Keyboard Characteristics

Especially for international applications, it is important for an application to find out what kind of keyboard is attached. For some applications, keyboard input in multiple languages, such as Japanese and English, is necessary. In that case, the program must be able determine which keyboard is attached.

Two Windows API functions, GetKeyboardLayoutName() and GetKeyboardType(), can be used.

- GetKeyboardLayoutName() returns a 8digit number which represents the keyboard's language specific layout, which can be looked up in the Language Identifier Table. The US English keyboard has the number **00001033** (hexadecimal **0x409**), as an example.
- GetKeyboardType() returns more detailed information about the keyboard, depending on the parameter passed to it. For common types of keyboards, you can find out the manufacturer, the number of regular keys supported, and the number of function keys supported. Thus, if your application allowed the user to press the **F12** function key, you could find out whether the keyboard in fact has this key.

The Code

Below is the code, in the CLICK event of a button, that will return these keyboard characteristics.

```
declare function GetKeyboardLayoutName, GetKeyboardType
kbLayoutName = Str(' ',9): char(0)

rv = GetKeyboardLayoutName( kbLayoutName)
call msg(@window , 'Keyboard layout identifier is ': kbLayoutName[1,8])

kbType = 0
rv = GetKeyboardType( kbType)
begin case
  case rv = 1
    call msg(@window, "Keyboard type: IBM PC/XT or compatible (83-key) keyboard")
  case rv = 2
    call msg(@window, "Keyboard type: Olivetti 'ICO' (102-key) keyboard")
  case rv = 3
    call msg(@window, "Keyboard type: IBM PC/AT (84-key) or similar keyboard")
  case rv = 4
    call msg(@window, "Keyboard type: IBM enhanced (101- or 102-key) keyboard")
  case rv = 5
    call msg(@window, "Keyboard type: Nokia 1050 and similar keyboards")
  case rv = 6
    call msg(@window, "Keyboard type: Nokia 9140 and similar keyboards")
  case rv = 7
    call msg(@window, "Keyboard type: Japanese keyboard")
  case 1
    call msg(@window, "Keyboard type: Unknown")
end case
kbType = 2
call msg(@window, 'This keyboard supports ': GetKeyboardType (kbType) : ' function keys.')
```

The call to GetKeyboardLayoutName() returns the language identifier of the keyboard in **kbLayoutName**.

The programs calls GetKeyboardType() twice. The first call, passing **0**, returns the numeric value of the keyboard type. The second call, passing **2**, returns the number of function keys supported.

The Windows API Declaration

The code above will not run until the declaration for GetKeyboardLayoutName() and GetKeyboardType() have been added. To add the declarations, do the following:

1. Log out of the application.
2. Log into the **SYSPROG** application.
3. Add a row, (call it **DLL_APICALLS_USER32**), with the first line as **USER32** and containing the declarations as shown below.

```
USER32
LONG  STDCALL GetKeyboardLayoutNameA( LPCHAR) As GetKeyboardLayoutName
LONG  STDCALL GetKeyboardType( LONG)
//....add any other declarations in USER32 here.....
```

4. Save the row.
5. Run Declare_FCNS at the System Editor Exec Line to create the declaration header, as shown below:

```
RUN DECLARE_FCNS 'DLL_APICALLS_USER32'
```

6. Exit the editor.
7. Log out of **SYSPROG**.
8. Log into your application.
9. Run the window.