

# **Visual Foxpro Tutorial 2**

By: David Hosick

## **Introduction**

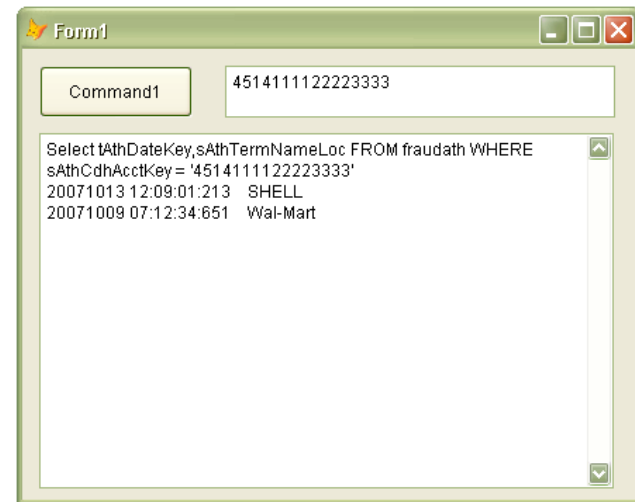
For this project we will make a small application that obtains a card number, and then shows all the fraudulent transactions done with this card.

### **Primary Aim:**

- How to create a new program
- Calling external code from buttons
- Demonstrates IntelliSense

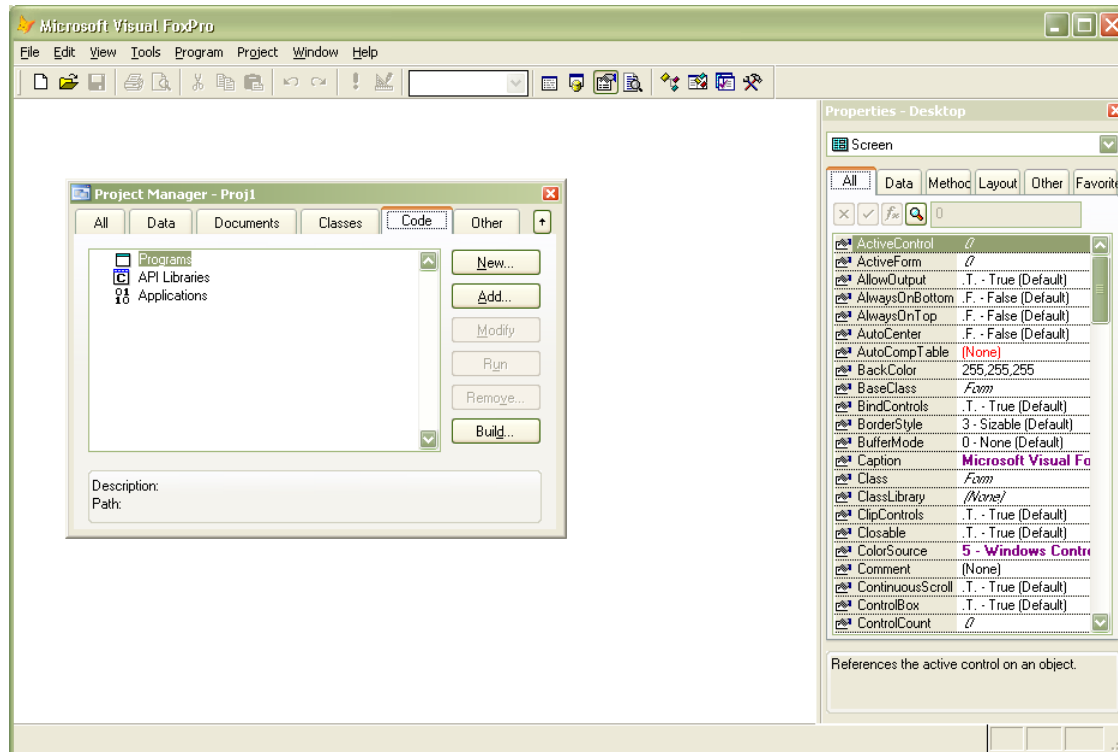
### **Secondary Aim**

- Shows how to connect to CMAPrism database
- How to prepare simple queries
- Briefly mentions how to process results of queries



## Section 1, Creating the GUI (Graphical User Interface)

Step 1. Create a new project. Once the project is saved, close FoxPro and reopen the project by navigating to the file's location in 'My Computer'. Then open the project file by double clicking on it. If you don't do this trick FoxPro will later have trouble finding ancillary files we create for this project.

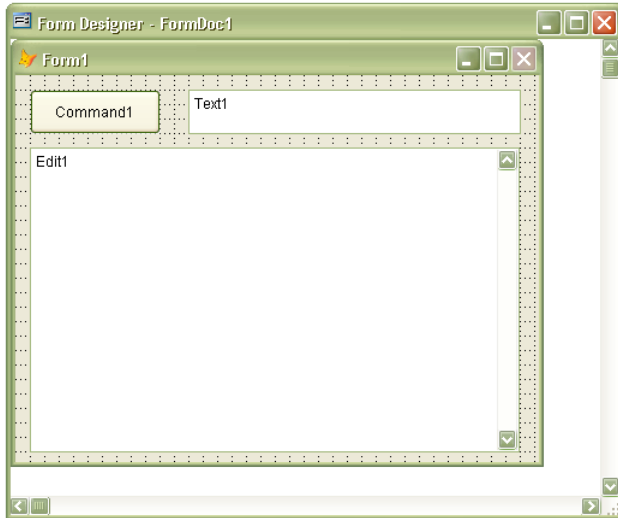


### Paths

FoxPro remembers the directory you are in when you started the program up. It looks in this directory for additional files.

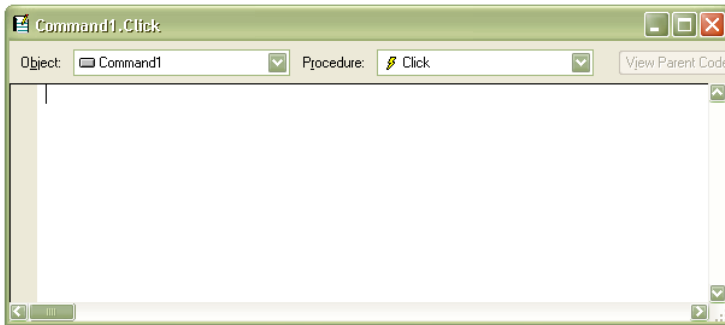
Step 2. Create a new form that is similar to the one shown in the image below. We will require the following objects.

- 1 Button
- 1 Textbox
- 1 Editbox



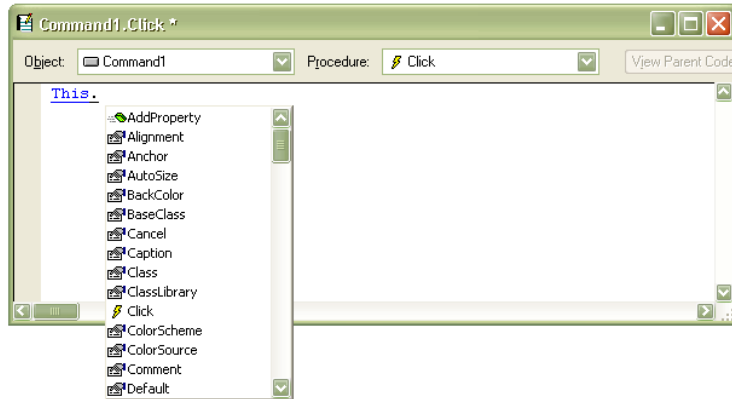
Test your work  
Why not test your form out? Hit CTRL-E to give your form a test run.

Step 3. Open the 'Click' method for the button called 'Command1'. Once it is opened you should see the following window.



Short Cut  
There is a quick way to open the 'click' method.  
  
Just double click on the object in the form designer window.

Step 4. Type the word 'This' and then a period, a pop-up will appear. This is known as IntelliSense. It is there to help by showing you all possible properties and functions an object can do.

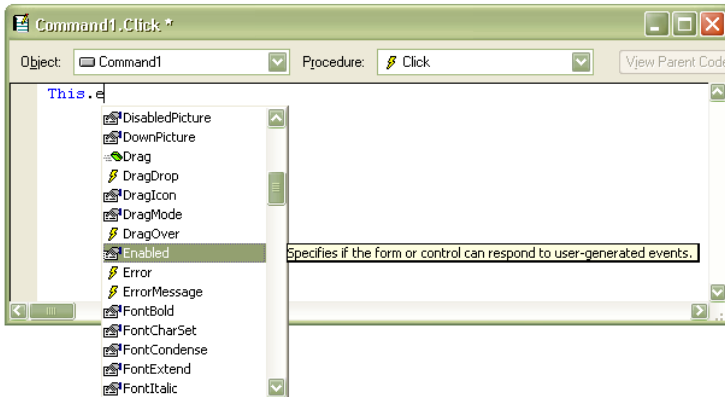


### IntelliSense

Hitting ESC will make the IntelliSense pop-up go away.

If you want it to come back either hit CTRL-J, or erase the last period and type it again.

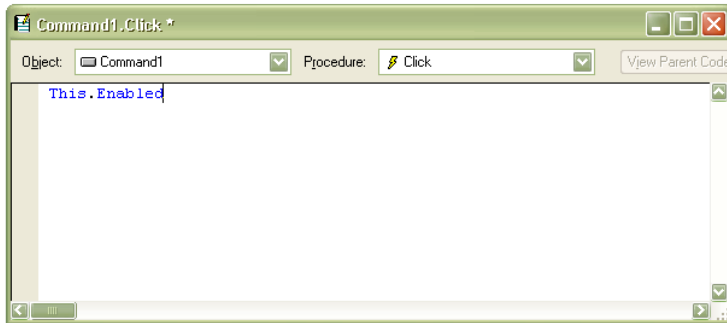
Step 5. We want to set this button's 'Enabled' property. Begin typing the word 'enabled'.



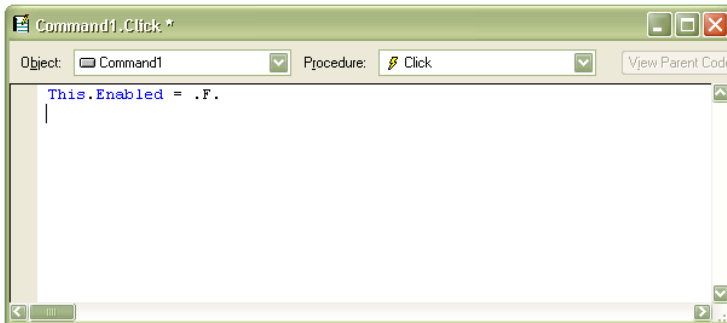
### Short-Cut

You can use the up and down arrows to scroll through the list.

Step 6. You will notice that 'Enabled' was highlighted after typing the first letter. Because the correct thing is highlighted, hit the 'Tab' key. IntelliSense will then write out the entire word for you.

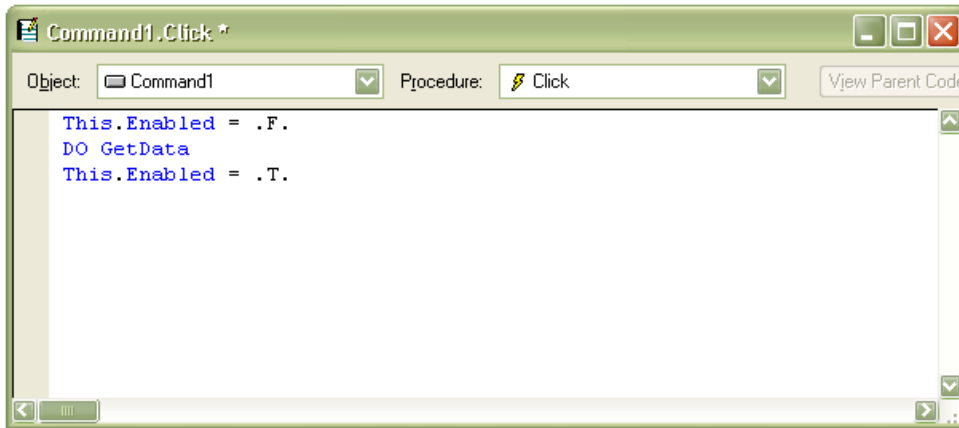


Step 7. Continue by putting a 'space' and then an equal sign '=', now another helpful pop-up appears showing you the possibilities for this property. It can either be True, or False. Use the up and down arrows to select False. Then hit Enter.



Step 8. Continue entering the lines as shown in the image below: In brief, those three lines will be run whenever a user pushes the button on your form. When a user does push it, this happens:

- The program disables the button, so it cannot be pushed again.
- The program runs the 'GetData' program (we will write this program next)
- Once the 'GetData' program is finished, the button is turned back on.



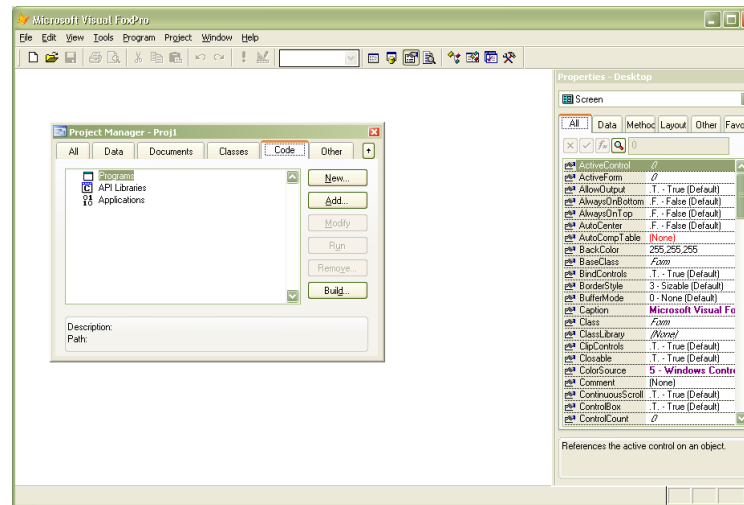
```
Command1.Click *
Object: Command1 Procedure: Click View Parent Code
This.Enabled = .F.
DO GetData
This.Enabled = .T.
```

### Don't Worry

It's alright that you have no idea what that 'DO GetData' line does.

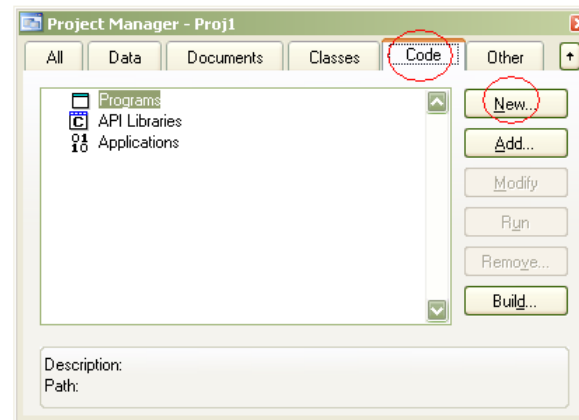
We haven't yet written the program called 'GetData'.

Step 9. You can now close the 'Click' method editor and also the form editor. You will be prompted to save your work, which you should agree to. After doing this, you should find yourself with an empty screen except for the 'Project Manager'.

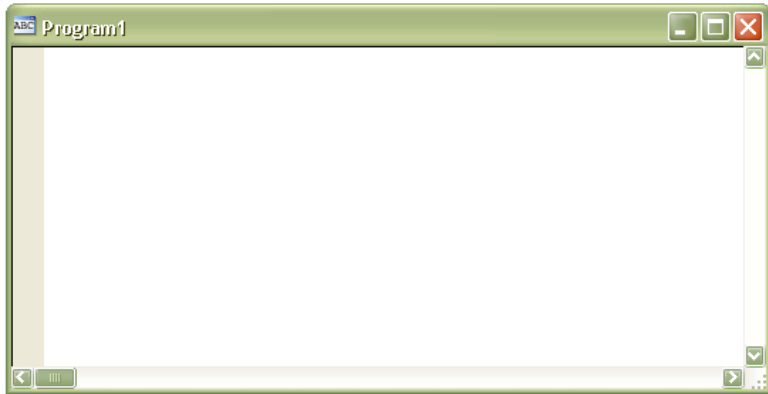


## Section 2, Writing the GetData Program

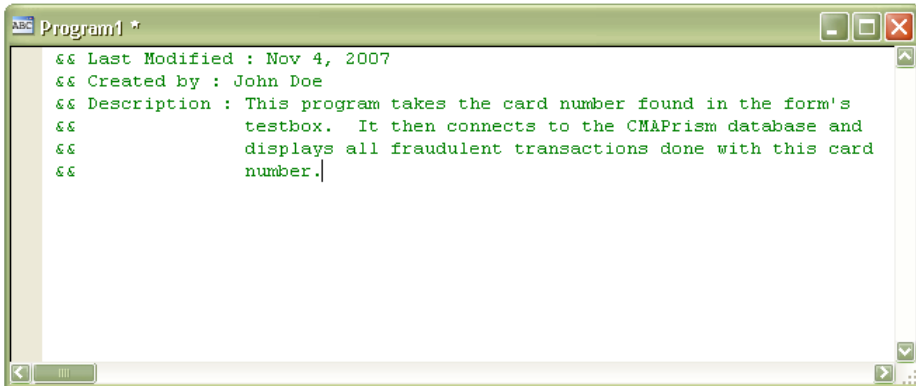
Step 10. We now need to make the program called 'GetData' that our form calls when a user hits the command button. Do this by selecting the 'code' tab in the project manager. Then, click 'new'.



Step 11. If you created a new program successfully, you will see a screen like the one below.



Step 12. We begin by writing some introductory comments that will help us, and future programmers who may need to modify the code. Granted, for such a short program this is really unnecessary, but it is nonetheless a good habit to get into.



### Comments

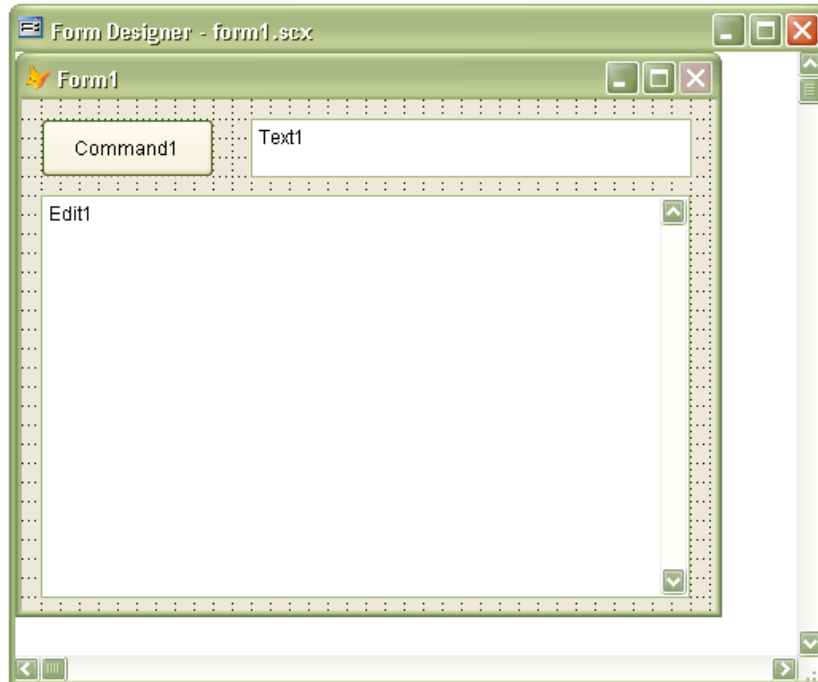
Make sure you are liberal with your comments. Other people will one day have to modify your code. And, it could be hard to ask you directly what your code does, you may no longer even work for the company.



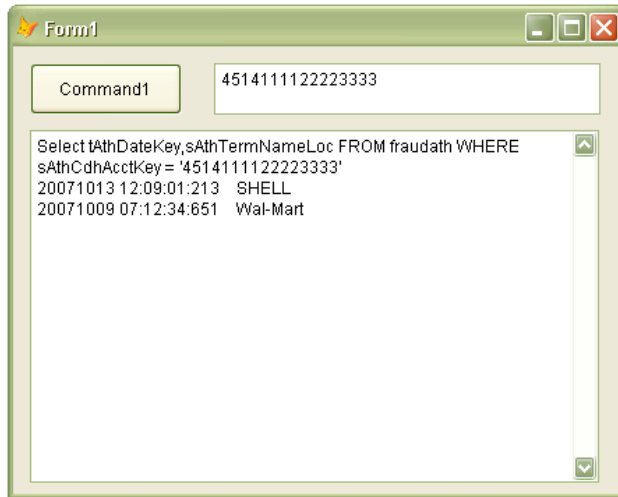




Step 16. You have now finished programming the tool. Save your code, and go back to the form (go to the screen shown below) and hit CTRL-E to run the application.



Step 17. Make sure you plug in a card number that has fraudulent transactions marked in the CMAPrism database. In other words, a card marked over 24 hours ago. Your results should be similar to the ones show in the screen shot bellow.



Step 18. The tutorial is now finished.