# URP 6275: Module 3, Homework 3 Relationship Classes

## ArcGIS Pro Instructions

**Before you begin the assignment please download the data from Canvas**

**Please email Kate Norris with any homework questions via Canvas Email.**

Creating relationships between objects, such as a feature class and non-spatial tables.

In this 4 Part Homework, we're going to look at a couple of different tools and techniques for importing data to a FGDB and creating relationship classes between Parcel Data, Owner Information, Buildings, and Sales Tables. This homework can best be accomplished by first watching the Demo, if you haven't already. The tools covered are Import Table (Multiple), and Create Relationship Class. The Techniques covered include; Building Relationship Statements, Using Relationship Classes in ArcGIS Pro, identifying features with relationships in ArcGIS Pro, querying tables and features with relationship classes in ArcGIS Pro, and QAQC.

## Part 1: Import a table

The OwnersPub.txt, BuildingPub.txt and SalesPub.txt CAMA (Computer Assisted Mass Appraisal) Database tables contains owner, building and sales information for the parcels in the alachua_pin feature class in the Alachua County Parcel 2016 File Geodatabase (FGDB). To create relationships between the parcels and their owners, buildings and sales information, the owner and building and sales information must be imported into the Alachua County Parcel 2016 FGDB.

### Step 1:

Add your downloaded data to the map. In the catalog window, Right-click the **Alachua County Parcel 2016 FGDB**, point to **Import**, and click **Table(s) (multiple)**.

### Step 2:

Using the shift and ctrl keys select and add the **OwnersPub.txt**, **BuildingPub.txt** and **SalesPub.txt** CAMA Data tables to the **Input Table** text box.

### Step 3:

Click **Run**.

The alachua_pin, OwnersPub, BuildingPub, and SalesPub tables are now in the Alachua County Parcel 2016 FGDB.

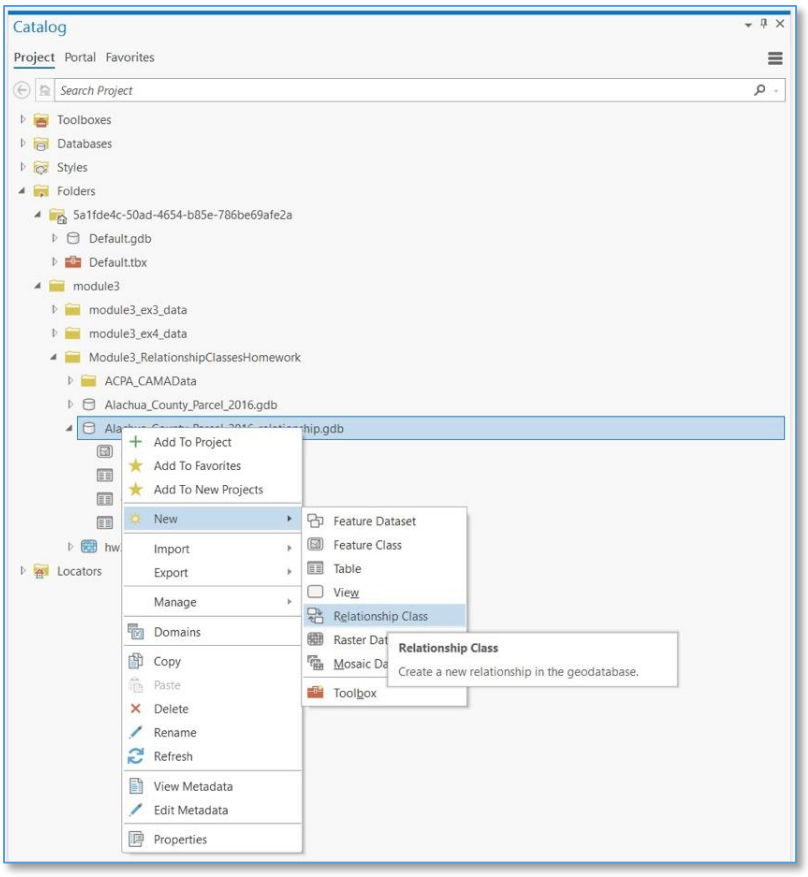
# Part 2: Create Relationship Classes

In Part 1, you imported .txt tables containing owner, building and sales objects into the Alachua County Parcel 2016 FGDB. The FGDB already has a feature class, alachua_pin that contains parcel objects. You will now create a relationship class between the parcels and the owners, building and sales tables so that when you use the data in ArcGIS Pro, you can easily find out who owns which parcels, what kind of buildings are located on each parcel, and what the parcels previous sold for.

***Tip*:** To build your relationship statements, start with the name of the origin feature class, followed with Has or Have, and end with the name of the destination feature class, such as; "Owners have Parcels," "Parcels have Buildings" and "Parcels have Sales Information." For example, in a parcel database, you might have a relationship class between parcels and owners in which owners "own" parcels and parcels are "owned by" owners. You could also have a relationship class between parcel and sales information in which parcels "contain sales" and the flip of that "sales contained by" parcels. Other examples include, AddressHasZones or ParcelsHaveOwners (Many-to-many relationship).

## Step 1:

Right-click the Alachua County geodatabase in the **Catalog** window, point to **New**, and click **Relationship Class**.



## Step 2:

Make the following selections within the Create Relationship Class tool (Example screen grab at end of text description)

Origin Table:

- Navigate to the **Alachua County file geodatabase**. Click **OwnersPub** in the Origin table/feature class

Destination Table:

- Click **alachua_pin** in the Destination table/feature class list.

Output Relationship Class:

- Replace the default name with the following, **ParcelOwners**, this will be the name of the Relationship Class that is stored in the FGDB.
    - This designates the Alachua Parcels feature class (alachua_pin) as the destination feature class.
    - When creating a one-to-many relationship, whether simple or composite, the one side must be the origin class. The many side must always be the destination class.

Relationship Type:

- Click **Simple**.
    - You are creating a simple relationship class, since owners and parcels can exist in the database independently of each other. If you were mapping fire hydrants and water mains you would have to choose a composite relationship as fire hydrants can't exist without a water main. The same is true for transformers and electrical poles.

Forward Path and Backward Path Labels:

    - You must now specify the path labels and the message notification direction. The forward path label describes the relationship as it is navigated from the origin class to the destination class—in this case, from Owners to Parcels. The backward path label describes the relationship when navigated in the other direction—from Parcels to Owners.
- Type **owns** for the Forward Path label.
- Type **is owned by** for the Backward Path label.

Message Direction:

- The message notification direction describes how messages are passed between related objects. Message notification is not required for this relationship class; therefore, you can accept the default of **None**.
- By default, the notification direction for a simple relationship is **None**.

Cardinality:

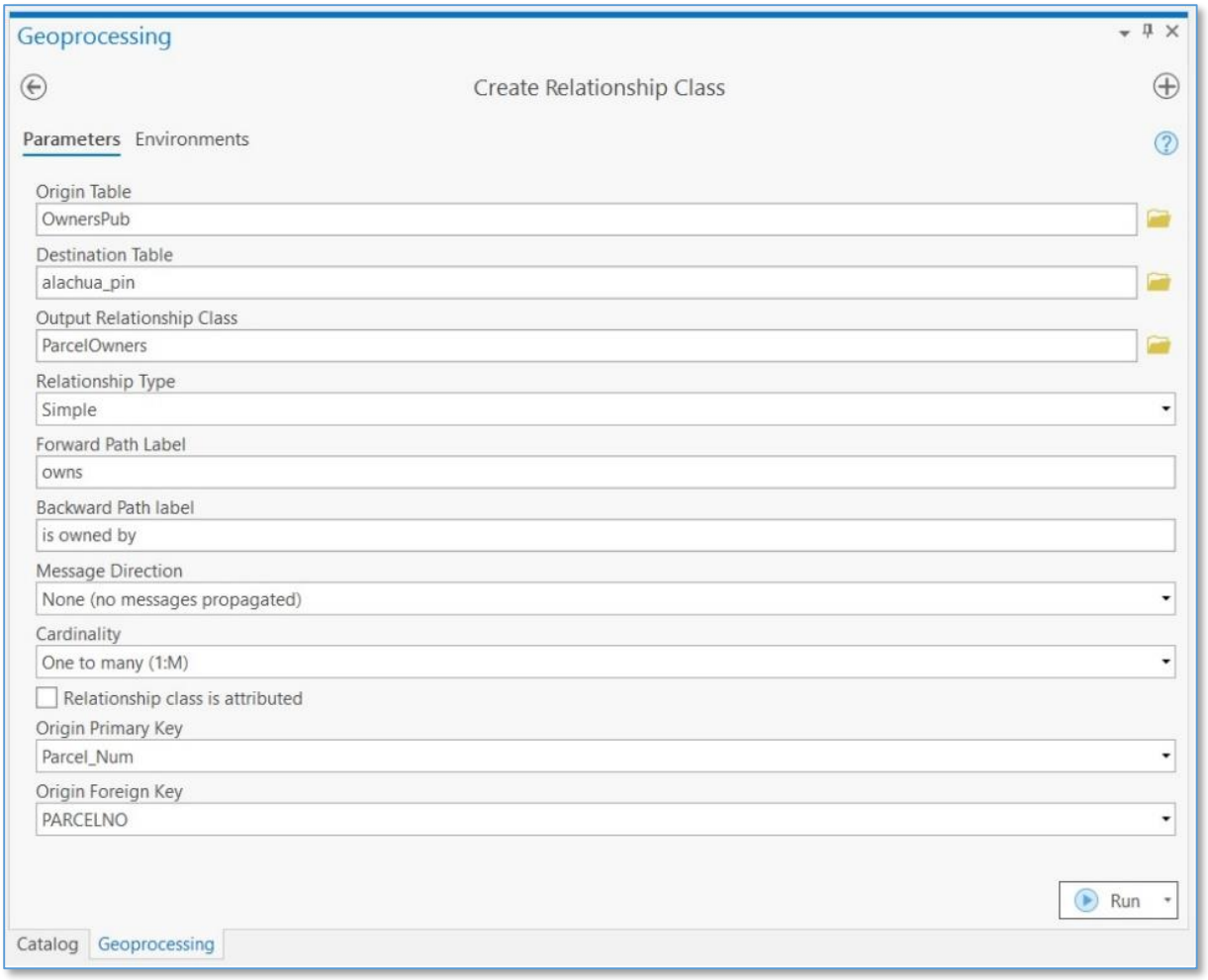
- You will now specify the cardinality of the relationship. The cardinality describes the possible number of objects in the destination feature class or table that can be related to an object in the origin feature class or table.
- Click **One to many (1:M)** to specify that one owner can own many parcels.
    - ***Tip*****:** Many-to-many (M:N) relationship classes require the relationship class to have its own table in the database. You can optionally add attributes to this table, or you can allow ArcGIS to manage the schema of the table for you.

Relationship class is attributed:

- You must now specify whether your new relationship class will have attributes. In this example, the ParcelOwners relationship class **does not require attributes**, which is the default (**unchecked**).
    - ***Tip*****:** In this example, you are not adding attributes to your relationship class, although any relationship class can have attributes. Relationship classes with attributes are stored in a table in the database. This table contains at least the foreign key to the origin feature class or table and the foreign key to the destination feature class or table. For example, in a simple relationship between a feature class that stores water laterals (a water lateral line is a small-diameter pipe that runs from the main line to a hydrant) and a feature class that stores hydrants. Water lateral objects have their own attributes, and hydrant objects have their own attributes. The relationship class in this example describes which water laterals feed which hydrants. Because you want to store some kind of information about that relationship—such as the type of riser connecting the two—you can store this information as attributes in the relationship class.

Origin Primary Key and Origin Foreign Key:

- The next step is to specify the primary key in the origin table (Owners) and the embedded foreign key field in the destination feature class (Parcels). Owners and Parcels that have the same value in these fields will be related to each other.
- Select: **Parcel_Num** as the Origin Primary Key.
- Select: **PARCELNO** as the Origin Foreign Key. This is the embedded foreign key in the destination feature class.




## Step 3:

Click **Run**.

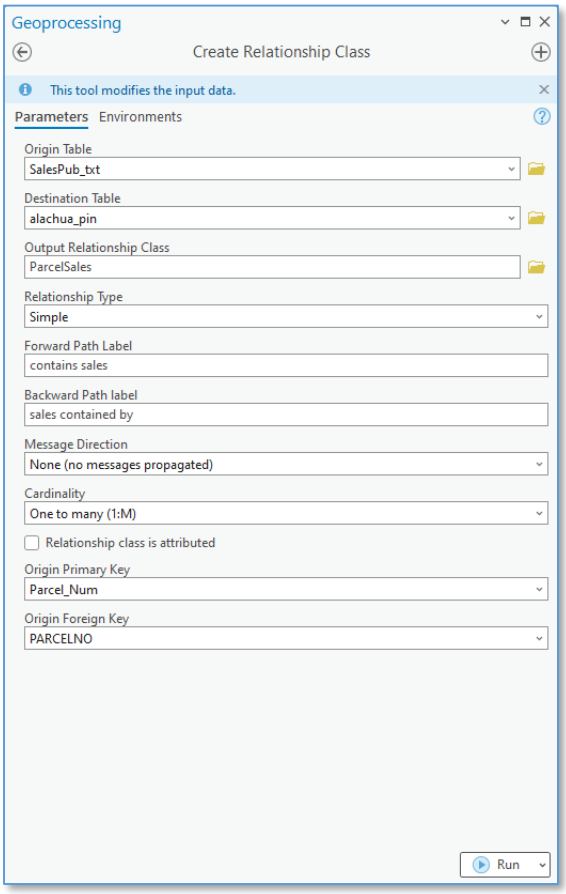
Once you've created the relationship class, it appears in the Catalog tree, and you can inspect its properties as well as relationships for any particular feature class.

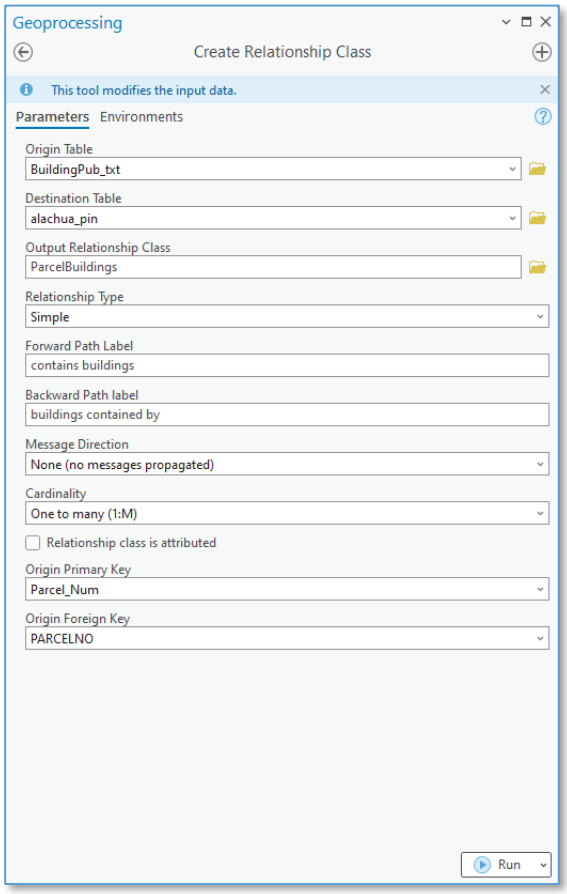
- ***Tip*:** You can't modify any of the properties you specified when you created the relationship class except for renaming. You can set and change relationship rules at your discretion. If you need to change other properties, it is easy to **delete the relationship class** and re-create it.
- ***Tip*:** The Catalog window does not allow you to change the properties of relationship classes or any other geodatabase datasets if any of the geodatabases' contents are being accessed in ArcGIS Pro. Exit any other instances of ArcGIS Pro before you make the changes.

# Part 3: Create Relationship Classes Continued

Next run through the same process again for the **Sales** and **Building** tables, but this time switch the origin and destinations, so that the Parcel data is the Origin and the other tables the Destinations.

**Please Note:** Because of the tables used in this assignment and their relationships the demo video shows another way of joining the tables. Either format will be accepted for the submission. Please, however, take this time to consider the cardinality relationship between the origin and destination tables.



# Part 4: Working with Relationship Classes in Map

In Map, you can explore what objects are related to any particular object in your geodatabase.

- When you identify a feature in your map, you can see the objects related to that feature in the Explore pop-up dialog box. If the related object that you navigate to in the Explore pop-up dialog box has objects related to it through other relationships, you can continue to navigate to those related objects.
- When you select one or more rows or features in a table, you can open the related table and select the related objects.
- **To support queries that return related objects, you must include the table and layer involved in the relationship class in the map. If either the origin or destination layer or table is not included in the map, the pop-up ignores the relationship.**

## Step 1:

Click **List By Data Source** tab in the Contents Pane

## Step 2:

Right-click the table that contains the objects whose related objects you want to explore and click **Open Attribute Table.**

## Step 3:

Select the objects whose related objects you want to explore.

## Step 4:

Click **Options** > **Related Data**, then click the path label for the relationship.

## Step 5:

Click **Show Selected** to display only those objects related to the selected objects in the first table.

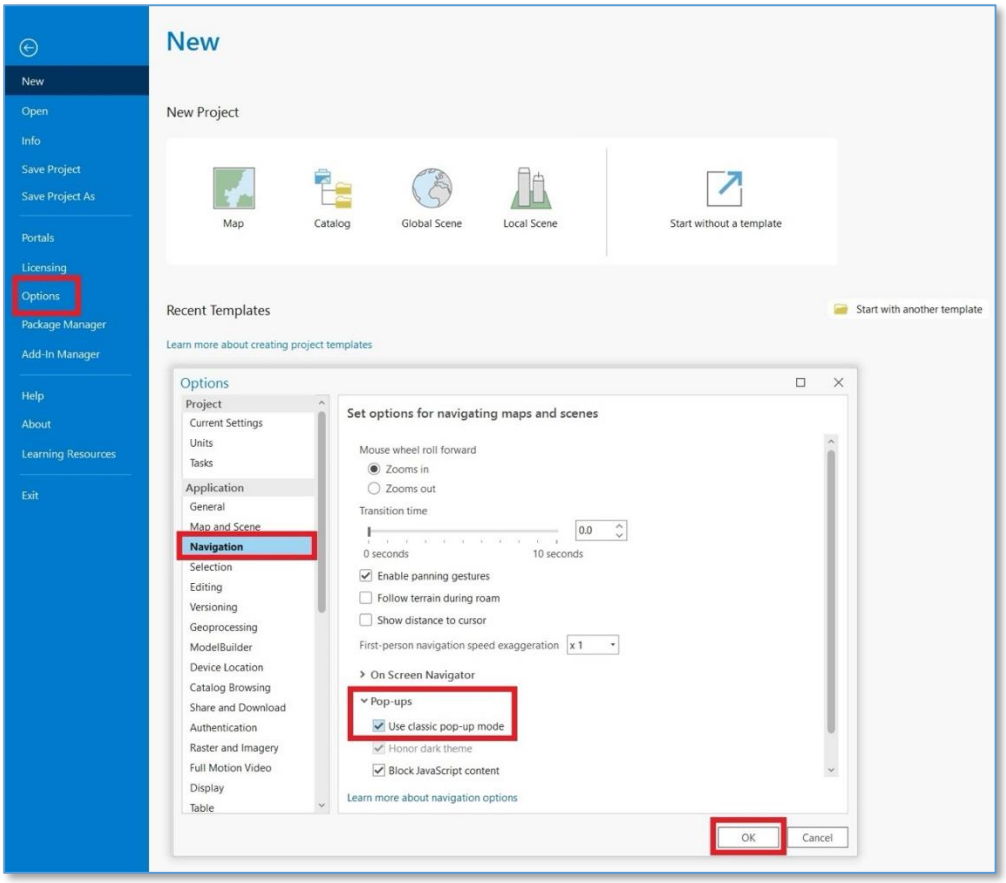
***Tip*: Why can't I see my Relationship Class in the Pop-up?**

Esri recently changed the way Pop-ups display information, follow the steps below to make your Relationship Classes visible using View pop-ups in classic mode

To view pop-ups in classic mode, perform the following steps:

1. In an open project, click the **Project** tab on the ribbon. Alternatively, on the ArcGIS Pro start page, click **Settings** in the lower left corner.
2. In the list on the left, click **Options**.
3. On the **Options** dialog box, under **Application**, click **Navigation**.
4. Under **Pop-ups**, uncheck the **Honor dark theme** option to display the light-themed coloring at all times regardless of the current application theme. Optionally, check the **Use classic pop-up mode** option to have pop-ups use classic styling.

*(Example screen grab on next page)*

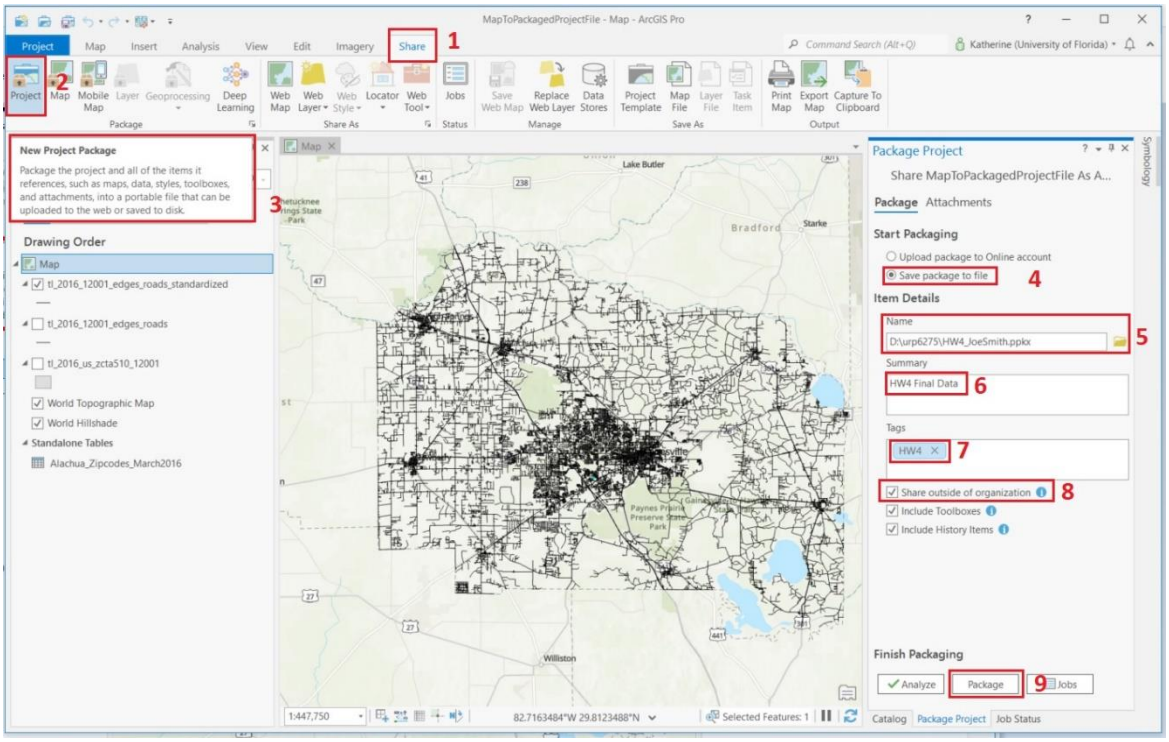



***Tip*:** If a relationship between two objects in a simple relationship class of 1:1, or 1:M is deleted, the value for the foreign key in the destination object is replaced with a null. Because of this the foreign key in the destination object must have fields that can be nulled. If this happens and the foreign key is set to Null, orphaned objects can exist in the relationship's destination class. This is allowed because simple relationship classes are designed to relate objects that can exist independently of each other. To identify these orphaned destination objects, relationship rules must be setup to prevent them, after which the Validate Features tool will identify any orphaned destination objects.

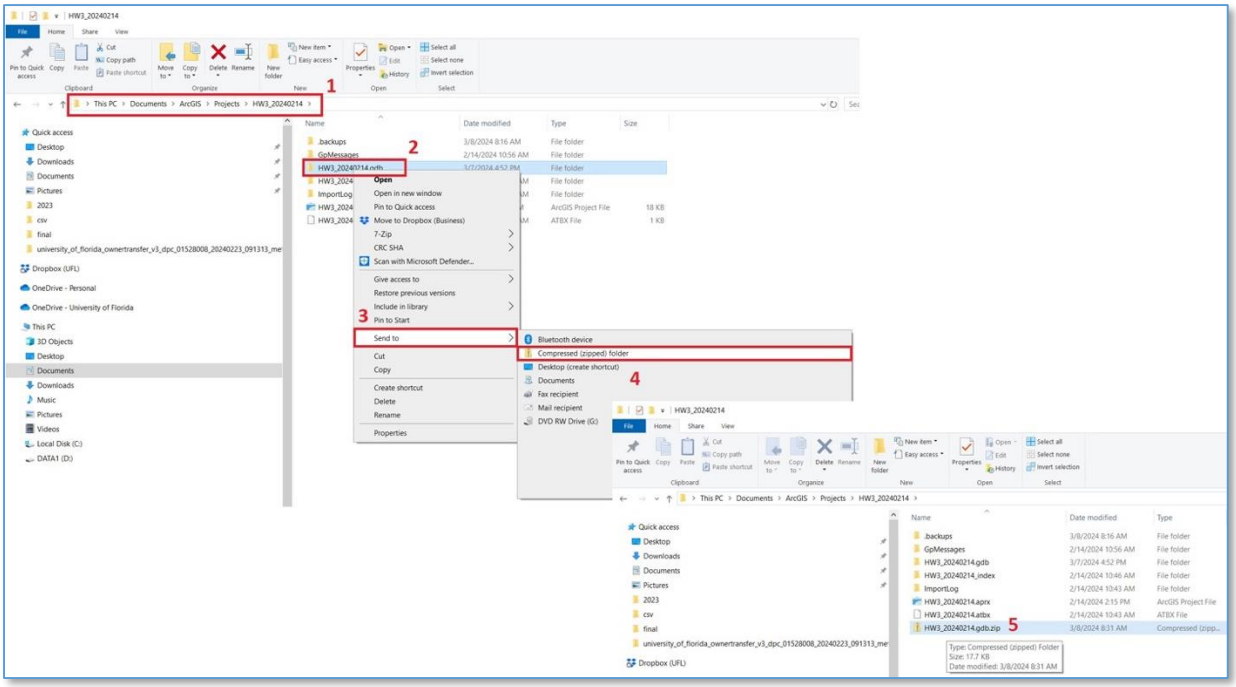
## For Submission

Submit your FGDB (Zip file or PPKX file) with all of your data, along with the three constructed relationship classes.

*The screen grabs on the next page show how to create a Zip or PPKX file of your FGDB*




*How to create a Project Package (PPKX)*




*How to zip a File Geodatabase*