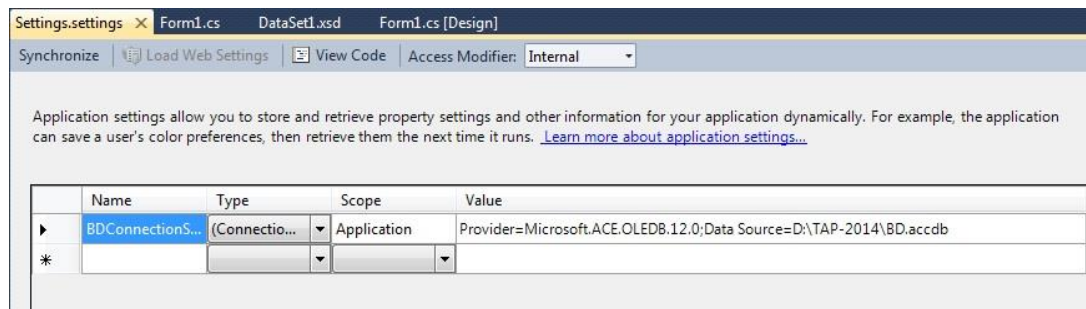


## TEHNICI DE PROGRAMARE AVANSATA – Laborator 4

- Schimbarea conexiunii la baza de date
- Realizarea unei legaturi 1-n intre doua multimi din cadrul unui form
- Protejarea unei baze de date Access prin parola
- Accesarea tabelor dintr-o baza de date Access protejata prin parola (cu obiect reader)
- Accesarea tabelor dintr-o baza de date Access protejata prin parola (cu obiect tableAdapter)

### 1. Schimbarea conexiunii la baza de date

- Deschidem gruparea *Properties* a aplicatiei (din Solution Explorer)
- Dublu Click pe optiunea *Settings.settings*



- Schimbam calea corespunzator noii baze de date
- Inchidem tab-ul Settings.settings

**Tema 1:** sa se testeze aceasta facilitate utilizand doua baze de date, avand o tabela dar cu continut diferit si comutand pe fiecare din ele. Sa se verifice schimbarea conexiunii la nivelul dataSet-ului. Se va utiliza o aplicatie care lucreaza cu un form de vizualizare a unei tabele.

### 2. Implementarea vizualizarii unei legaturi 1-n intre doua tabele

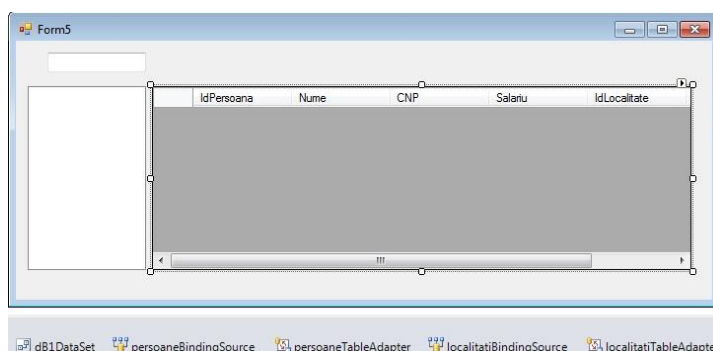
Obs. Vom folosi doua tabele din baza de date intre care avem o asociere many to one.

Persoane(IdPersoana, Nume, CNP, Salariu, IdLocalitate)

Localitati (IdLocalitate, DLocalitate)

Consideram deasemenea ca avem definite componentele Persoane si Localitati de tip DataTable in DataSet-ul aplicatiei.

Pas 1. Se creaza un form nou; se plaseaza un control de tip ListBox; un control de tip DataGridView si un control de tip textBox cu numele txtIdLocalitate.



Grid-ul nu va permite actualizari, ListBox-ul va afisa denumirile localitatilor iar txtIdLocalitate va fi legat de campul IdLocalitate din LocalitatiBindingSource.

Pas 2. Se creaza metoda *filtreazaLocalitati()*, ce seteaza proprietatea *filter* a obiectului *persoaneBindingSource*.

```
private void filtreazaLocalitati() {  
    try{  
        persoaneBindingSource.Filter = "IdLocalitate=" + txtIdLocalitate.Text;  
    }  
    catch { }  
}
```

Pas 3. Se creaza procedura eveniment: *localitatiBindingSource\_PositionChanged*, astfel:  
fiind in designer-ul form-ului, selectam obiectul *localitatiBindingSource* (facem click pe acel control) si alegem din fereastra de proprietati din grupul *Events*, evenimentul *PositionChanged*. Facem dublu-click pe acest eveniment.

Se scrie in aceasta metoda urmatoarea linie de cod:

```
filtreazaLocalitati();
```

Pas 4. Scriem aceiasi instructiune si in metoda eveniment *Form\_Load*, la sfarsitul ei.

Obs. Metoda prezentata se bazeaza pe filtrarea dinamica a componentei *PersoaneBindingSource* prin generarea unei conditii (la momentul executiei) in care *IdLocalitate* de la persoane sa fie egal cu *IdLocalitate* din pozitia curenta a lui *LocalitatiBindingSource*.

In felul acesta sincronizam liniile din grid cu linia curenta din *ListBox*.

**\*Tema 2** Sa se implementeze doua legaturi *many to one* intre 3 tabele. De exemplu Judete => Localitati => Persoane

### 3. Protejarea unei baze de date Access prin parola

Consideram o baza de date MS Access neprotejata prin parola.

Vom pune o parola de acces la aceasta.

Lansam in executie MS Access 2010. Solicitam Open din optiunea File din meniul principal.

Selectam baza de date la care vrem sa punem parola de acces.

Selectam din comboBox-ul Open (din partea din dreapta jos a ferestrei) optiunea Open Exclusive.

Selectam optiunea File din meniul principal.

Selectam optiunea Encrypt with password din fereastra din dreapta.

Tastam parola dorita atat in textBox-ul Password cat si in Verify.

Apasam butonul OK.

Obs. Pt. eliminarea parolei procedam similar.

### 4. Accesarea tabelor dintr-o baza de date Access protejata prin parola (cu obiect reader)

Vom crea o noua aplicatie.

Vom testa modul de accesare prin program a unei tabele din baza de date protejata prin parola.

Vom plasa pe form un obiect *ListBox*.

Vom utiliza tabela Persoane din baza de date.

Generam procedura eveniment *Form1\_Load* si o completam cu codul:

```
private void Form1_Load(object sender, EventArgs e)  
{  
    con.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0;" +  
        "Data Source=D:\\TAP-2017\\DB1.accdb;" +
```

```

        "Jet OLEDB:Database Password=xxxx";
cmd.Connection = con;
cmd.CommandText = "Select Nume from Persoane";
con.Open();
rdr = cmd.ExecuteReader();
while (rdr.Read())
    listBox1.Items.Add(rdr.GetString(0));
con.Close();
}

```

Testam aplicatia. ListBox-ul se va incarca cu numele persoanelor din baza de date.

## 5. Accesarea tabelor dintr-o baza de date Access protejata prin parola (cu obiect tableAdapter)

Pas 1 Se adauga aplicatiei o componenta DataSet.

Pas 2 Se solicita crearea unui table adapter:

Click dr (DataSet Designer) => Add = Table adapter

Choose Your Data Connection => New Connection => Browse Database file name

Completam Password => OK => Next

Copy data file in the project => No

Sensitive data => Yes

Save the connection string in the config.file => Yes

Next => etc

Pas 3. Se plaseaza pe form un DataGridView si se conecteaza la sursa de date

Pas 4. Se testeaza aplicatia.