

## Curs 6. CREAREA ȘI LUCRUL CU FORMULARE ȘI RAPOARTE

### 1. Crearea și lucrul cu formulare

### 2. Crearea și lucrul cu rapoarte

### 1. Crearea și lucrul cu formulare

- formularele oferă o interfață prietenoasă și elegantă prin care programatorul poate controla acțiunile utilizatorului
- formularele au peste 70 de proprietăți, la care se adaugă cele ale controalelor, între 38 și 58 fiecare
- principalul rol este introducerea datelor, regăsirea datelor, afișarea meniurilor, etc.
- sursele de înregistrări ale formularelor indică de unde provin sau unde vor fi stocate datele și sunt specificate printre proprietățile formularelor

Clasificare:

\* după sursa de date:

- formulare legate (bound) permit actualizarea și afișarea datelor din tabele
- formulare nelegate (unbound) pentru afișarea sau editarea unor date care nu sunt în tabele

\* după modul de afișare

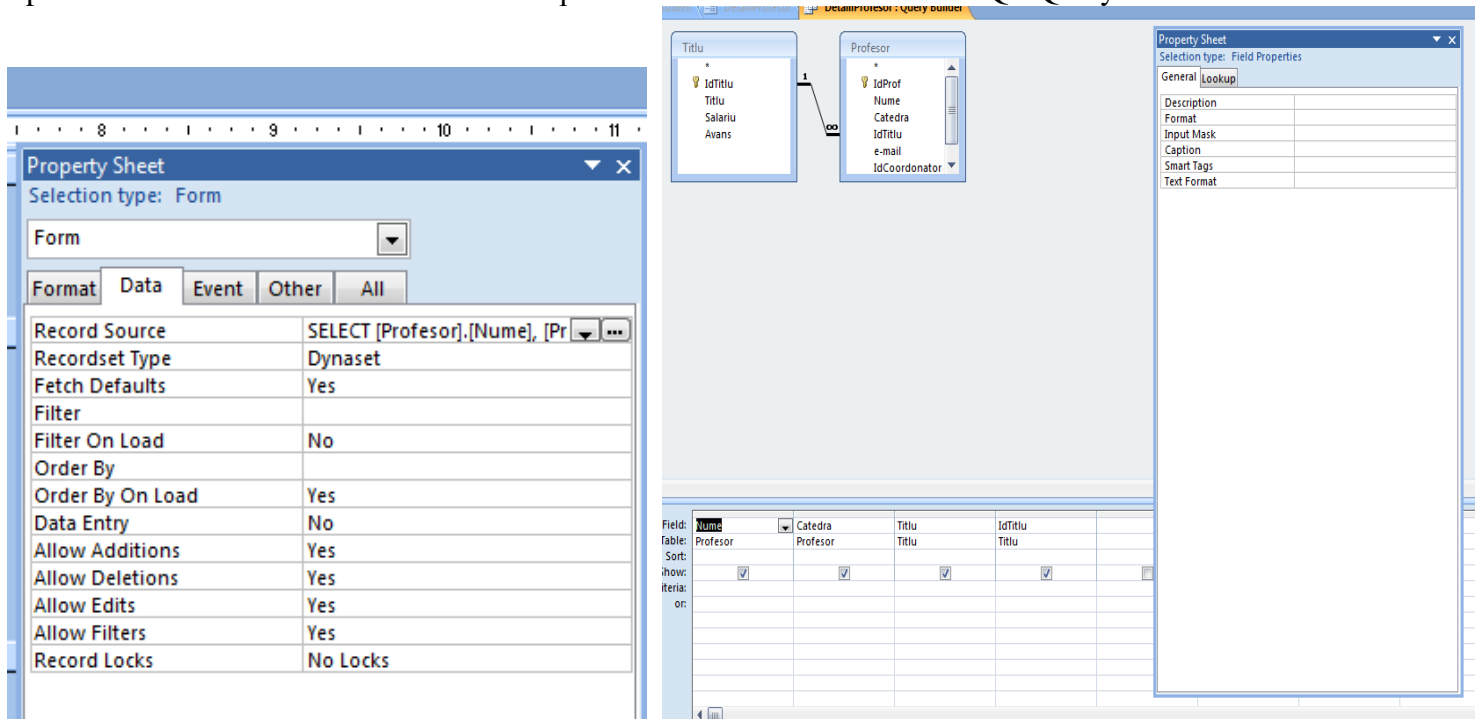
- single form: afișează o sg. înregistrare la un moment dat
- tabular form: permit vizualizarea concomitentă a mai multor înregistrări
- datasheet form: afișare asemănătoare foilor de calcul
- chart form: afișarea datelor sub forma graficelor

\* după modul de interacțiune cu alte ferestre:

- formulare modale : nu permit accesarea obiectelor conținute în alte ferestre până când nu sunt închise sau ascunse
- formulare nemodale: permit activarea/deschiderea altor ferestre

### 1.1. Sursele de înregistrări ale formularelor

- sursa poate fi o tabelă, interogare sau o instrucțiune SQL, specificată în proprietatea *Record Source*
- pentru a vedea pagina de proprietăți a unui formular deschis în *Design View*, se efectuează clic pe butonul selector al formularului, apoi clic dreapta *Properties*
- săgeata derulantă în dreapta proprietății *Record Source* deschide o listă cu tabelele și interogările din baza de date curentă
- pentru a modifica sursa de date se dă clic pe ... care va deschide fereastra SQL Query Builder



- dacă formularul se bazează pe o interogare, modificările aduse acestuia în fereastra SQL Builder se vor reflecta și în interogarea propriu-zisă
- numai interogările pentru regăsirea datelor de tip SELECT pot servi drept sursă de înregistrări pentru un formular
- formularele bazate pe interogări salvate sunt mult mai rapide dar aceste interogări sporesc numărul de obiecte din b.d. făcând-o să ocupe mai mult spațiu

## 1.2. Crearea unui formular simplu

- *Form Wizard*

- *Design View*

Secțiunile unui formular (meniul *View*)

**Detail:** conține datele din fiecare înregistrare a sursei de date; pentru a trece de la o înregistrare la alta se folosesc butoanele de navigare ale formularului

**Form Header:** partea de sus, include informații care nu se schimbă în funcție de fiecare înregistrare, cum ar fi un antet sau un control ce nu are legătură cu înregistrarea curentă, casetă combinată pentru filtrarea înregistrărilor

**Form Footer:** include informații despre data și ora creării, totaluri, funcții de agregare, etc

**Page Header:** formularele care conțin un număr mare de controale pot fi împărțite cu ajutorul unui control de tip *Page Break* în două sau mai multe pagini, pentru a fi afișate pe rând. Secțiunile *Page Header* și *Page Footer* vor rămâne nemodificate când se trece de la o pagină la alta a formularului

**Page Footer:** asemănătoare *Page Header*

- un formular odată creat poate fi deschis în 3 feluri: din obiectul *Forms*, din alt formular cu ajutorul unui buton, dintr-un macro sau cu ajutorul lb. VBA

- indiferent de cum a fost deschis un formular se poate face trecerea la oricare din cele trei moduri în care poate fi văzut formularul: *Design View* (pentru a lucra cu controalele), *DataSheet View* (pentru a avea acces numai la datele pe care se bazează formularul) sau *Form View* (pt a vedea formularul în forma sa finală).

- navigarea între controale și înregistrări se face cu ajutorul tastei *Tab*; schimbarea ordinii de navigare se face din meniul *View*, opțiunea *TabOrder*

- navigarea între înregistrări se face cu ajutorul butoanelor de navigare; aceste butoane apar în colțul din stânga jos al unui formular dacă în pagina de proprietăți a acestuia proprietatea *Navigation Buttons* are valoarea *Yes*.

- în pagina de proprietăți a formularului întâlnim 6 proprietăți ce ne permit controlul acțiunilor utilizatorului asupra datelor pe care se bazează formularul:

Allow Edits	Dacă are valoarea <i>Yes</i> , utilizatorul va putea modifica înregistrările, dacă are valoarea <i>No</i> , nu va putea schimba datele afișate în formular
Allow Deletions	Dacă are valoarea <i>Yes</i> , utilizatorul va putea șterge înregistrări, cu condiția ca să fie păstrată integritatea referențială
Allow Additions	Dacă are valoarea <i>Yes</i> , utilizatorul va putea adăuga înregistrări
Data Entry	Dacă are valoarea <i>Yes</i> , formularul deschide automat o înregistrare nouă, fără date. Utilizatorul nu are acces la înregistrările existente. Dacă proprietatea <i>Allow Additions</i> are valoarea <i>NO</i> iar <i>Data Entry</i> are <i>Yes</i> , va rezulta o eroare
Recordset Type	Poate fi <i>DynaSet</i> , <i>DynaSet (Inconsistent Updates)</i> sau <i>Snapshot</i> . Se referă la formularele ce au mai multe tabele ca sursă de date. Dacă valoarea este <i>dynaset</i> , datele din controalele asociate câmpurilor vor putea fi editate de către utilizator, ceea ce nu este valabil în cazul valorii <i>Snapshot</i> .
Record Locks	Se referă la aplicații multiuser. Dacă are valoarea <i>No Locks</i> , mai mulți utilizatori vor putea modifica aceleași înregistrări în același timp, fiind luate în considerare numai modificările ce au fost salvate primele. Opțiunea <i>Edited Records</i> împiedică pe ceilalți utilizatori de a opera schimbări asupra unei înregistrări care este deja în curs de modificare. Opțiunea <i>All Records</i> blochează toate înregistrările din formular și din tabelele de bază atâta timp cât un utilizator editează orice înregistrare din tabelă.

Proprietățile din fila *Format* controlează aspectul.

*Caption:* textul afișat în bara de titlu.

*Default View:* stabilește dacă formularul va fi afișat în modul *Single Form*, *Continuous Form* sau *Datasheet*.

*Views Allowed:* stabilește dacă utilizatorul poate trece din modul *Form View* în modul *DataSheet View*.

*ScrollBars*: stabilește dacă formularul are bare de derulare verticale și orizontale

*RecordSelectors*: Dacă valoarea sa este Yes, pe latura din stânga a secțiunii Detail va apare un selector pt.înregistrări

*Navigation Buttons*

*Record Dividers*: Dacă valoarea este Yes, în modul Continuos va apare o linie de separare între înregistrări

*AutoSize*: Dacă valoarea sa este Yes, fereastra formularului va fi dimensionată corespunzător la fiecare deschidere a acestuia.

*AutoCenter*: Dacă valoarea este Yes, la deschidere formularul va apare centrat pe ecran

*ControlBox*: Dacă valoarea este Yes, caseta de control va apare în partea din stânga a barei de titlu

*MaxMinButtons*: Precizează dacă butoanele Minimize și Maximize vor fi incluse în bara de titlu. În acest caz valoarea proprietății *WhatsThisButton* trebuie să fie No.

*CloseButton*: Dacă valoarea este Yes, butonul X pt închiderea formularului va fi inclus în partea din dreapta a barei de titlu

*WhatsThisButton*: Dacă valoarea este Yes, butonul ? va apare în partea din dreapta a barei de titlu în cazul în care și *CloseButton* are valoarea Yes.

Două proprietăți importante din fila *Other*:

*PopUp*: Dacă valoarea este Yes, formularul va fi afișat deasupra altor ferestre, putând fi mutat și înafara ferestrei

*Access*

*Modal*: dacă valoarea este Yes, alte ferestre deschise nu pot primi controlul până când formularul nu este închis.

### **Controale pe formulare**

- Design View

- View , Toolbox

**Controale nelegate ( unbound )** : folosite pentru a furniza sau primi de la utilizator informații ce nu sunt sau nu vor fi stocate în b.d. Exemple: etichete ce descriu alt control, butoane ce deschid formulare, fișiere de tip bitmap etc.

**Controale legate**: pentru a afișa sau edita informații din baza de date, le corespunde câte un câmp dintr-o tabelă sau un câmp returnat dintr-o interogare sau instrucțiune SQL. Orice control poate fi legat cu excepția dreptelor, dreptunghiurilor, sfârșitului de pagină , etichetelor și cadrelor pentru imagini.

- un control atașat va moșteni multe din proprietățile câmpului corespunzător ( *Caption, Description, Input Mask, Format* )
- pentru a plasa un control legat se alege meniul View, opțiunea *Field List*
- pentru a transforma un control nelegat în unul legat se folosește proprietatea *Control Source* în care se specifică câmpul corespunzător

**Controale calculate**: folosesc expresii pentru a obține datele; cele mai utilizate sunt casetele text; pentru expresii mai complicate se folosește *Expression Builder*.

**Control Wizard**: utilitar pentru lucrul cu controale de tip casetă combinată, listă și grup de opțiuni.

### **1.3. Subformulare**

- se folosesc atunci când între două tabele există o relație de tip 1:m. Ex. Tabelele Profesor și Titlu sunt legate și dacă dorim să aflăm informații despre titlul de lector, să vedem și toți profesorii care au acest titlu.

- putem avea maxim două nivele de imbricare a subformularelor și oricâte subformulare pe același nivel de imbricare.

*Exemplu*: Formular care arată toți profesorii care au un anumit titlu

- se creează formularul cu Form Wizard

- se alege tabela *Profesor* , câmpurile *Nume, Catedra, Data Angajării*

- se alege tabela Titlu, câmpurile *Titlu și Salariu*

Form Wizard

Which fields do you want on your form?  
You can choose from more than one table or query.

Tables/Queries  
Table: Titlu

Available Fields:  
IdTitlu  
Avans

Selected Fields:  
Nume  
Catedra  
DataAngajarii  
Titlu  
Salariu

Cancel < Back Next > Finish

- dacă se alege ca datele să fie prezentate în funcție de tabela Profesor, formularul va fi de tip Single Form, adică datele vor fi afișate într-un singur formular iar dacă se alege ca datele să fie prezentate în funcție de tabela Titlu, putem opta între două posibilități: Linked Forms (formulare legate, vor fi create două formulare separate) și Form with Subforms (formular cu subformulare)

Form Wizard

How do you want to view your data?

by Titlu  
by Profesor

Nume, Catedra, DataAngajarii, Titlu, Salariu

Single form Linked forms

Form Wizard

How do you want to view your data?

by Titlu  
by Profesor

Titlu, Salariu  
Nume, Catedra, DataAngajarii

Form with subform(s) Linked forms

Form Wizard

Titlu

Titlu  
Salariu

Profesor

Nume	Catedra	DataAn
Popescu Marin	Matematica	3/12/20
Mihu Dana	Informatica	3/12/20
Radulescu E	Limbii moderne	3/12/19

Record: 1 of 3 No Filter Search

- În Design apare un nou control de tip Subform/Subreport; legătura între cele două formulare se face prin intermediul a două proprietăți ale controlului de tip Subform/Subreport: *Link Child Fields* și *Link Master Fields* care specifică numele coloanelor de legătură ce formează cheia străină respectiv a coloanei ce formează cheia primară.
- Proprietatea *Source Object* are ca valoare numele subformularului ( Profesor Subform )

Selection type: Subform/Subreport

Profesor Subform1

Format Data Event Other All

Name	Profesor Subform1
Visible	Yes
Source Object	Profesor Subform1
Link Master Fields	IdTitlu
Link Child Fields	IdTitlu
Width	4.1667"
Height	1.3333"
Top	1"

Form

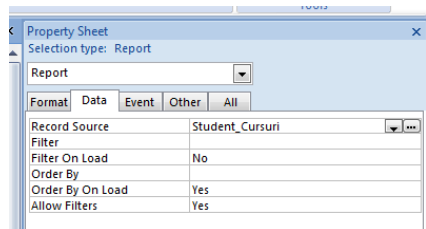
Format Data Event Other All

Record Source	Profesor
Caption	Profesor Subform1
Pop Up	No
Modal	No
Display on SharePoint Site	Follow Table Setting
Default View	Datasheet
Allow Form View	Yes
Allow Datasheet View	Yes
Allow PivotTable View	Yes
Allow PivotChart View	Yes

- proprietatea *Record Source* are ca valoare tabela *Profesor*, din care provin datele subformularului.
- un subformular poate fi adăugat unui formular și în modul Design View, se trage subformularul în formular, apoi se stabilesc proprietățile *Link Child Fields* și *Link Master Fields*.
- Altă metodă cu ajutorul controlului de tip Subform/Subreport.

## 2.Crearea și lucrul cu rapoarte

- Rapoartele reprezintă o metodă prin care datele pot fi prezentate într-un mod intuitiv și plăcut. Ele pot fi vizualizate sau exportate într-un alt format.
- Nu pot fi folosite pentru introducerea datelor.
- Pot conține aproape toate elementele unui formular, cu excepția controalelor care presupun acțiuni ale utilizatorului, cum sunt cele de tip *Combo Box*
- formularele se pot converti în rapoarte și invers



- datele unui raport pot proveni dintr-o tabelă, o interogare sau o instrucțiune SQL. Setul de înregistrări nu este un dynaset ci un snapshot.

### 2.1. Crearea unui raport simplu

- Report Wizard
- Design View
- sursa de date este specificată de proprietatea *Record Source*
- pentru ca informațiile stocate într-un câmp al sursei de date să poată fi afișate în raport, trebuie ca în grila QBE proprietatea Show a câmpului respectiv să fie activată
- dacă interogarea pe care se bazează raportul are parametri, pentru a putea vizualiza sau tipări raportul, utilizatorul trebuie să furnizeze valorile paramterilor respectivi.

#### Secțiunile raportului:

**Detail:** se repetă pentru fiecare înregistrare la vizualizarea sau la tipărirea raportului.

**Report Header:** apare numai pe prima pagină, ca un titlu pentru întregul raport. Dacă dorim să creăm o pagină separată numai pentru titlul raportului, alegem ca valoare a proprietății *ForceNewPage* a acestei secțiuni opțiunea *AfterSection*.

**Report Footer:** apare numai pe ultima pagină a raportului și poate conține informații despre data și ora creării, totaluri, funcții de agregare, etc

**Page Header:** apare în partea de sus a fiecărei pagini, cu excepția primeia, unde apare după antetul raportului. Cel mai adesea conține titlurile câmpurilor afișate.

**Page Footer:** apare în partea de jos a fiecărei pagini a raportului. Cel mai adesea conține data și ora și eventual numărul paginii.

Ex. Now ()

= "Page" & [Page] & " of " & [Pages]

- În plus, față de aceste secțiuni pe care le-am întâlnit și la crearea formularelor, un raport poate avea un număr de antete și subsoluri de grup, care să afișeze titlurile câmpurilor ce definesc grupul respectiv și totaluri în cadrul grupului.

### 2.2. Proprietățile unui raport

*Record Source*

*Filter:* numele unei interogări sau al unei instrucțiuni SQL care limitează numărul de înregistrări din tabela de bază ce vor fi afișate în raport

*Filter On:* dacă are valoarea Yes, filtrul definit de proprietatea Filter se va aplica

*Order By:* un câmp sau o listă de câmpuri separate prin virgule, din sursa de date a raportului în funcție de care va fi stabilită ordinea în care vor fi afișate datele

*Order By On:* dacă are valoarea Yes, se va face sortarea stabilită de proprietatea *Order By*.

*Caption*

*Record Locks*: dacă are valoarea All Records alți utilizatori nu vor putea modifica datele din tabele de bază ale raportului atâta timp cât acesta este vizualizat sau tipărit

*Page Header* și *Page Footer*: stabilește dacă antetul și subsolul de pagină va fi tipărit și pe paginile în care apare antetul și subsolul raportului

*Picture*: pentru fundal

*Picture Type*: determină dacă imaginea specificată de proprietatea Picture va fi salvată în baza de date sau va fi păstrată într-un fișier separat.

*Picture Size Mode*: determină în ce mod imaginea specificată de proprietatea Picture va fi micșorată în cazul în care nu încapă în pagină

*Picture Alignment*

*Picture Tiling*: pentru repetaRE pe toată pagina

*Picture Pages*: determină dacă imaginea specificată va fi tipărită pe toate paginile, numai pe prima sau pe niciuna.

*Layout for Print*: dacă are valoarea Yes, la proiectarea raportului nu se pot folosi decât fonturile native ale imprimantei și cele True type.

## 2.3. Sortarea și Gruparea datelor

*Group Header*: crează o secțiune în antet

*Group Footer*: creează o secțiune în subsol unde se pot include controale care să afișeze totaluri numai pentru grupul respectiv

*Group On*: dacă valoarea este *Each Value*, Access va considera ca făcând parte din același grup doar înregistrările pentru care valorile câmpului respectiv sunt egale. Access va crea, pentru fiecare valoare a câmpului un grup care va avea antet și subsol.

Dacă valoarea este Interval, grupurile se pot extinde la toate înregistrările pentru care valoarea câmpului respectiv se află într-un interval specificat.

*Group Interval*: dacă valoarea anterioară este alta decât *Each Value*, proprietatea *Group Interval* definește intervalul în care se poate afla valoarea câmpului respectiv pentru a face parte din grup.

*Keep Togheter*: dacă are valoarea NO, Access va trece la o pagină nouă numai atunci când pagina curentă este plină. Dacă are valoarea *Whole Group* access va face tot posibilul ca antetul, subsolul și secțiunea Detail să se afle pe aceeași pagină. Dacă are valoarea *With First Detail*, Access va trece la o pagină nouă dacă antetul și cel puțin prima înregistrare de la secțiunea Detail a grupului nu încap pe pagina curentă.

### Folosirea funcțiilor agregat

- în subsolul formularului
- casete de text pentru calcule
- dacă se folosesc funcții agregat în rapoarte, access restrânge numărul de înregistrări pentru care se face agregarea numai la grupul curent
- dacă este inclusă o funcție agregat în cadrul antetului sau subsolului unui grup, vor fi luate în considerare numai înregistrările care fac parte din acel grup.
- dacă este inclusă o funcție agregat în cadrul secțiunii *Page Header* sau *Page Footer*, funcția va lua în considerare numai înregistrările din pagina respectiv, iar dacă este inclusă la secțiunea *Report Header* sau *Report Footer*, funcția va calcula totalul pentru toate înregistrările raportului.

### Editarea rapoartelor

- în modul *Design View*
- se modifică *Caption*
- alinierea textului în coloane cu proprietatea *Text Align*
- folosirea a unui număr fix de variabile: *Format*, *Decimal Places*
- folosirea proprietății Font Weight pentru scriere boldată
- numerotarea automată a liniilor: se introduce o nouă casetă de text, proprietatea *Control Source*=1, pentru incrementare se folosește proprietatea *Running Sum* care are valoarea *Over All*; dacă se renumerează fiecare grupă atunci *Running Sum*=*Over Group*
- proprietatea *Can Grow*=Yes, pentru afișarea în întregime a textului în control