

[1420] Database Access Example

There are several different ways to access databases using C# with Windows. The following example shows you how you can access an existing local database using C#'s Entity Framework. **This only works with Windows.**

Here are the general steps for setting up this program:

- 1.) Download the database files by clicking on this link: [FoodStoreDatabase.zip](https://lwtech.instructure.com/courses/1841516/files/122992675/download?wrap=1)
(<https://lwtech.instructure.com/courses/1841516/files/122992675/download?wrap=1>)
- 2.) Open up your Windows "Downloads" directory in a file window, right-click on the FoodStoresDatabase.zip file and select "Extract All..." from the popup menu that appears.
- 3.) Press the "Extract" button to uncompress the database files. They should appear inside of the "FoodStore Database" folder.
- 4.) Create a new Visual Studio Console App project called "FoodStoreViewer"
- 5.) Pull out Visual Studio's "Server Explorer" pad (on the left side of the screen) and then right-click on "Data Connections" and select "Add Connection..." from the popup menu.
- 6.) Select "Microsoft SQL Server Database File" from the list of Data Sources that appears. (Or, if you aren't given a choice, click on "Change..." next to the Data source fields and then select SQL Server Database File.)
- 7.) Click the Browse... button, open up the Download folder and click down into the FoodStoresDatabase folders until you find "FoodStore.mdf" (If you cannot find it, use the Search box in the upper right corner of the window).
- 8.) Click on "FoodStore.mdf" and press "Open"
- 9.) Make sure "Use Windows Authentication" is selected, then click "OK"
- 10.) A message about "upgrading the database file" will appear. Just click "Yes"
- 11.) The "FoodStore.mdf" node should appear in the "Data Connections" area of the "Server Explorer" pad. Feel free to click on the triangle next to it and explore the Tables in the database to get a sense of what it contains.
- 12.) Now go over to the right side of the screen and right-click on the "FoodStoreViewer" project in the "Solution Explorer" panel (i.e., the second item from the top of the list).
- 13.) Select "Add" and then "New Item" from the popup menu that appears.
- 14.) Choose "ADO.NET Entity Data Model" and name it "FoodStoreModel", then click "Add"

15.) Click "Next". Click "Next". Answer "Yes" when it offers to copy the database to your Project. Click "Next" one more time.

16.) Finally, **click on the checkbox next to "Tables"** then click "Finish" to have Visual Studio create your database model. When the process is complete (it takes a little while), you'll see a diagram similar to the following one appear:

This is called an "Entity-Relationship Diagram" and it shows the tables and fields in the database as well as the relationships between tables.

In addition, several auto-generated Class files were added to the project. You can - and should - examine those classes to see how they work HOWEVER you should never edit those classes! (If your database needs to change after this step, you'll need to delete this model and create a new one from the new database files.)

You can now easily use those Classes to access the data in the database using code like this:

```
using System;
using System.Data.Entity;

namespace FoodStoreViewer
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("FoodStore Products and Prices:");
            Console.WriteLine("=====");

            FoodStoreEntities db = new FoodStoreEntities();

            DbSet products = db.Products;

            foreach (Product product in products)
            {
                decimal price = (decimal)product.price;
                string priceStr = price.ToString("C");
                Console.WriteLine($"{product.productID}: {product.name} - {product.vendor} - {priceStr}");
            }

            Console.ReadLine();
        }
    }
}
```

Output:

In-Class Group Exercise

How would we modify this program so that it also prints out the name and location of each store in the database?