Database Tables

get ready to get nerdy!

Terminology

- record = a row in a table
- field = a column in a table
- table = a "spreadsheet" in a database
- database = a collection of tables (a Geodatabase includes other stuff, too)

Tables

- two types of tables in ArcGIS
  - attribute tables
    - connected with explicit spatial data
  - standalone tables
    - not connected with explicit spatial data
    - can have *implicit* spatial data!

Tables

- attribute tables
  - has a set of mandatory fields (feature/object ID - FID/OID, etc.)
- standalone tables
  - has a single mandatory field (object ID - OID)

Tables

- standalone tables
  - what do I mean by explicit and implicit spatial data?
    - explicit = connected to feature or spatial object
    - implicit = spatial data that has not been recognized as such
      - GPS coordinates not in a shapefile
      - place names
      - can be turned into explicit data via several processes
        - georeferencing! (later in course)

Tables

- standalone tables
  - what do I mean by explicit and implicit spatial data?
    - explicit = connected to feature or spatial object
    - implicit = spatial data that has not been recognized as such
      - GPS coordinates not in a shapefile
      - place names
      - can be turned into explicit data via several processes
        - joining (later in lecture)
        - georeferencing! (later in course)

Records

- represents a single entity
  - attribute table = spatial entity
  - standalone = another entity
- must have a value or <null> for every field
- otherwise, exactly like an Excel spreadsheet row

Fields

- must be well-defined
  - big difference from Excel
  - means that each field can have only one data type and data length
    - when defining a field, must choose data type and data length

Fields
data types

Type	Definition	Example	Uses
Short	integers -32,000 to 32,000	15, 22, 87	integer, ordinal, nominal data (small numbers)
Long	Really big integers	230942323	integer (large numbers)
Float	smaller decimal numbers	1234023.22	ratio
Double	larger decimal numbers	2.35908E+13	ratio
String / Text	alphanumeric strings	Saint Louis	nominal, ordinal
Date	dates!	3/22/99	dates! (technically nominal)
Blob	binary large objects (images, documents, movies, etc.)		everything else (technically nominal)

Fields
data lengths

Don't worry about it (use default), except for strings

Make string field lengths as big as the biggest string you expect to store!

Lengths are defined by number of characters

Fields
data lengths

STATENAME field should be as long as the longest state name (~25)

Fields

- defining fields
  - traditional goal: choose the data type and data length that is just big enough, but not too big
    - too big = database will get huge
    - not big enough = won't be able to store your data
  - these days
    - most default values are fine
    - only worry about optimization on BIG databases

Fields

- ArcGIS table requirements
  - field names = 13 character max!
    - can make field aliases
    - saved with map project file
  - LAME!

Fields

- calculating fields
  - just like Excel
    - doing an example on the board...

Technology

- file formats
  - ArcGIS supports many types of tables (and databases)
    - DBase (DBF), MySQL, Oracle, Excel 2003 (.xls) (not .xlsx)
  - interface + requirements are always the same
    - abstraction!
   - don't have to worry about the pecularities of each format

Joining Tables

- situation
  - many times you have a standalone table that has *implicit spatial data*
  - you have feature layer that has the *same* spatial data, but in *explicit form*
  - you want to link them together

Joining Tables

- example
  - census data Zip+3 polygon shapefile
    - attribute table just has Zip+3 name, area, etc.
  - a census data table with lots of demographic info on each Zip+3 polygon
    - must be downloaded separately
    - need to link to the Zip+3 shapefile for spatial display, analysis, operations, etc.
- JOINING IS YOUR ANSWER!

Joining Tables

key field

OID	Zip+3 Name	Area
1	932	455
2	931	467
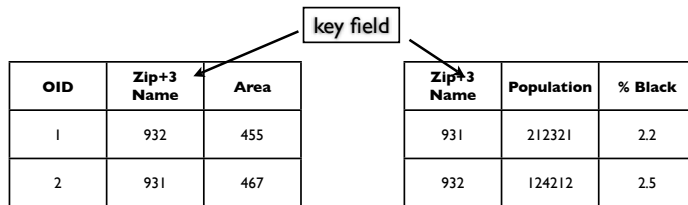
Attribute Table
Destination Table

Zip+3 Name	Population	% Black
931	212321	2.2
932	124212	2.5

Standalone Table
Source Table

OID	Zip+3 Name	Area	Population	% Black
1	932	455	212321	2.2
2	931	467	124212	2.5

Joining Tables

key field

OID	Zip+3 Name	Area
1	932	455
2	931	467
3	930	322

Zip+3 Name	Population	% Black
931	212321	2.2
932	124212	2.5
933	1131	2.1
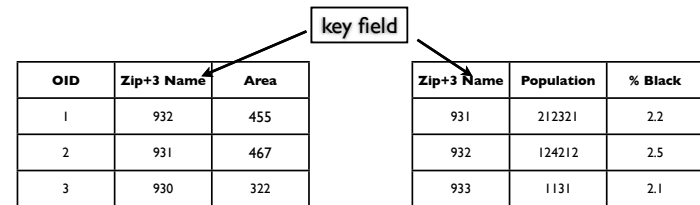
OID	Zip+3 Name	Area	Population	% Black
1	932	455	212321	2.2
2	931	467	124212	2.5
3	930	322	<null>	<null>

Joining Tables

key field

OID	Zip+3 Name	Area
1	932	455
2	931	467
3	931	467

Zip+3 Name	Population	% Black
931	212321	2.2
932	124212	2.5
933	1131	2.1

OID	Zip+3 Name	Area	Population	% Black
1	932	455	212321	2.2
2	931	467	124212	2.5
3	930	467	124212	2.5
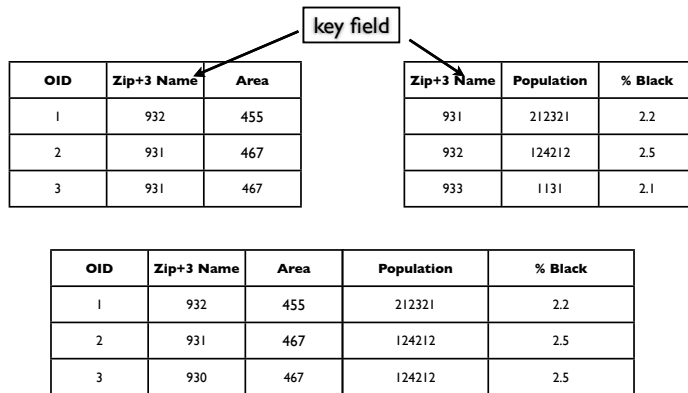
Joining Tables

- cardinality
  - joining is good for **one-to-one** cardinality and **one-to-many** cardinality
  - why not **many-to-one** or **many-to-many** cardinality?
    - we use relates for these cardinalities

Querying Tables

- Why query?
  - to **select** features (or raster cells) meeting a certain criteria
    - you can then perform operations on the selected features
      - make them into a new layer
      - count them, averages, other stats
      - pattern analysis
      - etc.