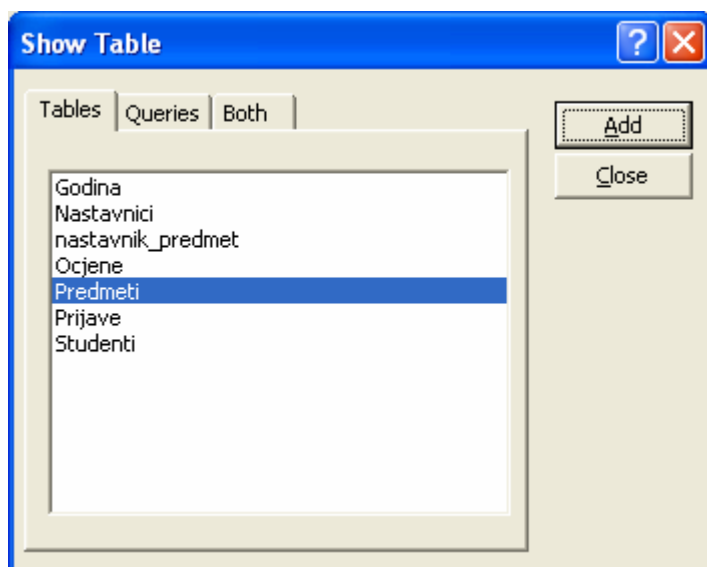


Formiranje relacija

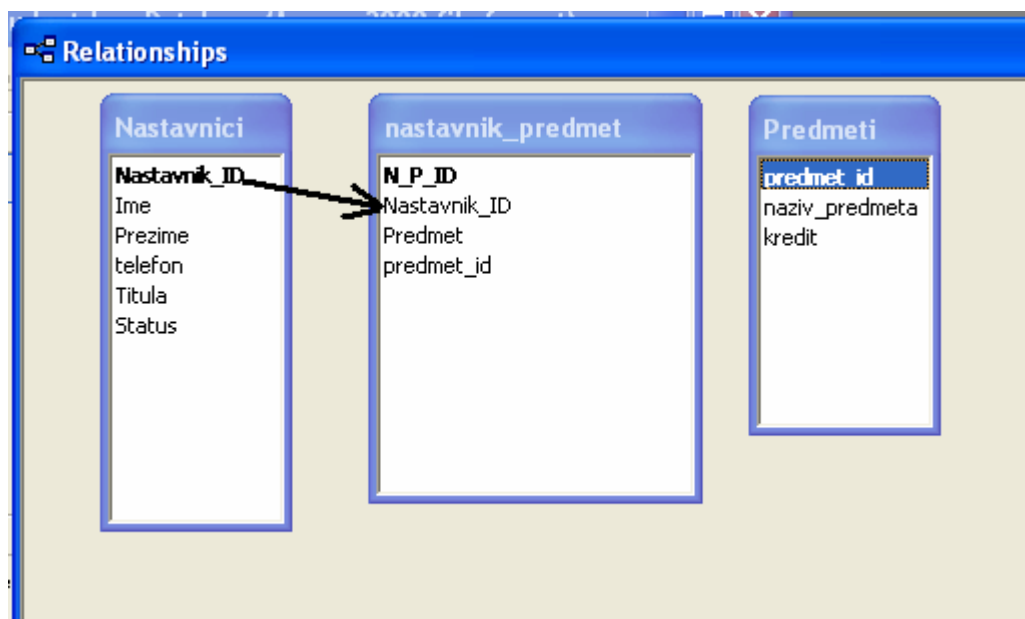
Najprije treba otvoriti prozor za formiranje relacija. To možemo učiniti klikom na dugme Relationships ili iz menija Tools->Relationships.

U otvorenom prozoru Relationships, kliknućemo desnim dugmetom miša i u padajućem meniju izabrati Show table. Dobićemo dijalog kao na slici.

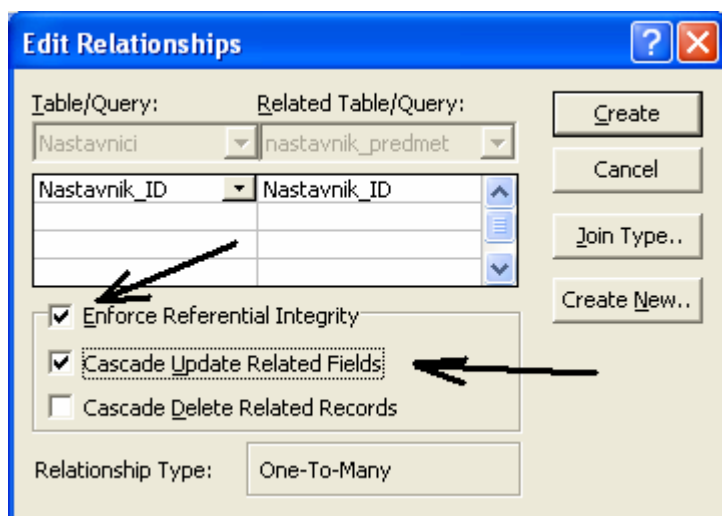


U datom dijalogu izabraćemo tabele koje će biti vezane relacijom. Označićemo tabelu Nastavnici, pa kliknuti na dugme Add. To ćemo ponoviti za tabele nastavnik_predmet i Predmeti. Zatim ćemo zatvoriti dijalog pomoću dugmeta Close.

U prozoru Relationships, relacije formiramo pomoću miša, jednostavnom operacijom dovlačenja i puštanja. Dakle, pritisnemo desno dugme miša iznad polja **Nastavnik_Id** u tabeli **Nastavnici** i ne puštajući dugme povučemo pokazivač miša do polja **Nastavnik_Id** u tabeli **nastavnik_predmet**.



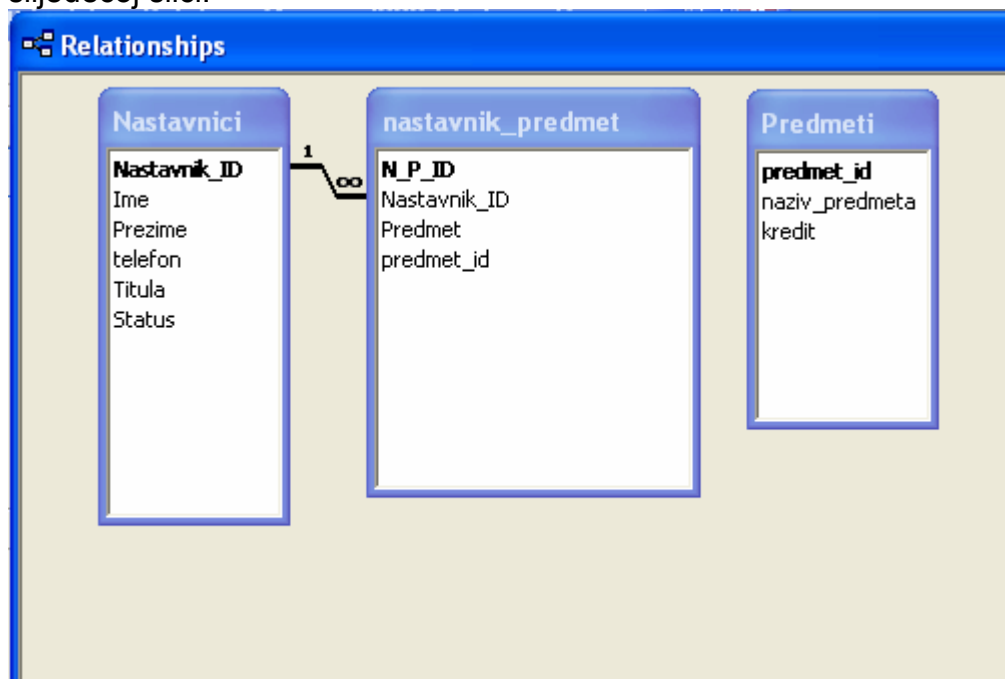
Access će nam ponuditi da napravimo relaciju pomoću dijaloga **Edit relationships**, kao na slici:



Access je već prepoznao da relaciju kreiramo pomoću polja **Nastavnik_Id** u obje tabele, te da je to relacija tipa jedan-prema-više (**One-To-Many**).

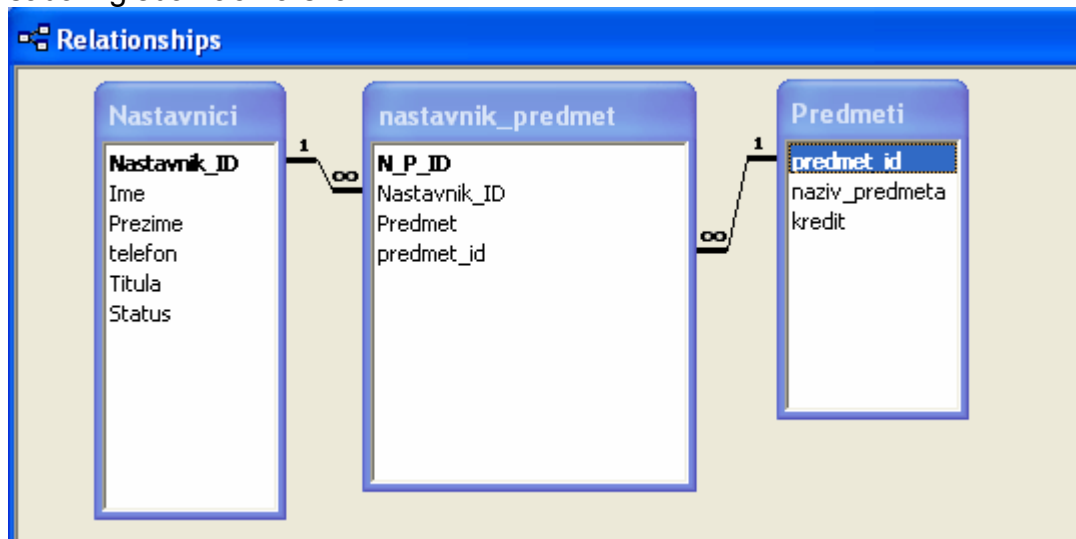
Naše je još i da mu kažemo i da želimo da primjeni vezani integritet podataka (**Enforce referential integrity**) i da samostalno popunjava vezana polja (**Cascade update related fields**). Značenje oba pojma biće razjašnjeno u nastavku teksta.

Nakon što kliknemo dugme **Create**, prozor relationships će izgledati kao na slijedećoj slici:

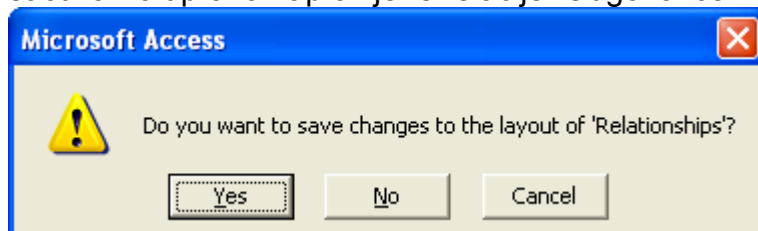


Dakle, pojavila se linija koja spaja tabele **Nastavnici** i **nastavnik_predmet** po poljima **Nastavnik_Id** i koja pokazuje da se radi o relaciji tipa **jedan-prema-više**.

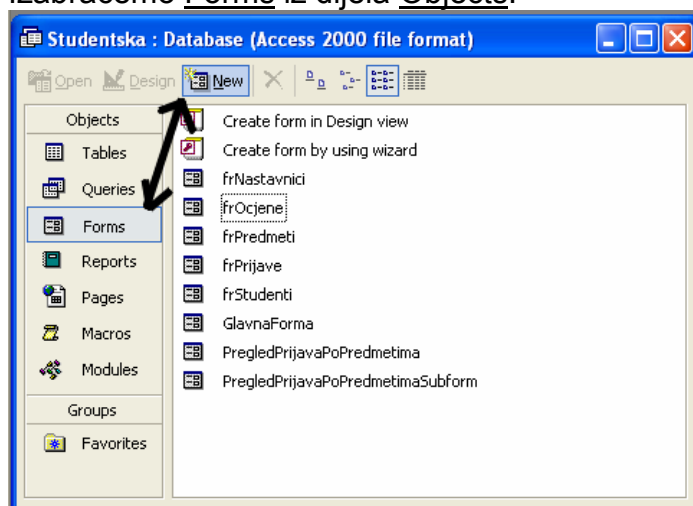
Isti postupak treba ponoviti i za polje **predmet_id** tako što ćemo ga povući iz tabele **Predmeti** i pustiti u tabeli **nastavnik_predmet**. Prozor Relationships sada izgleda kao na slic



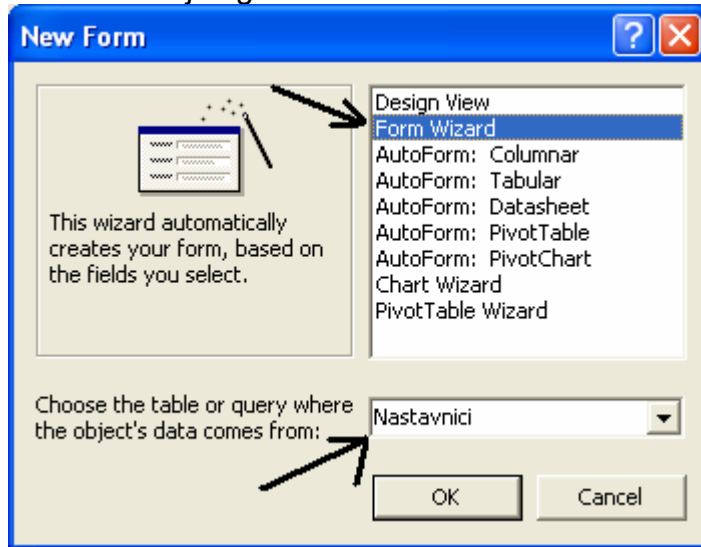
Sada ćemo zatvoriti prozor Relationships. Access će nas pitati želimo li da sačuvamo upravo napravljene relacije. Odgovorićemo da hoćemo.



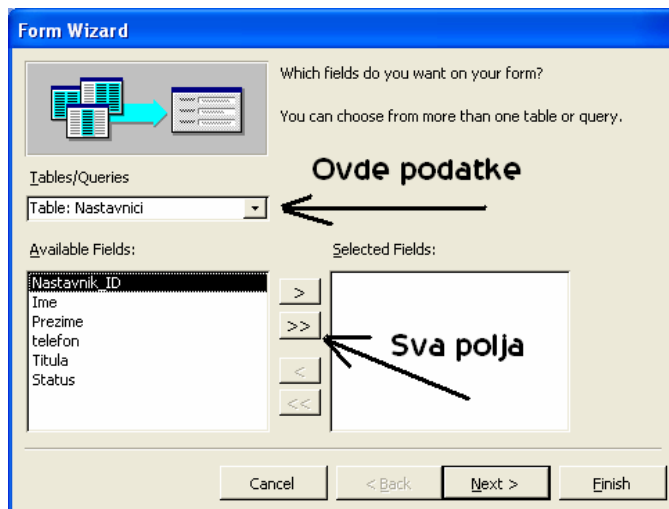
Da bi mogli da koristimo naše relacije, napravićemo jednu novu formu. Izabraćemo Forms iz dijela Objects.



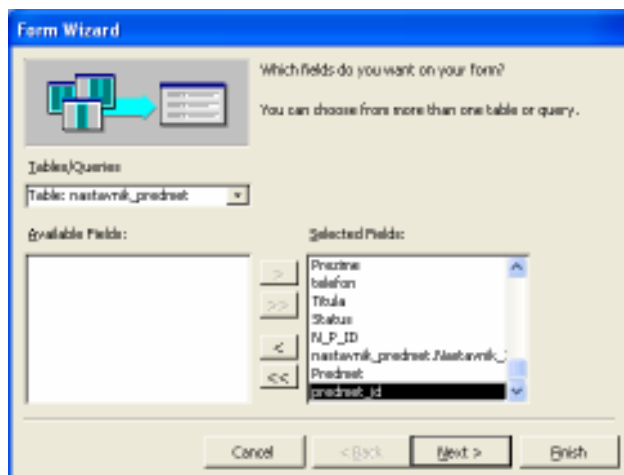
Dobićemo dijalog kao na slici:



Dobićemo dijalog koji nas pita na koji način ćemo praviti formu i odakle dolaze podaci. Izabraćemo Form Wizard, da formu napravimo pomoću čarobnjaka. Podatke ćemo uzeti iz tabele Nastavnici. Kliknućemo dugme OK.



U slijedećem dijalogu, Access nas pita koja polja želimo da prikazemo na formi. Da bi objasnili značenje vezanog integriteta podataka (**Enforce referential integrity**), odabraćemo sva polja.



Zatim ćemo iz padajućeg kombo-boxa izabrati tabelu **nastavnik_predmet** i takođe odabrati sva polja

Postupak ćemo ponoviti i za tabelu **Predmeti** (takođe odabrati sva polja).

Nakon što pritisnemo dugme Next, dobićemo dijalog kao na slici:

Form Wizard

How do you want to view your data?

by **Nastavnici**
by nastavnik_predmet
by Predmeti

Nastavnici_Nastavnik_ID, Ime, Prezime, telefon, Titula, Status

N_P_ID, nastavnik_predmet_Nastavnik_ID, Predmet, nastavnik_predmet_predmet_id, Predmeti_predmet_id, naziv_predmeta, kredit

Form with subform(s) Linked forms

Cancel < Back Next > Finish

Ovdje nas Access pita na koji način želimo da povežemo podatke koji će biti prikazani na formi. Odabraćemo da osnovna tabela bude **Nastavnici** (**by Nastavnici**) i da vezani podaci budu prikazani pomoću podforme (**Form with subforms**). Kliknućemo Next.

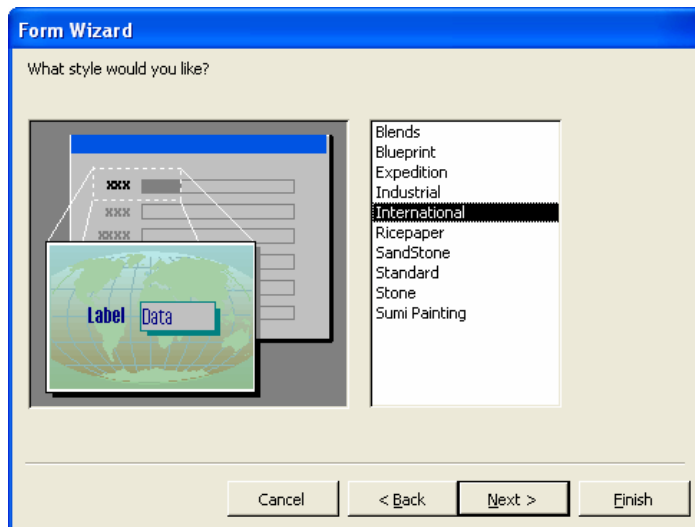
Form Wizard

What layout would you like for your subform?

Tabular Datasheet PivotTable PivotChart

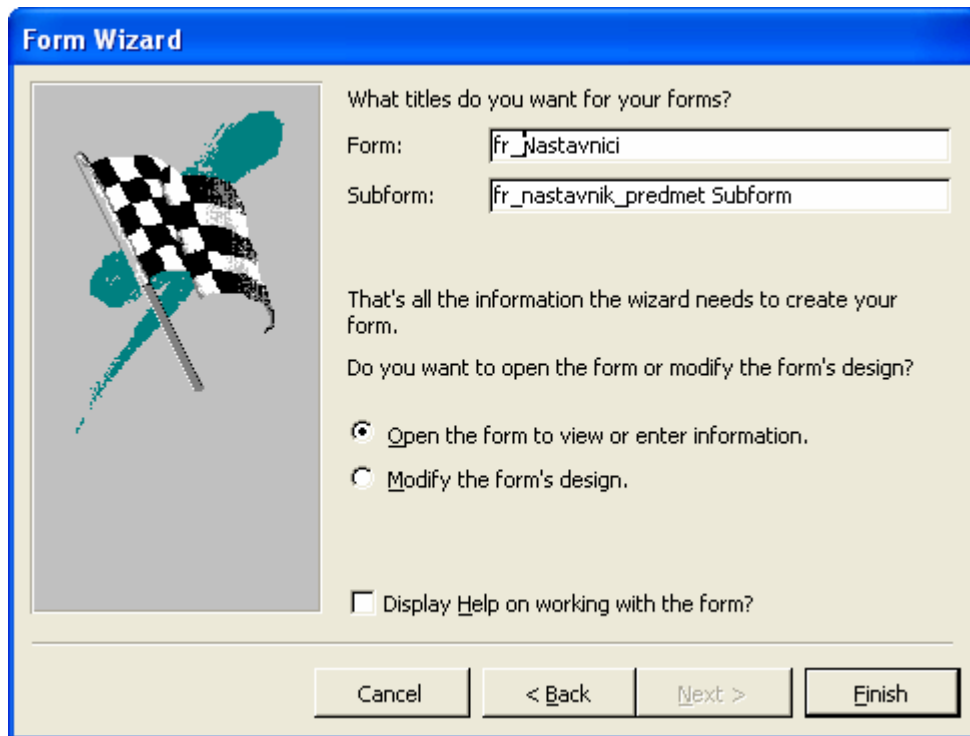
Cancel < Back Next > Finish

Access nas sada pita kako želimo da nam prikaže odabrane podatke. Odabraćemo **Datasheet**, kao na slijedećoj slici:



Nakon što kliknemo na dugme Next, Access će nas pitati i za izgled forme (boja pozadine, sjene itd.). Odabraćemo International.

Zadnji dijalog ovog čarobnjaka pitaće nas da damo imena našim novim formama. Da bi se kasnije lakše snalazili, pošto se radi o formama, ubacićemo slova **fr_** pred ponuđeni naziv.

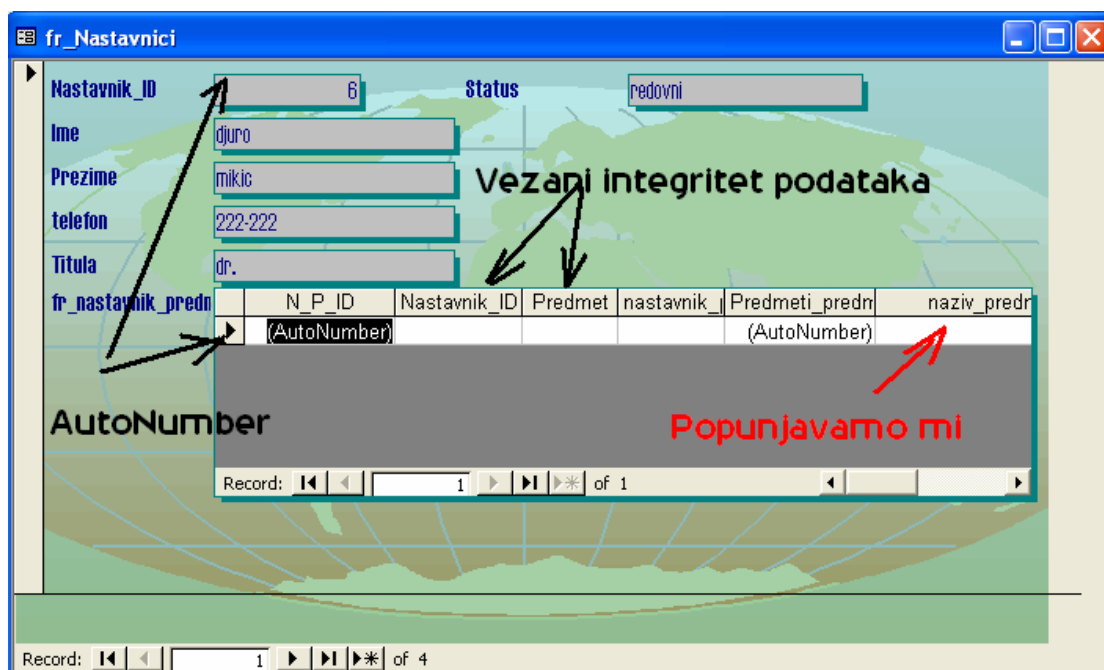


Pritisnućemo dugme Finish. Access će nam napraviti dvje forme - **fr_Nastavnici** i **fr_nastavnik_predmet Subform**, te otvoriti formu fr_Nastavnici.

U nju možemo unositi osnovne podatke o profesorima, ali i koji predmet oni predaju.

Na strani 2, tokom formiranja relacija, govorili smo o primjeni vezanog integriteta podataka (**Enforce referential integrity**). Sada je vrijeme da taj pojam razjasnimo.

Vezani integritet znači da će Access sam pronaći i popuniti potrebna polja u našoj formi. Naš zadatak je, u ovom slučaju, samo da mu upišemo naziv predmeta, a on će sam popuniti ostala polja.



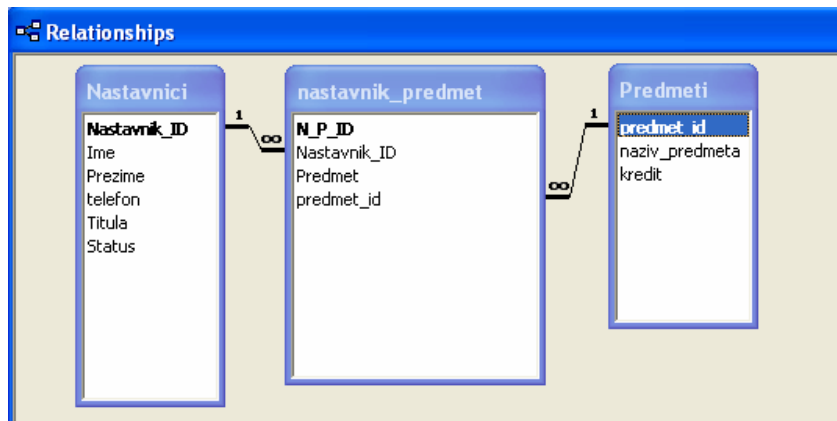
Kako Access može da popuni ostala polja?

AutoNumber polja Access popunjava samostalno, nevezano za referential integrity.

Čim smo unijeli ime i prezime nastavnika, Access je sam odredio vrijednost polja **Nastavnik_Id**, jer je ono tipa **AutoNumber**. Međutim, ovo još nije referential integrity.

Čim smo naziv predmeta, Access je sam odredio vrijednost polja **Predmet_Id**, jer je ono tipa **AutoNumber**. Međutim, ni ovo još nije referential integrity.

Vezani integritet podataka (referential integrity) Access vadi iz zadane relacije, kao na slijedećoj slici:



Dakle, u tabeli **nastavnik_predmet** se nalaze polja koja Access treba da popuni na osnovu pravila koja smo mu zadali. Polje **Nastavnik_Id** će popuniti na osnovu istoimenog polja u tabeli **Nastavnici**, a polje **Predmet_Id** iz tabele **Predmeti**.

Npr. Ako u polje **naziv_predmeta** unesemo Statistika, pa otvorimo sve tri tabele u DataSheet pogledu, imaćemo ovakav raspored:

Nastavnici					
Nastavnik_ID	Ime	Prezime	telefon	Titula	Status
6	djuro	mikic	222-222	dr.	redovni

nastavnik_predmet		
N_P_ID	Nastavnik_ID	predmet_id
37	6	8

Predmeti		
predmet_id	naziv_predmeta	kredit
8	Statistika	30

Dakle, Access je sam popunio polja **Nastavnik_Id** u tabelama **Nastavnici** i **nastavnik_predmet**, i ona u obje tabele imaju vrijednost 6. Takođe je popunio i polja **predmet_id** u tabelama **predmeti** i **nastavnik_predmet** i postavio oba na vrijednost 8.

Ovi brojevi služe nam za povezivanje podataka u različitim tabelama. Npr. Interesuje nas koje predmete predaje Đuro Mikić. Kada ga Access nađe u tabeli **Nastavnici**, on će pogledati u pomoćnu tabelu **nastavnik_predmet** i vidjeti da je u pitanju **predmet_id** broj 8. Zatim će u tabeli **Predmeti** potražiti **predmet_Id** broj 8 i vidjeti da je naziv predmeta **Statistika**.

Još jedan primjer. U formi **fr_Nastavnici** unijet ćemo nastavnika Oliveru Jefrić, a zatim i predmete koje predaje – Engleski jezik I i Engleski Jezik II, kao na slici:

Ako otvorimo sve tri tabele u DataSheet pogledu, imaćemo ovakav raspored:

Nastavnici					
Nastavnik_ID	Ime	Prezime	telefon	Titula	Status
10	Olivera	Jefrić	333333	mr.	redovni

nastavnik_predmet		
N_P_ID	Nastavnik_ID	predmet_id
37	6	8
38	10	9
39	10	10

Predmeti		
predmet_id	naziv_predmeta	kredit
9	Engleski jezik I	30
10	Engleski jezik II	30

Id nastavnika Olivera Jefrić je 10. U pomoćnoj tabeli nastavnik_predmet, se dva puta nalazi Nastavnik_Id broj 10.

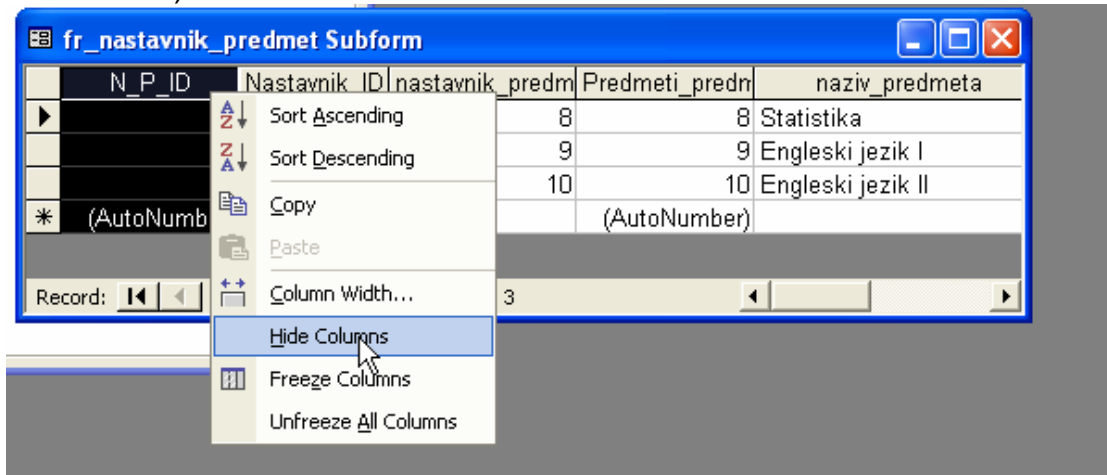
U jednom redu je predmet_id broj 9, pa iz tabele predmeta znamo da se radi o predmetu Engleski jezik I.

U drugom redu je predmet_id broj 10, pa iz tabele predmeta znamo da se radi o predmetu Engleski jezik II.

Pošto smo objasnili pojam vezanog integriteta podataka (**Enforce referential integrity**), trebamo ukloniti **ID** polja iz forme, da ne zbunjuju korisnika.

Otvorićemo formu `fr_nastavnik_predmet` Subform kao DataSheet View (dvoklikom).

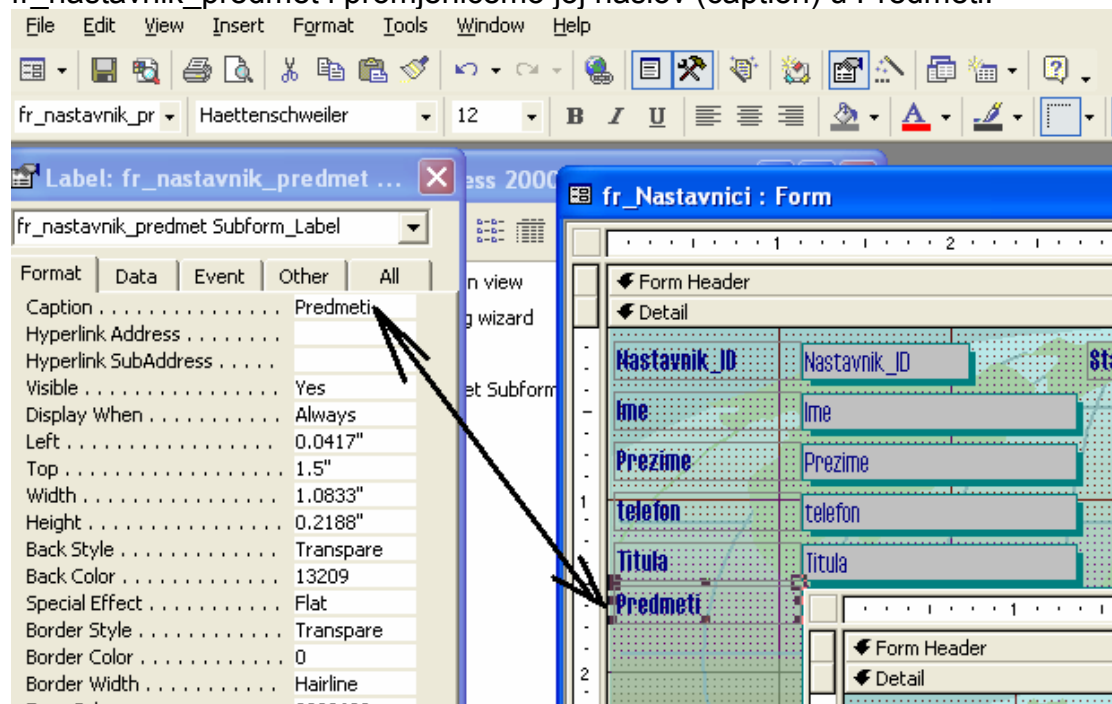
Kliknućemo desnim tasterom miša na vrh kolone `N_P_ID` (na polje gdje se nalazi naziv).



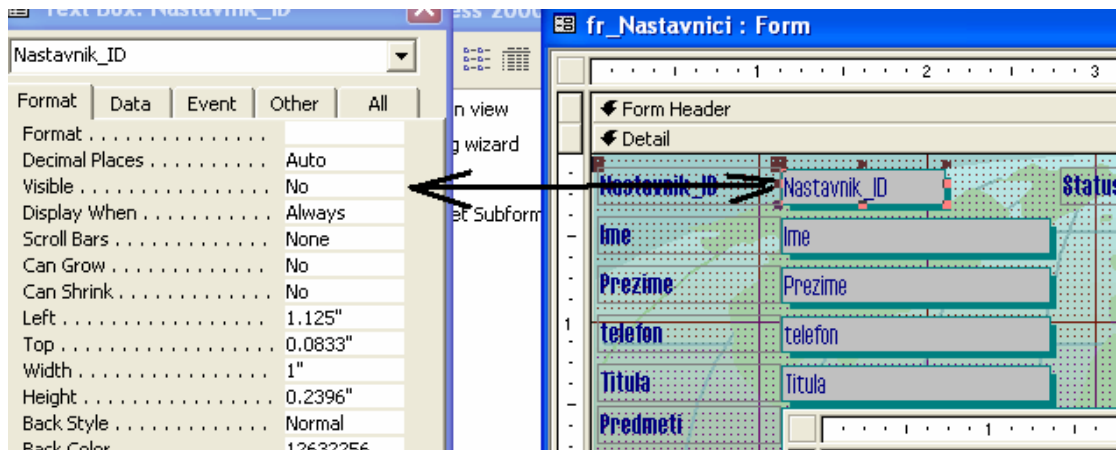
Iz padajućeg menija odabraćemo **Hide Columns (sakrij kolone)**. Nakon kolika, kolona je sakrivena. Postupak treba ponoviti i za ostale ID kolone.

Još nam je ostalo da promjenimo labelu `fr_nastavnik_predmet` i sakrijemo polje `Nastavnik_Id` u formi `fr_Nastavnici`.

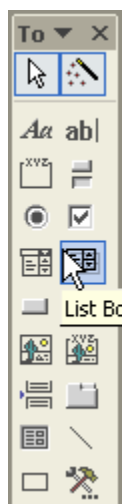
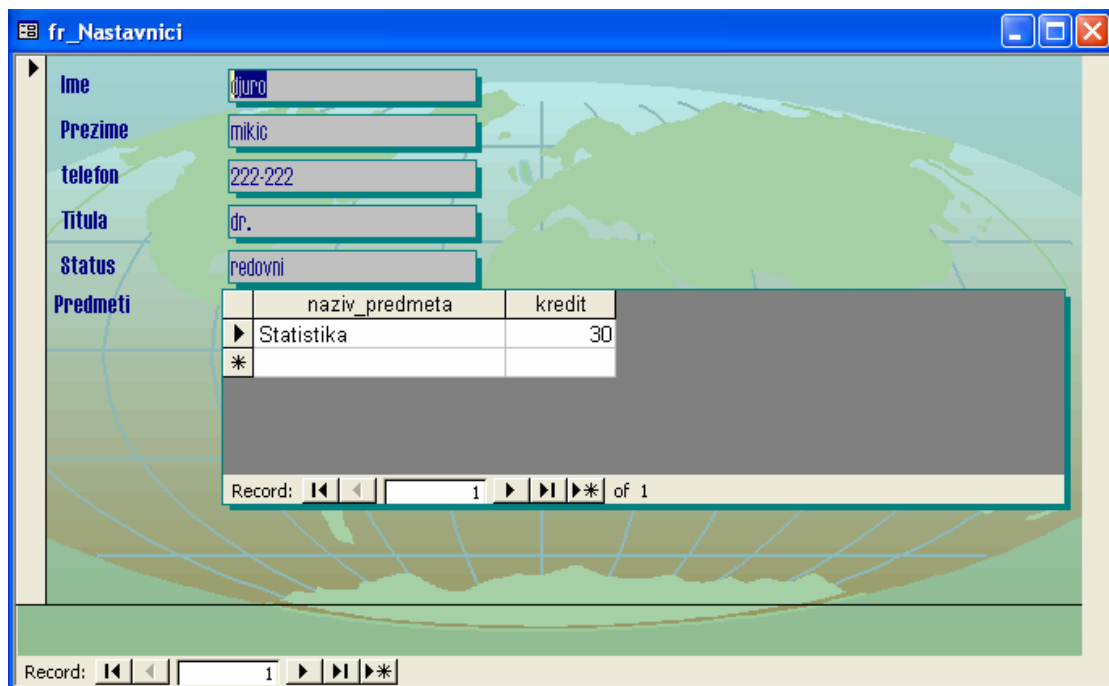
Otvorićemo formu **fr_Nastavnici** u dizajn režimu. Kliknućemo na labelu `fr_nastavnik_predmet` i promjenićemo joj naslov (caption) u `Predmeti`.



Zatim ćemo kliknuti na input-box `Nastavnik_Id` i podesiti osobinu **Visible** na No, nakon čega ovo polje više neće biti vidljivo u radnom režimu.



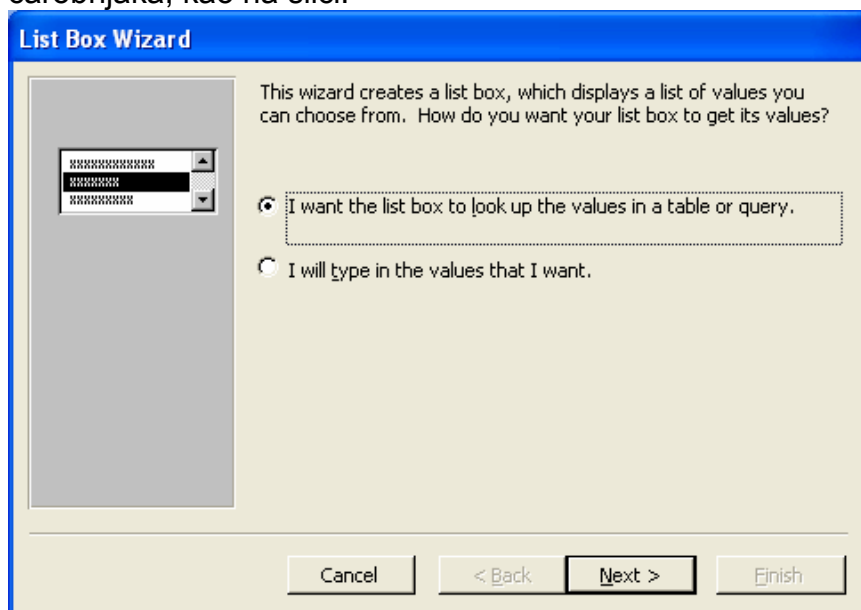
Možemo još rasporediti polja po formi radi ljepšeg izgleda, pa će naša forma izgledati kao na slici:



Sada smo dobili formu u koju možemo unijeti ime nastavnika i **nove** predmete koji on predaje. Međutim, šta ako neki predmet predaju dva nastavnika, kao profesor i asistent? Ako pokušamo unijeti postojeći predmet u oavkvu tabelu, Access nas neće razumjeti, nego će svaki put unijeti novi predmet. Da bi izbjegli dupliranje imena predmeta, moramo u podformu `fr_nastavnik_predmet` Subform ubaciti još jednu kontrolu.

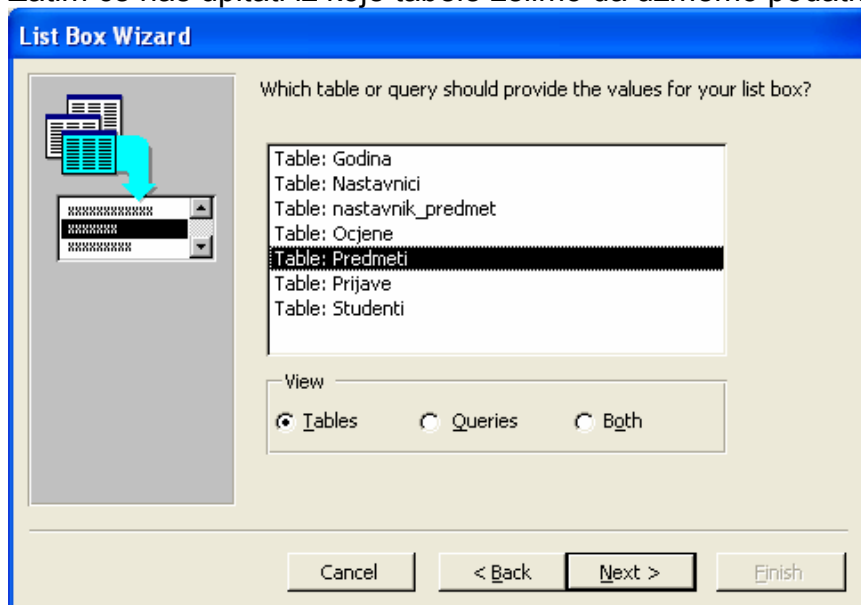
Otvorićemo podformu u dizajn režimu. U ToolBox-u (slika lijevo) odabraćemo ListBox i umetnuti u podformu.

Čim kliknemo mišom u tijelo podforme, Access će nam ponuditi pomoć u vidu čarobnjaka, kao na slici:



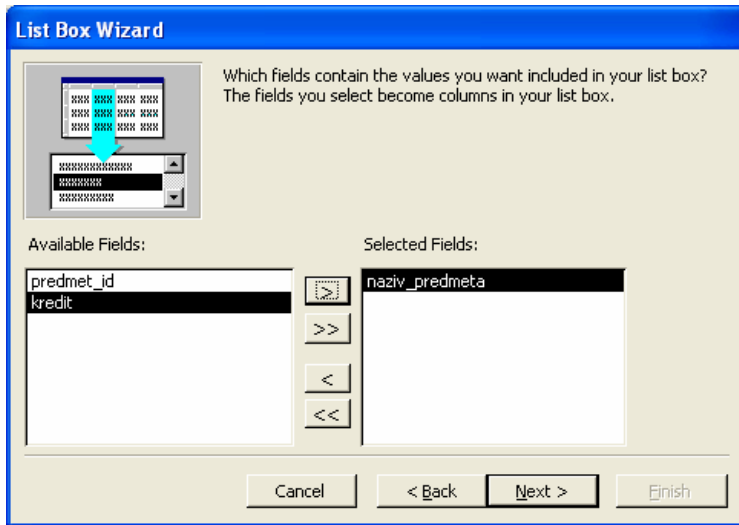
U ovom dijalogu, on nam nudi da ListBox sadrži vrijednosti iz Tabela i Upita ili da sami zadajemo vrijednosti. Odabraćemo prvu varijantu – iz tabela ili upita.

Zatim će nas upitati iz koje tabele želimo da uzmemo podatke.

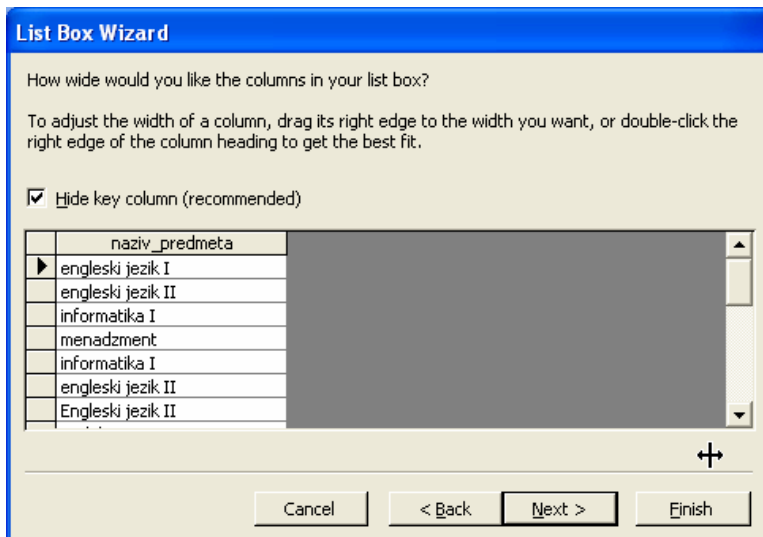


Odabraćemo tabelu Predmeti, i kliknuti Next.

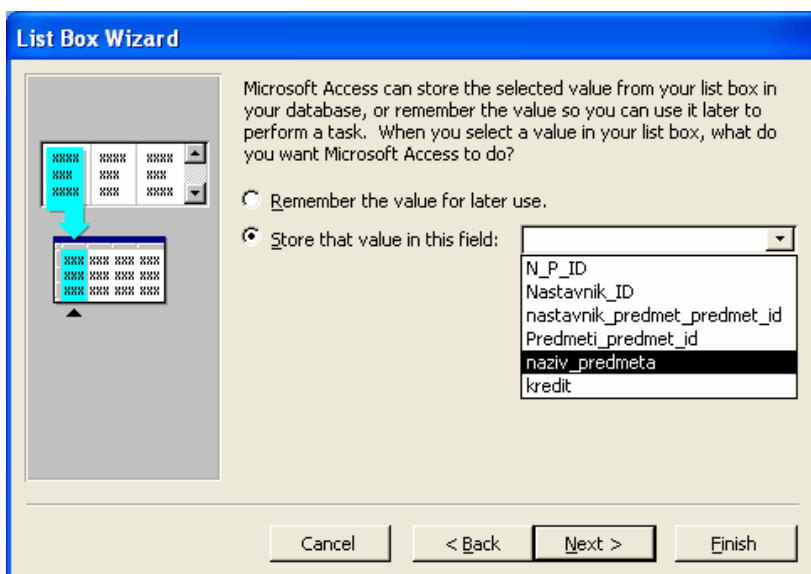
Access će nas pitati koju kolonu želimo da koristimo kao izvor podataka.



Odabraćemo kolonu **naziv_predmeta** i kliknuti next.

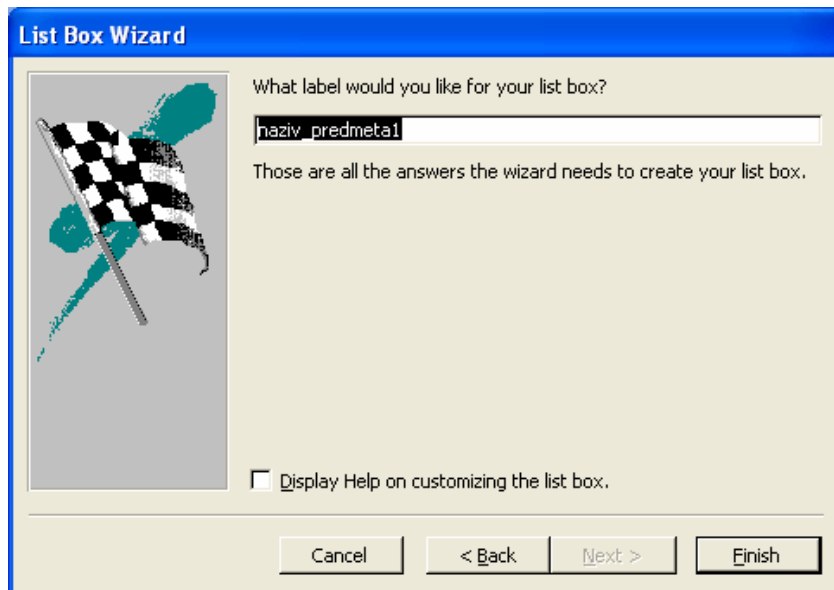


U sljedećem dijalogu, Access će nas pitati da li treba da sakrije ključno polje (gdje ćemo prihvatiti podrazumjevanu vrijednost), i pružiće nam priliku da podesimo širinu vidljivih kolona.

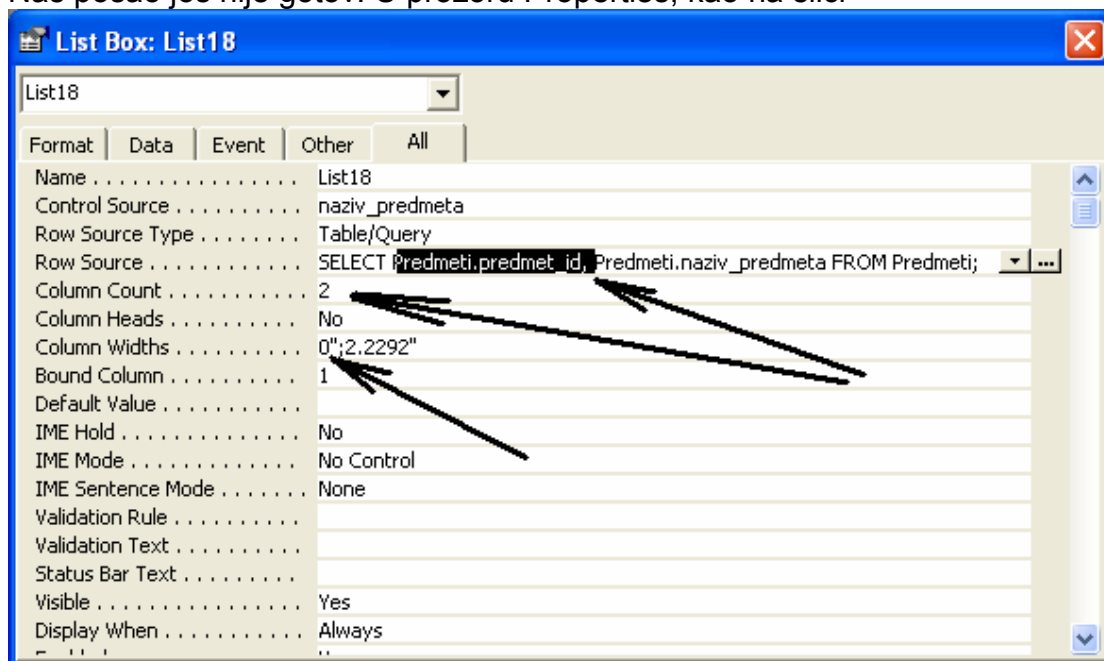


U sljedećem dijalogu, Access će nam ponuditi da odabrane vrijednosti upisuje u bazu ili da ih pamti. Odabraćemo da ih smjesti u bazu.

U zadnjem dijalogu, ponudićete nam da damo naslov labeli našeg novog list boxa. Prihvaćemo njegov izbor i kliknuti dugme Finish.



Naš posao još nije gotov. U prozoru Properties, kao na slici



treba promjeniti tri stvari.

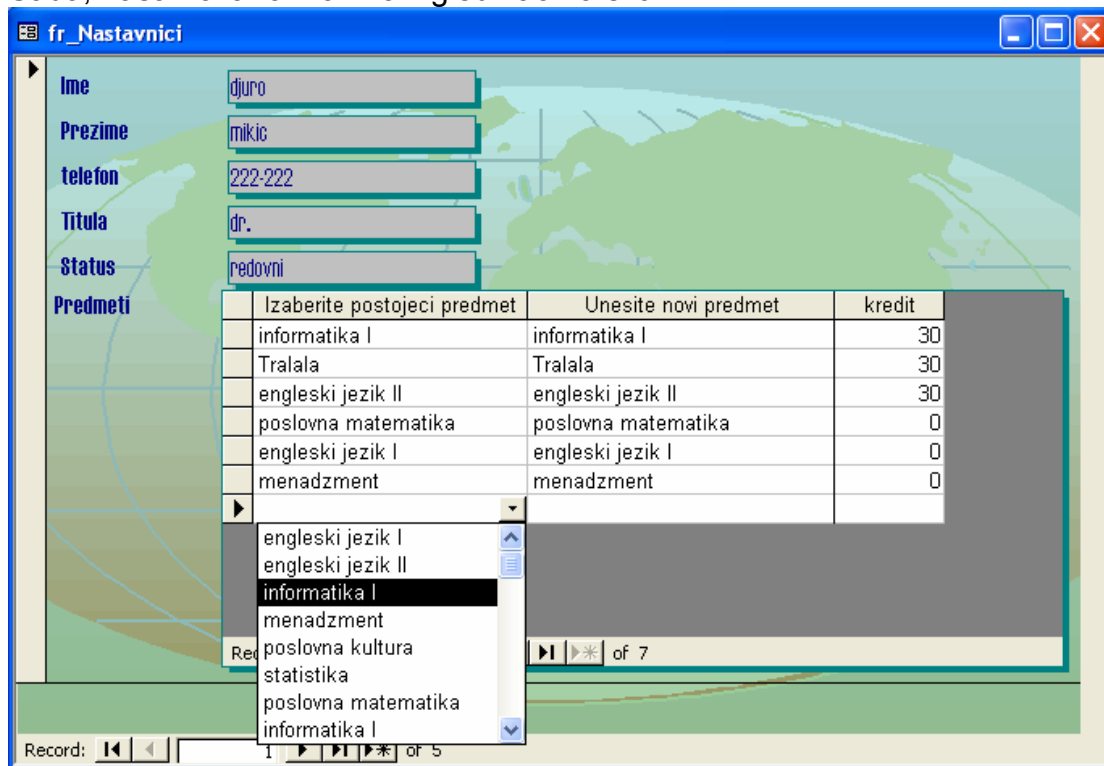
Iz linije **RowSource** u kojoj se nalazi text upita koji je formirao čarobnjak, trebamo obrisati dio **Predmeti.Predmet_Id**. Ovo radimo jer nam nije u ovom list boxu potrebno polje **predmet_id**.

U liniji **Column Count** trebamo Accessu reći da želimo samo jednu kolonu u našem list-boxu.

U liniji **Column Widths**, trebamo izbrisati prvi broj **0"**; koji predstavlja širinu prve kolone, koju smo prethodnim akcijama izbrisali.

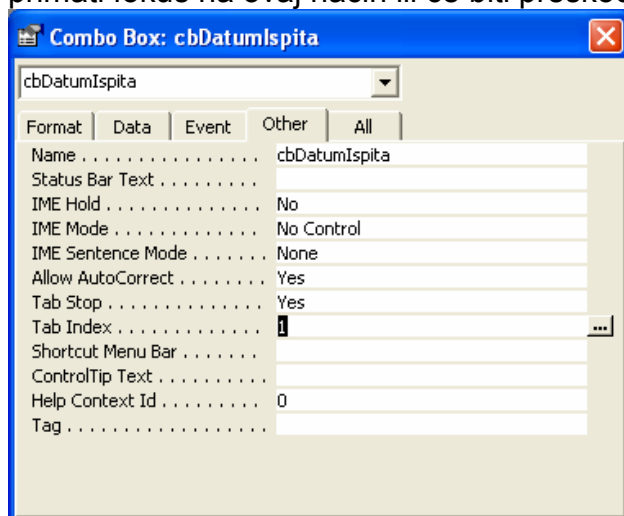
Nakon toga, potrebno je dotjerati labele (naslove) naših polja, tako da **list-box** ima naslov **Izaberite postojeći predmet**, a **naziv_predmeta** ima naslov **Unesite novi predmet**.

Sada, naša nova forma ima izgled kao na slici.



U polju **Izaberite postojeći predmet**, biramo postojeći a u polju **Unesite novi predmet**, unosimo novi.

Kretanje od kontrole do kontrole moguće je pomoću TABa ili ENTERa. Redoslijed kretanja određujemo u prozoru Properties u polju **Tab index** gdje brojanje počinjemo od nule. U polju Tab Stop određujemo da li će kontrola primiti fokus na ovaj način ili će biti preskočena.



Programiranje

U formi frPrijave_po_predmetima, pored standardnog pretraživanja, stavljamo i filter u list box u kom se nalaze datumi ispita i predmeti:

```
Sub traziPredmet()
```

```
Me.Naziv_Predmeta.SetFocus
```

```
'GoToControl cesto pravi probleme, za koje na internetu nema rjesenja
```

```
'DoCmd.GoToControl "Prezime1"
```

```
DoCmd.FindRecord Forms![frPrijave_po_predmetima]![cbTrazi], acEntire, , , ,  
acCurrent
```

```
'DoCmd.GoToControl "Broj_Indexa"
```

```
Me.cbDatumIspita.RowSource = "SELECT IspitniRokovi.Id_Ispitnog_roka, " _  
    & "IspitniRokovi.datum_ispita, Predmeti.naziv_predmeta "
```

```
—
```

```
    & "FROM Predmeti INNER JOIN IspitniRokovi ON " _  
    & " Predmeti.predmet_id=IspitniRokovi.predmet_id " _  
    & "where IspitniRokovi.predmet_id = " & _  
    Me.predmet_id & ";"
```

```
End Sub
```