

Projektowanie baz danych - Ukryte mechanizmy kontroli

Każda profesjonalna aplikacja powinna zawierać wiele ukrytych mechanizmów kontroli.

Do podstawowych mechanizmów kontroli można zaliczyć:

- Rejestr sesji badający, kto, w jakich godzinach pracował z aplikacją i czy aplikacja została zamknięta prawidłowo;
- Rejestr uruchamianych procedur modyfikujących dane;
- Rejestr błędów generowanych przez aplikację;
- Rejestr edycji danych;
- Rejestr wprowadzanych zmian danych;
- Rejestr uruchamianych wydruków;
- Rejestr czasu trwania złożonych procedur.

Mechanizmy kontroli są szczególnie przydatne:

- Do obsługi reklamacji związanych z wydajnością systemu;
- Do optymalizacji pracy aplikacji;
- Do badania przyczyn błędnych danych;
- Do analizy przyczyn błędów aplikacji.

Przykładowa uniwersalna procedura rejestruje rekord w tabeli SYS_Rejestr_zmian.

Procedura może być uruchamiana w różnych formularzach.

```
Sub SYS_Rejestr_zmian(formularz As String, pole As String, _
stara_wartosc As Variant, nowa_wartosc As Variant)
Dim zr As DAO.Recordset

Set zr = CurrentDb.OpenRecordset("SYS_Rejestr_zmian", _
dbOpenDynaset, dbAppendOnly)

zr.AddNew
  zr!formularz = formularz
  zr!pole = pole
  zr!stara_wartosc = Left(CStr(Nz(stara_wartosc)), 255)
  zr!nowa_wartosc = Left(CStr(Nz(nowa_wartosc)), 255)
  zr!czas_zmiany = Now()
  zr!komputer = "xxx"
  zr!uzytkownik = "xxx"
zr.Update

End Sub
```

Niżej zamieszczono przykład zastosowania procedury do rejestracji zmian nazwy klienta w formularzu KLI_Klienci_rejestracja.

```
Private Sub Form_BeforeUpdate(Cancel As Integer)

If (Me.nazwa_klienta <> Me.nazwa_klienta.OldValue) And Not NewRecord Then
  SYS_Rejestr_zmian Me.Name, "nazwa_klienta", _
  Me.nazwa_klienta.OldValue, Me.nazwa_klienta
End If

End Sub
```

Do sprawdzenia zmiany danych wykorzystano właściwość formantu **OldValue** (stara wartość).

Jak budować i stosować poszczególne mechanizmy kontroli, można dowiedzieć się z kursu PSP.