Geodatabase

7

Introduction

ArcGIS works with different GIS and non-GIS file formats. Some were created by ESRI such as coverage, shapefile, geodatabase and non-ESRI products such as AutoCAD, MIF/MID, and others. The geodatabase is the built-in data structure for ArcGIS and is the primary data format used for editing and data management. Geodatabase combines "geo" (spatial data) with "database" to create a central data repository for spatial data storage and management. It can be leveraged in desktop, server, or mobile environments and allows you to store GIS data in a central location for easy access and management.

A Geodatabase in ArcGIS can be defined as a collection of diverse types of data kept in either Microsoft Access database or multiuser relational database (Oracle, Microsoft SQL Server, PostgreSQL, Informix, or IBM DB2) file systems. If it is held in a Microsoft Access database file system it is called a personal Geodatabase and if it is kept in a relational database file system it is called a file geodatabase. The differences between them are: (1) File geodatabases have unlimited storage space while personal geodatabases have a limit of 2 GB, (2) File geodatabases save their data in an ArcGIS folder format while the personal geodatabases save their data in the format of Microsoft access, and (3) File geodatabases can be compressed to save space.

The geodatabase in ArcGIS contains three primary dataset types: (1) Feature classes, (2) Raster datasets, and (3) Tables.

A Geodatabase can be generated from scratch by creating or collecting dataset types. After building a number of these fundamental dataset types, one can add to or extend their geodatabase with more advanced capabilities (such as adding topologies, networks, or subtypes) to model GIS behavior, maintain data integrity, and work with an important set of spatial relationships.

Geodatabases work across a range of database management systems (DBMS). Which include architectures and file systems that come in many sizes, and have varying numbers of users. They can scale from small, single-user databases built on files up to larger workgroup, department, and enterprise geodatabases accessed by many users.

Creating a Geodatabase

This section will allow you to capture data using digital data to create feature classes and store it in a geodatabase.

Scenario 1: You are working as a hydrogeologist for Water Authority and you are asked to create a file geodatabase and fill it with point, line, and polygon feature classes. You are also going to use the "**Image_Rectify.tif**" image as a source to capture groundwater wells as a point feature class, the fault as a line feature class, and plant as a polygon feature class.
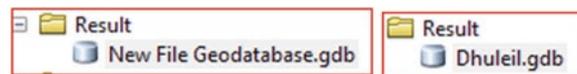
Create File Geodatabase

1. Start ArcMap
2. Open Catalog window\connect to Ch07
3. R-click **Result** in **Ch07** folder/select New/File Geodatabase.
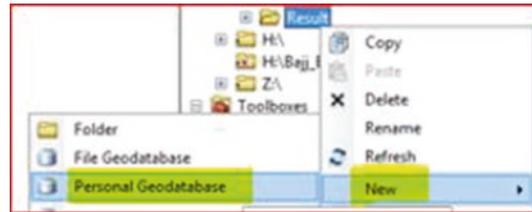
4. Change the name from New File Geodatabase.gdb to "**Dhuleil.gdb**"

**Create a Feature Class and Assign for it WGS84-GCS**

To create a feature class, the file geodatabase must already exist. You are going to create **3-new feature** classes and call them **Well (point)**, **Fault (line),** and **Plant (polygon)**.

5. R-click on the "**Dhuleil.gdb**" and select New/Feature Class
   a. Name: Well
   b. Type: Point Feature
6. Click Next
7. D-click Geographic Coordinate Systems/scroll down and D-click World/scroll down and select WGS1984
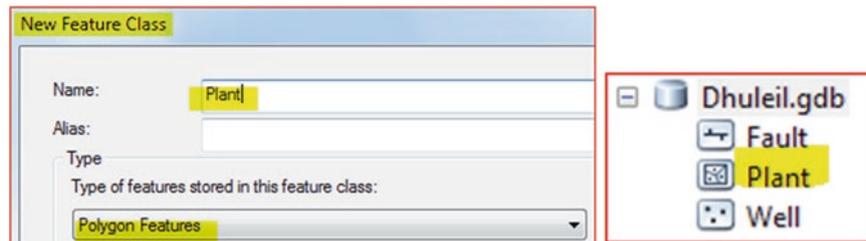8. Click Next/Next/Next/
9. Click Finish
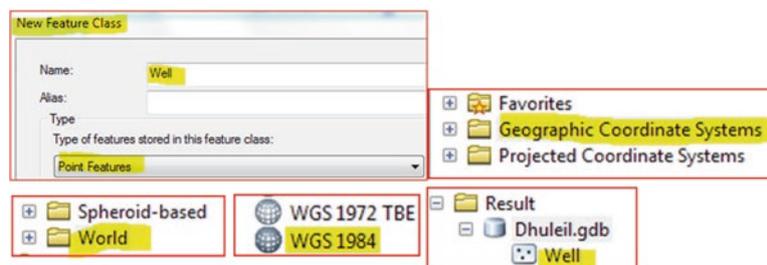
   Result: The Well feature class is created and it will be stored in the Dhuleil.gdb and added into TOC.

10. Repeat the previous steps and create a Fault (line) Feature Class and assign to it the GCS_WGS_1984
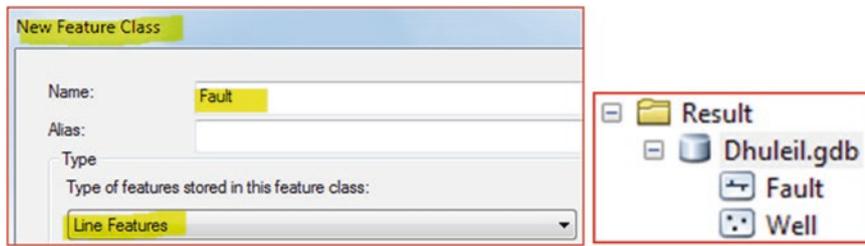
11. Repeat the previous steps again and create a Plant (polygon) Feature Class and assign to it the GCS_WGS_1984


Result: The Fault and Plant feature classes are integrated into the TOC.

Capture the Feature Classes Using an Image

You are going to digitize the two Wells, the Fault, and the Plant using a rectified image. The image is called "**Image_Rectify. tif**" and it is registered in latitude-longitude and associated with Jordanian datum (D_Jordan).

12. Click Add Data/browse to \\Ch07\Data\Image folder and integrate "**Image_Rectify.tif**"
13. If you don't see the image/r-click "**Image_Rectify.tif**" it in the TOC and point to Zoom To Layer
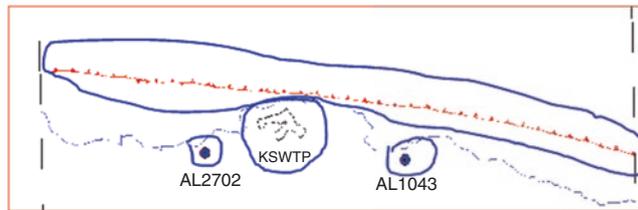
14. On the Editor tool/click on the drop-down arrow/Start Editing

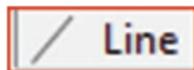15. Click the Well feature template in the Create Features window
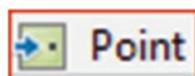
16. Click the Point tool ▸ **Point** under Construction Tools window
17. Click in the center of the well (AL2702) on the image "**Image_Rectify.tif**." then click on the second well (AL1043)

Result: The two points are created on the map and the second point is selected.
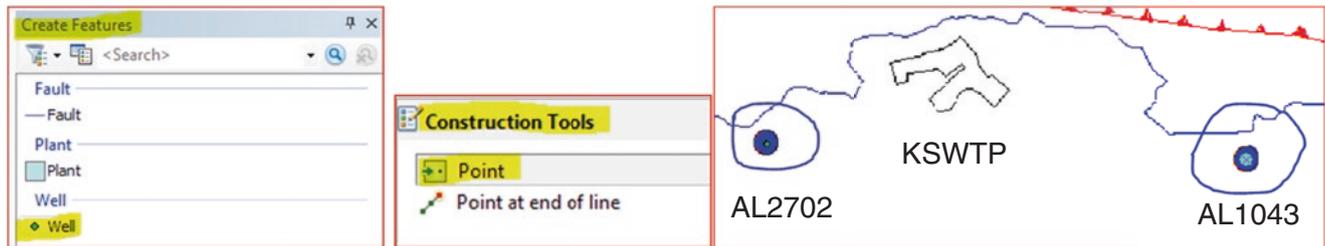
18. For the "Fault" layer make sure you select Fault feature template in the Create Features window
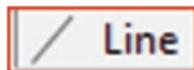
19. Click the Line tool under Construction Tools
20. Rest your pointer over the endpoint of the existing fault in the eastern portion of the image and click once
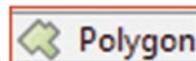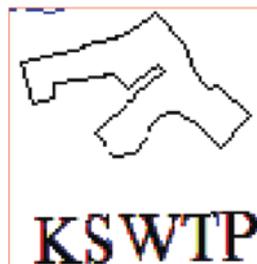21. Using the image as a guide, digitize the new line by clicking the map each place you want to add a vertex
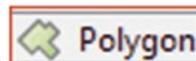22. Press F2 or double-click at the end of the fault to finish digitizing

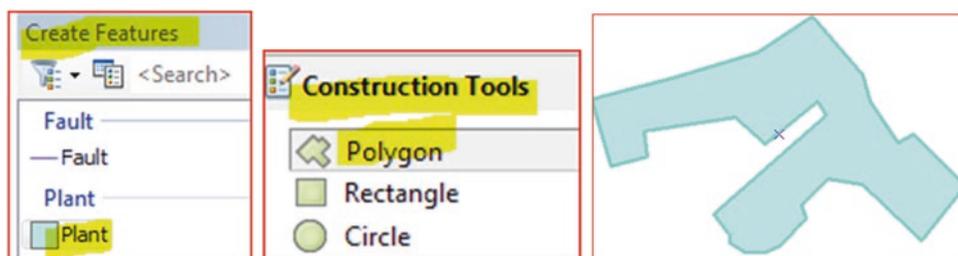**Result**: The line is created on the map and it is selected.

23. Zoom In around the KSWTP in the "**Image_Rectify.tif**"

24. Click the Plant from the Create Features window and Polygon under the Construction Tools
25. Place your pointer over the right-east corner of the KSWTP and click once, then click along the outside edge of the KSWTP in the image to create the polygon
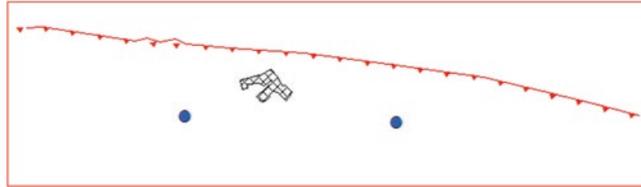26. Press F2 or double-click at the end of the plant on the image to finish digitizing the plant

**Result**: The Plant is created on the map and it is selected.

27. On the Editor toolbar/click Editor drop-down/click Save Edits

28. Change the symbols of the Wells, Fault and Plant based on your taste

Create Feature Dataset and Import Shapefiles into It

29. Open Catalog window/browse to \\Ch07\Result folder
30. R-click "Dhuleil.gdb"/select New/Feature Dataset
31. Name: Water
32. Next/click Projected Coordinate System/select UTM/Select WGS1984/
33. Open Northern Hemisphere/select WGS1984 UTM Zone 36N
34. Click Next/Next/Finish
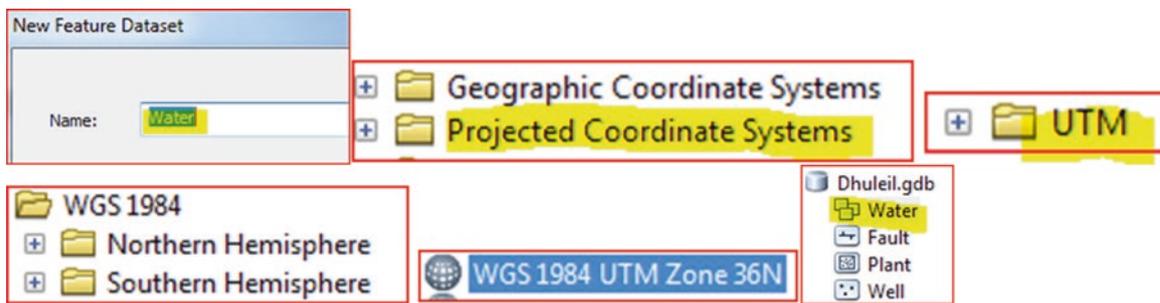
Result: The Water Feature Dataset is created in the "**Dhuleil.gdb**" and it is registered in UTM zone 36 N and associated with the WGS 1984 datum.
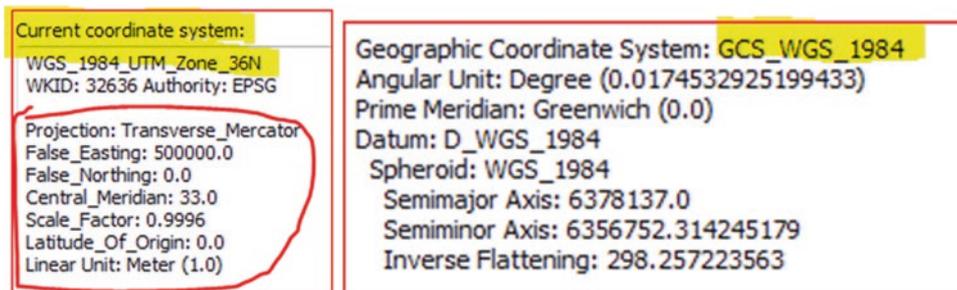

Important Note: Any file that will be imported into the Water Feature Dataset, will have its coordinate system converted into the coordinate system of the Water Feature dataset automatically.

## Import the Stream, StudyArea, and Dam into the Water Feature Dataset

This step will import three shapefiles (Dam.shp, Stream.shp, and StudyArea.shp) into the Water Feature dataset. The original coordinate of the three shapefiles are in latitude and longitude and associated with the WGS 1984. Once these files are imported into the Water Feature Dataset, they will be converted into WGS_1984_UTM zone 36 N.

35. In the Catalog window/R-click Water Feature Dataset/Import/Feature Class (multiple)
36. Input Features/browse to \\Ch07\Data\Shapefile folder, highlight **Dam.shp**, **Stream.shp**, and **StudyArea.shp** and click Add
37. Click OK
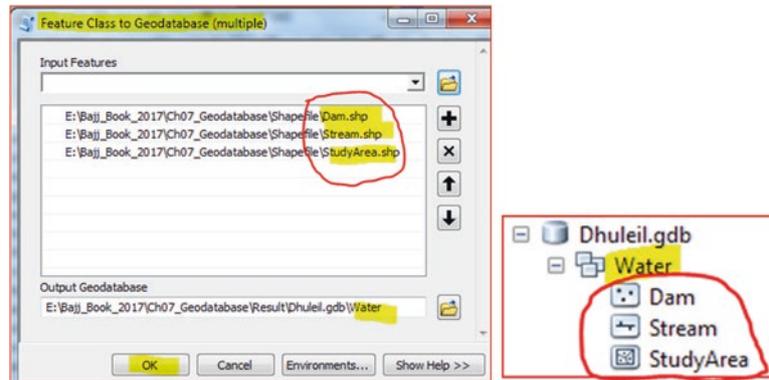
Result: The Dam, Stream, and StudyArea is imported in the Water Feature dataset and the three feature classes are displayed in the TOC.

38. Make the StudyArea Hollow, and change the symbols of the Dam and Stream feature classes as in the image below
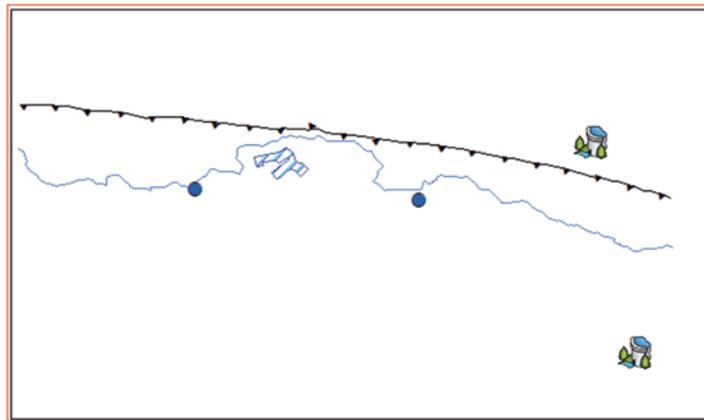
39. In Catalog window/R-click the Water Feature Dataset and find out what is the coordinate system?
40. In Catalog window/R-click the Dam feature class and find out what is the coordinate system? Repeat this for the Stream and StudyArea

Create a Relationship Class

A relationship class can be created between any feature class or table within a geodatabase using tools either in ArcCatalog or the Catalog window in ArcMap. These tools can be used to create simple, composite, and attributed relationship classes. In this section, you will create a relationship class between a well feature class and two tables. The first table contains water chemical analysis and the second table contains hydrological information about the wells. The relationship class will be created first between the well feature class and Table1 feature class based on the Well Inventory NO. (INVEN_No). The second relationship will be created between Table1 and Table2 also based on INVEN_No. Relationship classes support all cardinalities: one-to-one, one-to-many, and many-to-many. In this section one-to-one will be used. Relationship classes provide many advanced capabilities not found in ArcMap that joins and relates. A relationship class can be set up so that when the user modifies an object, the related objects update automatically. This can involve physically moving related features, deleting related objects, or updating an attribute.

Scenario 2: You are a hydrogeologist working in Jizi catchment area in Sultanate of Oman. You have been asked by your superior to perform an analysis dealing with the following files in ArcGIS:

1. Well.shp (groundwater wells)
2. Catchment.shp (Jizzi watershed area)

3. Table1.txt (table containing chemical water analysis from the wells)
4. Table2.txt (table containing hydrological information about the wells)

In order to do this job you have to do the following: create a Personal Geodatabase, integrate the proper files into it and create a relationship class.

Create Personal Geodatabase

1. In ArcMap/Insert Data Frame and call it Jizzi
2. Open Catalog window/R-click \\Result folder/New/Personal Geodatabase
3. Rename the New Personal Geodatabase.mdb "**Jizzi.mdb**"

Integrate Shapefiles and Tables into Personal Geodatabase

4. R-click "**Jizzi.mdb**"/Import/Feature Class (Multiple)
5. Browse to \\Data\Jizzi folder/highlight Catchment.shp and Well.shp
6. Click Add then OK

Result: Catchment and Well feature classes are now imported into Jizzi.mdb.

7. R-click "**Jizzi.mdb**"/Import/Table (Multiple)
8. Browse to \\Data\Jizzi folder/highlight Table1.txt and Table2.txt
9. Click Add then OK

**Resul**t: Table1 and Table2 feature classes are now imported into Jizzi.mdb.
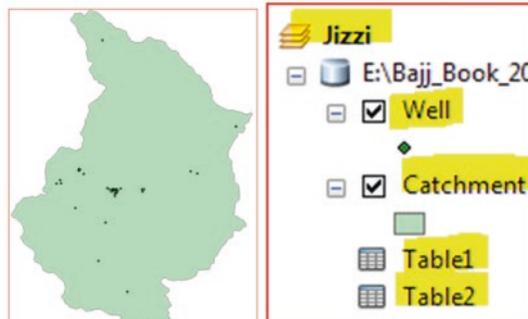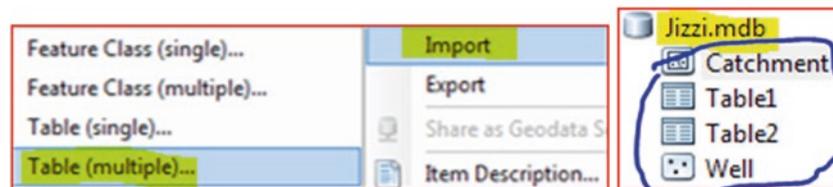
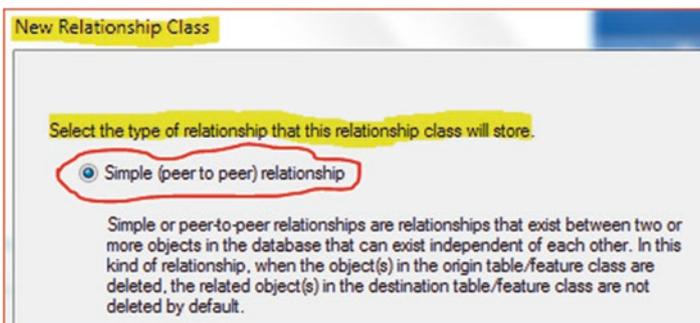10. Add Well, Catchment, Table1, and Table2 feature classes into the TOC.

Create Relationship Class

First create Relationship Class in Jizzi.gdb between Well and Table1

11. In Catalog window/R-click **Jizzi.mdb**/New/Relationship Class
12. Name: Well_Table1
13. Origin table: Well
14. Destination table: Table 1
15. Next
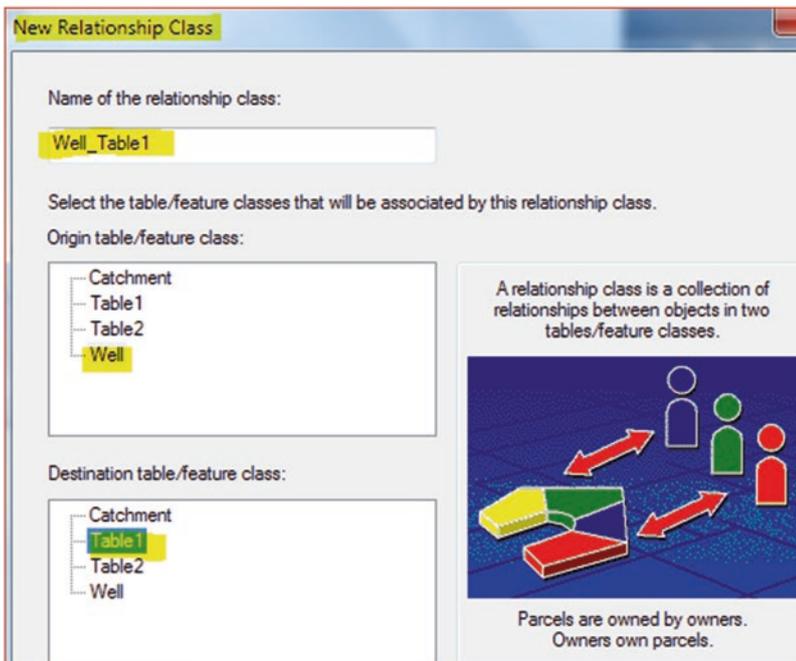
16. Select Simple (peer to peer) relationship

17. Click Next/Next/Select 1 – 1 (one to one)

18. Click Next (select/No, I do not want to add attributes to this relationship class)

19. Click Next (select the Primary & Foreign key, which is INVEN_NO)

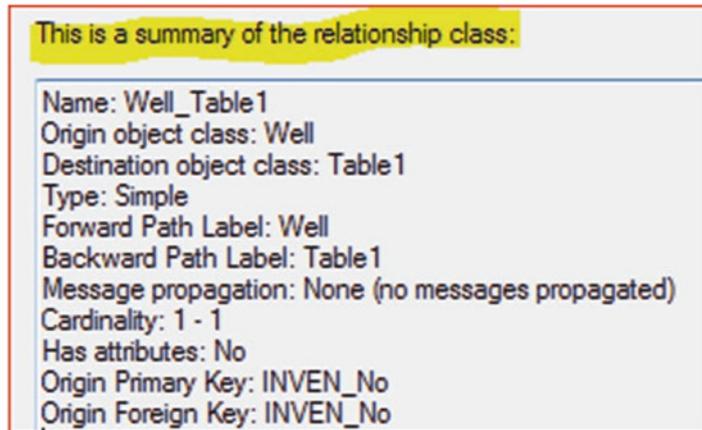20. Click Next and Finish

This is a summary of the relationship class:

Name: Well_Table1
Origin object class: Well
Destination object class: Table1
Type: Simple
Forward Path Label: Well
Backward Path Label: Table1
Message propagation: None (no messages propagated)
Cardinality: 1 - 1
Has attributes: No
Origin Primary Key: INVEN_No
Origin Foreign Key: INVEN_No
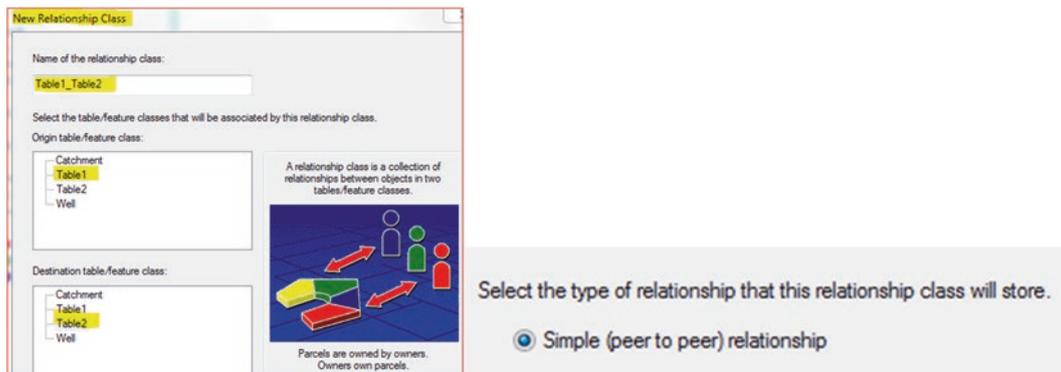
Result: The Well_Table1 Relationship Class is now established.

Create Relationship Class in Jizzi.gdb Between Table1 and Table2

21. R-click **Jizzi.gdb**/New/Relationship Class
22. Name: Table1_Table2
23. Origin table: Table1
24. Destination table: Table 2
25. Next

26. Next

27. Next Select 1 – 1 (one to one)

28. Check No, I do not want to add attributes to the relationship class

29. Next
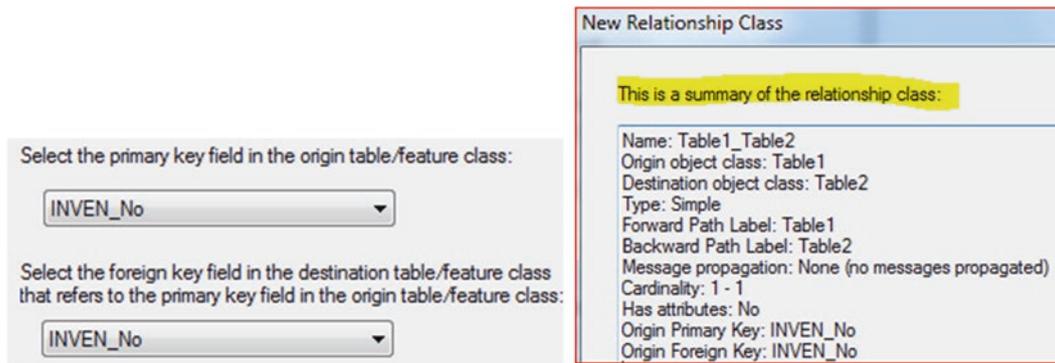
30. Click Finish
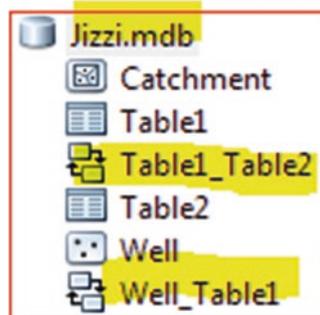
Result: The Table1_Table2 Relationship Class is now established.

Find the Wells that have Salinity Higher than 500 mg

31. In ArcMap make sure that the Well, Catchment, Table1, and Table2 feature classes are already in the TOC
32. If you don't see the tables, click the List By Source (above the table of content)
33. R-click Table1/Open/click Table Options icon/Select by Attributes
34. D-click TDS click > and type 500
    [TDS] > 500
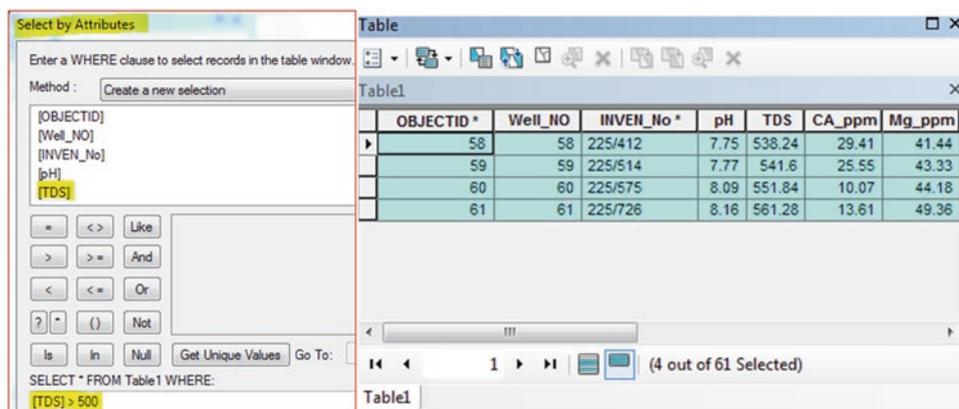
35. Click Apply/Close
36. Click Show Selected Records (at the bottom of the table)

Result: Four-wells are selected.

37. Click again Table Options/point to Related Tables/select Well_Table1: Table1

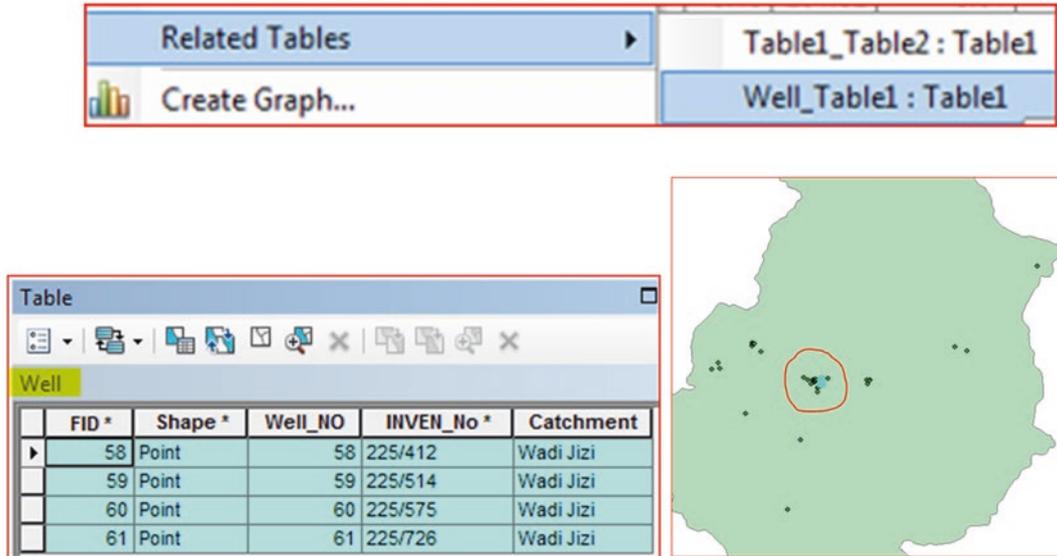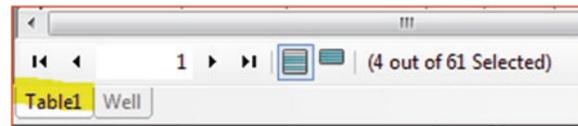**Result**: Four-wells in the Well Attribute table are selected and these wells are selected in the catchment area.

38. Highlight the Table1

39. In Table 1/click again Table Options/Related Tables/select **Table1_Table2: Table1**

Table2

OBJECTID *	INVEN_No *	Drilling	Elevation_m	Depth_m	Aquifer	Yield_m3_h
53 225/412		Dec-87	115	500	Quaternary	43
54 225/514		Jun-05	178	316	UeR	90
55 225/575		Mar-02	175	311	UeR	200
58 225/726		Jun-05	50	1126	Alluvium	90

Result: Four wells are selected and these wells are penetrating the following aquifers:

a. Quaternary: 1 well
b. UeR: 2 wells
c. Alluvium: 1 well

Projection and Datum Conflict

In the datum conflict in Chap. 5, you encountered a problem when you projected the well feature class from Jordanian Transverse Mercator into the Palestine Transverse Mercator. In order to perform the projection, the Jordanian datum had to be converted into WGS 84 and then the WGS 84 converted into the Palestine datum. The result was not a perfect projection as the output projected file didn't align correctly with the original file that registered in the Palestine Transverse Mercator. The discrepancies between the same well from the two files was around 177 m. To solve these issues the projection was performed in the geodatabase environment.

GIS Approach

40. Insert Data Frame and call it Datum Conflict
41. Create File Geodatabase in the Result folder and call it Projection.gdb
42. R-click Projection.gdb/New/Feature Dataset and call it "**Datum**"

43. Next/click the Add Coordinate System drop-down arrow ![icon] icon
44. Click Import browse to \\Projection folder and select Well_PTM.shp and click Add
45. Click Next/Next/Finish

Result: The Datum Feature Dataset is created.

46. R-click Datum Feature Dataset/Import/Feature Class (Multiple)
47. Browse to \\Data\Projection folder and select Dhuleil.shp, Well_JTM, and Well_PTM.shp
48. Click Add/OK
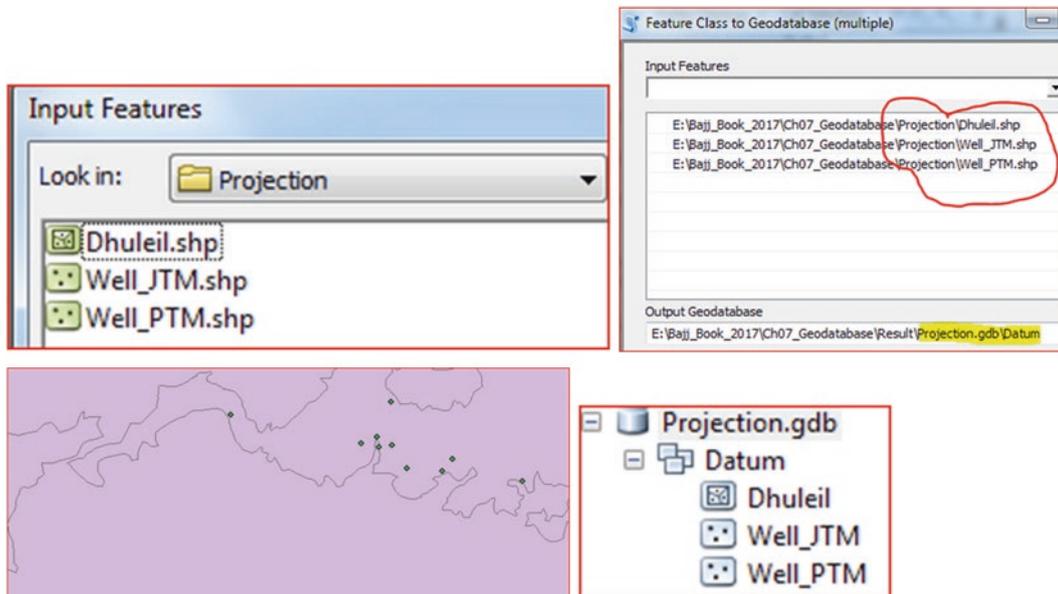
Result: The three feature classes **Dhuleil**, **Well_JTM**, and **Well_PTM** are reside in the Datum Feature Dataset and integrated into the TOC. The **Well_JTM**, and **Well_PTM** are both projected into the Palestine Transverse Mercator and they are aligned together perfectly.