Exercise 7: Creating and Working With File Geodatabases

Geodatabases offer a well-organized means to store your geospatial data layers. This is one of the primary reasons we have tried to use them throughout the lab exercises. However, it is not always necessary to store your files in a database structure. For example, you can use standalone shapefiles and raster grids to store your vector and raster data, respectively. Other than just organization, databases do offer added benefits. For example, you can establish relationships between spatial data layers and tables. Within ArcGIS, three types of databases can be created: personal geodatabases, file geodatabases, and enterprise geodatabases. A personal geodatabase is stored inside of a Microsoft Access database while a file geodatabase is stored in a folder that houses binary files. Both of these databases are designed to be used by a single editor and multiple users and do not support versioning. So, they would not be appropriate for a large working group. Enterprise geodatabases are stored on a server in an enterprise-level database. They offer multiple editors and users along with versioning. These are commonly used to support data access and editing needs of large working groups.

In this exercise, you will create a file geodatabase for use within a desktop GIS Application. Specifically, you will design a database that stores trail data for the state of West Virginia. Within the database, you will create a table in which multiple reviews of a trail can be added and associated with a specific trail feature.

The trail data were made available by the West Virginia GIS Technical Center as part of the West Virginia Trail Inventory. The data are available here: http://www.wvgis.wvu.edu/data/dataset.php?ID=413. A web map of the trail inventory is available here: http://www.mapwv.gov/trails/#.
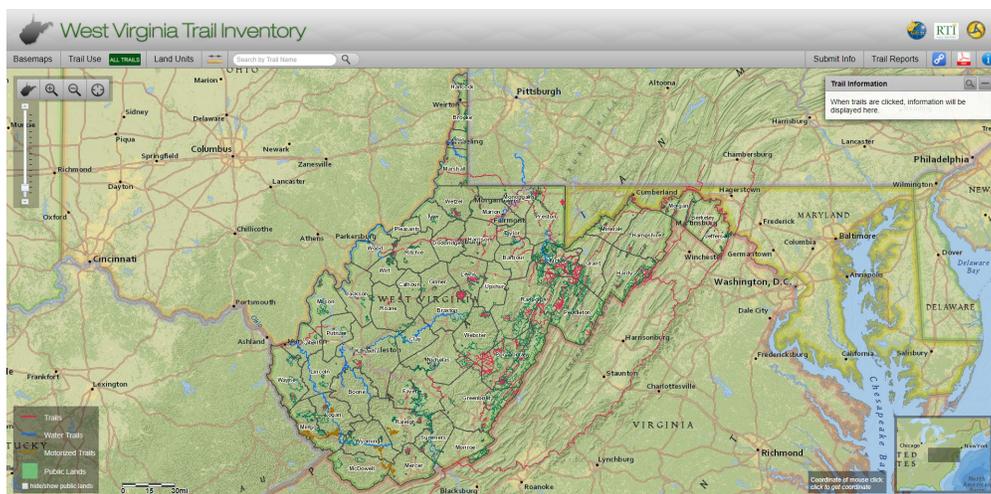
> Topics covered in this exercise include:
>
> 1. Create a new file geodatabase
> 2. Import spatial data into a geodatabase
> 3. Define domains
> 4. Define subtypes
> 5. Create a new table inside a geodatabase
> 6. Define a relationship
> 7. Edit a geodatabase

☐ Download the **Exercise_7** data from https://www.wvview.org/. All lab materials are available on the course webpage and linked to the exercise. You will need to extract the compressed files and save it to the location of your choosing. The following layers have been provided:

counties_detailed: polygon features representing county boundaries for West Virginia

public_lands: polygon boundaries representing the extent of public land (state and federal) in West Virginia

wv_trails: line features representing trails in West Virginia mapped by the West Virginia Trail Inventory

wv_trails_points: point features associated with trails mapped by the West Virginia Trail Inventory (for example, overlooks, parking area, and historic signs)

wv_hillshd: hillshade or shaded relief surface for West Virginia at a 30 meter cell size.

Step 1. Create a New File Geodatabase

You will begin this exercise by creating a new file geodatabase. This can be done within ArcGIS Pro.

☐ Open ArcGIS Pro.
☐ Click on the View Tab followed by Catalog View to Open the Catalog.

- In the Catalog View, right-click on Folders followed by Add Folder Connection. Connect to the **Exercise_7** folder that you downloaded. This folder should now be available under Folders in the Catalog.
- In the **Exercise_7 folder** create an empty **Exercise_7_working** folder in the downloaded and unzipped **Exercise_7 folder**, where you will save your work. Do this by right-clicking in the folder then selecting New followed by Folder.
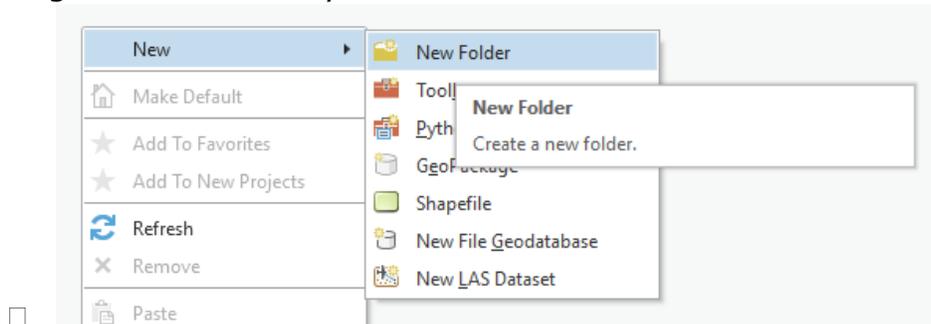
- Right click on the new **Exercise_7_working** to open it.
- Right-click in the folder space then choose New followed by File Geodatabase.
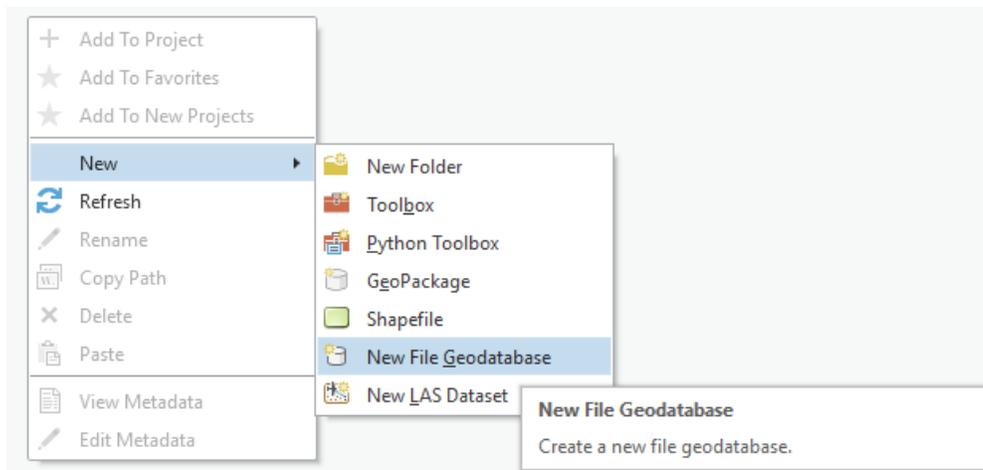
- Name the file geodatabase **wv_trails_database.gdb**.
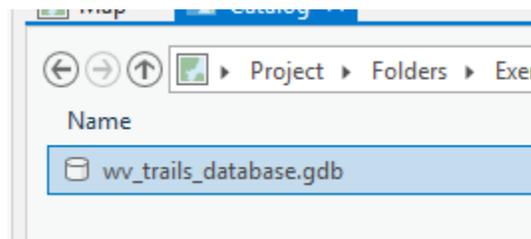
- Double-click on the new database in the Catalog to enter it.

Step 2. Import Vector Data

You have now created a file geodatabase. However, there is nothing in it. There are a variety of features that can be added to a database including the following:

Feature Class: stores vector data

Table: stores tabulated or aspatial

Raster Dataset: stores raster data, including images

Raster Catalog: a location to store multiple raster datasets

Mosaic Dataset: allow you to store raster datasets, like images, as a catalog and view them as an image mosaic

Toolbox: a place to store tools, such as those derived from ModelBuilder or Python scripts.

Feature Dataset: a location to store related feature classes

Relationship Class: establishes a relationship between data layers in the geodatabase, such as feature classes and tables

Note: It is now always necessary to store your vector data inside of a feature dataset. However, sometimes it is. For example, feature datasets must be used when building topology, creating a network dataset, or producing a terrain dataset. They can also be used to simply organize data. For example, if a lot of vector features will be added to the geodatabase, they can be organized based on theme. Note that all features in a feature dataset must share a common coordinate system.
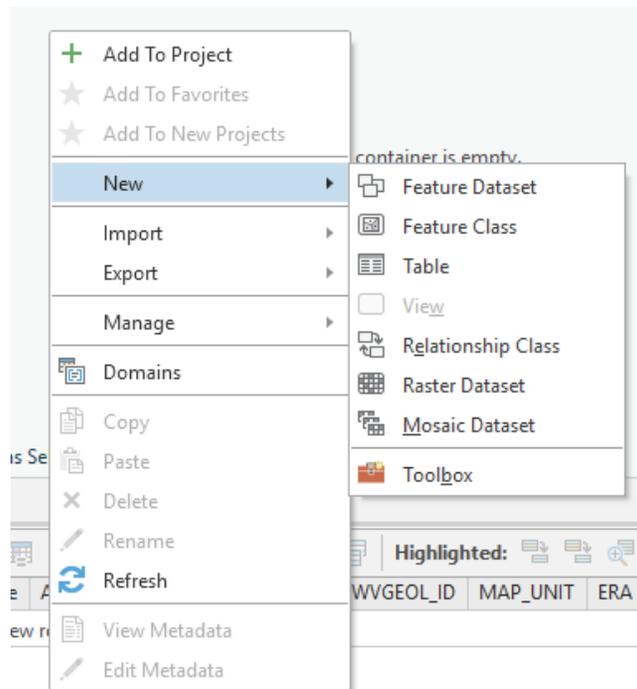
Other than creating new features, it is also possible to Import features. For example, a shapefile can be imported into a geodatabase and converted to a feature class. A raster grid or image file can be imported into a geodatabase as a raster dataset.

Tools are available under Import to import in a variety of data. For example, the **Feature Class to Feature Class** and **Feature Class to Geodatabase tools** allow you to import vector data into the geodatabase, such as shapefiles. The **Table to Table and Table to Geodatabase tools** allow you to import tables or aspatial data, such as DBF and CSV files.
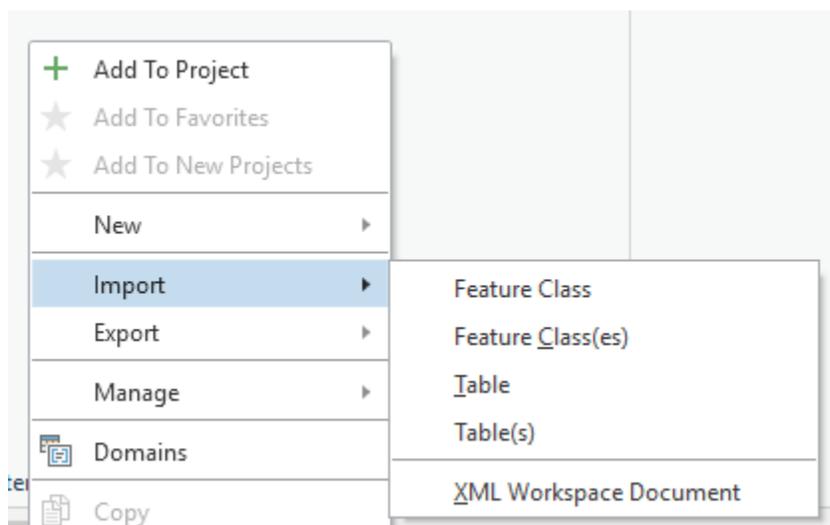
You will start populating the database by importing in some shapefiles using the **Feature Class to Geodatabase Tool**.

- ☐ Make sure you are inside of the new geodatabase.
- ☐ Right-click in the Catalog then select Import followed by the Feature Class(es). This will open the **Feature Class to Geodatabase Tool**.
- ☐ From the **Data** folder within the **Exercise_7** folder you downloaded from the website, add the following shapefiles to the Input Features list: **counties_detailed.shp**, **public_lands.shp**, **wv_trails.shp**, and **wv_trails_points.shp**.
- ☐ Click Run to execute the tool.
- ☐ Once the tool executes, the four feature classes will be added to the **wv_trails_database.gdb** geodatabase. If you can't see them in the geodatabase, you may need to right-click then select Refresh to update the Catalog.
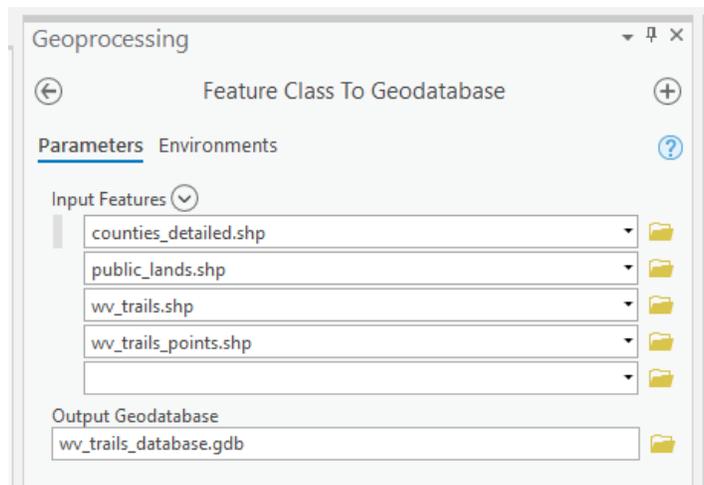
You should see that the four vector layers have been converted to feature classes and have been added to the geodatabase.
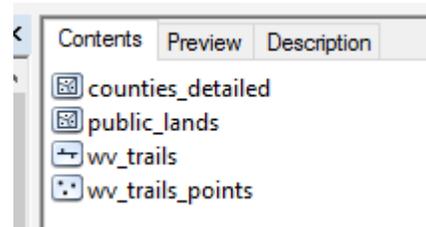
Step 3. Import Raster Data

You will now import a raster data layer into your file geodatabase using the **Copy Raster Tool**.

- ☐ Select the Analysis Tab followed by Tools. This will open the Geoprocessing Pane. Search for the **Copy Raster Tool** and open it.
- ☐ From the **Data** folder within the **Exercise_7** folder, add the **wv_hillshd** raster grid. Save it to the **wv_trails_database.gdb** geodatabase as **wv_hillshd**. You do not need to change anything else.
- ☐ Click Run to execute the tool.
- ☐ Once the tool executes, you will be returned to the **Exercise_7** folder that contains the **wv_trails_database.gdb** geodatabase. Click on the database to enter it.
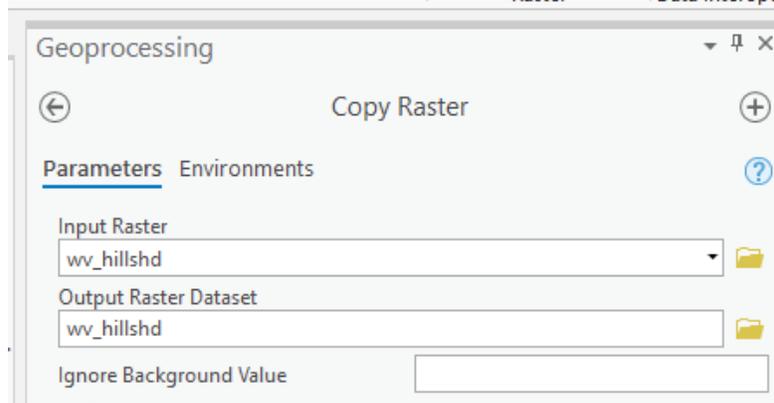
Step 4. Define Domains

You will add a new table to your database in which users can review a specific trail. This table will contain six fields:

"Residency" = whether or not the reviewer is a resident of West Virginia (possible values are yes and no)

"Date" = the date the trail was used

"Difficulty" = the difficulty level of the trail (easy, moderate, challenging, difficult, very difficult, extreme)

"Rating" = rating of the trail from one to five stars.

"Review" = allows the reviewer to provide a review of the trail.

"Trail_ID" = field that will hold the trail ID to associate the review with a specific trail

Notice that the "Residency" and "Difficulty" fields have a set of allowed values. Or, you don't want the reviewer to be able to enter values that are different from the allowed values.

You can define these allowed values by using domains. A domain is a property that is shared by all features in the geodatabase. They can be created as coded values, in which only discrete values can be entered or as ranged values, in which a continuous number can be entered between a high and low value. You will set up domains for the "residency" and "difficulty" fields using coded values. You will use ranged values for "rating."

- Right-click on the geodatabase in the Catalog View then select Domains.
- Navigate to the Domains Tab.
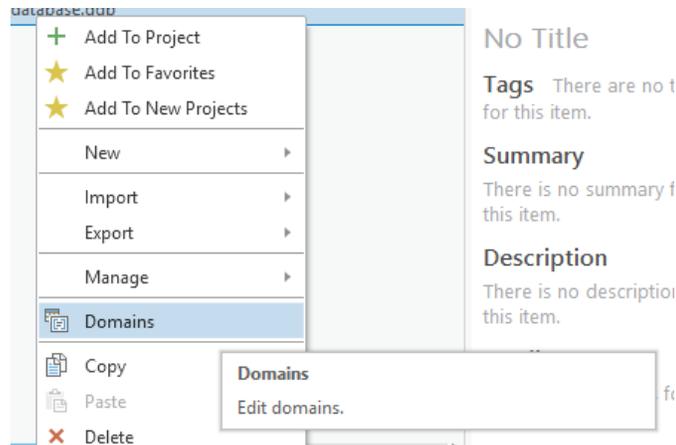
□ Create three domains with the following characteristics (do not include the quotes):

    a. Domain 1

        i. Domain Name = "Residency"

        ii. Description = "West Virginia Resident?"

        iii. Field Type = Short Integer

        iv. Domain Type = Coded Values Domain

        v. Split Policy = Default

        vi. Merge Policy = Default

        vii. Codes

            1. Code = 0, Description = "Yes"

            2. Code = 1, Description = "No"

Domain Name	Description	Field Type	Domain Type	Split Policy	Merge Policy		Code	Description
Residency	West Virginia Resident?	Short	Coded Value Domain	Default	Default		0	Yes
		Text	Coded Value Domain	Default	Default		1	No

    b. Domain 2

        i. Domain Name = "Difficulty"

        ii. Description = "Trail Difficulty Level"

        iii. Field Type = Short Integer

        iv. Domain Type = Coded Values Domain

        v. Codes

            1. Code = 0, Description = "Easy"

            2. Code = 1, Description = "Moderate"

            3. Code = 2, Description = "Challenging"

            4. Code = 3, Description = "Difficult"

            5. Code = 4, Description = "Very Difficult"

            6. Code = 5, Description = "Extreme"

        vi. Split Policy = Default

vii. Merge Policy = Default

Domain Name	Description	Field Type	Domain Type	Split Policy	Merge Policy		Code	Description
Residency	West Virginia Resident?	Short	Coded Value Domain	Default	Default		0	Easy
							1	Moderate
Difficulty	Trail Difficulty Level	Short	Coded Value Domain	Default	Default		2	Challenging
							3	Difficulty
							4	Very Difficult
							5	Extreme

c. Domain 3
    i. Domain Name = "Rating"
    ii. Description = "Rating: 0 to 5 stars"
    iii. Field Type = Double
    iv. Domain Type = Range Domain
    v. Split Policy = Default
    vi. Merge Policy = Default
    vii. Minimum Value = 0
    viii. Maximum Value = 5

Domain Name	Description	Field Type	Domain Type	Split Policy	Merge Policy		Code	Description
Residency	West Virginia Resident?	Short	Coded Value Domain	Default	Default		1	Moderate
							2	Challenging
Difficulty	Trail Difficulty Level	Short	Coded Value Domain	Default	Default		3	Difficulty
							4	Very Difficult
Rating	Rating: 0 to 5 Stars	Double	Range Domain	Default	Default		5	Extreme

☐ Click save under the Domains Tab to save the Domains.

Note: You are using Double as the Field Type for the last Domain so that you can store non-integer ratings, like 4.5 stars.

Step 5. Create a Table

You are now ready to create a table to store the trail reviews.

☐ Double-click on the **wv_trails_database.gdb** in the Catalog View to open it.
☐ Right-click within the geodatabase then select New followed by Table. This will open the Create Table window.
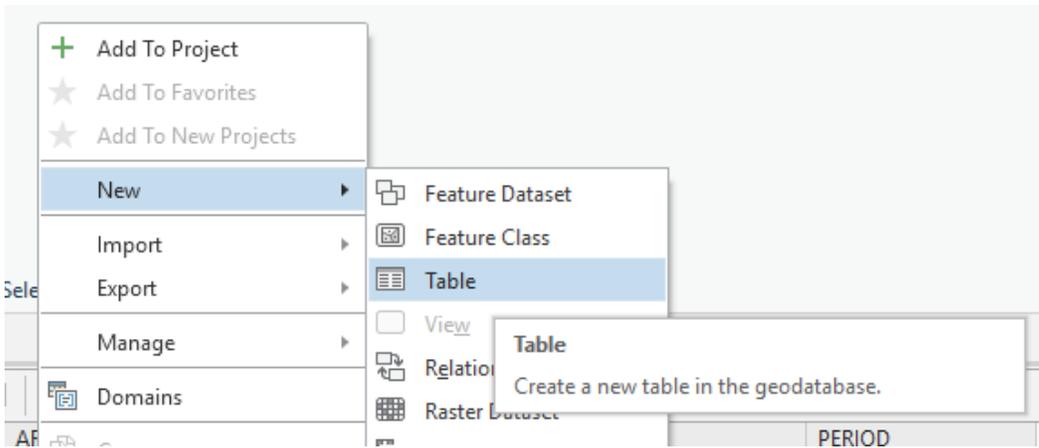
- Name the table **Trail_Reviews**.
- Set the Alias to "Review of WV Trails" (without the quotes).
- Click Next.
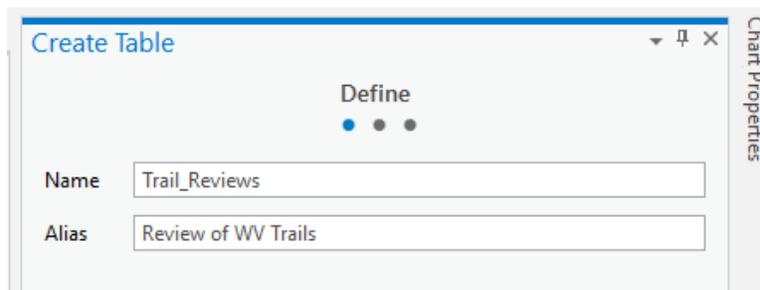
- You will need to create the following fields with the following settings.
  - a. "Residency"
    - i. Field Name = "Residency"
    - ii. Data Type = Short Integer
    - iii. Default Value = 0
  - b. "Date"
    - i. Field Name = Date
    - ii. Data Type = Date
  - c. "Difficulty"
    - i. Field Name = Difficulty
    - ii. Data Type = Short Integer
  - d. "Rating"
    - i. Field Name = Rating
    - ii. Data Type = Double
  - e. "Review"
    - i. Field Name = Review
    - ii. Data Type = Text
    - iii. Length = 2000
  - f. "Trail_ID"

  i. Field Name = Trail_ID
  ii. Data Type = Short Integer
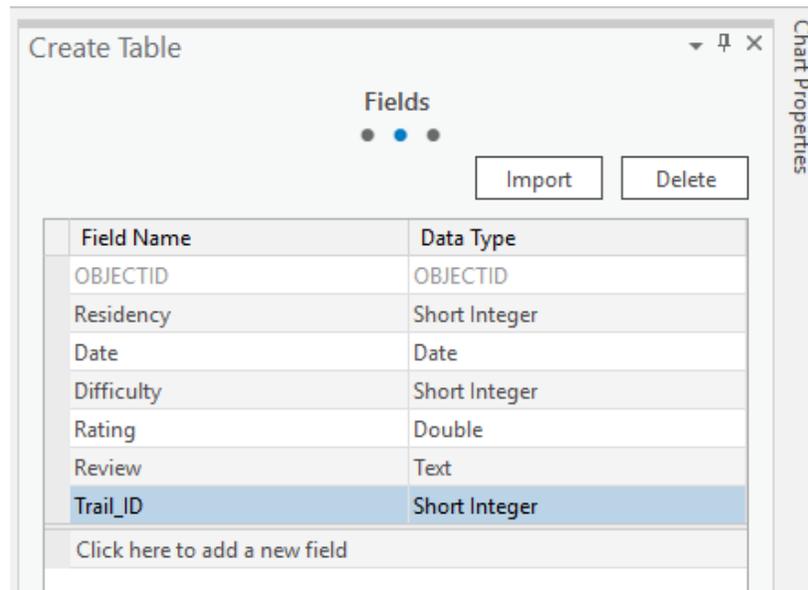
  ☐ Select Next.
  ☐ You do not need to make any changes to the storage configurations. Click Finish.

You will now need to associate the domains with the fields to which they should be applied.

  ☐ In the Geoprocessing Pane search for the **Assign Domain to Field Tool**.
  ☐ Create the following domains:
   a. For Residency:
    i. Input Table = Trail Reviews
    ii. Field Name = Residency
    iii. Domain Name = Residency
   b. For Difficulty:
    i. Input Table = Trail Reviews
    ii. Field Name = Difficulty
    iii. Domain Name = Difficulty
   c. For Rating:
    i. Input Table = Trail Reviews
    ii. Field Name = Rating
    iii. Domain Name = Ration

Notes: The domains created above are used here to specify the allowed values for each field. Now, a user will not be able to input values that are

not allowed. The only options for residency are yes and no, they must enter one of the predefined difficulty levels, and they must provide a rating between 0 and 5. Note that the data type of a field has to be the same as the data type of the associated domain. We also had you specify the Default Value for the "Residency" field as 0, which means yes or that the reviewer is a West Virginia resident. Note that this field was defined as a short integer as opposed to text because you are using coded values.

Step 6. Create a Relationship Class

When a review is created, it should be linked to a specific trail. To do this, you will need to set up a relationship class between the **wv_trails** layer and the **Trail_Reviews** table. You might be thinking that reviews could simply be saved in the attribute table of the **wv_trails** layers; however, you want to be able to store multiples reviews for each trail, which would be difficult to implement in the attribute table. In contrast, a relationship class allows for the creation of one-to-many relationships. So, the reviews will be stored in the table and the table will be associated with the trails so that it is clear what trail is being reviewed.

- Right-click on the **wv_trails_database.gdb** in the Catalog View then select New followed by Relationship Class. This will open the **Create Relationship Class Tool**.
- Define the following settings:
  - a. Origin Table = **wv_trails**
  - b. Destination Table = **Trail_Reviews**
  - c. Output Relationship Class = **wv_trails_Trail_Reviews**
  - d. Relationship Type = Simple
  - e. Forward Path Label = **Trail_Reviews**
  - f. Backward Path Label = **wv_trails**
  - g. Message Direction = None (no message propagated)
  - h. Cardinality = One to many (1:M)
  - i. Relationship Class is attributed = don't select
  - j. Origin Primary Key = Trail_ID
  - k. Origin Foreign Key = Trail_ID
- Click Run to create the relationship class.

Step 7. Create Subtypes

Your last task will be to created subtypes for the trails. Subtypes define categories of the data. Here, we will define subtypes for the trails as trails, water trails, and motorized trails.

☐ In the geodatabase, right-click on the **wv_trails** feature class then select Design followed by Subtypes.
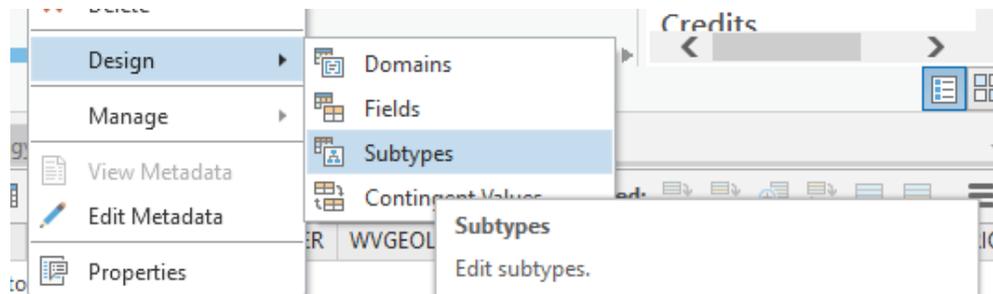
☐ From the list of fields, right-click on Type and select Set as subtype.
☐ Define the following Codes and Descriptions:
  a. Code = 0, Description = Trail

b. Code = 1, Description = Motorized

c. Code =2, Description = Water

☐ Click OK to apply the subtypes. Click Save in the Subtypes Tab to finalize the subtypes.

Note: The "Type" field was already populated with 0, 1, 2 values to code the three trail types. Subtypes are generally specified using coded values as opposed to a text field.

☐ You have now prepared the file geodatabase.

Step 8. Investigate the File Geodatabase

☐ Navigate to the Map tab. Use the Add Data button to add all of the layers from the database you just created.

☐ Take some time to change the symbology of the data layers.
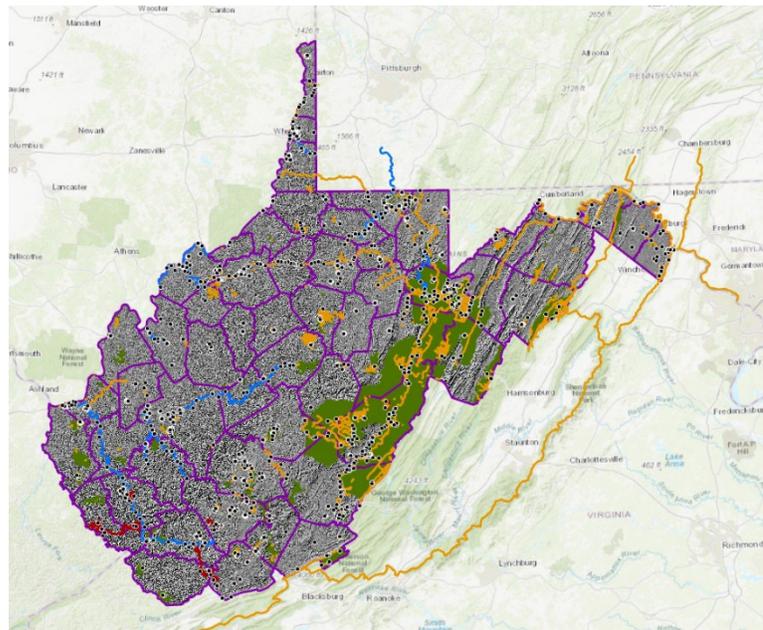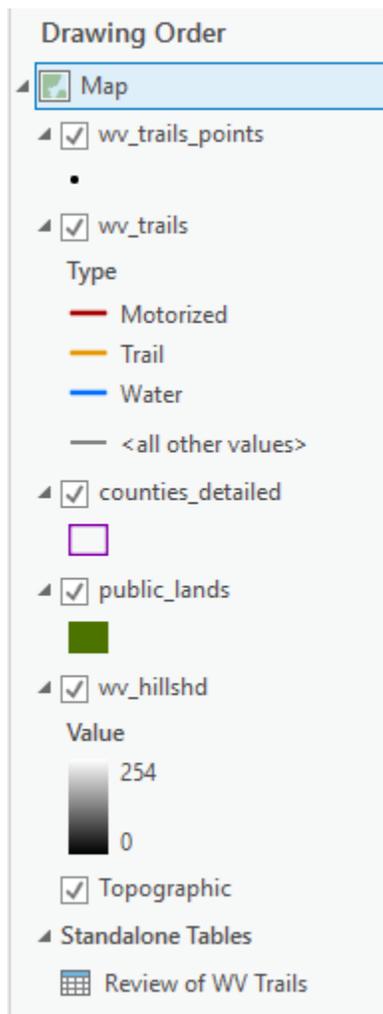
Here are a few comments about the data. First, noticed that different lines symbologies were automatically assigned to the three different types of trails. This is because you defined subtypes. Also note that the table you created does not contain any records yet. Here we will provide an example of how to add a review that is associated with a specific trail.

☐ Navigate to the Edit Tab.
☐ Click on the Create button in the Features area. This will open the Create Features Pane.

Note: You can now edit any feature in the geodatabase. The three different types of trails are available for editing. This is one of the benefits of adding subtypes.

☐ Click on the Select button in the Selection area of the Edit Tab.
☐ Click on a trail line feature. It doesn't matter which one. This will select it.
☐ Click on the Attributes button in the Selection area of the Edit Tab. This will open the Attributes Pane.
☐ Expand all of the arrows under **wv_trails**.
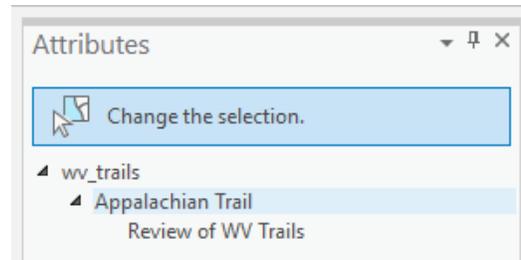
Note: The review table appears under the trails.

☐ In the Attributes Pane, right-click on Reviews of WV Trails. Select Add New to Relationship. You can now provide a review for the trail.
☐ Fill out the review. It doesn't matter if the information is correct. This is just to confirm that your table is working correctly. You should notice that the "Residency" field is already populated with Yes, since this was set as the default. Since you did not set a default for the other fields, they are populated with <NULL> until a value is provided. The last field ("Trail_ID") is populated with a number that acts as a unique ID for
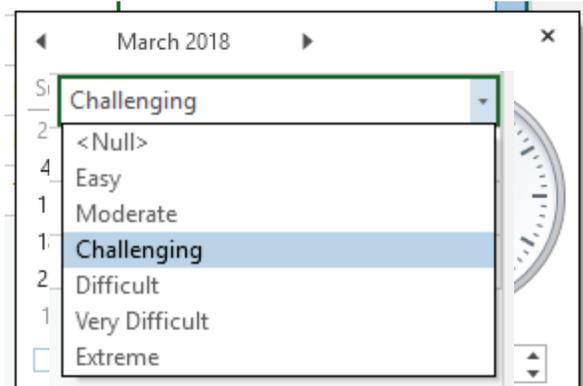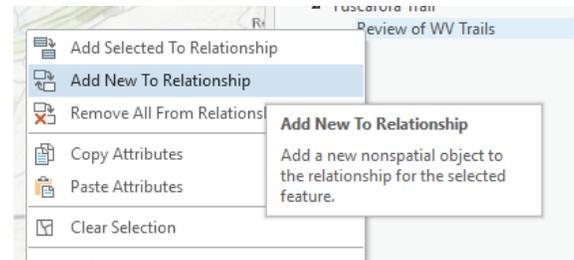
the trail. This relationship was defined when setting up the relationship class. When you try to edit the date, a calendar is provided to make this selection. Drop down lists are provided relative to the domains defined.

Below is an example review.

OBJECTID	1
Residency	Yes
Date	3/9/2018 12:00:00 AM
Difficulty	Challenging
Rating	3.5
Review	Great hike, but it rained.
Trail_ID	2480

Deliverables 1. Write a review for a trail. The information does not need to be correct. Provide a screen capture of the information you provided. (10 Points)

Deliverables 2. Provide a screen capture of your map space that shows how you symbolized the data layers. (10 Points)

Deliverables 3. Write a short paragraph describing some of the advantages of using a geodatabase instead of individual files. (20 Points)

END OF EXERCISE