober Mulch	2,69 €
		10	Rasenbesen "Bert"	Markenprodukt, lange Haltbarkeit	5,99 €

Wirtschaftsinformatik 2 - LE 07 - Recordsets

109

Hinweis für Online-Lehrveranstaltung

BÄT

Hier die Übung Ü7.2 mit ca. 15 min Bearbeitungszeit einplanen

Wirtschaftsinformatik 2 - LE 07 - Recordsets

110

Inhalt

BÄT

Ziel und Einordnung

Rückblick

Ausgangssituation

Datenbankschnittstelle

Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen

Fallstricke

Zusammenfassung und Ausblick



Wirtschaftsinformatik 2 - LE 07 - Recordsets

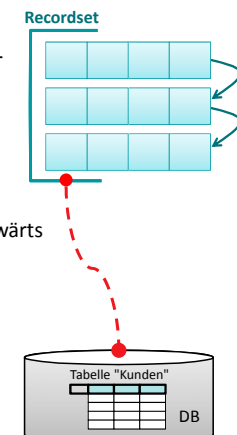
111

Suchen und Finden im Recordset

BÄT

Recordset-Typen, die beim Befüllen angegeben werden können

- Dynaset: Änderungen am Datenbestand in der Datenbank spiegeln sich im Recordset wider
`db.OpenRecordset ("tblPersonen", RecordsetTypeEnum.dbOpenDynaset)`
- Table: Keine Aktualisierung, hat Index; Keine Suche/Sortierung möglich
`db.OpenRecordset ("tblPersonen", RecordsetTypeEnum.dbOpenTable)`
- ForwardOnly: Recordset kann nur einmal vorwärts durchlaufen werden; spart Speicherplatz
`db.OpenRecordset ("tblPersonen", RecordsetTypeEnum.dbOpenForwardOnly)`
- Snapshot: Kopie der Daten zum Zeitpunkt des Befüllens
`db.OpenRecordset ("tblPersonen", RecordsetTypeEnum.dbOpenSnapshot)`



Wirtschaftsinformatik 2 - LE 07 - Recordsets

112

Suchen und Finden im Recordset

Navigationen

Navigationen

Beginn Recordset Field

Zeiger

Ende

dbOpenDynaset

Tabelle "Kunden" DB

Wirtschaftsinformatik 2 - LE 07 - Recordsets 113

Suchen und Finden im Recordset

Navigationen

Navigationen

- ersten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindFirst()`

Beginn Recordset Field

Zeiger

FindFirst

Ende

dbOpenDynaset

Tabelle "Kunden" DB

Wirtschaftsinformatik 2 - LE 07 - Recordsets 114

Suchen und Finden im Recordset

Navigationen

Navigationen

- ersten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindFirst()`
- nächsten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindNext()`

Beginn Recordset Field

Zeiger

FindNext

Ende

dbOpenDynaset

Tabelle "Kunden" DB

Wirtschaftsinformatik 2 - LE 07 - Recordsets 115

Suchen und Finden im Recordset

Navigationen

Navigationen

- ersten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindFirst()`
- nächsten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindNext()`

Beginn Recordset Field

Zeiger

FindNext

Ende

dbOpenDynaset

Tabelle "Kunden" DB

Wirtschaftsinformatik 2 - LE 07 - Recordsets 116

Suchen und Finden im Recordset

BÄT

Navigationmöglichkeiten

- ersten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindFirst()`
- nächsten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindNext()`

Positionsbestimmung, ob Datensätze gefunden werden konnten

`<einRecordset>.NoMatch`

117

Suchen und Finden im Recordset

BÄT

Navigationmöglichkeiten

- ersten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindFirst()`
- nächsten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindNext()`

Positionsbestimmung, ob Datensätze gefunden werden konnten

`<einRecordset>.NoMatch`

118

Suchen und Finden im Recordset

BÄT

Ziel: Aus dem Recordset aller Personen sollen die Kunden gefunden und ausgegeben werden, die "Meier" heißen.

Ansatz

- Recordset als Dynaset befüllen, Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende

119

Suchen und Finden im Recordset

BÄT

Ziel: Aus dem Recordset aller Personen sollen die Kunden gefunden und ausgegeben werden, die "Meier" heißen.

Ansatz

- Recordset als Dynaset befüllen, Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Recordset nach erstem Kunden "Meier" durchsuchen

120

Suchen und Finden im Recordset

Ziel: Aus dem Recordset aller Personen sollen die Kunden gefunden und ausgegeben werden, die "Meier" heißen.

Ansatz

- Recordset als Dnyaset befüllen, Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Recordset nach erstem Kunden "Meier" durchsuchen
- Wenn gefunden, dann Kunden ausgeben

Wirtschaftsinformatik 2 - LE 07 - Recordsets 121

Suchen und Finden im Recordset

Ziel: Aus dem Recordset aller Personen sollen die Kunden gefunden und ausgegeben werden, die "Meier" heißen.

Ansatz

- Recordset als Dnyaset befüllen, Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Recordset nach erstem Kunden "Meier" durchsuchen
- Wenn gefunden, dann Kunden ausgeben
- Weitersuchen in Schleife, bis keine Treffer (mehr)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 122

Suchen und Finden im Recordset

Ziel: Aus dem Recordset aller Personen sollen die Kunden gefunden und ausgegeben werden, die "Meier" heißen.

Ansatz

- Recordset als Dnyaset befüllen, Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Recordset nach erstem Kunden "Meier" durchsuchen
- Wenn gefunden, dann Kunden ausgeben
- Weitersuchen in Schleife, bis keine Treffer (mehr)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 123

Suchen und Finden im Recordset

Ziel: Aus dem Recordset aller Personen sollen die Kunden gefunden und ausgegeben werden, die "Meier" heißen.

Ansatz

- Recordset als Dnyaset befüllen, Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Recordset nach erstem Kunden "Meier" durchsuchen
- Wenn gefunden, dann Kunden ausgeben
- Weitersuchen in Schleife, bis keine Treffer (mehr)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 124

Suchen und Finden im Recordset

Ziel: Aus dem Recordset aller Personen sollen die Kunden gefunden und ausgegeben werden, die "Meier" heißen.

Ansatz

- Recordset als Dnyaset befüllen, Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Recordset nach erstem Kunden "Meier" durchsuchen
- Wenn gefunden, dann Kunden ausgeben
- Weitersuchen in Schleife, bis keine Treffer (mehr)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 125

Suchen und Finden im Recordset

Ziel: Aus dem Recordset aller Personen sollen die Kunden gefunden und ausgegeben werden, die "Meier" heißen.

Ansatz

- Recordset als Dnyaset befüllen, Zeiger steht auf erstem Datensatz oder wenn leer, dann auf Ende
- Recordset nach erstem Kunden "Meier" durchsuchen
- Wenn gefunden, dann Kunden ausgeben
- Weitersuchen in Schleife, bis keine Treffer (mehr)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 126

Suchen und Finden im Recordset

Modul

```
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset

End Sub
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 127

Suchen und Finden im Recordset

Modul

```
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset ("tblKunden",
dbOpenDynaset)

End Sub
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 128

Suchen und Finden im Recordset

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden",
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")

End Sub
    
```

129

Suchen und Finden im Recordset

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden",
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")

End Sub
    
```

130

Suchen und Finden im Recordset

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden",
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch

Loop
End Sub
    
```

131

Suchen und Finden im Recordset

BÄT

```

Modul
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden",
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch
' Name und Vorname ausgeben
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")

Loop
End Sub
    
```

132

Suchen und Finden im Recordset

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden",
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch
' Name und Vorname ausgeben
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Nächsten finden
rcsKnd.FindNext ("kndName = 'Meier'")
Loop
End Sub
```

133

Suchen und Finden im Recordset

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden",
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch
' Name und Vorname ausgeben
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Nächsten finden
rcsKnd.FindNext ("kndName = 'Meier'")
' Weiterer Schleifendurchlauf
Loop
End Sub
```

134

Suchen und Finden im Recordset

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden",
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch
' Name und Vorname ausgeben
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Nächsten finden
rcsKnd.FindNext ("kndName = 'Meier'")
' Weiterer Schleifendurchlauf
Loop
End Sub
```

135

Suchen und Finden im Recordset

BÄT

Modul

```
Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden",
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch
' Name und Vorname ausgeben
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Nächsten finden
rcsKnd.FindNext ("kndName = 'Meier'")
' Weiterer Schleifendurchlauf
Loop
End Sub
```

136

Suchen und Finden im Recordset

Modul

```

Option Compare Database
Option Explicit

Sub sucheKundenMeier()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden",
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch
' Name und Vorname ausgeben
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Nächsten finden
rcsKnd.FindNext ("kndName = 'Meier'")
' Weiterer Schleifendurchlauf
Loop
End Sub
    
```

Diagram: A Recordset is shown as a grid of cells. A pointer labeled 'Zeiger' points to the first record. The database table 'tblKunden' is shown below with a 'dbOpenDynaset' connection.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 137

Suchen und Finden im Recordset

Modul

```

Option Compare Database
Option Explicit

Sub ausgebenAlleKunden()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset("tblKunden", _
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch
' Name und Vorname ausgeben
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Nächsten finden
rcsKnd.FindNext ("kndName = 'Meier'")
Loop
End Sub
    
```

Diagram: A Recordset is shown as a grid of cells. A pointer labeled 'Zeiger' points to the last record. The database table 'tblKunden' is shown below with a 'dbOpenDynaset' connection.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 138

Suchen und Finden im Recordset

Modul

```

Option Compare Database
Option Explicit

Sub ausgebenKundenMeier()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset("tblKunden", _
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch
' Name und Vorname ausgeben
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Nächsten finden
rcsKnd.FindNext ("kndName = 'Meier'")
Loop
End Sub
    
```

Table View Data:

knldPk	kndName	kndVorname
1	Meier	Thomas
2	Müller	Sven
3	Meier	Mike
4	Müller	Silke

Wirtschaftsinformatik 2 - LE 07 - Recordsets 139

Suchen und Finden im Recordset

Modul

```

Option Compare Database
Option Explicit

Sub ausgebenKundenMeier()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset("tblKunden", _
dbOpenDynaset)
' Suchen nach Kunden "Meier"
rcsKnd.FindFirst ("kndName = 'Meier'")
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.NoMatch
' Name und Vorname ausgeben
Debug.Print rcsKnd.Fields("kndName") & _
", " & rcsKnd.Fields("kndVorname")
' Nächsten finden
rcsKnd.FindNext ("kndName = 'Meier'")
Loop
End Sub
    
```

Table View Data:

knldPk	kndName	kndVorname
1	Meier	Thomas
2	Müller	Sven
3	Meier	Mike
4	Müller	Silke

Wirtschaftsinformatik 2 - LE 07 - Recordsets 140

Suchen und Finden im Recordset

BÄT

Such-/Findemöglichkeiten

- ersten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindFirst`
- nächsten Datensatz finden, der ein bestimmtes Kriterium erfüllt
`<einRecordset>.FindNext`

Positionsbestimmung, ob Datensätze gefunden werden konnten

`<einRecordset>.NoMatch`

141

Ändern und Hinzufügen: Demo 07.02

BÄT

D07.02: Verwenden Sie ein Recordset, um

- die Bezeichnung und den Preis aller Produkte im Direktbereich auszugeben, deren Preis größer als 2 EUR ist
- Ausgangspunkt Tabelle "tblProdukte" mit Spalte "prdPreis"

tblProdukte	tblProduktKategorie	prdBezeichnung	prdBeschreibung	prdPreis
1		Blumenzwiebeln Sorte XY	Rote Blumen, sehr schön	0,59 €
2		Blumensamen, Sorte ABC	Gelbe Blumen, klein	0,99 €
3		Sonnenblumensamen, Sorte XXL	Große gelbe Blüten, sehr schön	1,99 €
4		Rasendünger Maxi	5kg, Lanzeitdünger	5,99 €
5		Rasendünger Mega	10kg, Sofortwirkung	9,99 €
6		Blumendünger "Sabine"	500 ml, Grün- und Zimmerpflanzen	0,99 €
7		Blumendünger "Dörte"	500 ml, Bio-Dünger	2,99 €
8		Rindenmulch fein	6 kg, feiner Mulch	2,49 €
9		Rindenmulch grob	6 kg, grober Mulch	2,69 €
10		Rasenbesen "Bert"	Markenprodukt, lange Haltbarkeit	5,99 €

142

Inhalt

BÄT

Ziel und Einordnung

Rückblick

Ausgangssituation

Datenbankschnittstelle

Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen

Fallstricke

Zusammenfassung und Ausblick

143

Ändern und Hinzufügen im Recordset

BÄT

Bearbeitungsmodus

- Änderungsmodus aktivieren
`<einRecordset>.Edit`
- Einfügemodus aktivieren
`<einRecordset>.AddNew`
- Bearbeitungsmodus abschließen
`<einRecordset>.Update`
- Löschmodus aktivieren
`<einRecordset>.Delete`

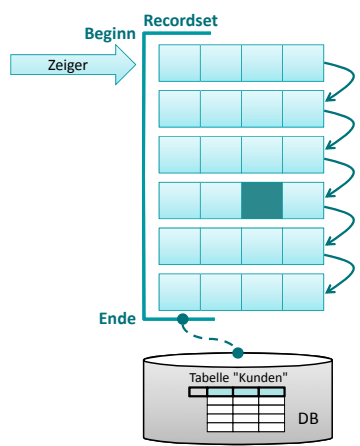
144

Ändern und Hinzufügen im Recordset BÄT

Ziel: Ein Datensatz des Recordset soll geändert werden

Ansatz

- Recordset als Dynaset füllen



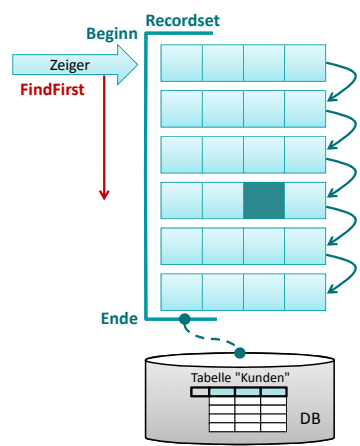
Wirtschaftsinformatik 2 - LE 07 - Recordsets 145

Ändern und Hinzufügen im Recordset BÄT

Ziel: Ein Datensatz des Recordset soll geändert werden

Ansatz

- Recordset als Dynaset füllen
- nach zu änderndem Datensatz suchen



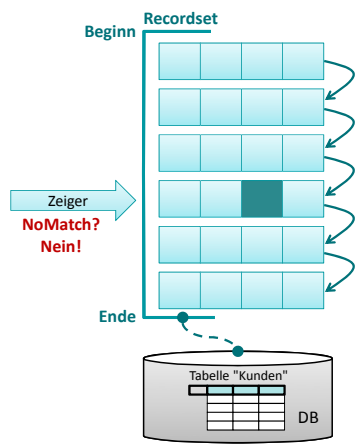
Wirtschaftsinformatik 2 - LE 07 - Recordsets 146

Ändern und Hinzufügen im Recordset BÄT

Ziel: Ein Datensatz des Recordset soll geändert werden

Ansatz

- Recordset als Dynaset füllen
- nach zu änderndem Datensatz suchen
- Prüfen, ob Datensatz gefunden



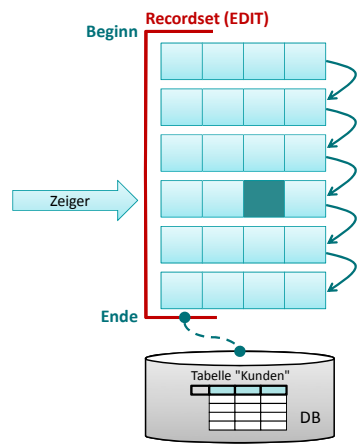
Wirtschaftsinformatik 2 - LE 07 - Recordsets 147

Ändern und Hinzufügen im Recordset BÄT

Ziel: Ein Datensatz des Recordset soll geändert werden

Ansatz

- Recordset als Dynaset füllen
- nach zu änderndem Datensatz suchen
- Prüfen, ob Datensatz gefunden
- Wenn ja, dann Änderungsmodus des Recordset aktivieren



Wirtschaftsinformatik 2 - LE 07 - Recordsets 148

Ändern und Hinzufügen im Recordset

Ziel: Ein Datensatz des Recordset soll geändert werden

Ansatz

- Recordset als Dynaset füllen
- nach zu änderndem Datensatz suchen
- Prüfen, ob Datensatz gefunden
- Wenn ja, dann Änderungsmodus des Recordset aktivieren
- Änderung des Feldes vornehmen

Wirtschaftsinformatik 2 - LE 07 - Recordsets 149

Ändern und Hinzufügen im Recordset

Ziel: Ein Datensatz des Recordset soll geändert werden

Ansatz

- Recordset als Dynaset füllen
- nach zu änderndem Datensatz suchen
- Prüfen, ob Datensatz gefunden
- Wenn ja, dann Änderungsmodus des Recordset aktivieren
- Änderung des Feldes vornehmen
- Aktualisierung abschließen

Wirtschaftsinformatik 2 - LE 07 - Recordsets 150

Ändern und Hinzufügen im Recordset

Ziel: Ein Datensatz des Recordset soll geändert werden

Ansatz

- Recordset als Dynaset füllen
- nach zu änderndem Datensatz suchen
- Prüfen, ob Datensatz gefunden
- Wenn ja, dann Änderungsmodus des Recordset aktivieren
- Änderung des Feldes vornehmen
- Aktualisierung abschließen

```

Modul
Option Compare Database
Option Explicit

Sub aendernKunde ()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden", _
dbOpenDynaset)
' Suchen nach Kunde mit ID4
rcsKnd.FindFirst ("kndIdPk = 4")
If rcsKnd.NoMatch Then
Debug.Print "Kein Datensatz!"
Else
' Änderungsmodus aktivieren
rcsKnd.Edit
' Änderung vornehmen
rcsKnd.Fields("kndName") = "Schmidt"
' Aktualisierung durchführen
rcsKnd.Update
End If
End Sub
    
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 151

Ändern und Hinzufügen im Recordset

Vorher

kndIdPk	kndName	kndVorname
1	Meier	Thomas
2	Müller	Sven
3	Meier	Mike
4	Müller	Silke

Nachher

```

[Allgemein] | vorgebenKundenMeier
Option Compare Database
Option Explicit

Sub aendernKunde ()
' Recordset deklarieren
Dim rcsKnd As Recordset
' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset ("tblKunden", _
dbOpenDynaset)
' Suchen nach Kunde mit ID4
rcsKnd.FindFirst ("kndIdPk = 4")
'Schleife bis zum Ende des Recordset
If rcsKnd.NoMatch Then
Debug.Print "Keinen Datensatz gefunden."
Else
' Änderungsmodus aktivieren
rcsKnd.Edit
' Änderung vornehmen
rcsKnd.Fields("kndName") = "Schmidt"
' Aktualisierung durchführen
rcsKnd.Update
End If
End Sub
    
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 152

Ändern und Hinzufügen im Recordset

Vorher

kndIdPk	kndName	kndVorname
1	Meier	Thomas
2	Müller	Sven
3	Meier	Mike
4	Müller	Mike

Nachher

kndIdPk	kndName	kndVorname
1	Meier	Thomas
2	Müller	Sven
3	Meier	Mike
4	Schmidt	Mike

```

Sub aendernKunde()
' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset("tblKunden", _
    dbOpenDynaset)

' Suchen nach Kunde mit ID4
rcsKnd.FindFirst ("kndIdPk = 4")

' Schleife bis zum Ende des Recordset
If rcsKnd.NoMatch Then
    Debug.Print "Keinen Datensatz gefunden."
Else
    ' Änderungsmodus aktivieren
    rcsKnd.Edit
    ' Änderung vornehmen
    rcsKnd.Fields("kndName") = "Schmidt"
    ' Aktualisierung durchführen
    rcsKnd.Update
End If
End Sub
    
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 153

Ändern und Hinzufügen: Demo 07.03

D07.03: Verwenden Sie ein Recordset, um

- die Preise aller Produkte in Tabelle "tblProdukte" um 0,50 EUR zu erhöhen
- Ausgangspunkt Tabelle "tblProdukte" mit Spalte "prdPreis"

tblProdukte	tblProduktKategorie	prdIdPk	prdBezeichnung	prdBeschreibung	prdPreis
		1	Blumenzwiebeln Sorte XY	Rote Blumen, sehr schön	0,59 €
		2	Blumensamen, Sorte ABC	Gelbe Blumen, klein	0,99 €
		3	Sonnenblumensamen, Sorte XXL	Große gelbe Blüten, sehr schön	1,99 €
		4	Rasendünger Maxi	5kg, Lanzeitdünger	5,99 €
		5	Rasendünger Mega	10kg, Sofortwirkung	9,99 €
		6	Blumendünger "Sabine"	500 ml, Grün- und Zimmerpflanzen	0,99 €
		7	Blumendünger "Dörte"	500 ml, Bio-Dünger	2,99 €
		8	Rindenmulch fein	6 kg, feiner Mulch	2,49 €
		9	Rindenmulch grob	6 kg, grober Mulch	2,69 €
		10	Rasenbesen "Bert"	Markenprodukt, lange Haltbarkeit	5,99 €

Wirtschaftsinformatik 2 - LE 07 - Recordsets 154

Hinweis für Online-Lehrveranstaltung

Hier die Übung Ü7.3 mit ca. 15 min Bearbeitungszeit einplanen

Wirtschaftsinformatik 2 - LE 07 - Recordsets 155

Ändern und Hinzufügen im Recordset

Ziel: Ein Datensatz soll Hinzugefügt werden

Ansatz

- Recordset als Dynaset befüllen
- Anfügemodus des Recordset aktivieren
- Neuen Eintrag feldweise belegen (Reihenfolge egal)
- Aktualisierung abschließen

Wirtschaftsinformatik 2 - LE 07 - Recordsets 156

Ändern und Hinzufügen im Recordset BÄT

Ziel: Ein Datensatz soll Hinzugefügt werden

Ansatz

- Recordset als Dnyaset befüllen
- Anfügemodus des Recordset aktivieren
- Neuen Eintrag feldweise belegen (Reihenfolge egal)
- Aktualisierung abschließen

Wirtschaftsinformatik 2 - LE 07 - Recordsets 157

Ändern und Hinzufügen im Recordset BÄT

Ziel: Ein Datensatz soll Hinzugefügt werden

Ansatz

- Recordset als Dnyaset befüllen
- Anfügemodus des Recordset aktivieren
- Neuen Eintrag feldweise belegen (Reihenfolge egal)
- Aktualisierung abschließen

Wirtschaftsinformatik 2 - LE 07 - Recordsets 158

Ändern und Hinzufügen im Recordset BÄT

Ziel: Ein Datensatz soll Hinzugefügt werden

Ansatz

- Recordset als Dnyaset befüllen
- Anfügemodus des Recordset aktivieren
- Neuen Eintrag feldweise belegen (Reihenfolge egal)
- Aktualisierung abschließen

Wirtschaftsinformatik 2 - LE 07 - Recordsets 159

Ändern und Hinzufügen im Recordset BÄT

Ziel: Ein Datensatz soll Hinzugefügt werden

Ansatz

- Recordset als Dnyaset befüllen
- Anfügemodus des Recordset aktivieren
- Neuen Eintrag feldweise belegen (Reihenfolge egal)
- Aktualisierung abschließen

Wirtschaftsinformatik 2 - LE 07 - Recordsets 160

Ändern und Hinzufügen im Recordset

Ziel: Ein Datensatz soll hinzugefügt werden

Ansatz

- Recordset als Dnyaset befüllen
- Anfügemodus des Recordset aktivieren
- Neuen Eintrag feldweise belegen (Reihenfolge egal)
- Aktualisierung abschließen

Modul

```
Option Compare Database
Option Explicit

Sub hinzufuegenKunde()
' Verbindung zur DB vorbereiten
Dim db As Database
Set db = CurrentDb
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd=db.OpenRecordset("tblKunden", _
dbOpenDynaset)

' Anfügemodus aktivieren
rcsKnd.AddNew

' Felder des neuen Eintrags belegen
rcsKnd.Fields("kndName") = "Schmidt"
rcsKnd.Fields("kndVorname") = "Udo"

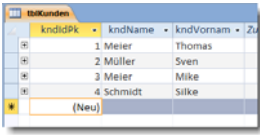
' Aktualisierung durchführen
rcsKnd.Update

End Sub
```


Wirtschaftsinformatik 2 - LE 07 - Recordsets 161

Ändern und Hinzufügen im Recordset

Vorher



Nachher



```
[Allgemein] | hinzufuegenKunde
Option Compare Database
Option Explicit

Sub hinzufuegenKunde()
' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset("tblKunden", _
dbOpenDynaset)

' Anfügemodus aktivieren
rcsKnd.AddNew
' Felder belegen
rcsKnd.Fields("kndName") = "Schmidt"
rcsKnd.Fields("kndVorname") = "Udo"


' Aktualisierung durchführen
rcsKnd.Update

End Sub
```


Wirtschaftsinformatik 2 - LE 07 - Recordsets 162

Ändern und Hinzufügen im Recordset

Vorher



Nachher



```
[Allgemein] | hinzufuegenKunde
Option Compare Database
Option Explicit

Sub hinzufuegenKunde()
' Recordset deklarieren
Dim rcsKnd As Recordset

' Verbindung zur Datenbank vorbereiten
Dim db As Database
Set db = CurrentDb

' Recordset initialisieren und dadurch füllen
Set rcsKnd = db.OpenRecordset("tblKunden", _
dbOpenDynaset)

' Anfügemodus aktivieren
rcsKnd.AddNew
' Felder belegen
rcsKnd.Fields("kndName") = "Schmidt"
rcsKnd.Fields("kndVorname") = "Udo"

' Aktualisierung durchführen
rcsKnd.Update

End Sub
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 163

Ändern und Hinzufügen im Recordset

Bearbeitungsmodus

- Änderungsmodus aktivieren
`<einRecordset>.Edit`
- Einfügemodus aktivieren
`<einRecordset>.AddNew`
- Bearbeitungsmodus abschließen
`<einRecordset>.Update`
- Löschmodus aktivieren
`<einRecordset>.Delete`

Wirtschaftsinformatik 2 - LE 07 - Recordsets 164

Hinweis für Online-Lehrveranstaltung



Hier die Übungen

- Ü7.4 mit ca. 15 min oder Ü7.5 mit ca. 20 min
 - Ü7.6 mit ca. 15 min
- einplanen.

Inhalt



Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle

Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen

Fallstricke

Zusammenfassung und Ausblick



Recordsets mit SQL-Abfragen kombinieren



Initialisierung von Recordsets

- mit einer Tabelle: lädt alle Datensätze aus der Tabelle

```
' Alle Kunden  
Set rcsKunden = db.OpenRecordset("tblKunden")
```

- mit einer grafischen Abfrage

```
' Alle Kunden  
Set rcsKunden = db.OpenRecordset("qryKundenAusBerlin")
```

- mit einer SQL-Abfrage

```
' Alle Kunden  
Set rcsKunden = db.OpenRecordset("SELECT * FROM tblKunden")  
' Kunden aus Berlin (auf Leerzeichen in SQL-Abfrage achten)  
Set rcsKunden = db.OpenRecordset("SELECT * FROM tblKunden" & _  
" WHERE ort='Berlin'")
```

Inhalt



Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle

Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen

Fallstricke

Zusammenfassung und Ausblick



Inhalt BÄT

Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle
Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen
Fallstricke
Zusammenfassung und Ausblick

Wirtschaftsinformatik 2 - LE 07 - Recordsets 169

Weitere VBA-Funktionen zum Datenzugriff BÄT

Bisher: Zugriff auf mehrere Datensätze und deren Werte mittels Recordsets
Jetzt: Ermittlung eines Ergebniswertes aus der Datenbank mittels Domänenfunktionen

- verschiedene Funktionen, z.B.
 - zum Ermitteln eines Wertes aus einem Datensatz
 - zum Ermitteln eines Wertes des ersten/letzten Datensatzes
 - zum Zählen von Datensätzen (anhand einer nicht leeren Spalte)

```
' Generelle Syntax
Let <Var> = DLookup(<Spalte>, <Tabelle>, <Bedingung>)
Let <Var> = DFirst(<Spalte>, <Tabelle>, <Bedingung>)
Let <Var> = DLast(<Spalte>, <Tabelle>, <Bedingung>)
Let <Var> = DCount(<PrimärschlüsselSpalte>, <Tab>, <Beding>)
```

- weitere Funktionen in Zusammenhang mit Auswertung (z.B. Minimum, Maximum) in LE09

Wirtschaftsinformatik 2 - LE 07 - Recordsets 170

Weitere VBA-Funktionen zum Datenzugriff BÄT

Beispiele

```
' Deklaration
Dim strName As String
Dim strErsterName As String
Dim strLetzterName As String
' Initialisierung
Let strName = DLookup("kndName", "tblKunden", "kndIdPk=2")
Let strErsterName = DFirst("kndName", "tblKunden")
Let strLetzterName = DLast("kndName", "tblKunden")
' Ausgabe
Debug.Print strName
Debug.Print strErsterName
Debug.Print strLetzterName
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 171

Weitere VBA-Funktionen zum Datenzugriff BÄT

Beispiele

```
' Deklaration
Dim lngAnzahl As Long
Dim lngNurKat2 As Long
' Initialisierung
Let lngAnzahl = DCount("prdIdPk", "tblProdukte")
Let lngNurKat2 = DCount("prdIdPk", "tblProdukte", "prdkatIdFk=2")
' Ausgabe
Debug.Print "Anzahl Produkte: " & lngAnzahl
Debug.Print "Anzahl Produkte in Kategorie 2: " & lngNurKat2
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets 172

Inhalt BÄT

Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle
Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen
Fallstricke
Zusammenfassung und Ausblick



Wirtschaftsinformatik 2 - LE 07 - Recordsets 173

Inhalt BÄT

Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle
Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen
Fallstricke
Zusammenfassung und Ausblick

Wirtschaftsinformatik 2 - LE 07 - Recordsets 174

Fallstricke BÄT

Endlosschleifen vermeiden

- Vergessen Sie nicht MoveNext oder FindNext, wenn Sie in einer Do Until-Schleife arbeiten

Keine Zählerschleife

- Verwenden Sie keine For-Schleife, um Recordsets vom Typ Dynaset (anhand der zuvor gezählten Elemente) zu bearbeiten.
- Anzahl im Recordset kann sich ändern, u.a. durch Einfügen und Löschen

Aufräumen nicht vergessen

- Denken Sie daran, das Recordset mit Close zu schließen, wenn Sie es nicht mehr benötigen, die Prozedur aber noch nicht endet.
- ...

Wirtschaftsinformatik 2 - LE 07 - Recordsets 175

Fallstricke BÄT

RecordCount-Eigenschaft

- liefert nur dann den genauen Wert der enthaltenen Datensätze, wenn zuvor mit MoveLast an das Ende des Recordsets gegangen wurde

```
' Achtung: Fallstricke bei Recordset: Genaue Anzahl  
' der Datensätze erst nach MoveLast in RecordCount  
If rcsBenutzer.RecordCount > 0 Then  
' MoveLast nur, wenn initial der Recordcount > 0 ist  
' (sonst Fehler)  
    rcsBenutzer.MoveLast  
End If  
' Jetzt liefert RecordCount die genau Anzahl Datensätze
```

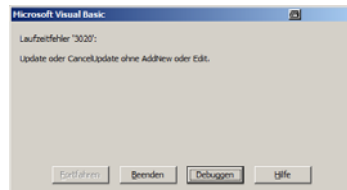
Wirtschaftsinformatik 2 - LE 07 - Recordsets 176

Fallstricke

BÄT

Edit in einer Schleife

- Änderungen müssen vor dem MoveNext, FindNext usw. mit Update abgeschlossen werden, (unmittelbar) vorher Änderungsmodus aktivieren



– ...

Wirtschaftsinformatik 2 - LE 07 - Recordsets

177

Fallstricke

BÄT

Edit in einer Schleife

- Änderungen müssen vor dem MoveNext, FindNext usw. mit Update abgeschlossen werden, (unmittelbar) vorher Änderungsmodus aktivieren

```
' Variable für Geburtsjahr deklarieren
Dim intGebJahr As Integer
' Recordset deklarieren, initialisieren
Dim rcsKnd As Recordset
Set rcsKnd = CurrentDb.OpenRecordset("tblKunden", dbOpenDynaset)
' rcsKnd.Edit 'geht hier schief
' Schleife bis zum Ende des Recordset
Do Until rcsKnd.EOF
' rcsKnd.Edit ' So geht es
Let intGebJahr = rcsKnd.Fields("kndGebJahr")
rcsKnd.Edit ' So gehts es auch
rcsKnd.Fields("kndAlter") = 2010 - intGebJahr
rcsKnd.Update ' So gehts
rcsKnd.MoveNext
'rcsKnd.Update 'geht hier schief
Loop
'rcsKnd.Update geht hier auch schief
```

Wirtschaftsinformatik 2 - LE 07 - Recordsets

178

Fallstricke

BÄT

Bisher ...

- Recordsets mit allen Daten einer Tabelle initialisiert
- allerdings ist es nicht sinnvoll,**
- Recordset mit sehr vielen Daten füllen und darin per FindNext zu suchen

stattdessen...

- Recordsets mit Abfragen kombinieren
 - Nutzung gespeicherter grafischer Abfragen
 - Nutzung von SQL-Abfragen

aber auch nicht sinnvoll per Recordset nachzubauen

- Standardfunktionen von Access (z.B. zum Füllen von Oberflächenelementen)
- SQL-Funktionen (wenn diese beherrscht werden)

Wirtschaftsinformatik 2 - LE 07 - Recordsets

179

Inhalt

BÄT

Ziel und Einordnung

Rückblick

Ausgangssituation

Datenbankschnittstelle

Recordsets

- Was sind Recordsets
- Mit Recordsets Daten lesen
- Navigation im Recordset per Zeiger
- Suchen und Finden im Recordset
- Ändern und Hinzufügen im Recordset
- Kombination mit grafischen Abfragen und SQL

Weitere VBA-Funktionen

Fallstricke

Zusammenfassung und Ausblick



Wirtschaftsinformatik 2 - LE 07 - Recordsets

180

BÄT

Inhalt

Ziel und Einordnung
Rückblick
Ausgangssituation
Datenbankschnittstelle
Recordsets
– Was sind Recordsets
– Mit Recordsets Daten lesen
– Navigation im Recordset per Zeiger
– Suchen und Finden im Recordset
– Ändern und Hinzufügen im Recordset
– Kombination mit grafischen Abfragen und SQL
Weitere VBA-Funktionen
Fallstricke
Zusammenfassung und Ausblick

Wirtschaftsinformatik 2 - LE 07 - Recordsets 181

BÄT

Zusammenfassung

Was stimmt an der Definition von „Recordset“ in der Wikipedia nicht?



The screenshot shows the Wikipedia article for 'Recordset'. The text defines a Recordset as a copy of data rows from a database table. The slide points out that this definition is incorrect because a Recordset is not a copy of data but a pointer to data in a database table. The article also mentions that a Recordset can be changed and that its contents can change over time.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 182

BÄT

Zusammenfassung

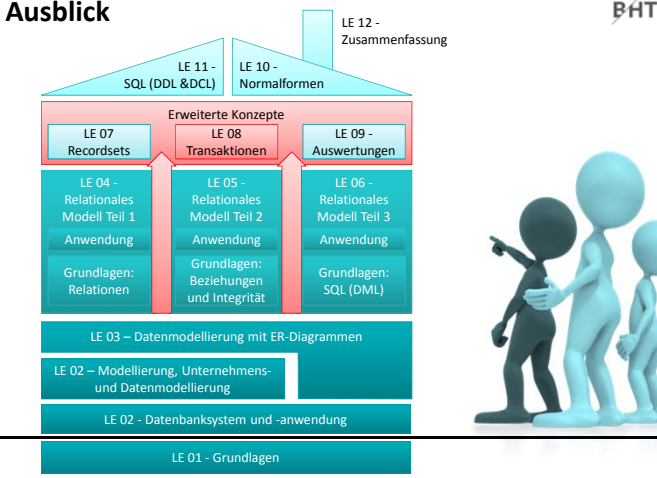
Ein Recordset

- ist eine geordnete Menge von Datensätzen, die aus einer oder mehreren Tabelle einer Datenbank geladen werden.
- kann abhängig von seinem Typ
 - einen Schnappschuss des Datenbestandes repräsentieren
 - mit der Datenbank verbunden sein und Auswirkungen von Änderungen widerspiegeln
- besitzt einen Zeiger, mit dem über Datensätze navigiert werden kann (MoveNext, MovePrevious, ...)
- bietet Möglichkeit auf Werte des Elementes zuzugreifen, auf das der Zeiger zeigt (Fields)
- lässt nach Datensätzen suchen (FindFirst, FindNext, ...)
- kann in verschiedene Modus geschaltet werden
 - Änderungsmodus (Edit)
 - Hinzufüge-Modus (AddNew)
 - Löschmodus (Delete)

Wirtschaftsinformatik 2 - LE 07 - Recordsets 183

BÄT

Ausblick



The diagram is shaped like a house. The roof contains LE 11 (SQL (DDL & DCL)) and LE 10 (Normalformen). The main body of the house is divided into three columns: LE 07 (Recordsets), LE 08 (Transaktionen), and LE 09 (Auswertungen). Below these are LE 04 (Relationales Modell Teil 1), LE 05 (Relationales Modell Teil 2), and LE 06 (Relationales Modell Teil 3). The base of the house consists of LE 03 (Datenmodellierung mit ER-Diagrammen), LE 02 (Modellierung, Unternehmens- und Datenmodellierung), and LE 01 (Grundlagen). A small illustration of three figures is on the right.

Wirtschaftsinformatik 2 - LE 07 - Recordsets 184

Wirtschaftsinformatik 2
LE 07 –Recordsets als
Datenbankschnittstelle

Prof. Dr. Thomas Off

<http://www.ThomasOff.de/lehre/beuth/wi2>