

District Data Coordinator Toolbox: Implementing Database Connections in SPSS

Jason Schoeneberger, Ph.D.
Senior Researcher & Task Lead



Mid-Atlantic: Delaware, Maryland,
New Jersey, Pennsylvania, Washington, D.C.

Data, data, everywhere

The volume of and the push to make use educational data is growing:

- More people must become data savvy (teachers, coordinators, etc.)
- Leadership may request cyclical reporting to establish and monitor trends
- Little time to document business rules or standardize data storage practices
- Quality control can take time or be difficult to manage

Teachers, principals, administrators and analysts often have difficulty keeping pace.



Some familiar scenarios

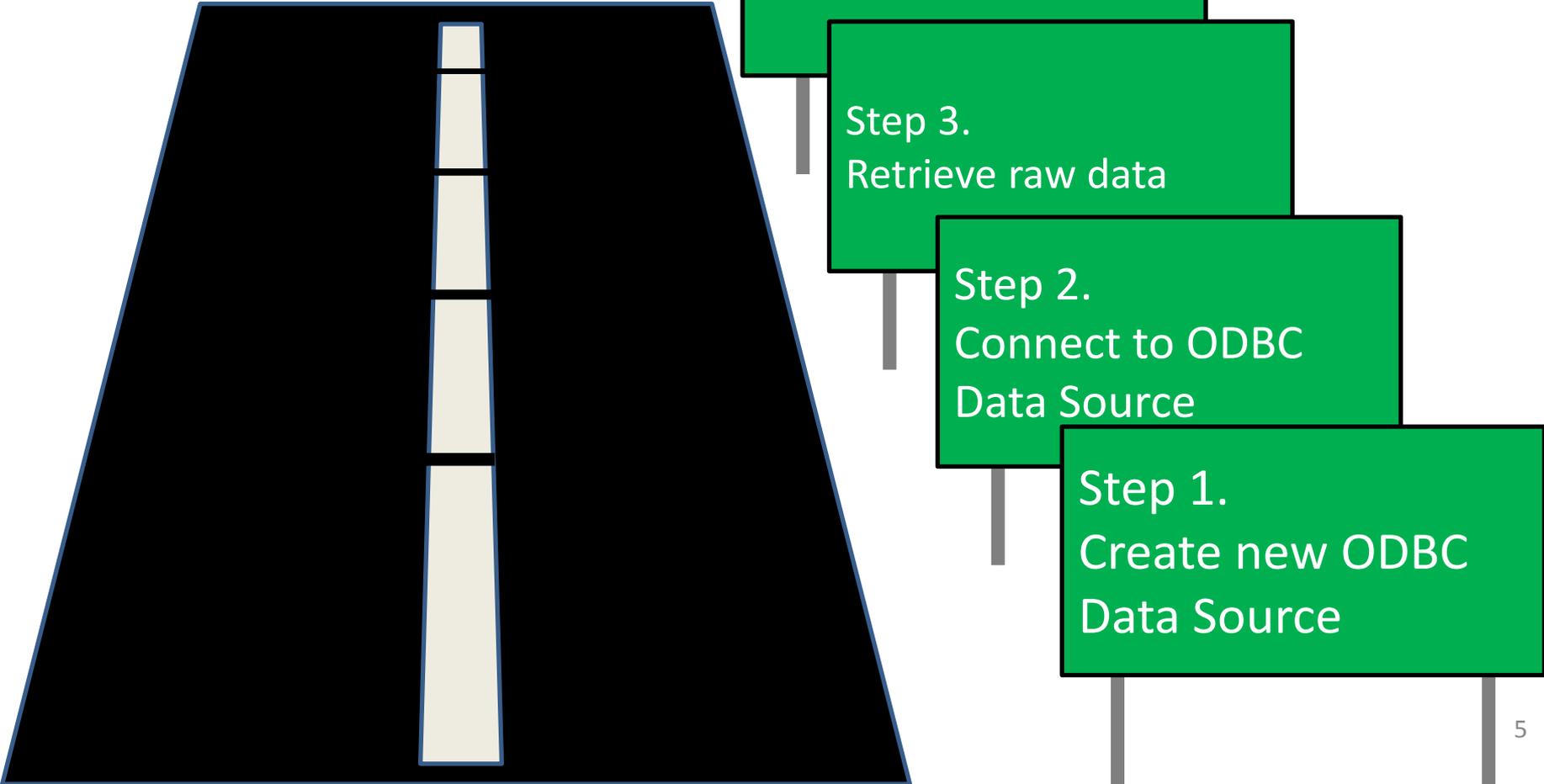
(using data stored in SQL, Oracle, Access, etc.)

- The same data points are necessary across reporting cycles
- Process to acquire and report data is repetitive across reporting cycles
- A non-technical person may be tasked with reporting responsibility
- Lack of documentation
- Analysts report shortage of storage space on network or external hard drives
- Analysts are maintaining idiosyncratic versions of various data elements (e.g. test score files, student attendance files, etc.)
- Idiosyncratic versions have commonalities across analyst versions
- Separate data requests completed by different analysts yield conflicting results (e.g. a school mean test score)

Database connections

- Databases (e.g., SQL, Oracle, Access, etc.) allow for basic data base connectivity:
 - Open Database Connectivity (ODBC)
 - Object Linking and Embedding Database (OLEDB)
 - These are often standard on computers
- ODBC/OLEDB connections are frameworks to allow data manipulation software (e.g. Excel, SPSS, SAS) to communicate with databases

Road map to data connectivity



Step 4.
Summarize raw data

Step 3.
Retrieve raw data

Step 2.
Connect to ODBC
Data Source

Step 1.
Create new ODBC
Data Source

Traveling the road by example

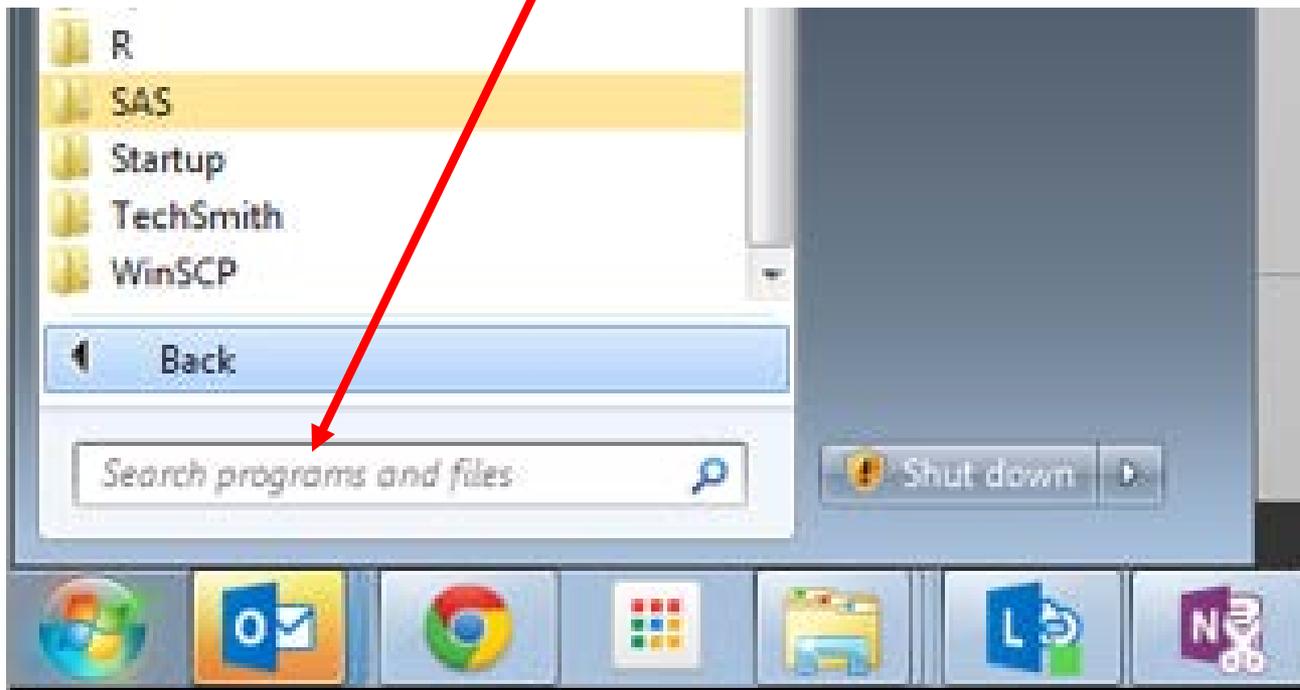
- To follow the steps in our road map to connectivity, let's assume the following example:
 - District leadership has asked us to examine reading achievement as measured by reading assessment achievement levels
 - Leadership is specifically interested in 6th grade student performance
 - They want to examine performance by student Limited English Proficient (LEP) status.
 - The data we need to obtain are stored in an Access database

Creating an ODBC data source

- The first step is to create an ODBC Data Source centered on an existing database such as Access, SQL, or Oracle. ODBC Data Sources are frameworks, or linkages for software packages such as SPSS to communicate with databases

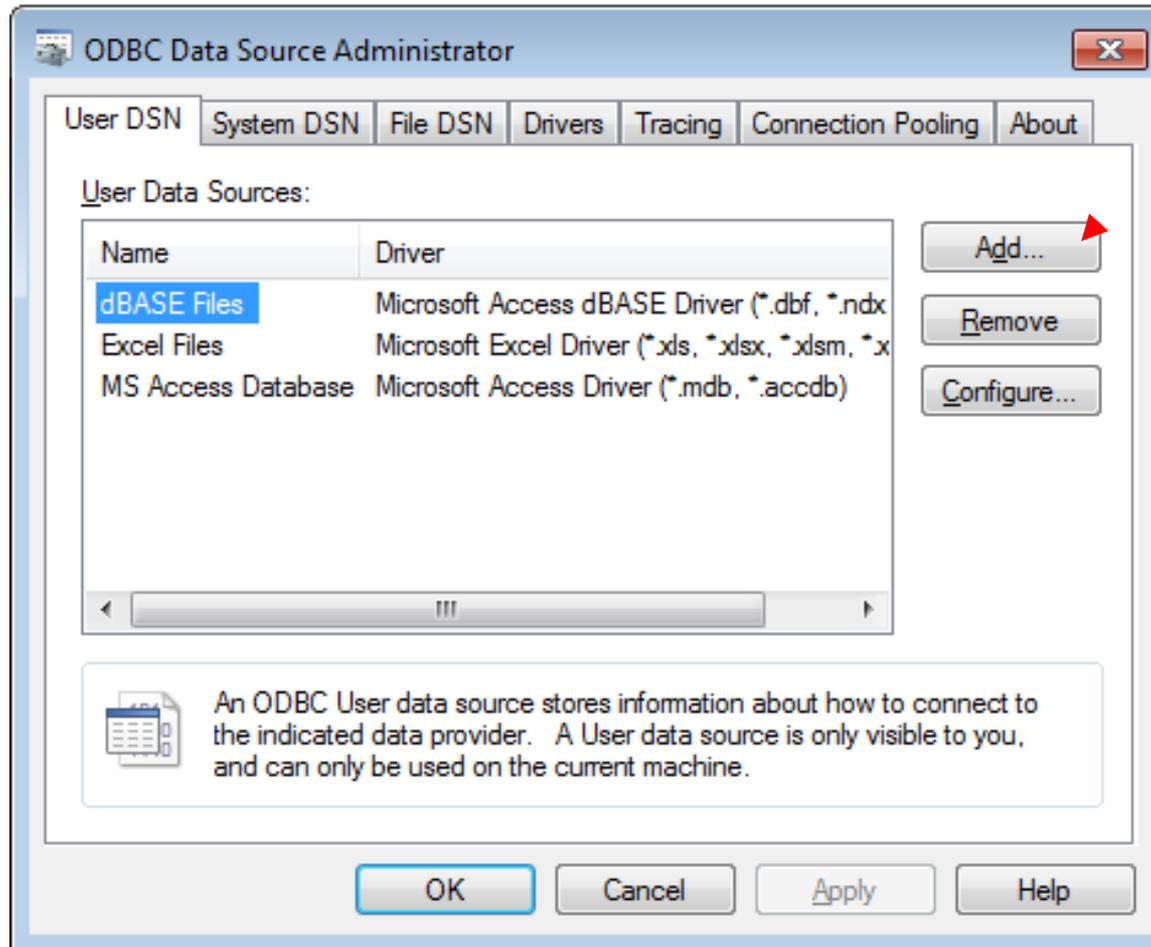
Open ODBC administrator window

- Type 'ODBC' in Search Box and press Enter



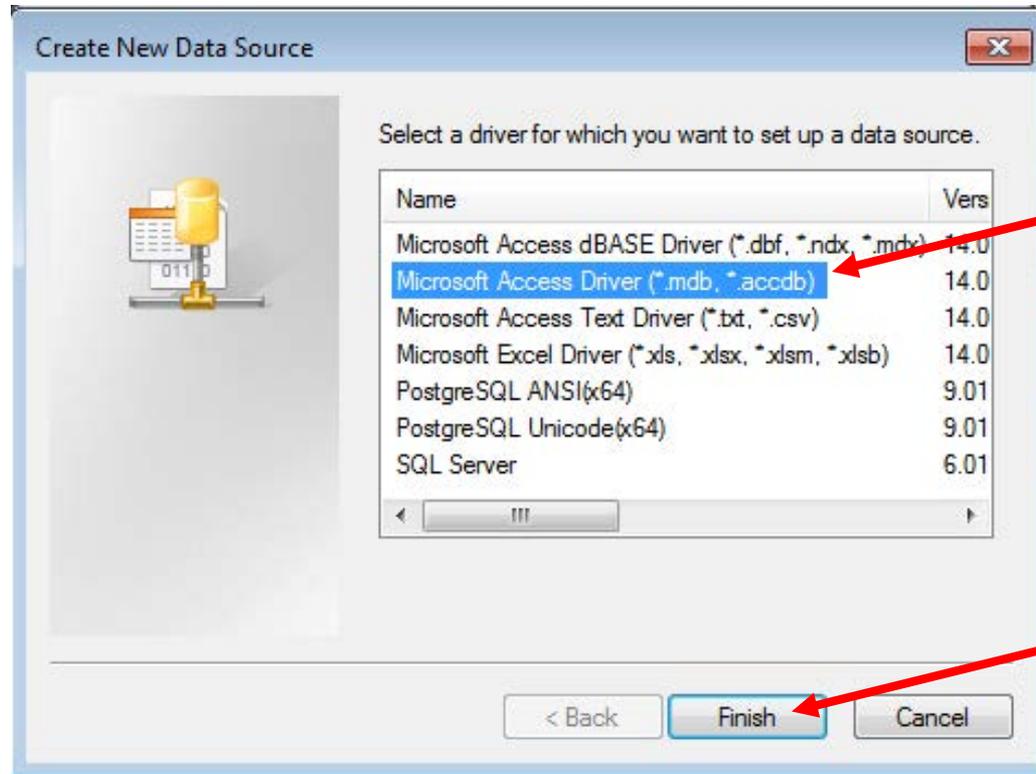
Add a new data source

- Click 'Add' to begin adding a new ODBC data source



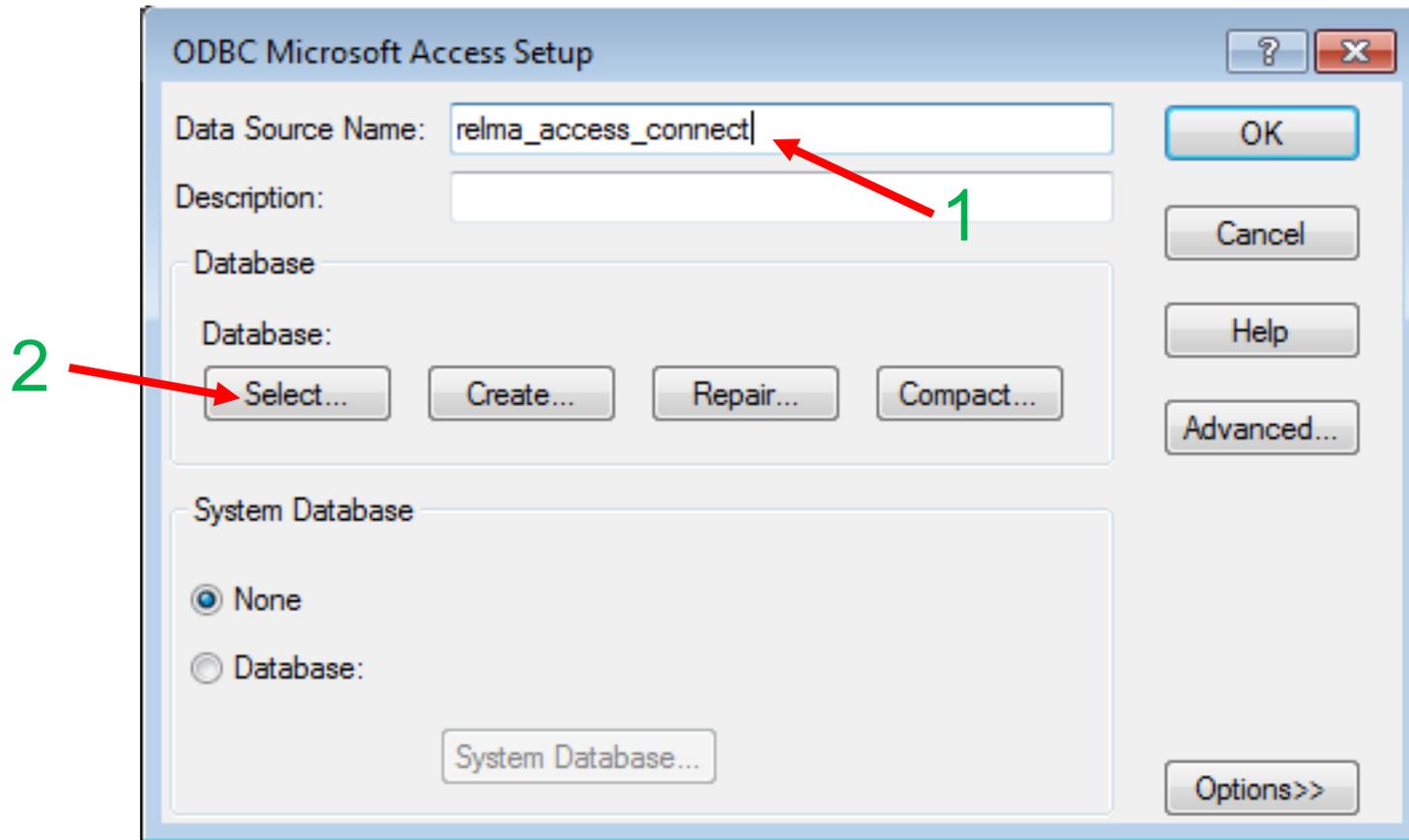
Choose a driver for the data source

1. Choose driver for connection to a source (in this example, we connect to an Access database)
2. Click Finish



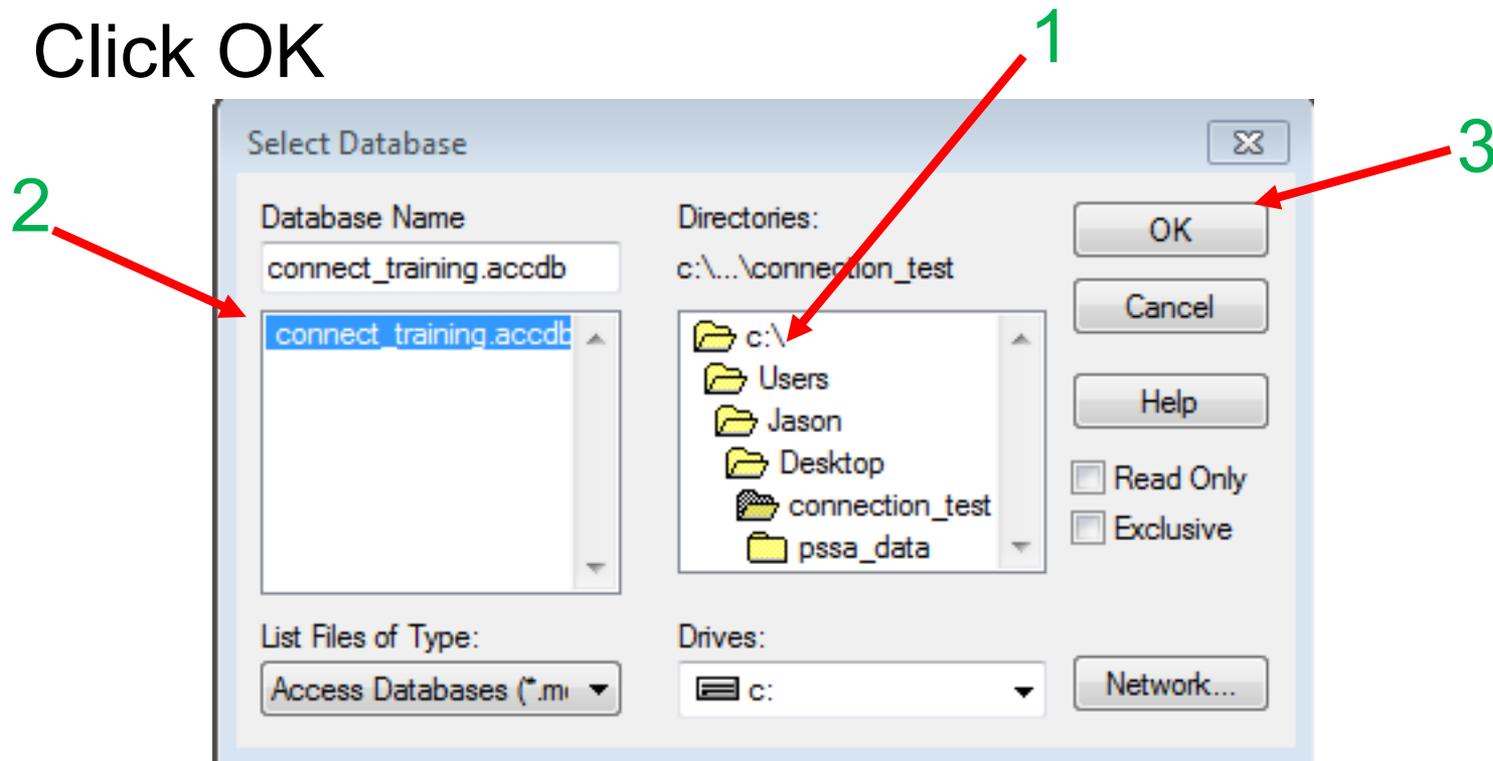
Name the database connection

1. Name the connection to the database
2. Click 'Select' button under Database



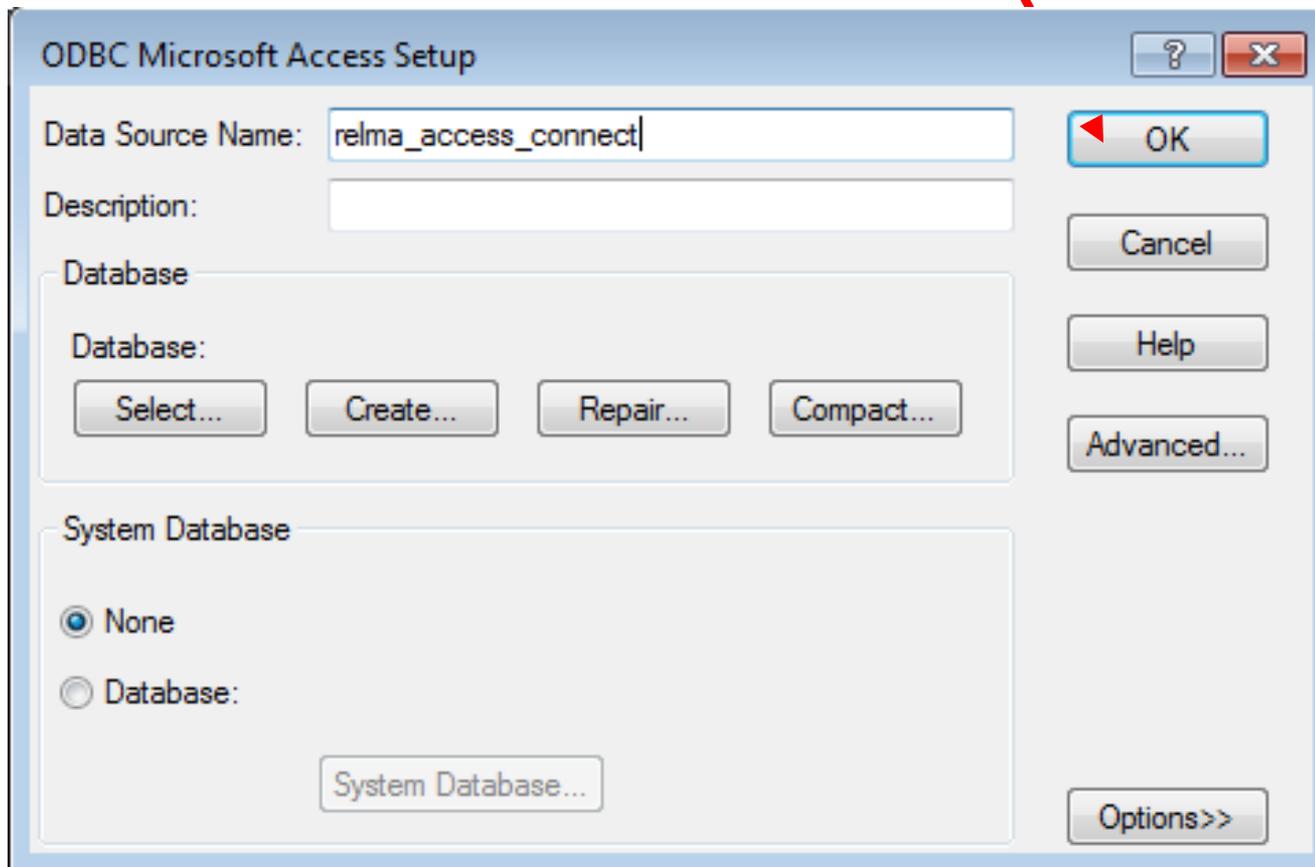
Select source database

1. Navigate to location of the database (the Access database we want to connect to in this example)
2. Select source database
3. Click OK



Click OK for data source name

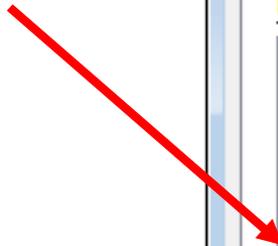
- Click OK button under Database



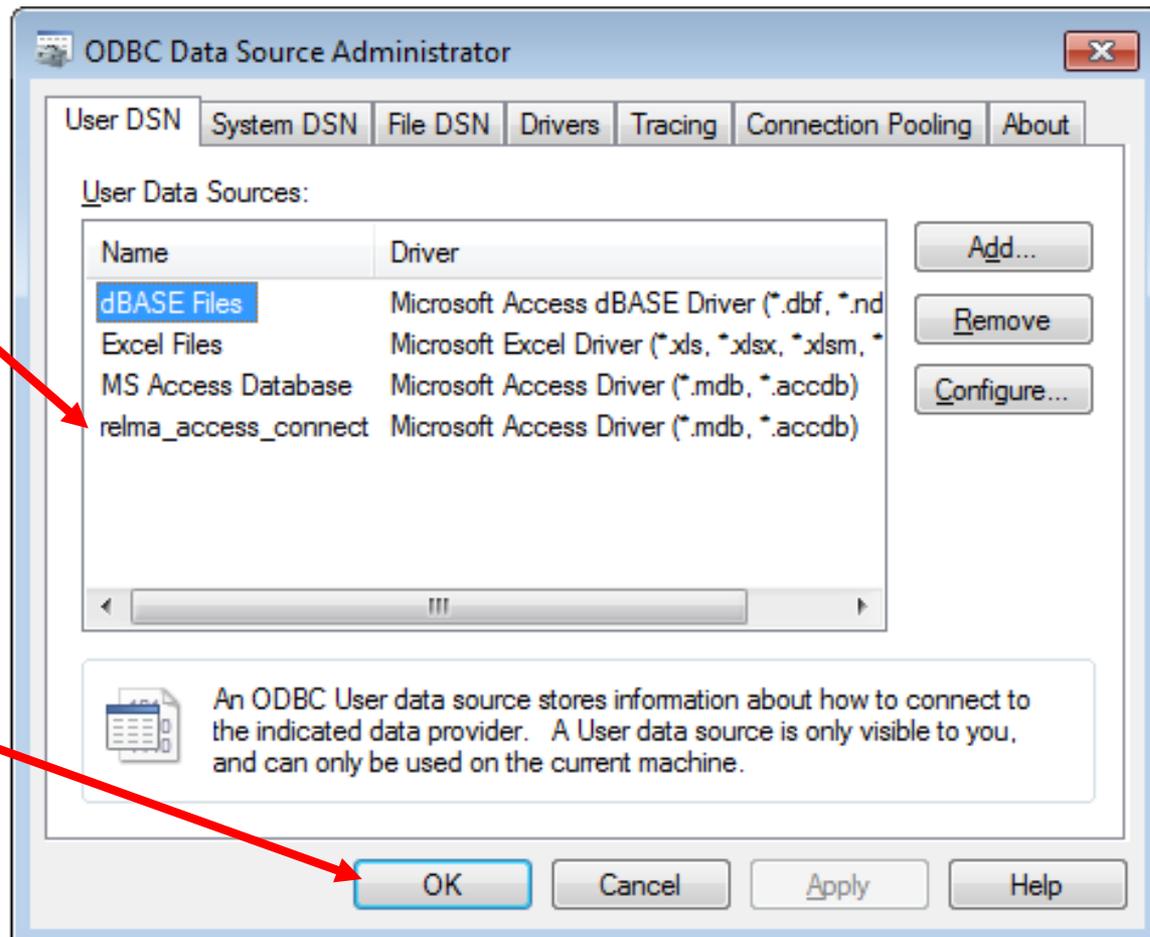
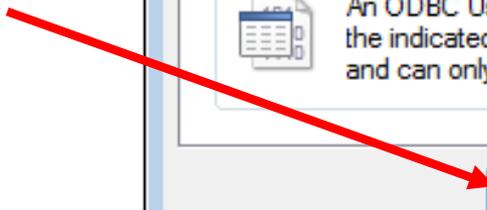
New data source appears in ODBC directory

1. New data source listed in ODBC directory
2. Click OK

1



2

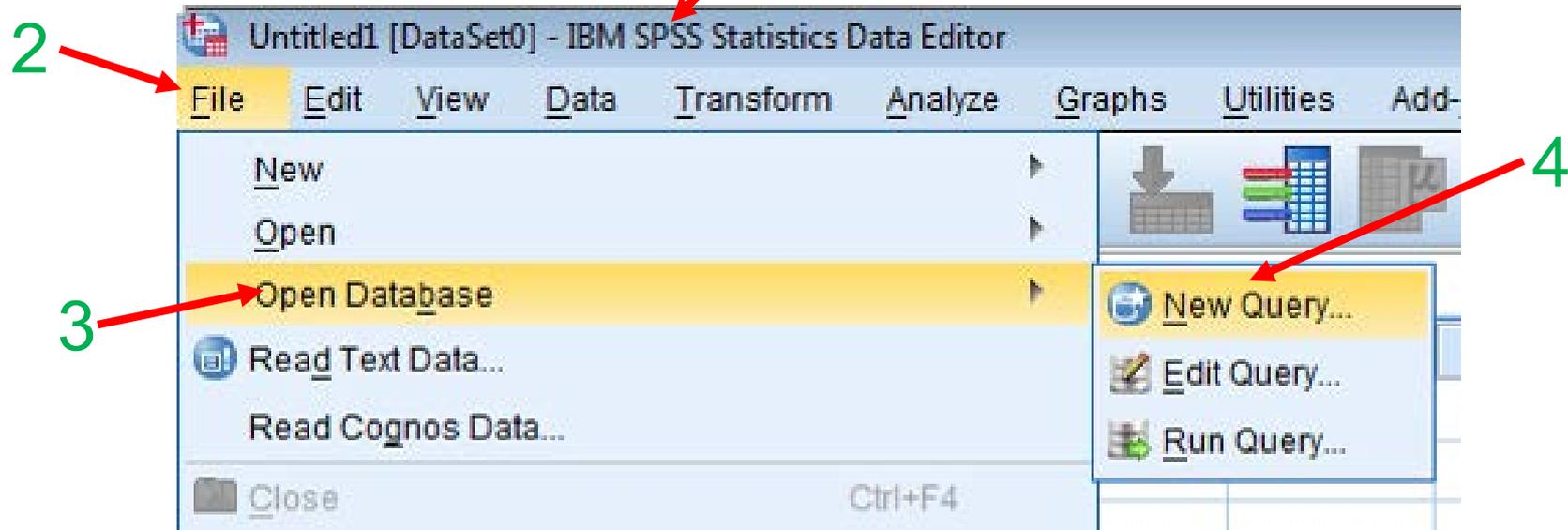


Connecting to a Database

- Now that our ODBC data source exists for communicating with the database, the information in the database can be extracted directly into other software packages (e.g. SPSS) for further manipulation

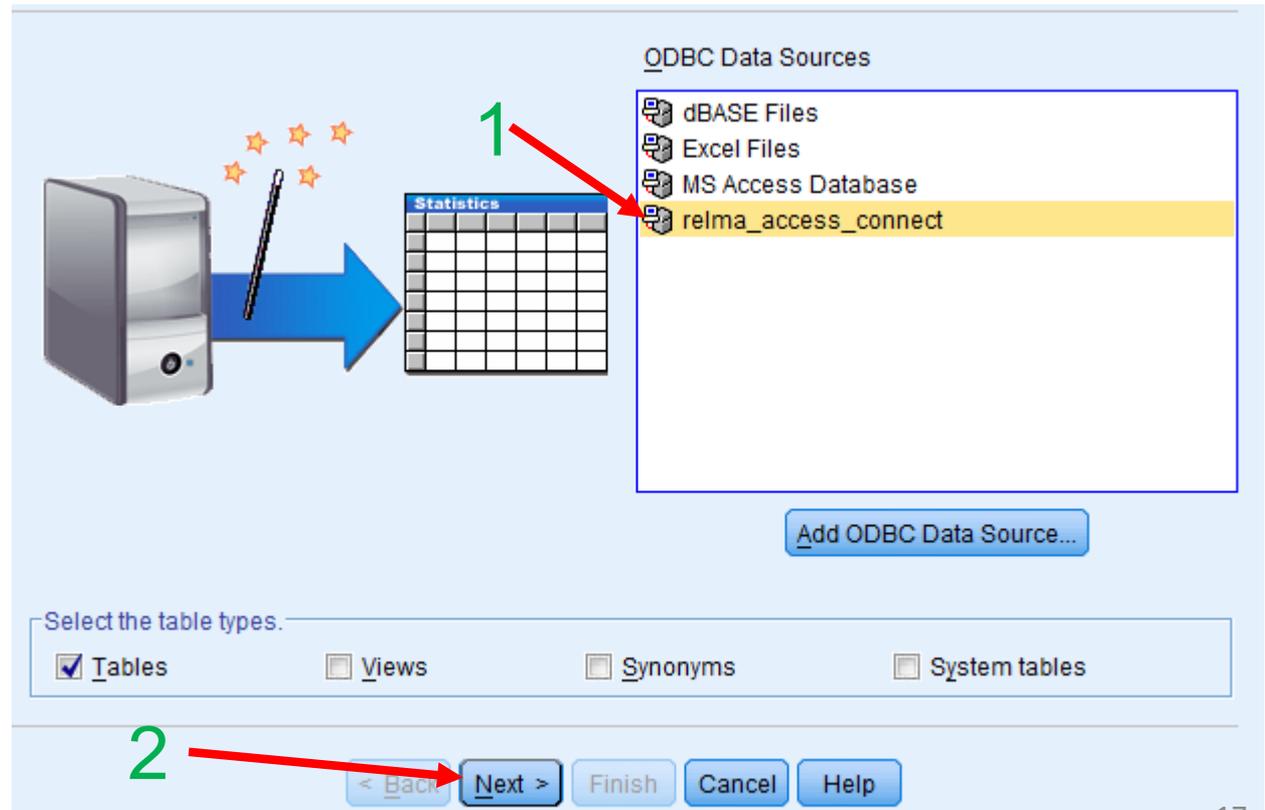
Connect to database using SPSS

1. Open an instance of SPSS (screenshots use SPSS 22.0)
2. Navigate to the File menu in SPSS
3. Click on 'Open Database'
4. Select 'New Query'



Choose your data source

1. Select the data source of interest (in this example, the relma_access_connect data source)
2. Click Next



View available tables in data source

- Now we can view data tables in the database
 - student_assessments
 - student_demographics
- Press ‘+’ button to list variables in student_demographics table

Select Data

Select the fields you want to retrieve. Then click the arrow button or drag the fields to the Retrieve Fields list.

Tip: Selecting a table selects all of its fields.

Available Tables:

student_assessments
 student_demographics

Retrieve Fields in this Order:

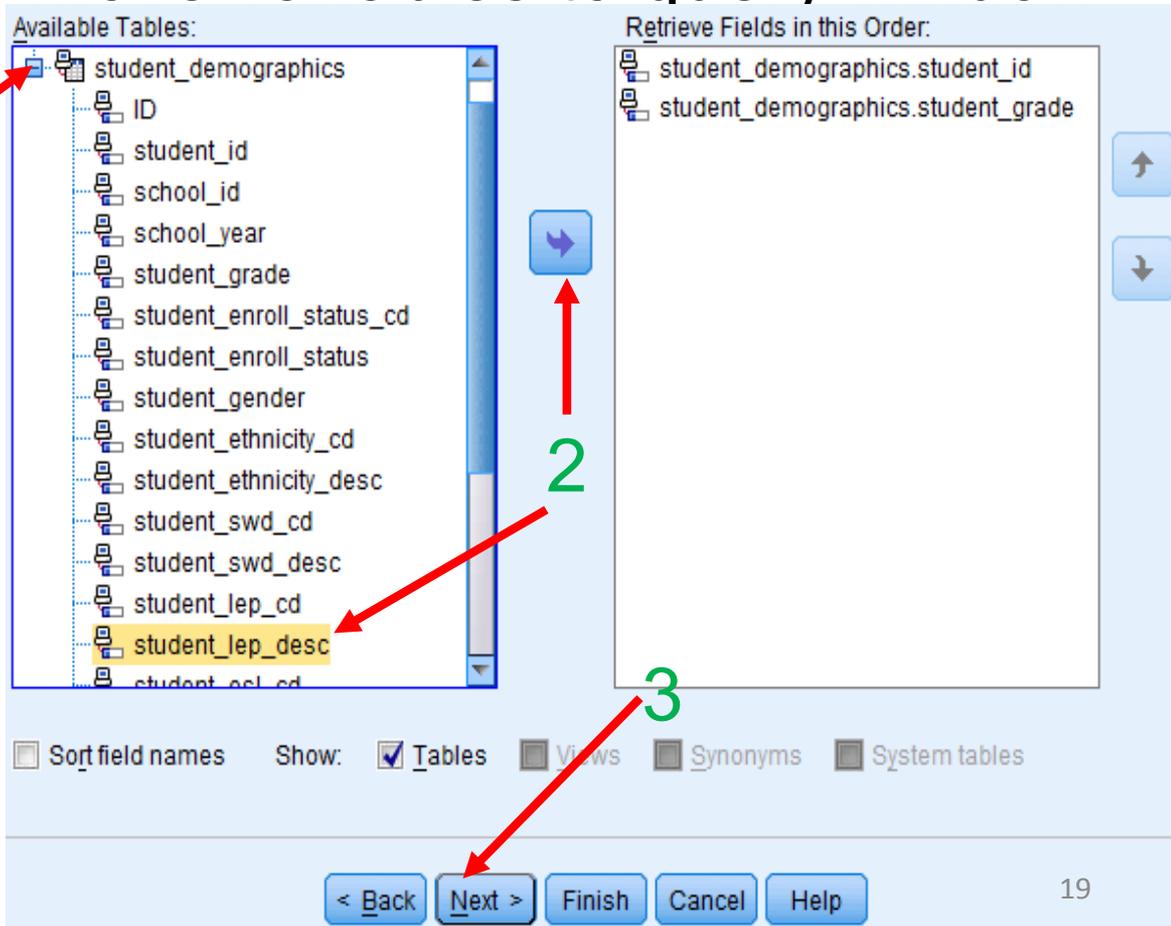
Sort field names Show: Tables Views Synonyms System tables

< Back Next > Finish Cancel Help

Select demographic variables

1. Click on variables in student_demographics table
2. Click  arrow to move variables to query window

- a) Student_id
(student identification number)
- b) Student_grade
(student grade level)
- c) Student_lep_desc
(student LEP status description)



Available Tables:

- student_demographics
 - ID
 - student_id
 - school_id
 - school_year
 - student_grade
 - student_enroll_status_cd
 - student_enroll_status
 - student_gender
 - student_ethnicity_cd
 - student_ethnicity_desc
 - student_swd_cd
 - student_swd_desc
 - student_lep_cd
 - student_lep_desc
 - student_race_cd

Retrieve Fields in this Order:

- student_demographics.student_id
- student_demographics.student_grade

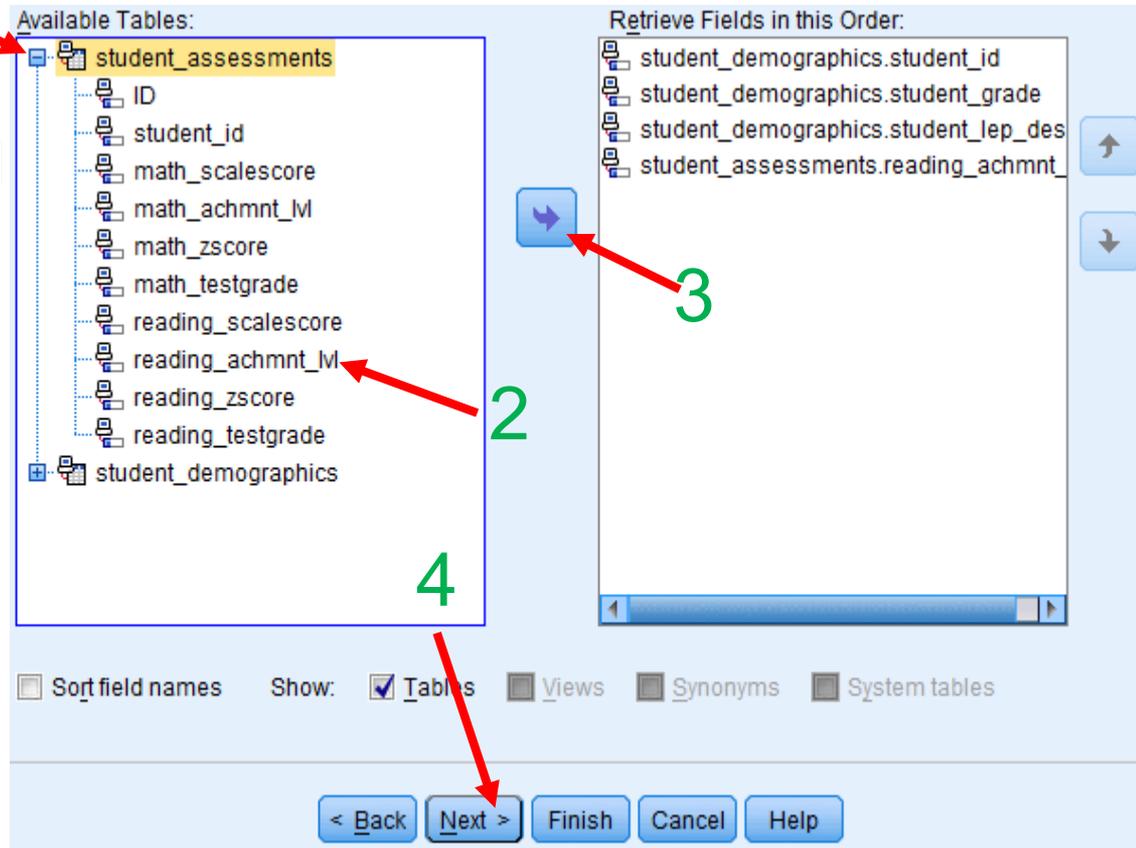
Sort field names Show: Tables Views Synonyms System tables

< Back Next > Finish Cancel Help

3. Click Next

Select assessment variables

1. Press '+' to list variables in student_assessments table
2. Select reading_achmnt_lvl from student_assessments table
3. Click  to move variable to query window
4. Click Next



Adding/deleting relationships

1. Auto Join will find variables in common among tables and join them automatically
2. Un-Check the Auto Join Tables button (all joins are removed)

Specify Relationships

In order to retrieve data from multiple tables, you must specify how to join them. To indicate a join, first select the type of join that you want. Then drag a field from one table over the corresponding field in the other table. Alternatively, you can select the two fields and press the Join button.

The screenshot shows the 'Specify Relationships' dialog box with two tables displayed:

student_demographics		student_assessments	
ID	COUNTER	ID	COUNTER
school_id	DOUBLE	math_achmnt_lm	DOUBLE
school_year	DOUBLE	math_scalescore	DOUBLE
student_birthdate	DATETIME	math_testgrade	DOUBLE
student_enroll_status	VARCHAR	math_zscore	DOUBLE
student_enroll_status_cd	VARCHAR	reading_achmnt_lm	DOUBLE
student_entry_cd	VARCHAR	reading_scalescore	DOUBLE
student_entry_date	DATETIME	reading_testgrade	DOUBLE
student_esl_cd	VARCHAR	reading_zscore	DOUBLE
student_esl_desc	VARCHAR	student_id	VARCHAR
student_ethnicity_cd	VARCHAR		
student_ethnicity_desc	VARCHAR		
student_gender	VARCHAR		
student_gifted_cd	VARCHAR		
student_grade	VARCHAR		
student_id	VARCHAR		
student_lep_cd	VARCHAR		

At the bottom of the dialog, the 'Join Type' is set to 'Inner'. The 'Join' button is visible. The 'Auto Join Tables' checkbox is checked, and a red arrow labeled '2' points to it. A green '1' is also present in the center of the dialog.

Join types

1. Change type of join using drop-down button

- Inner Join returns *only* records with matching student_ids in both tables
- Right Join returns *all* records from student_assessments and records with matches in student_demographics
- Left Join returns *all* records from student_demographics and records with matches in student_assessments

Specify Relationships

In order to retrieve data from multiple tables, you must specify how to join them. To indicate a join, first select the type of join that you want. Then drag a field from one table over the corresponding field in the other table. Alternatively, you can select the two fields and press the Join button.

Join Type: Auto Join Tables

< Back Next > Finish Cancel Help

Specifying join types

1. Click and hold on variable in one table, drag mouse to corresponding field in other table (in this example, create relationship from student_id-to-student_id)
2. Be sure that 'Inner' is specified in the Joint Type dropdown box
3. Click Next

The screenshot shows a software interface for defining a relationship between two tables: **student_demographics** and **student_assessments**.

student_demographics		student_assessments	
ID	COUNTER	ID	COUNTER
school_id	DOUBLE	math_achmnt_lvl	DOUBLE
school_year	DOUBLE	math_scalescore	DOUBLE
student_birthdate	DATETIME	math_testgrade	DOUBLE
student_enroll_status	VARCHAR	math_zscore	DOUBLE
student_enroll_status_cd	VARCHAR	reading_achmnt_lvl	DOUBLE
student_entry_cd	VARCHAR	reading_scalescore	DOUBLE
student_entry_date	DATETIME	reading_testgrade	DOUBLE
student_esl_cd	VARCHAR	reading_zscore	DOUBLE
student_esl_desc	VARCHAR	student_id	VARCHAR
student_ethnicity_cd	VARCHAR		
student_ethnicity_desc	VARCHAR		
student_gender	VARCHAR		
student_gifted_cd	VARCHAR		
student_grade	VARCHAR		
student_id	VARCHAR		
student_lep_cd	VARCHAR		

At the bottom of the interface, the **Join Type** dropdown menu is open, showing the following options:

- Inner (selected)
- Right Outer
- Left Outer

Buttons for **Join**, **Next >**, **Finish**, **Cancel**, and **Help** are visible at the bottom right. An **Auto Join Tables** checkbox is also present.

Filter data during data retrieval

- Returned data can be filtered during retrieval

1. Select student_grade from Column to filter window
2. Use drop-down to select criteria operation ('=')
3. Enter '06' to return only 6th grade student data
4. Click Next

The screenshot shows the SPSS Filter dialog box. The 'Fields:' list on the left has 'student_grade' selected, indicated by a red arrow labeled '1'. The 'Criteria:' table has 'student_demographics.s...' in the 'Expression 1' column, '=' in the 'Relation' column, and ''06'' in the 'Expression 2' column. A red arrow labeled '2' points to the 'Relation' column, and another red arrow labeled '3' points to the 'Expression 2' column. The 'Functions:' list is visible below the fields. At the bottom, the 'Next >' button is highlighted with a red arrow labeled '4'. The 'Finish' button is also visible.

Criteria:	Expression 1	Relation	Expression 2
1	student_demographics.s...	=	'06'

Define variables during data retrieval

- The query can recode character string data to numeric during retrieval

1. Check the box to recode chosen variables to numeric (not in this example)
2. Change width for character string variables (or minimize based on observed values)
3. Click Next

	Result Variable Name	Data Type	Recode to Numeric
student_demographics.student_id	student_id	String	<input type="checkbox"/>
student_demographics.student_grade	student_grade	String	<input type="checkbox"/>
student_demographics.student_lep_de...	student_lep_desc	String	<input type="checkbox"/>
student_assessments.reading_achmnt...	reading_achmnt_M	Numeric	

Width for variable-width string fields: Minimize string widths based on observed values

< Back Next > Finish Cancel Help

Retrieving results

- The query wizard generates SQL code for use inside SPSS syntax windows

1. First option will pull data directly into SPSS
2. Second option will paste in SPSS syntax window
3. Click Finish

The screenshot shows the 'Query Wizard' dialog box in SPSS. The top section contains the generated SQL code: `SELECT T0.student_id, T0.student_grade, T0.student_lep_desc, T1.reading_achmnt_M FROM student_demographics T0, student_assessments T1 WHERE T0.student_id = T1.student_id AND (T0.student_grade = '06')`. Below the code, the question 'What would you like to do with this query?' is followed by two radio button options: 'Retrieve the data I have selected.' (which is selected) and 'Paste it into the syntax editor for further modification'. A third option, 'Save query to file', is partially visible with an empty text box and a 'Browse...' button. At the bottom, there are navigation buttons: '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'. Red arrows with numbers 1, 2, and 3 point to the selected radio button, the 'Paste it into the syntax editor...' option, and the 'Finish' button, respectively.

Retrieved data in table form

- Resulting data returned to SPSS, ready for analysis



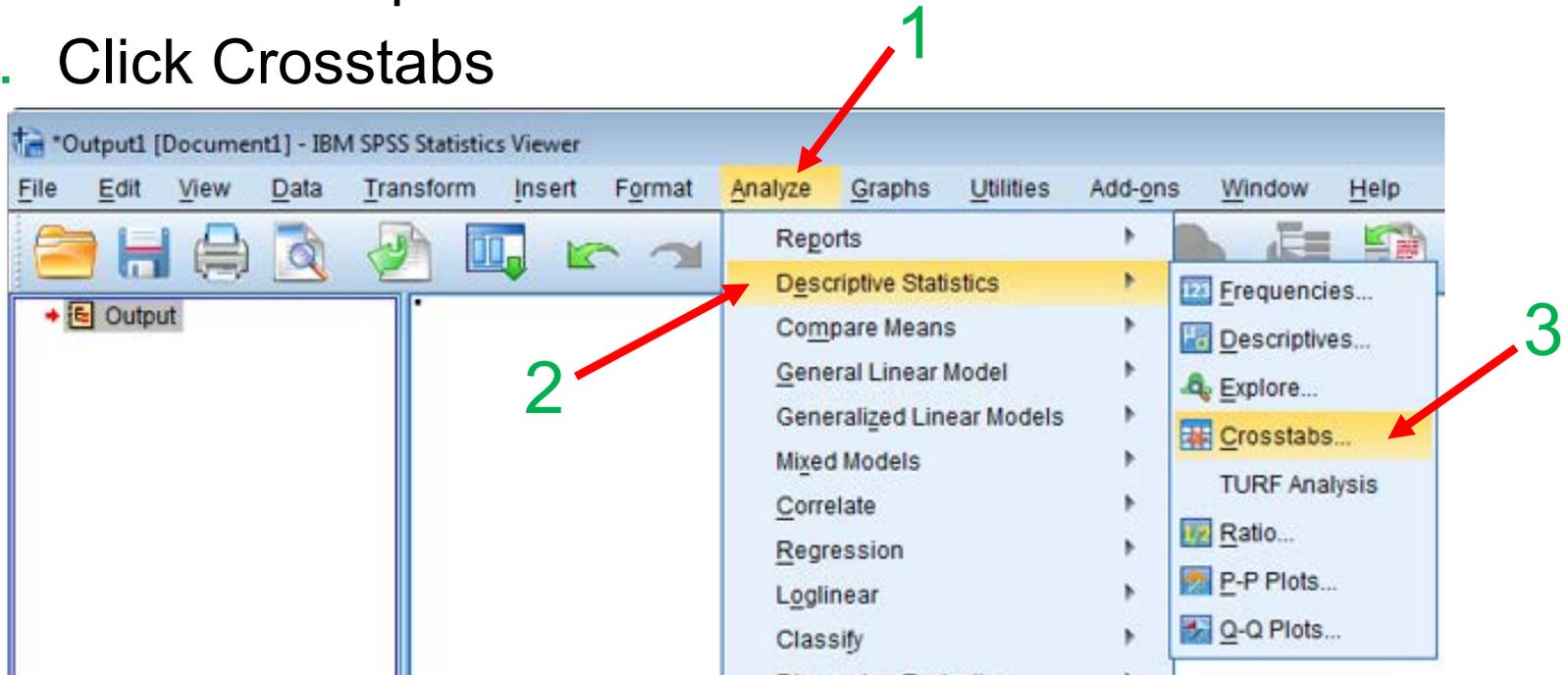
	student_id	student_grade	student_lep_desc	reading_achmnt_lm	var
1	1668822	06	LEP with End Date	3.00	
2	8281080	06	LEP with End Date	3.00	
3	8282887	06	LEP with End Date	2.00	
4	8284052	06	LEP with End Date	2.00	
5	8289278	06	LEP with End Date	2.00	
6	8294530	06	LEP with End Date	3.00	
7	8298963	06	LEP with End Date	3.00	
8	8299248	06	LEP with End Date	3.00	
9	8300223	06	LEP with End Date	3.00	
10	8300224	06	LEP with End Date	3.00	
11	8300231	06	LEP with End Date	3.00	

Summarize data retrieved from connection

- If we want something that is more friendly for leadership, we may want to summarize the raw data
- For this example, we will use a Crosstabs analysis to summarize the data

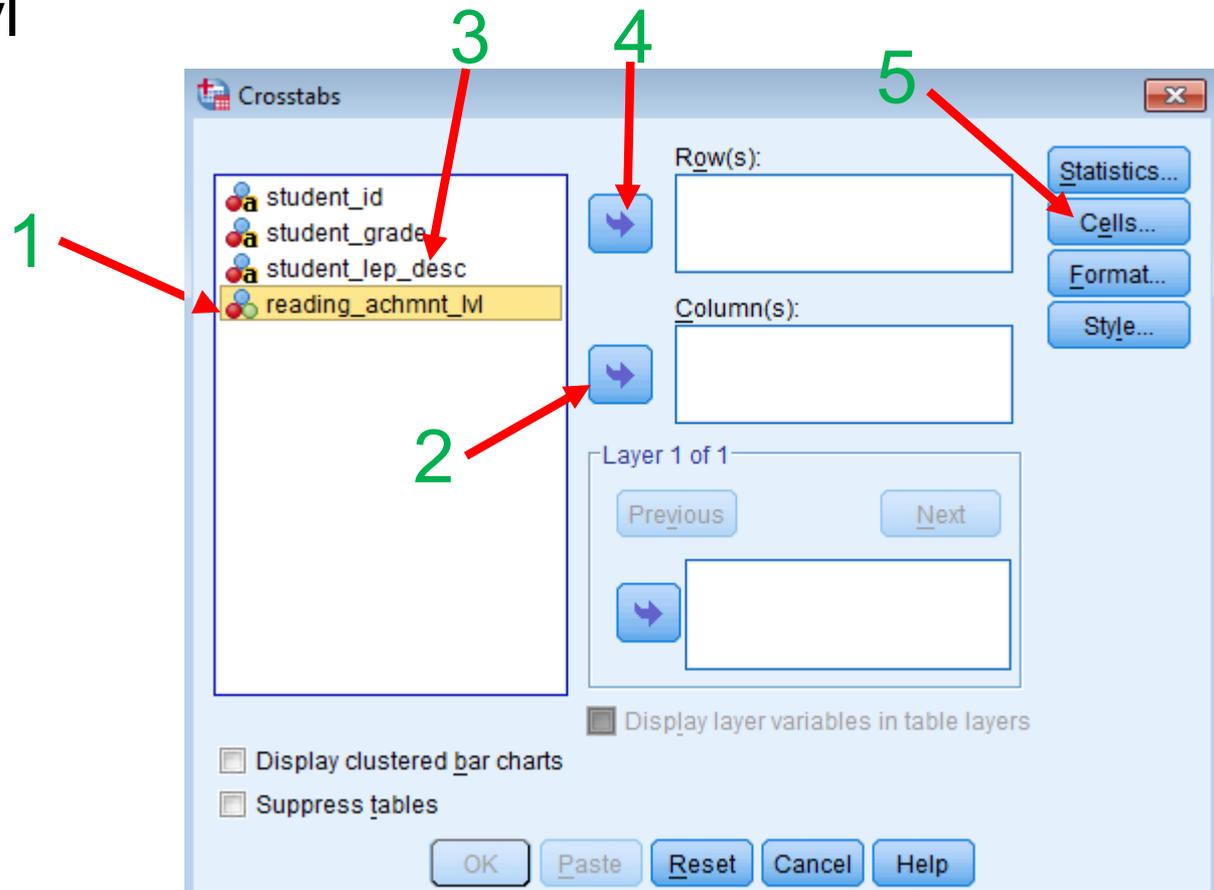
Generating crosstabs

- Let's create a Crosstabs table using the returned data
 - We'll specify reading achievement levels as columns and student LEP status as rows
- Click Analyze
 - Click Descriptive Statistics
 - Click Crosstabs



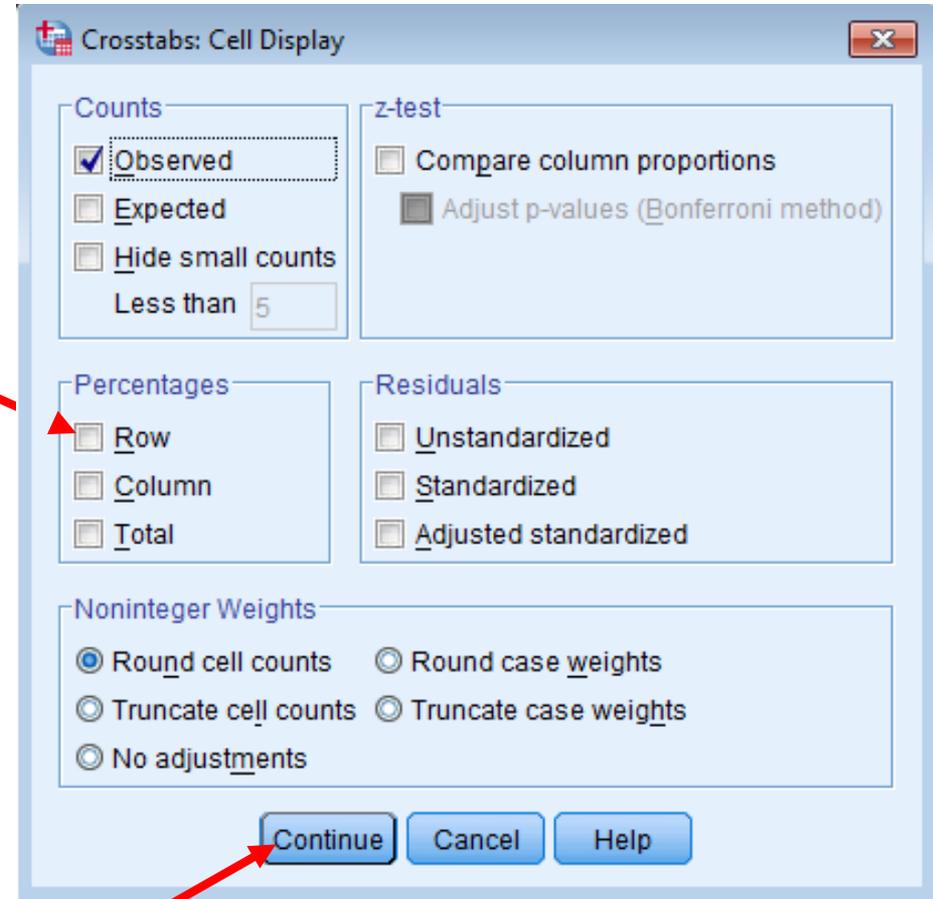
Crosstabs – Specify rows & columns

1. Click reading_achmnt_lvl
2. Click arrow next to Column(s) box
3. Click student_lep_desc
4. Click arrow next to Row(s) box
5. Click Cells box



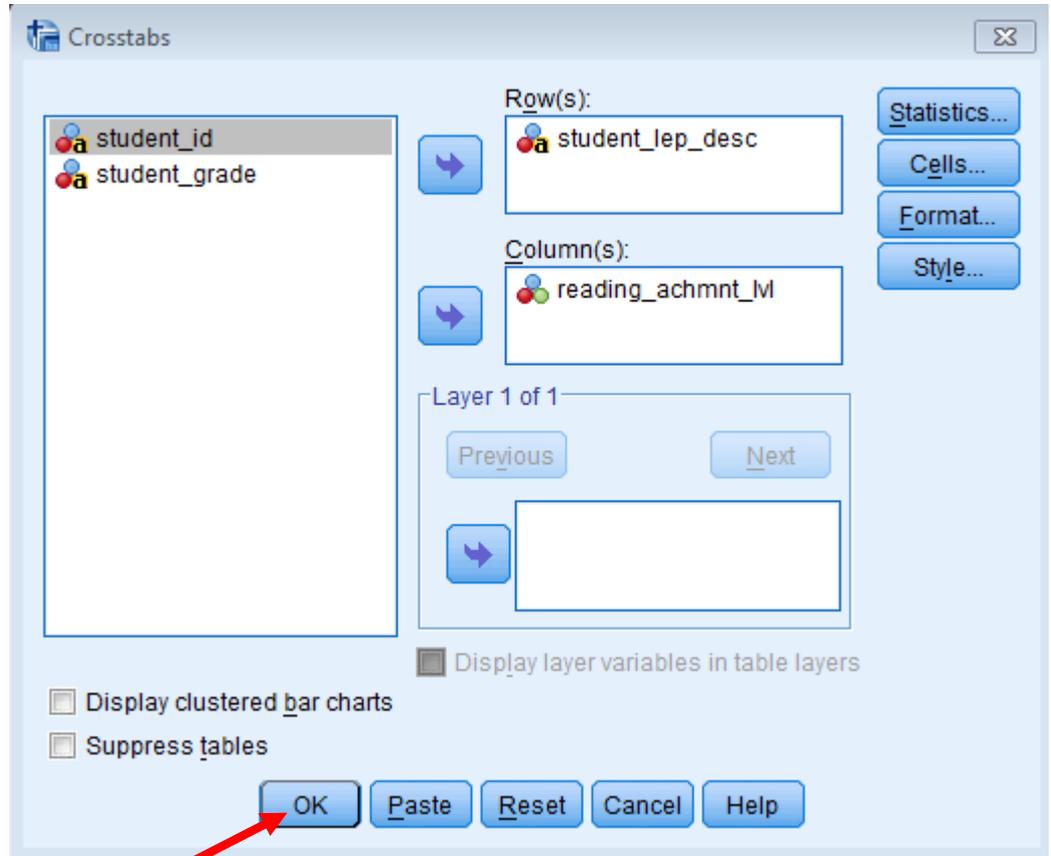
Crosstabs – Specify summaries

1. Check the Row box under Percentages (this will show us the distribution within LEP category across achievement levels)
2. Click Continue



Crosstabs – Generate table

1. Click OK to generate table immediately (Clicking Paste will insert Crosstabs syntax into an SPSS syntax file)



1

PivotTable – Finished product

- Now we have counts and percentages within each LEP Status across Reading achievement levels
- Should the parameters of the request change, or new data become available, the data can quickly be refreshed using the connection

			reading_achmnt_lvl				Total
			1.00	2.00	3.00	4.00	
student_lep_desc	Currently LEP	Count	109	88	68	3	268
		% within student_lep_desc	40.7%	32.8%	25.4%	1.1%	100.0%
	LEP with End Date	Count	18	73	241	46	378
		% within student_lep_desc	4.8%	19.3%	63.8%	12.2%	100.0%
	Never classified LEP	Count	471	542	1196	389	2598
		% within student_lep_desc	18.1%	20.9%	46.0%	15.0%	100.0%
Total		Count	598	703	1505	438	3244
		% within student_lep_desc	18.4%	21.7%	46.4%	13.5%	100.0%

Pasting the syntax

- Pasting the syntax can be advantageous for repetitive tasks
 1. Syntax for retrieving data from database
 2. Syntax for generating crosstab table

```
1  
2 GET DATA  
3 /TYPE=ODBC  
4 /CONNECT='DSN=relma_access_connect;DBQ='+  
5 'C:\Users\Jason\Desktop\connection_test\connect_training.accdb;DriverId=25;FIL=MS '+  
6 'Access;MaxBufferSize=2048;PageTimeout=5;'  
7 /SQL="SELECT T0.student_id, T0.student_grade, T0.student_lep_desc, T1.reading_achmnt_M FROM "+  
8 "student_demographics T0, student_assessments T1 WHERE T0.student_id = T1.student_id AND "+  
9 "(T0.student_grade = '06')"  
10 /ASSUMEDSTRWIDTH=255.  
11  
12 CACHE.  
13 EXECUTE.  
14 DATASET NAME DataSet1 WINDOW=FRONT.  
15 DATASET ACTIVATE DataSet1.  
16 CROSSTABS  
17 /TABLES=student_lep_desc BY reading_achmnt_M  
18 /FORMAT=AVALUE TABLES  
19 /CELLS=COUNT ROW  
20 /COUNT ROUND CELL.
```

Editing the query syntax

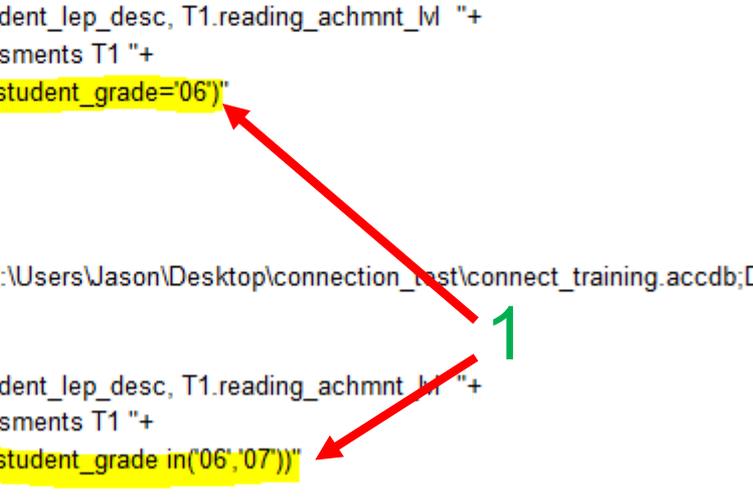
- Maybe leadership wants data for 6th and 7th graders
1. Edit SQL syntax to pull both grade levels (compare highlighted sections)

```
22 GET DATA
23 /TYPE=ODBC
24 /CONNECT='DSN=relma_access_connect;DBQ=C:\Users\Jason\Desktop\connection_test\connect_training.accdb;DriverId=25;FIL=MS'+
25 'Access;MaxBufferSize=2048;PageTimeout=5;'
26 /SQL=
27 "SELECT T0.student_id, T0.student_grade, T0.student_lep_desc, T1.reading_achmnt_lm "+
28 "FROM student_demographics T0, student_assessments T1 "+
29 "WHERE T0.student_id = T1.student_id AND (T0.student_grade='06')"
```

30 /ASSUMEDSTRWIDTH=255.

```
32 GET DATA
33 /TYPE=ODBC
34 /CONNECT='DSN=relma_access_connect;DBQ=C:\Users\Jason\Desktop\connection_test\connect_training.accdb;DriverId=25;FIL=MS'+
35 'Access;MaxBufferSize=2048;PageTimeout=5;'
36 /SQL=
37 "SELECT T0.student_id, T0.student_grade, T0.student_lep_desc, T1.reading_achmnt_lm "+
38 "FROM student_demographics T0, student_assessments T1 "+
39 "WHERE T0.student_id = T1.student_id AND (T0.student_grade in('06','07'))"
```

40 /ASSUMEDSTRWIDTH=255.



Retrieved data from edited query

- Resulting data, with 6th and 7th graders, returned to SPSS, ready for analysis



	student_id	student_grade	student_lep_desc	reading_achmnt_lm	var
1	1668822	06	LEP with End Date	3.00	
2	8260933	07	LEP with End Date	3.00	
3	8261910	07	LEP with End Date	4.00	
4	8265445	07	LEP with End Date	1.00	
5	8277585	07	LEP with End Date	3.00	
6	8278256	07	LEP with End Date	2.00	
7	8279101	07	LEP with End Date	2.00	
8	8279372	07	LEP with End Date	2.00	
9	8279434	07	LEP with End Date	3.00	
10	8280342	07	LEP with End Date	2.00	

Questions/Need help

Contact:

Jason Schoeneberger, Ph.D.

Senior Researcher and Task Lead

REL Mid-Atlantic at ICF International

jason.schoeneberger@icfi.com

704-307-9395



Please visit www.relmidatlantic.org for other data tools!