

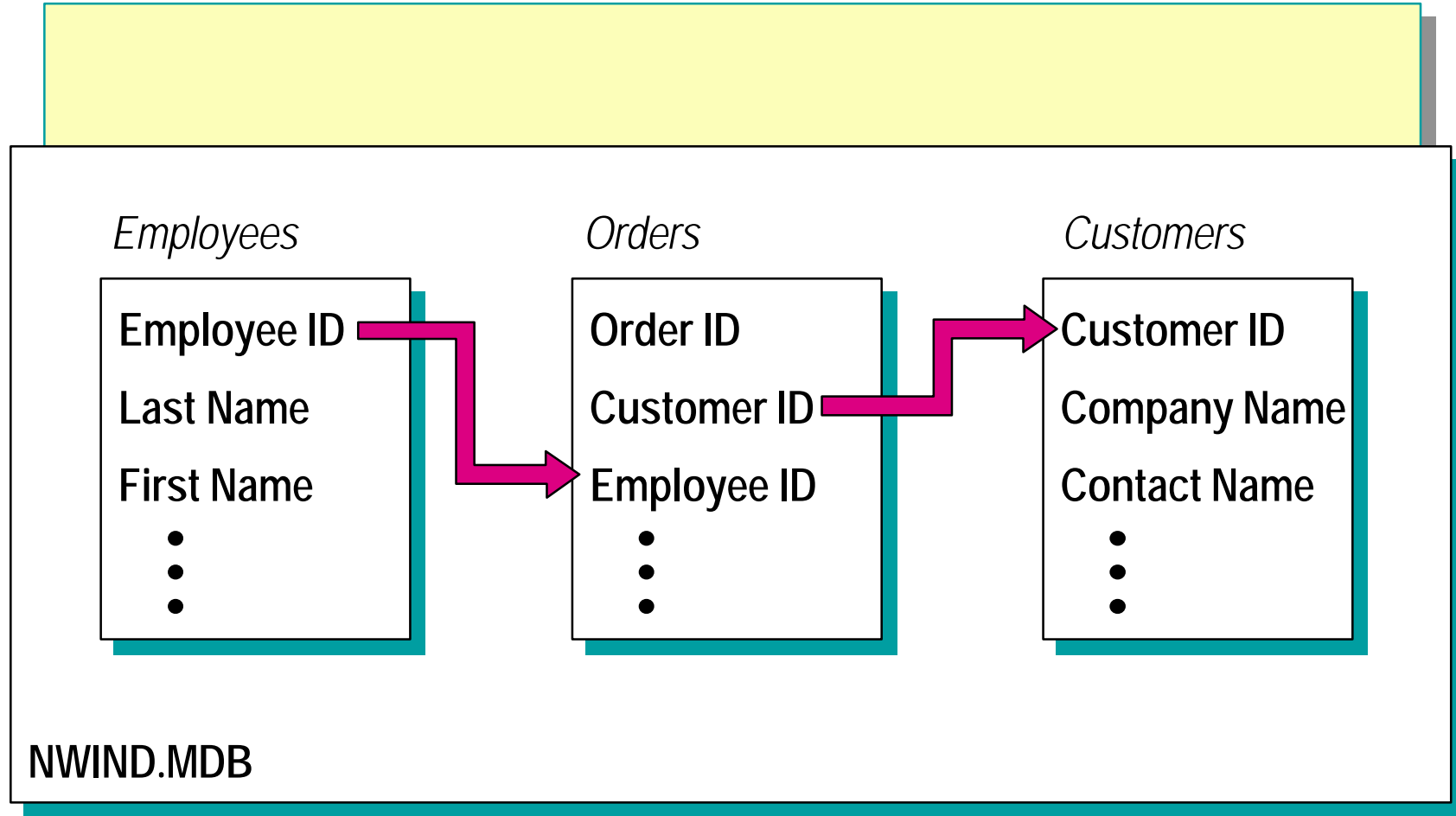
Data Access Using the Data Control

12

◆ Overview

- Overview of a Database
- How Does Visual Basic Access Databases?
- What Is the Data Control?
- The Current Row and the Dynaset
- Finding Records
- Updating, Deleting, and Adding Records
- Data Control Events
- How Locking Is Handled
- Filling a Grid with Data from a Table

Overview of a Database



Overview of a Table

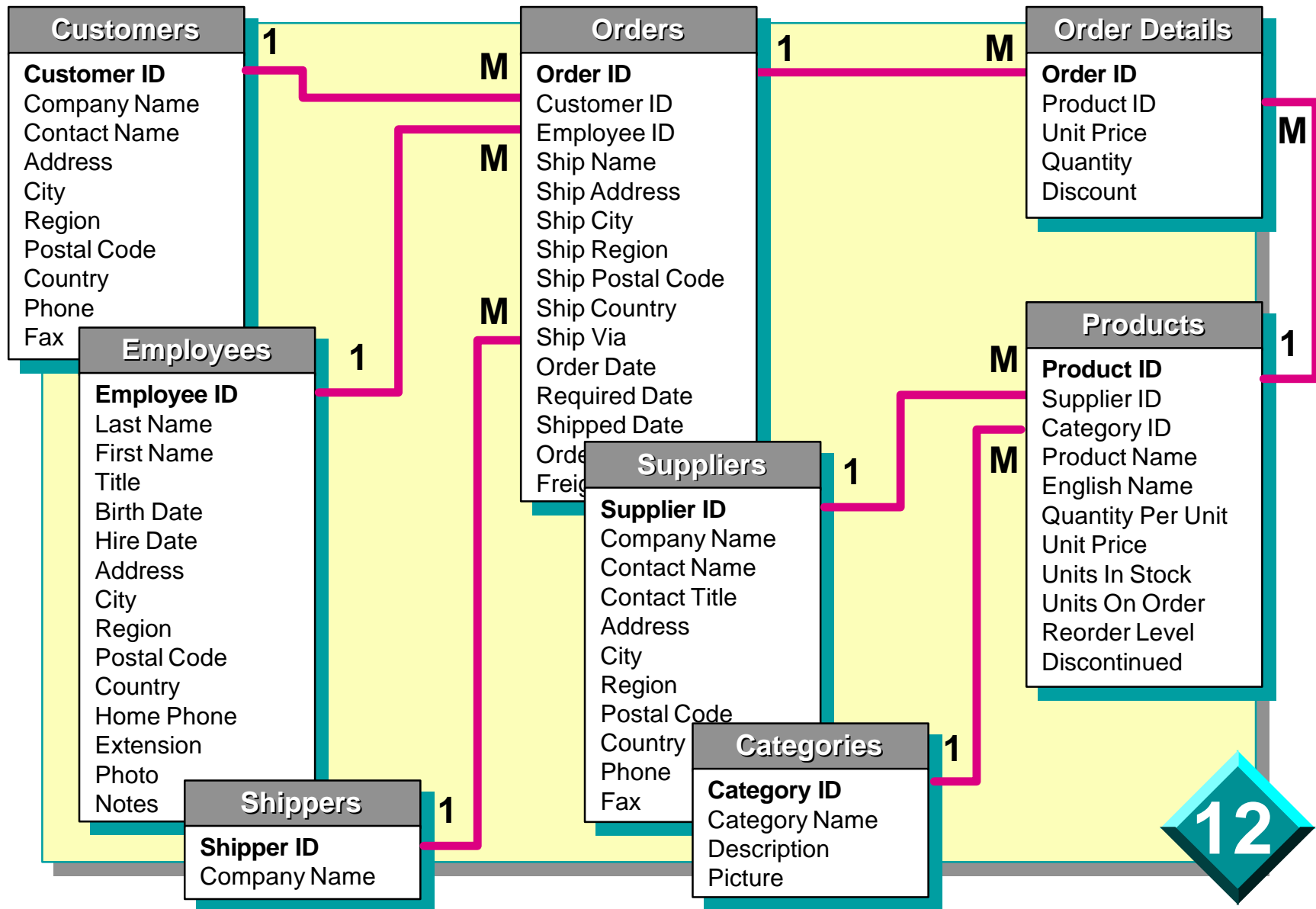
Employees Table

Employee ID	Last Name	First Name
135	Leverling	Tim
284	Buchanan	B.L.
⋮	⋮	⋮

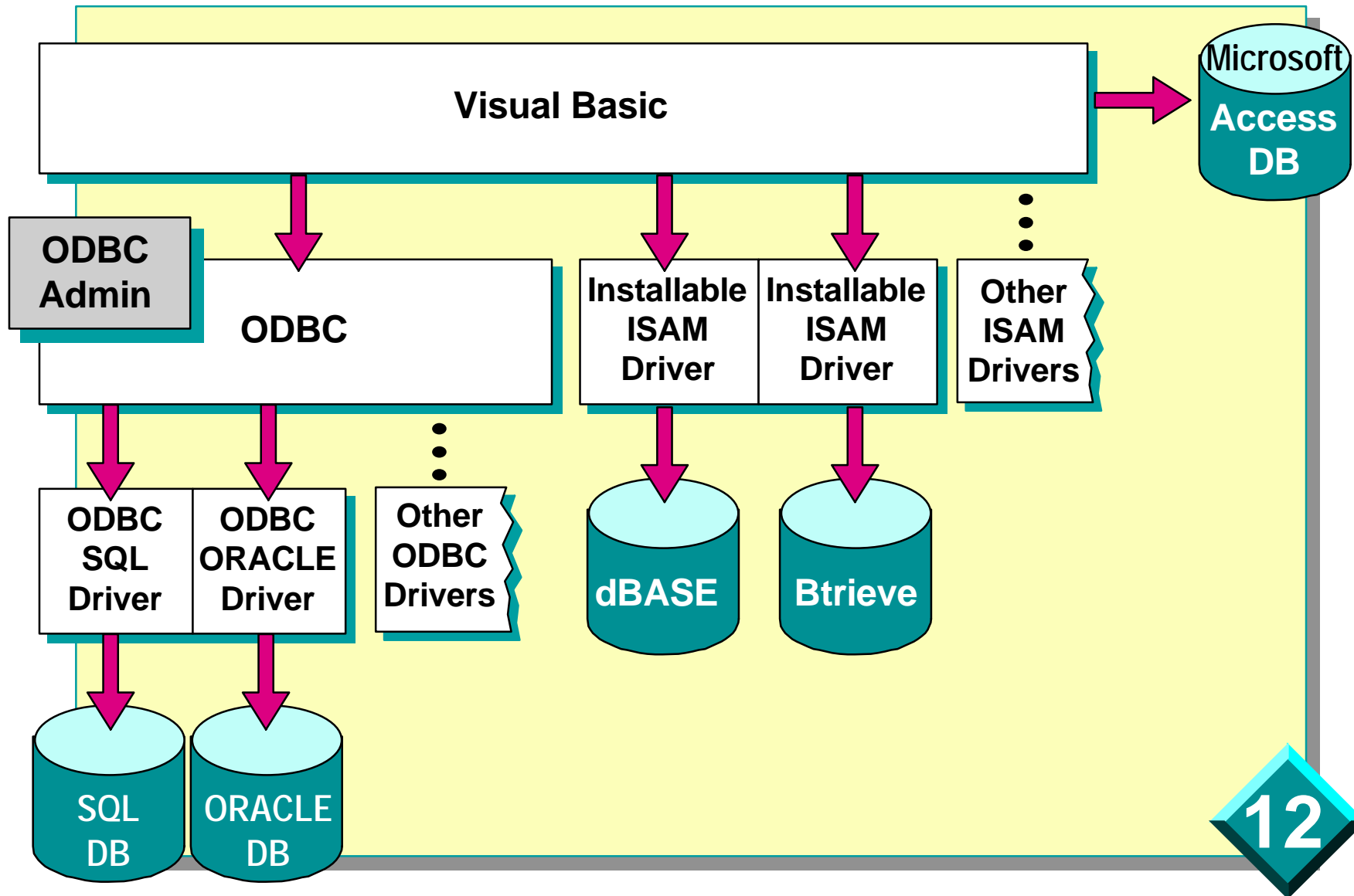
Rows (records)

Columns (fields)

Northwind Database Diagram

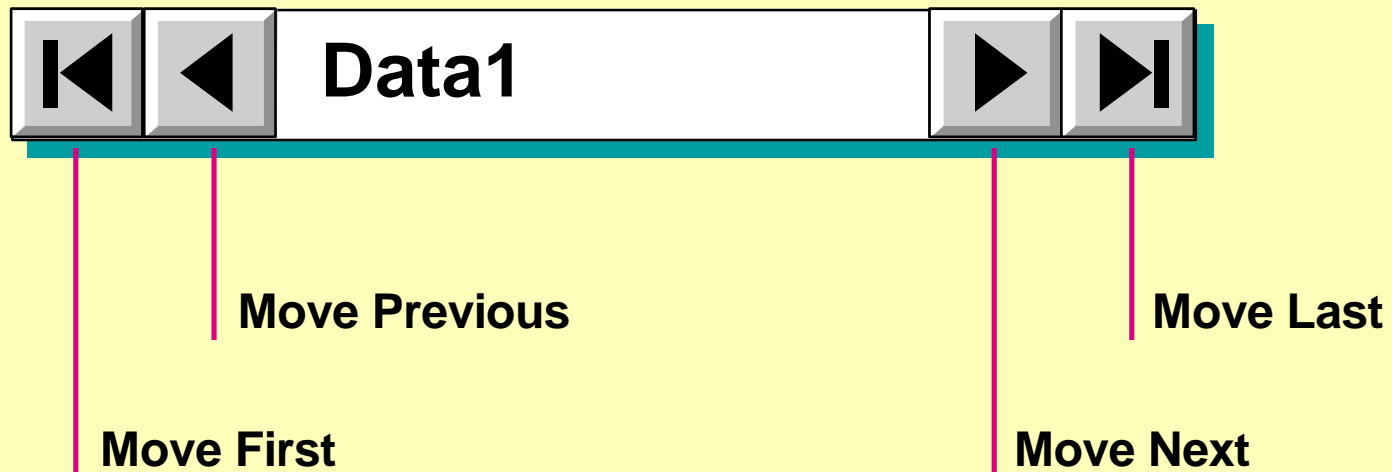


How Does Visual Basic Access Databases?

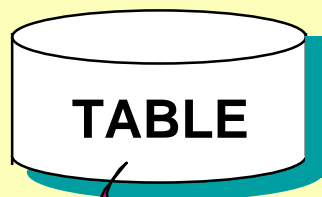


The Data Control

- Creates DYNASET
- Reflects DB Table or SQL Query



Characteristics of a DYNASET Object




**Populated 2048 Bytes
worth of Records at
a time on demand**

**Default ordered by
Primary Index Key
e.g. Emp ID**

Emp ID	Last Name	First Name
1		
2		
3		
4		
.		
.		
.		
.		
.		
.		
.		
N		

Binding Controls to the Data Control

```
.DataSource = Data1  
.DataField = 'Last Name'
```



Form1

Last Name:

◀ Data1 ▶

Data Control Walkthrough

The image shows a screenshot of a Windows application window titled "Form1". The window has a standard Windows 95-style title bar with a minimize button, a maximize button, and a close button. The main area of the form contains two text input fields. The first field is labeled "Last Name:" and the second field is labeled "First Name:". Below these fields is a data control labeled "Data1". The data control has a series of navigation buttons: a "Go to First" button (represented by two left-pointing arrows), a "Go to Previous" button (represented by a single left-pointing arrow), a "Go to Next" button (represented by a single right-pointing arrow), and a "Go to Last" button (represented by two right-pointing arrows).

Displaying a Calculated Field

The screenshot shows a form window titled "Form1". Inside the form, there are three input fields: "Quantity:", "Price:", and "Total:". Below these fields is a control labeled "Data1" with navigation buttons (back, forward, first, last). A pink arrow points from the "Data1" control to a text box containing a SQL query. Another pink arrow points from the text box to the "Total:" field.

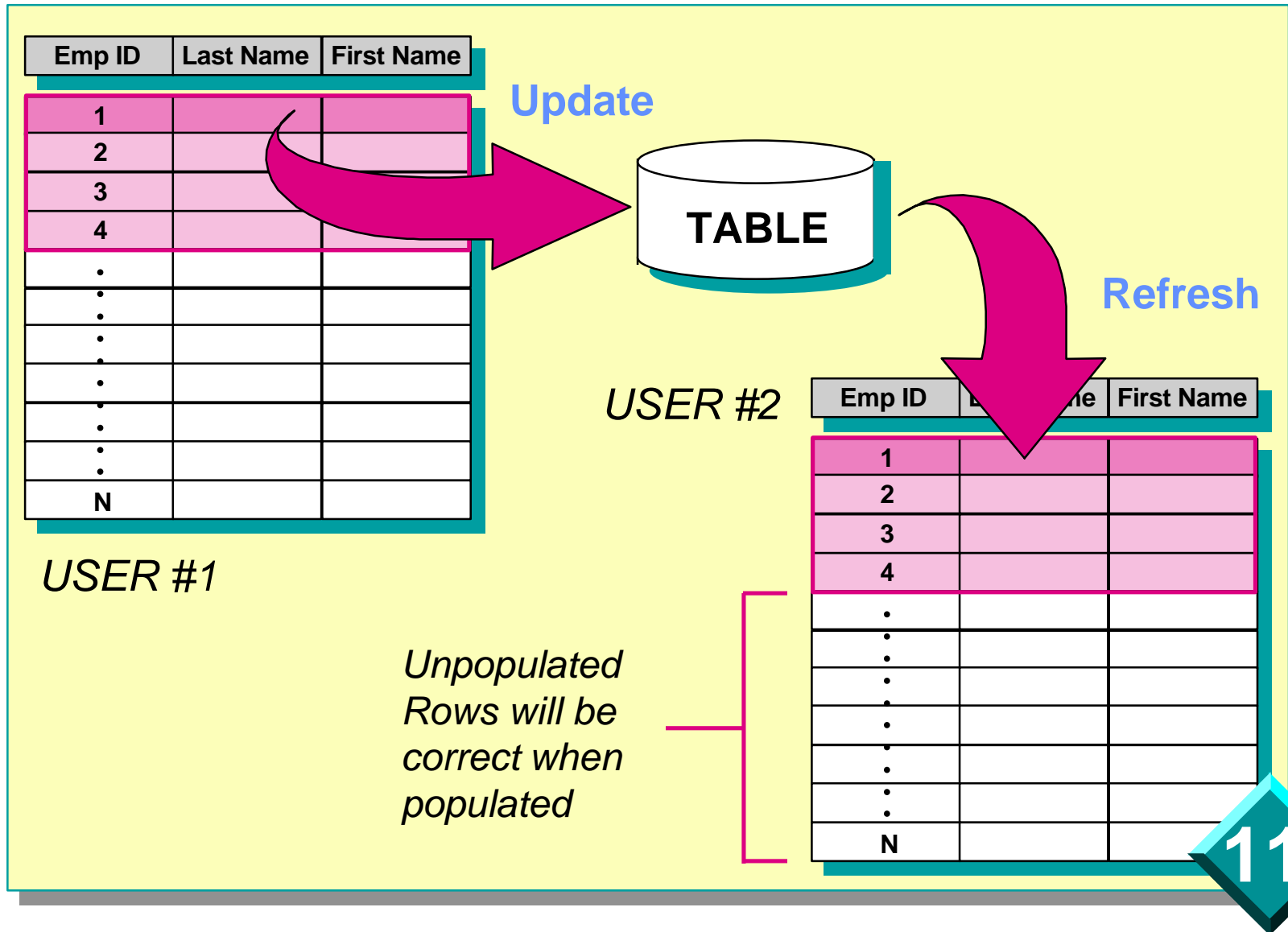
`total = [Unit Price] * Quantity`

```
.RecordSource = "Select [Unit Price], quantity,  
    ↪[Unit Price] * quantity As total from [Order Details]"
```

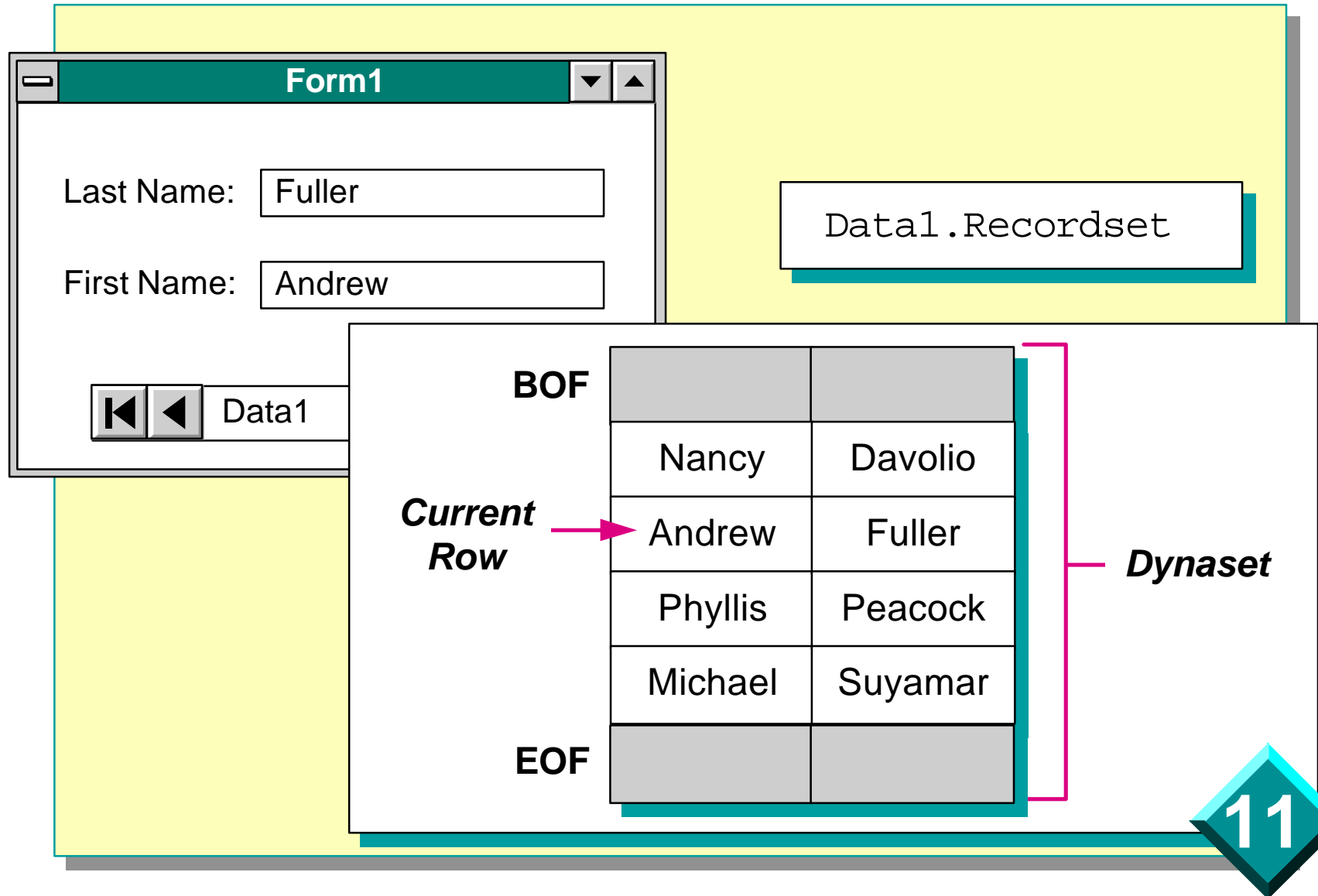
Data Control Methods and Properties

- Refresh Method
- Connect Property
- Exclusive Property
- ReadOnly Property

Multiple Concurrent TABLE Users



The Current Row and The Dynaset



Finding Records

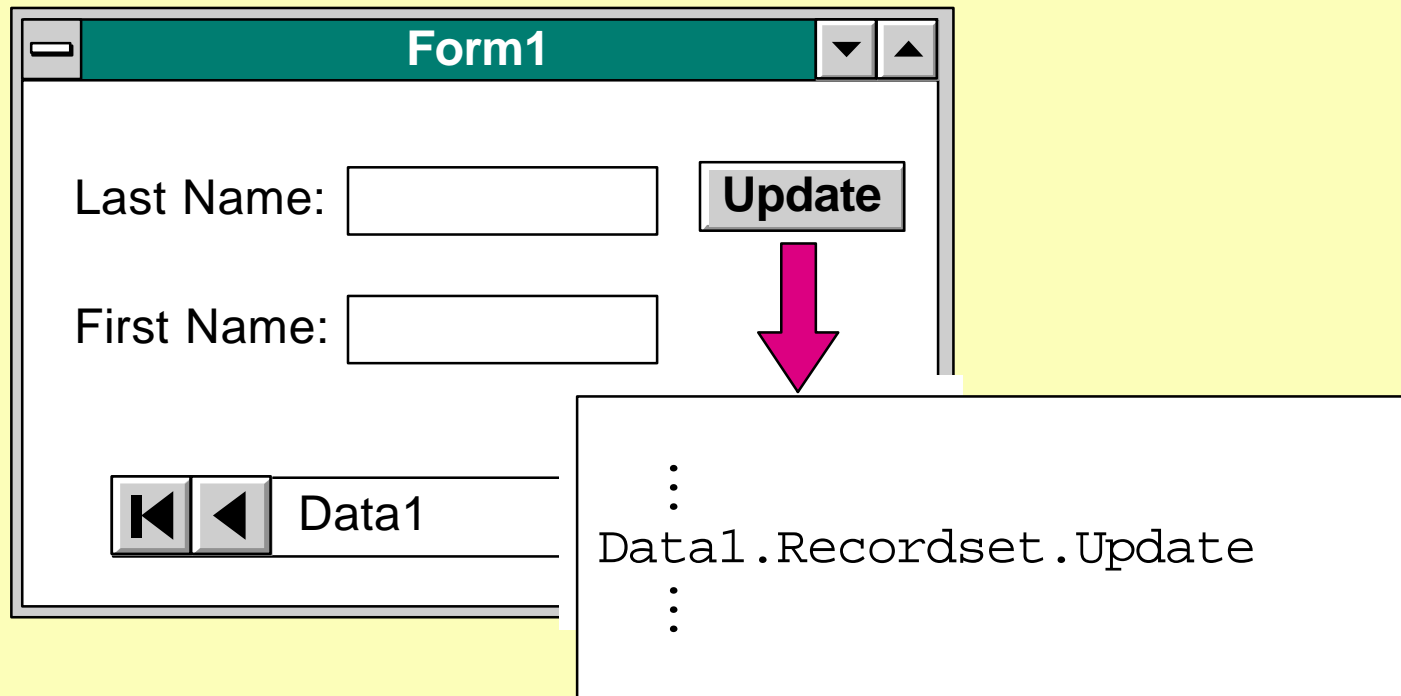
- Find Methods: `FindFirst`, `FindLast`, `FindPrevious`, `FindNext`

The screenshot shows a form window titled 'Form1'. It contains two text input fields labeled 'Last Name:' and 'First Name:'. To the right of the 'Last Name' field is a button labeled 'Find'. A large pink arrow points from the 'Find' button down towards a code block. Below the input fields are navigation buttons: a double left arrow, a single left arrow, and a 'D.' label.

```
⋮  
Data1.Recordset.FindFirst criteria  
⋮
```

- Criteria: Any Valid SQL “Where” Clause
- Use Bookmark Property to Save Current Position

Updating Records



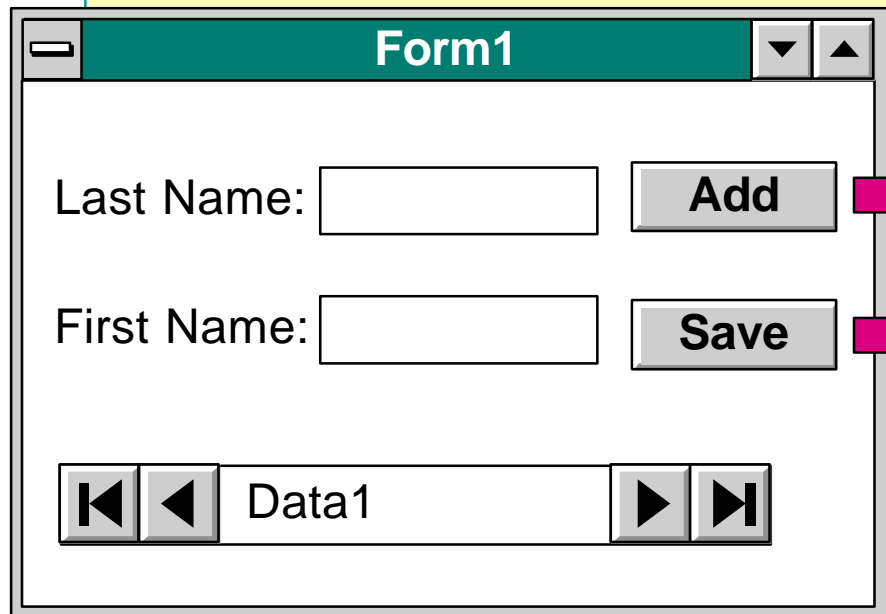
Deleting Records

The image shows a screenshot of a Visual Basic application window titled "Form1". Inside the form, there are two text boxes labeled "Last Name:" and "First Name:". To the right of the "Last Name:" text box is a button labeled "Delete". A large pink arrow points from the "Delete" button down to a code window. The code window contains the following code:

```
⋮  
Data1.Recordset.Delete  
⋮
```

Below the text boxes, there is a data grid control labeled "Data1" with navigation buttons (back, forward, etc.).

Adding New Records



A screenshot of a Windows-style window titled "Form1". Inside the window, there are two text input fields. The first is labeled "Last Name:" and the second is labeled "First Name:". To the right of the "Last Name:" field is a button labeled "Add". To the right of the "First Name:" field is a button labeled "Save". At the bottom of the window, there is a data control bar for "Data1" with navigation buttons (back, forward, first, last).



```
⋮  
Data1.Recordset.AddNew  
⋮
```

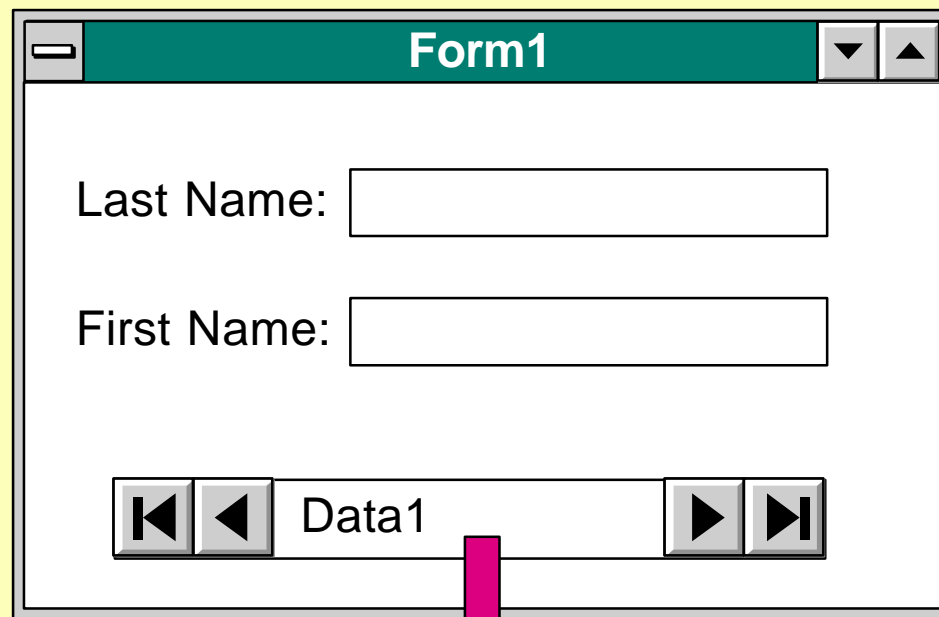


```
⋮  
Data1.Recordset.Update  
⋮
```

◆ Data Control Events

- Validate
- Reposition
- Error

The Validate Event



Form1

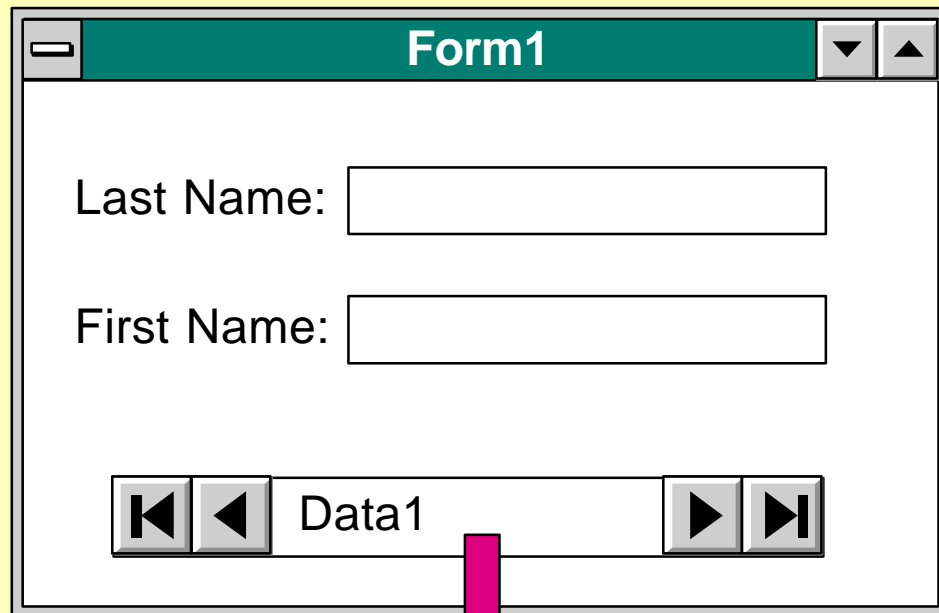
Last Name:

First Name:

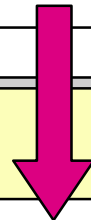
Data1

```
Sub Data1_Validate (Action As Integer, Save As Integer)  
    'Occurs when current record changes
```

The Reposition Event

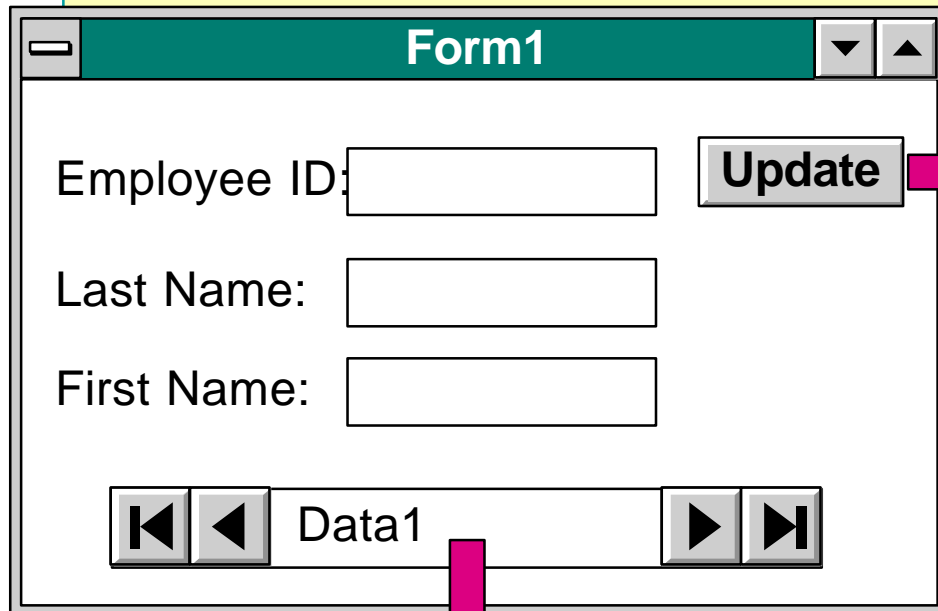


A screenshot of a Windows form titled "Form1". The form contains two text boxes labeled "Last Name:" and "First Name:". Below these is a control labeled "Data1" which has four navigation buttons: a double left arrow, a single left arrow, a single right arrow, and a double right arrow.



```
Sub Data1_Reposition()  
    'Occurs when current record changes
```

The Error Event



A screenshot of a Windows-style window titled "Form1". Inside the window, there are three text input fields labeled "Employee ID:", "Last Name:", and "First Name:". To the right of the "Employee ID:" field is a button labeled "Update". Below these fields is a data grid control labeled "Data1" with navigation buttons (back, forward, first, last) on its left and right sides. A pink arrow points from the "Update" button to a code block on the right, and another pink arrow points from the "Data1" control to a code block at the bottom.

```
Sub cmd_Update  
On Error...  
:  
'Handle Duplicate Key
```

```
Sub Data1_Error ()  
'If error occurs, for example: Duplicate Key
```

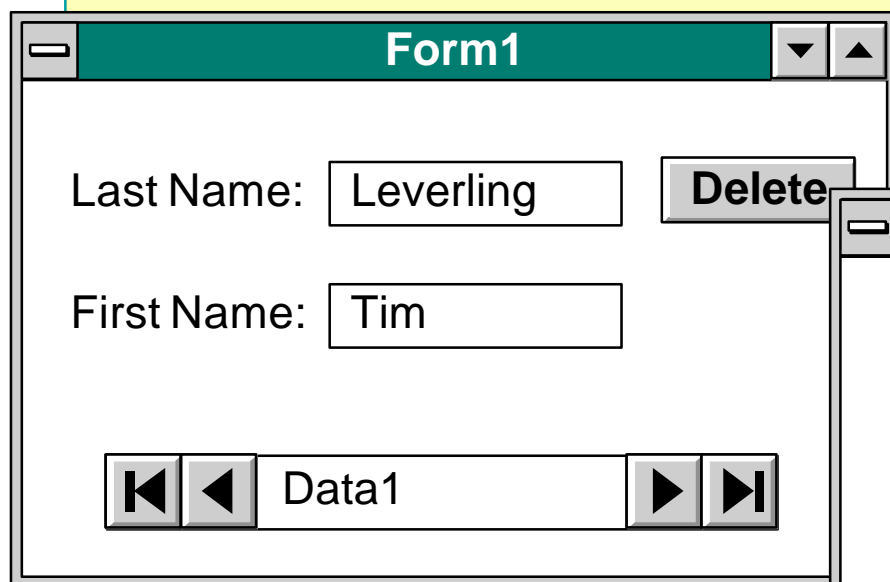
Handling Multiple Users

User #1

- Deletes "Leverling"

User #2

- "Record is Deleted" message when moving to "Leverling"



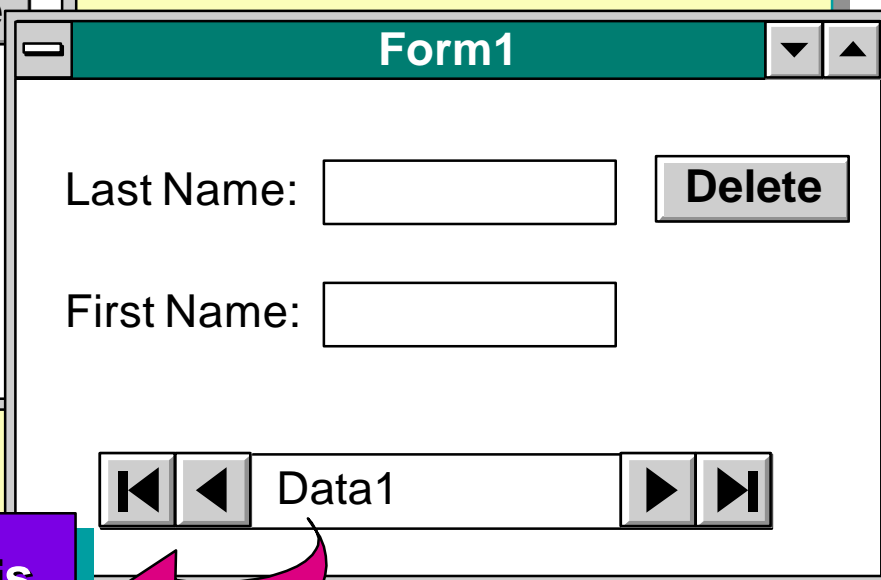
Form1

Last Name: Leverling

First Name: Tim

Data1

Buttons: Delete, Previous, Next, First, Last



Form1

Last Name:

First Name:

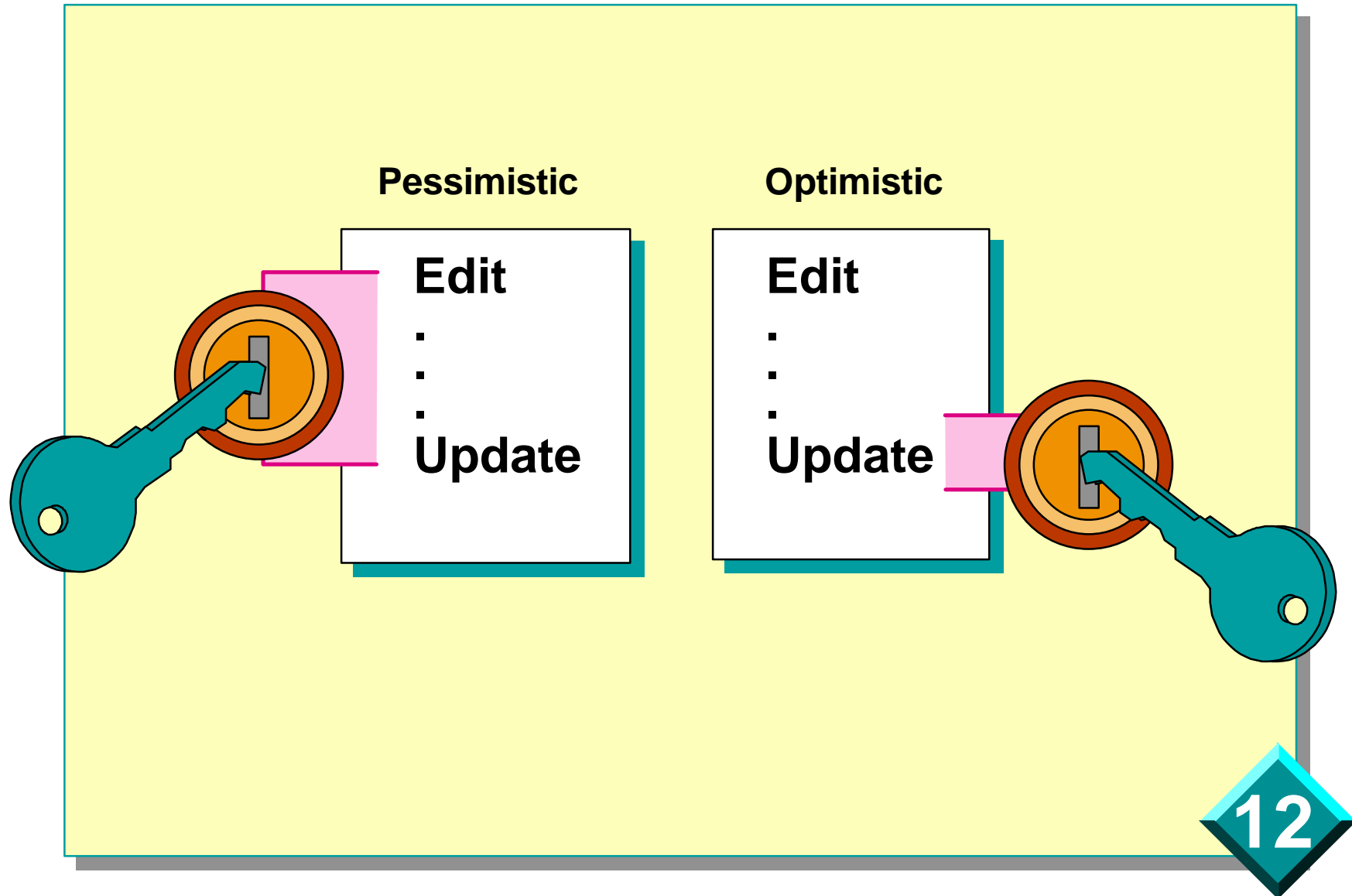
Data1

Buttons: Delete, Previous, Next, First, Last

"Record is Deleted"

12

How Locking Is Handled



Accessing Data Programmatically

```
Data1.Recordset("Last Name")  
Data1.Recordset![Last Name]  
Data1.Recordset.Fields(0).Value
```

Filling a Grid with Data

Single Record

Last Name:

First Name:

Multiple Records

Last Name	First Name
Leverling	Tim
Buchanan	B.L.
⋮	⋮

Note: Many 3rd Party Grid Control are Bound Controls

Determining the Files You Need to Distribute

■ Special DLLs Are Required For

- Data control
- Accessing Btrieve, Paradox, or Microsoft FoxPro® files
- Accessing ODBC format databases
- Accessing SQL Server (Professional Edition only)

Summary

- Overview of a Database
- How Does Visual Basic Access Databases?
- What Is the Data Control?
- The Current Row and the Dynaset
- Finding Records
- Updating, Deleting, and Adding Records
- Data Control Events
- How Locking Is Handled
- Filling a Grid with Data from a Table