

KURS ACCESS-a

Ovo je još nedovršena verzija - moguće su izmene

(Baza koja ide uz ovo uputstvo nalazi se u direktorijumu Ort pod nazivom EvidencijaIspita).

Pršić Dragan

MS Access je moćan i intuitivan program firme Microsoft namenjen radu sa bazama podataka. Kao sastavni deo paketa MS Office ovaj sistem za upravljanje bazama podataka (DBMS - *database management system*) upotpunjuje mozaik poslovnih aplikacija namenjenih manjim i srednjim preduzećima.

Zbog lakoće korišćenja a istovremeno velikih mogućnosti ovaj program je popularan i među početnicima i među programerima. Naime, za kreiranje jednostavnih aplikacija ne zahteva se veliko poznavanje programa - po principu za jedno veće. Sa druge strane, da bi ste osetili svu snagu Access-a (saradnja sa drugim programima Office paketa, korišćenje API funkcija Windowsa, rad sa SQL serverima u većim mrežama, Internet, ...) biće Vam potrebno i veće znanje i iskustvo.

Naravno, ovo uputstvo je namenjeno početnicima, a svi ostali mogu pogledati dodatnu literaturu. Za praćenje kursa potrebno je poznavanje rada u operativnom sistemu Windows odnosno poznavanje osnovnih aktivnosti kao što su: kreiranje, brisanje, uređivanje fascikli i dokumenata, korišćenje sistema za pomoć, pretraživanje, podešavanje radnog okruženja, itd.

Baza podataka

Baza podataka je skup informacija vezanih za određenu temu ili namenu. To je skladište opšte namene za smeštaj informacija bilo koje vrste. Npr. pomoću nje možemo pratiti stanje robe na skladištu, finansijske tokove u nekom preduzeću, postignute rezultate na sportskim takmičenjima, evidentirati dolaske/odlaske na posao, položene ispite studenata... Praktično od Vas i Vaše mašte zavisi mogućnost korišćenja baza podataka. Baza podataka se ne mora voditi na računaru, ako uđete u neku arhivu ili kartoteku koja ne koristi računar, videćete primere takvih baza. Međutim, vođenje baze na računaru pruža mogućnost lakše obrade podataka koji se čuvaju. Zamislite da tražite podatke o položenim ispitima u periodu 1950-1960. god. u nekoj klasičnoj arhivi i u elektronskoj bazi.

Pretpostavimo da želimo da vodimo evidenciju o ispitima na fakultetu. Neka se za svaki ispit, radi evidencije, popunjava obrazac poput onog na slici Sl.1

Posmatrajući obrazac možemo saznati opšte podatke o ispitu (*ispitni rok, datum polaganja, predmet*) kao i zasebne podatke o svakom studentu pojedinačno (*broj indeksa, prezime, ime, ocena*). Sve ove podatke obrazac povezuje u jednu smislenu celinu i daje nam *informaciju* o datom ispitu. Ako skupimo obrasce svih ispita u jednu kartoteku dobićemo bazu podataka.

Kako da sada ovu bazu podataka prebacimo na računar - formiramo računarsku bazu podataka? Jedno rešenje bi moglo biti da svaki obrazac skeniramo i sačuvamo ga na disku računara. Kada nam trebaju podaci o nekom ispitu onda moramo naći sliku odgovarajućeg obrasca i eventualno ga odštampati. Sve što nam treba u ovom slučaju je skener i neki program za pregled slika, ali ne i sistem za rad sa bazama podataka - DBMS.

Šta možemo da uradimo sa sačuvanim informacijama? Kod ovakvog načina čuvanja podataka, malo toga. Jedino možemo da nađemo sliku obrasca i da ga pogledamo ili odštampamo. Međutim, pronalaženje neke slike u fascikli sa hiljadu slika može biti zamorno i sporo. Dalje, ako nam je potrebna informacija o broju studenata koji su položili ispit, prosečnoj oceni, broju položenih ispita u nekom roku, broju studenata koji su prijavili/izašli na ispit, o prolaznosti iz nekog predmeta itd. do svih ovih informacija možemo doći ako pogledamo jedan po jedan obrazac a onda ručno obradimo podatke.

Postavlja se pitanje da li računar može uraditi taj posao? Za sada, računar može samo da mehanički, velikom brzinom, uradi poslove za koje je isprogramiran ali ne i da obavlja kreativne poslove. On ne može da iz skeniranog obrasca izvuče korisne podatke kao što je *ispitni rok, datum polaganja, predmet...* - taj posao moramo *MI* da uradimo. Dakle ako želimo da nam računar i obrađuje a ne samo čuva informacije, moramo svaku tu informaciju razbiti na elementarne podatke (*ispitni rok, datum polaganja, predmet, broj indeksa, prezime, ime, ocena*) i tek onda za njihovu obradu možemo koristiti sistema za rada sa bazama kao što je Access.

Međutim, tu treba imati na umu jednu vrlo važnu stvar. Access je moćna alatka - ali samo alatka za *realizaciju* baze podataka. Naravno za rešavanje nekog problema nije Vam dovoljan samo alat pomoću koga će te taj problem rešiti.

Kreiranje baze počinje mnogo pre Access-a. Prvo morate razumeti problem (zahteve i ciljeve) koji želite rešiti korišćenjem baze podataka. Taj problem treba zatim analizirati i napraviti plan kako da se on reši. Od složenosti zahteva zavisi da li će te taj plan formalizovati kroz neki projekat. Za manje zahtevne baze dovoljna će biti neka skica ili makar misaona predstava o tome kako rešiti zadatak. U svakom slučaju, nemojte tražiti od Accessa da Vam da ideju za rešavanje problema, on će Vam samo pomoći da tu ideju realizujete na računaru.

Dakle, u ovom uputstvu nećete naći odgovor na pitanje kako praviti baze podataka već samo odgovor (uvodni) kako ih implementirati u Access-u.

EVIDENCIJA ISPITA

Ispitni rok:

Datum:

Predmet:

Rezultati ispita

R.b.	Broj indeksa	Prezime	Ime	Ocena
1.	11/80	Antić	Zoran	7
2.	14/80	Jović	Jasmina	5
3.	16/80	Marković	Drinka	8
4.	18/80	Petrović	Petar	8
5.	20/80	Tomić	Milan	9

Sl.1 Obrazac za evidenciju ispita

Terminologija baze podataka

U Accessu, sve informacije koje želimo da čuvamo smeštamo u *tabele* (Sl.2). Tabele su fizički nosioci podataka - jedino mesto u bazi gde se oni skladište. Elementarni podaci koje čine jednu informaciju definišu kolone - *polja* tabele. Na primer, polja tabele sa slike Sl.2 su: *BrojIndeksa*, *Ime*, *Prezime*, *DatumRodjenja*, *MestoRodjenja*. Skup svih podataka u jednom redu tabele predstavlja jedan *zapis*. Zapis predstavlja jediničnu informaciju koja se čuva u tabeli. U tabeli sa slike Sl.2 prvi red predstavlja jedan zapis (informaciju) o studentu *Antić Zoranu*.

Skup svih tabela zajedno sa ostalim objektima koji služe za unos, filtriranje, pretraživanje, prikazivanje..., predstavlja Accessovu *bazu podataka*.

TStudenti : Table					
	BrojIndeksa	Prezime	Ime	DatumRodjenja	MestoRodjenja
▶	11/80	Antić	Zoran	5/3/60	Požega
	14/80	Jović	Jasmina	12/4/61	Trstenik
	16/80	Marković	Drinka	24/12/60	Kraljevo
	18/80	Petrović	Petar	1/7/60	Kraljevo
	20/80	Tomić	Milan	15/6/60	Užice
*					

Record: 1 of 5

Kreiranje baze podataka

Početak rada na jednoj bazi sastoji se u definisanju njene arhitekture. Arhitektura predstavlja skelet (noseću konstrukciju) jedne baze. Ona mora biti dovoljno fleksibilna da omogući uvodjenje novih zahteva (proširivanje baze) ili izmenu postojećih. Sve što se kasnije radi u bazi oslanja se na nju.

Arhitekturu jedne baze čine strukturni (gradivni) elementi i relacije između njih. Kao osnovni gradivni elementi u bazama podataka koriste se tabele.

Izrada tabela baze podataka

Svi podaci sa kojima želite da radite moraju da budu smešteni u tabele. To su jedini objekti u bazi koji fizički čuvaju podatke. Zato je prvi korak u kreiranju neke baze izrada tabela. Tabele predstavljaju temelj Vaše aplikacije i zato im je potrebno posvetiti najviše pažnje. Naravno potrebno je dobro razumeti svrhu baze koju pravimo i šta se želi njome postići.

Želja nam je da napravimo elektronsku bazu kroz koju će mo voditi evidenciju o polaganju ispita na fakultetu. Koje su nam tabele za to potrebne? Da bi smo odgovorili na to pitanje prvo treba uočiti entitete (koncepte, pojmove) iz domena problema. Ti entiteti predstavljaju smislaone celine podataka koje treba realizovati kroz tabele. Kako ih odrediti?

Ne postoje precizna pravila koja Vas mogu dovesti do dobrog izbora entiteta-tabela. To je stvar veštine, znanja i iskustva onoga ko osmišljava (projektuje) bazu. Od toga zavisi i kvalitet jedne baze, ono što jednu bazu razlikuje od druge. Ako ovde napravite loš izbor teško da će krajnja realizacija biti dobra. I obratno, dobrim izborom ispunjavate potrebne uslove da aplikacija bude uspešna. Za početak, koristimo preporuku da u opisu problema treba uočiti imenice koje se koriste. One predstavljaju polazne kandidate za tabele.

Npr. u opisu problema datom slikom Sl.1. mogu se uočiti imenice: *ispiti, studenti, predmeti*. To znači da nam trebaju tabele za ispite, studente i predmete. Prva tabela koju ćemo napraviti je tabela sa opštim podacima (*ispitni rok, datum polaganja, predmet*) o ispitu. Nju ćemo nazvati *TIspiti**. Pošto za svaki ispit moramo da evidentiramo rezultate ispita napravićemo i tabelu *TRezultatIspita*. U njoj ćemo da čuvamo podatke o studentima koji su polagali ispitn i o ocenama koje su dobili. Tu se može postaviti pitanje zašto formirati dve tabele za podatke koje se tiču jednog ispita? Ukoliko bi smo sve podatke čuvali u jednoj tabeli ona bi izgledala kao ona na slici Sl.3.

IspitniRok	DatumPolagai	Predmet	BrojIndeksa	Prezime	Ime	Oцена
januar	24/1/82	Matematika	11/80	Antić	Zoran	7
januar	24/1/82	Matematika	14/80	Jović	Jasmina	5
januar	24/1/82	Matematika	16/80	Marković	Drinka	8
januar	24/1/82	Matematika	18/80	Petrović	Petar	8
januar	24/1/82	Matematika	20/80	Tornić	Milan	9

Sl.3 Izgled tabele *TIspiti* za slučaj da se svi podaci o ispitu čuvaju u njoj

Vidimo da bi u tom slučaju kod svakog studenta morali da unosimo i opšte podatke o ispitu. To ne samo da otežava unos podataka u bazu nego i nepotrebno duplira podatke. Zato je mnogo racionalnije zajedničke podatke o ispitu čuvati samo na jednom mestu u zasebnoj tabeli a rezultate ispita (student, ocena) u drugoj. Međutim, to nas obavezuje da kasnije između ove dve tabele uspostavimo vezu (relaciju) kako bi smo održali integritet podataka.

Kako studenti predstavljaju zaseban entitet i za njih ćemo kreirati tabelu *TStudenti* gde ćemo čuvati podatke kao što su *BrojIndeksa, Ime, Prezime, DatumRodjenja, MestoRodjenja*. Konkretno za ovu bazu podaci o datumu i mestu rođenja studenta nam nisu potrebni, ali zbog mogućih proširenja baze (npr. vođenje statistike o starosnoj strukturi studenata, mestima iz kojih se studenti najviše upisuju) uvodimo i ove podatke.

I četvrta tabela koju ćemo formirati je *TPredmeti* gde ćemo unositi podatke o nazivu predmeta.

Zašto se koristimo posebne tabele za studente i ispite? Recimo zašto podatke o studentima (ime i prezime, broj indeksa, itd.) koji su polagali ispit ne čuvamo u tabeli *TRezultatIspita*? Pa zamislite koji bi to posao bi da u svakom ispitnom roku kucamo ime i prezime ponovo. Mnogo je jednostavnije napraviti samo jednu tabelu sa studentima pa te podatke onda *pozivati* po potrebi. Time se izbegava i jedan potencijalno vrlo opasan problem (teško ga je otkriti) koji bi se javio

* Napomena o imenovanju objekata. Svi objekti istog tipa koje kreirate u bazi moraju imati jedinstveno ime koje jednoznačno identifikuje dati objekat. To znači da npr. može postojati tabela sa imenom *Ispiti* i obrazac sa istim imenom (identifikacija se vrši na osnovu tipa objekta) ali ne mogu postojati dve tabele sa istim imenom. Mi ćemo ime svakog objekta započinjati velikim slovom koje ukazuje na vrstu objekta: T za tabele (*TIspiti*), Q za upite (*QIspiti*), F za obrasce (*FIspiti*) itd. Ukoliko se neko ime sastoji od dve reči npr. Položeni ispiti uvodimo ime *PolozeniIspiti*, znači spojeno, bez naših slova i sa velikim početnim slovom za svaku reč.

ukoliko bi pri tom ponovnom prekucavanju imena i prezimena studenta napravila greška u kucanju. Ukoliko bi recimo, zamenili samo jedno slovo u imenu (to za korisnika i nije neka greška ali je za računar ogromna) pojavi bi se potpuno novi student u bazi koji realno ne postoji. Slična priča važi i za tabelu *TPredmeti*.

Uopšte, ideja je sledeća: sve podatke o nekom entitetu ("predmetu") čuvati na jednom pestu - u njegovoj sopstvenoj tabeli. Time se izbegava dupliranje podataka i olakšava održavanje baze. Pored toga, dalje proširivanje baze je mnogo lakše.

Jedno praktično pravilo pri formiranju tabela: ukoliko između dva podatka postoji relacija "jedan prema više" tada te podatke treba smestiti u različite tabele. Na primer, pošto za jednog studenta postoji više položenih ispita - podatke o studentima i položenim ispitima treba čuvati u različitim tabelama.

Planiranje tabela

Pošto smo odredili svrhu i broj tabela (*TIspiti*, *TRezultatiIspita*, *TStudenti* i *TPredmeti*) prelazimo na njihovo projektovanje. To znači da za svaku tabelu treba definisati polja (kolone) i tipove podataka koji se u njima mogu smestiti. Kako tabele *TIspiti* *TRezultatiIspita* objedinjavaju podatke iz drugih tabela prvo ćemo definisati tabele *TStudenti* i *TPredmeti*.

Tabela: <i>TStudenti</i>			
<i>Naziv polja</i>	<i>Tip polja</i>	<i>Podvrsta/veličina polja</i>	<i>Komentar</i>
BrojIndeksa	Text ¹	6	Ovo je polje koje jednoznačno određuje svakog studenta, unosi se u formatu ####/# gde # označava cifru
Prezime	Text	20	Prezime
Ime	Text	15	Ime
DatumRodjenja	Date/Time	Short Date ²	Datum rođenja studenta
MestoRodjenja	Text	30	Mesto rođenja studenta

¹ Text je tip podatka koji sadrži slova i/ili brojeve sa kojima se ne vrše računске operacije, dužine do 255 karaktera.

² Short Date je tip podatka za čuvanje datuma u formatu dd/mm/yy.

Definisanjem veličine polja (npr. za BrojIndeksa 6) određujete maksimalni broj znakova koji se mogu uneti u to polje. Ako pokušate da unesete više Access vam to neće dozvoliti. Na taj način možete kontrolisati unos podataka (npr. dužina matičnog broja u ličnoj karti je tačno određena). Ako ipak ustanovite da Vam treba veće polje to se naknadno (posle unetih podataka) može izmeniti. Problem može da nastane ako želite da smanjite veličinu polja. To u principu treba izbegavati jer se može izgubiti deo podataka.

Znači pri definisanju veličine polja računajte sa maksimalnim vrednostima.

Iz gornje tabele se vidi da je informacija o jednom studentu određena sa pet podataka. Postavlja se pitanje na koliko polja razbiti jednu informaciju? Da li ime i prezime staviti u jedno polje ili razbiti na dva kao što je to ovde urađeno? Preporuka je da se informacija razbije na što veći broj polja. Time se dobija mnogo veća fleksibilnost u obradi podataka. Na primer, mogu se javiti obrasci gde je potrebno prvo uneti ime pa prezime a u drugom slučaju obratno. To će te lakše uraditi ako podatke o imenu i prezimenu čuvate u dva polja.

Tabela: <i>TPredmeti</i>			
<i>Naziv polja</i>	<i>Tip polja</i>	<i>Podvrsta/veličina polja</i>	<i>Komentar</i>
PredmetID	Text	4	Ovo je polje jednoznačno određuje svaki predmet i unosi se u formatu 0000 gde 0 označava cifru, npr. 0001
NazivPredmeta	Text	50	Naziv predmeta

Tabela: <i>TIspiti</i>			
<i>Naziv polja</i>	<i>Tip polja</i>	<i>Podvrsta/veličina polja</i>	<i>Komentar</i>
IspitID	AutoNumber ³	LongInteger	Ovo je polje jednoznačno određuje svaki ispit, Access ga automatski popunjava
IspitniRok	Text	20	Naziv ispitnog roka: januar, april, jun, septembar, oktobar
DatumPolaganja	Date/Time	Short Date	Datum polaganja ispita
PredmetID	Text	4	U ovom polju se unosi šifra ispita koji se polaže

³AutoNumber je poseban tip podatka koji se automatski dodeljuje i nikada se ne menja.

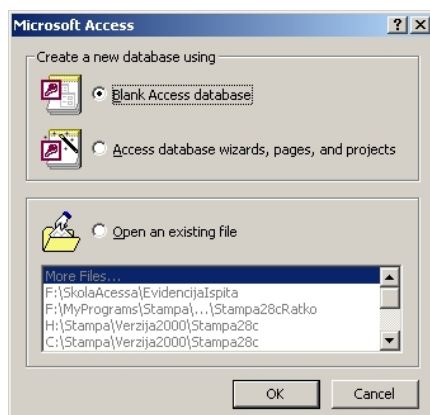
Tabela: <i>TRezultatiIspita</i>			
<i>Naziv polja</i>	<i>Tip polja</i>	<i>Podvrsta/veličina polja</i>	<i>Komentar</i>
IspitID	Number ⁴	LongInteger	Služi za vezu sa tabelom <i>TIspti</i>
BrojIndeksa	Text	6	Šifra studenta koji je polagao ispit
Ocena	Number	Byte	Ocena sa ispita

⁴ Number je tip podatka za čuvanje brojeva (numeričkih vrednosti) koji se koriste u aritmetičkim izrazima. Veličina broja zavisi od tipa: LongInteger (celi brojevi od -2,147,483,648 do 2,147,483,647), Byte (celi brojevi 0 - 255).

U tabeli *TIspti* uveli smo podatak *IspitID* koji ne postoji na obrascu sa slike S11. Preporuka je da svaka tabela ima jedno (ili više) polje (polje ključa) koje jednoznačno određuje svaki zapis u tabeli. Pored toga što nam omogućava da u tabeli nademo određeni zapis to polje nam omogućava da se povežemo sa drugim tabelama.

Pokretanje programa

Sada kada smo osmislili tabele pokrećemo Access da bi smo pomoću njegovih alata te tabele i napravili. Microsoft Access se pokreće kao i svaki drugi program u Windowsu - klikom (odnosno dvoklikom) na ikonu programa. Posle pozdravnog ekrana aplikacije, pojavljuje se ekran na kome možete birati jednu od tri opcije (Sl.4): kreirati novu bazu (Blank database), iskoristiti jedan od čarobnjaka za automatizovano kreiranje baze biranjem jednog od predefinisanih tipova baza (Access database wizards, pages, and projects) ili otvoriti već postojeću bazu (Open an existing file). Pošto kreiramo novu aplikaciju, biramo prvu opciju Blank database.

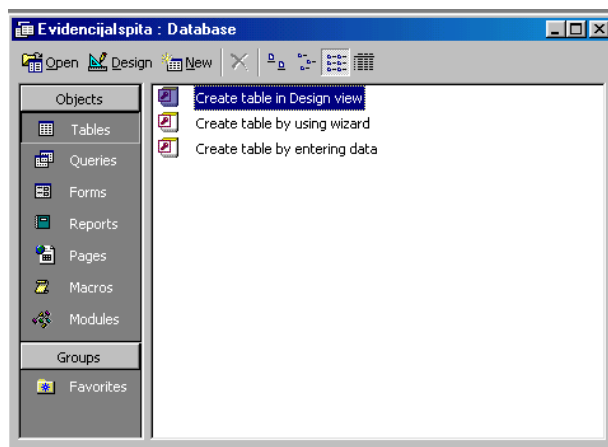


Sl.4 Izbor opcija pri pokretanju Accessa

Posle klika na *OK* program traži da snimimo datoteku, iako još ništa nismo radili! Kod programa za rad sa bazama podataka integritet podataka je na prvom mestu. To znači da svaki podatak koji unesete, izmenite ili obrišete trenutno zapisuje na disk, a naredba *Save* pri tom nema isto značenje kao u drugim programima. Aktiviranjem naredbe *Save* snima se struktura baze podataka, a ne sami podaci! Otuda zahtev za snimanjem datoteke ne predstavlja iznenađenje, već razumnu pojavu. Datoteci dajte ime *EvidencijaIspita* i snimite je u Vašu radnu fascikulu.

Posle kreiranja datoteke, pojavljuje se radno okruženje Accessa koje podseća na okruženje ostalih Office programa. Ispod naslovne trake na kojoj piše koje vlasnik prozora (Microsoft Access) nalazi se linija menija sa komandama koje su Vam na raspolaganju u Accessu. Ispod nje, nalazi se paleta sa najčešće korišćenim alatima u vezi sa poslom koji trenutno obavljate. Na dnu prozora smeštena je statusna linija gde Access ispisuje trenutno stanje u radu programa. Kao i kod ostalih Office programa izgled ovog radnog okruženja se može poprilično menjati (možete npr. napraviti svoje menije, svoje palete sa altima itd.).

Centralni deo ekrana zauzima prozor (kontejner) baze podataka (na naslovnoj liniji prozora baze piše *EvidencijaIspita: Database*) sa objektima baze (Tables, Queries, Forms, Reports, Pages, Macros i Modules) kao na slici S1.5.



Sl.5 Prozor baze podataka

Objekti baze podataka

Pre nego što nešto kažemo o objektima (elementima) baze treba reći da se u Accessu pod bazom podrazumevaju ne samo podaci koji se čuvaju već i svi objekti za rad sa podacima. Za razliku od drugih programa za rad sa bazama podataka kod kojih se svaka tabela nalazi u posebnoj datoteci, kompletna Access baza (sve tabelle) se nalazi u jednoj datoteci.

Tabele

Tabele su osnovne strukture u bazi podataka u koje se fizički smeštaju podaci. U njima se skadište svi podaci sa kojima ćete raditi i to su prvi objekti koje treba napraviti. Ostali objekti služe za obardu podataka koji se nalaze na jednom mestu - tabelama. U tabelama podaci su smešteni u redove (zapise) i kolone (polja). Jedna baza se obično sastoji od više tabela u koje se razdvojeno čuvaju podaci vezani za jedan entitet iz domena aplikacije. Na primer, mi ćemo napraviti jednu tabelu za studente i jednu tabelu za ispite. Naravno, podaci iz različitih zabela se kasnije po potrebi mogu kombinovati.

Upiti

Upiti su alati koji služe za dobijanje odgovora na određena pitanja o podacima iz tabela. Na primer, možemo dobiti odgovor na pitanja "Koliko je studenata iz Kraljeva?", "Koliko je prosečna starost studenata?". Upit možemo koristiti za kombinovanje podataka iz više odvojenih ali povezanih tabelle. Recimo povezivanjem (spajanjem) tabela o studentima i ispitima možemo odgovoriti na pitanja: "Koja je srednja ocena iz nekog predmeta?", "Koliko je puta student polagao neki ispit?", "Koliko je studenata položilo neki ispit", itd. Upite možemo koristiti i za izvršavanje operacija nad podacima (ažuriranje velike količine podataka, prepisivanje iz jedne tabelle u drugu, brisanje). Kao i tabelle i oni se mogu koristiti kao podloga za obrasce (forme) i izveštaje.

Obrasci

Obrasci (forme) omogućavaju da komuniciramo sa aplikacijom - unosimo, pregledamo, menjamo i brišemo podatke iz tabela. Sve ovo možemo raditi i direktno na tabeli međutim, pomoću obrazaca se mogu napraviti prikladni formati (korisnički interfejsi) koji podsećaju na papirne obrasce što krajnjem korisniku omogućava intuitivan rad i lakše prihvatanje vaše aplikacije. Na primer, za unos podataka sa neke fakture mnogo je efikasnije napraviti obrazac koji liči na odgovarajuću papirnu formu tako da korisniku ne morate mnogo objašnjavati gde šta treba uneti. Obrasci mogu da sadrže i druge obrasce (podobrasce) što omogućava istovremeni rad sa podacima iz više tabela.

Izveštaji

Izveštaji su elektronski oblik klasičnih papirnih obrazaca koje možemo gledati na ekranu ili štampati na papir. Npr. ako je potrebno da odštampano spisak studenata koji su položili ispit to ćemo uraditi uz pomoć izveštaja. Izveštaji mogu biti kreirani u različitim oblicima, recimo možemo štampati koverte sa adresama studenata. Podaci prikazani u izveštaju mogu biti izvedeni iz tabela ili iz upita (filtrirani podaci). Dizajniranje izveštaja je veoma slično dizajniranju formi - izgled zavisi od Vaše mašte.

Strane za pristup podacima

Spona između baze i Interneta je izvedena pomoću strana za pristup podacima (Data Access Pages). Za razliku od Accessa 97, gde se pogled na podatke iz baze može izvesti eksportovanjem podataka u HTML kod, Data Access strane

u Accessu 2000 predstavljaju dinamičku vezu između baze i HTML koda. U razvijenoj aplikaciji kojoj se pristupa preko Internet Explorera možete pristupiti podacima u tabelama MDB baze, menjati ih ili pregledati. Za intranet okruženje je vama na volji da li ćete bazi pristupati kroz front-end deo aplikacije ili preko Data Access strana i Internet Explorera.

Makroi

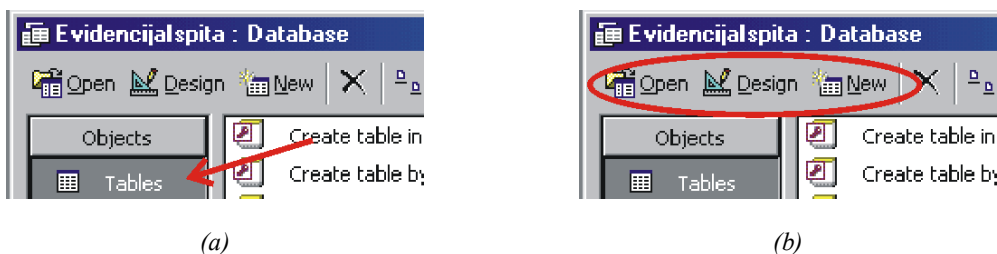
Ako često treba da obavite neke poslove koji se sastoje od niza koraka (radnji) možete da napišete makro. Makro je skup komandi koji automatizuje obavljanje nekog posla - kada se makro pokrene izvršavaju se sve komande po redosledu kojim su zapisane. Bez pisanja programa možete recimo da automatizujete slanje nekog izveštaja elektronskom poštom. I mada su makroi veoma korsini za automatizovanje pojedinih poslova bez pisanja ijednog programskog reda njihovo korišćenje se u novijim aplikacijama se ne preporučuje. Oni, zapravo, postoje isključivo zbog kompatibilnosti sa prethodnim verzijama Accessa. Za programiranje se preporučuje upotreba modula.

Moduli

Kao i makroi, moduli omogućavaju automatizaciju i prilagođavanje Accessa. Moduli međutim, pružaju mnogo više mogućnosti za precizniju kontrolu akcija i formiranje funkcija koje ne postoje u Accessu. Mana im je što zahtevaju malo više programerskog iskustva i veštine u programiranju VBA (Visual Basic for Application). Za pisanje profesionalnih programa korišćenje VBA je neophodno dok se jednostavnije aplikacije mogu napraviti i bez njihove upotrebe.

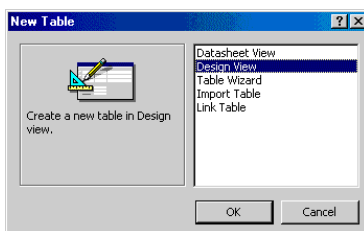
Kreiranje tabela

Počecemo sa kreiranjem tabele *TStudenti*. U prozoru baze na traci Objects selektujte Tables (Sl.6a). U vrhu ekrana prozora baze se nalaze tri dugmeta: Open (otvara postojeću tabelu za pregled), Design (izmene u strukturi tabele) i New (kreiranje nove tabele) (Sl.6b).



Sl. 6 Kreiranje nove tabele

Izborom komande New otvara se okvir za dijalog *New Table* kao na slici Sl.7.

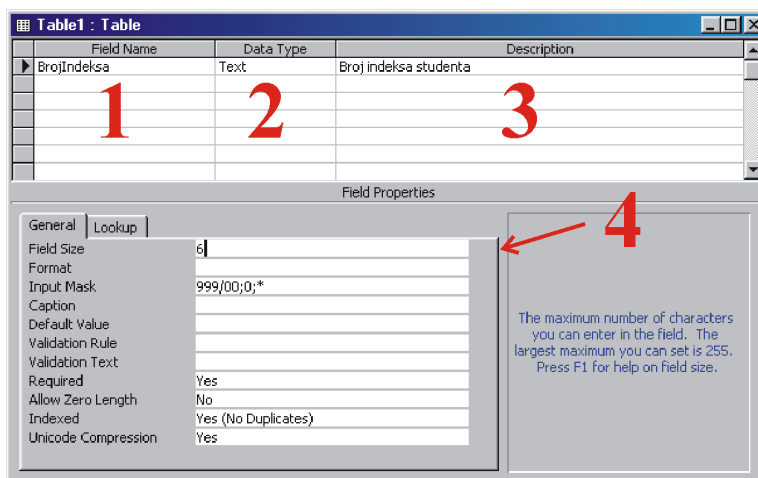


Sl. 7 Opcije pri kreiranju nove tabele

Značenje opcija je sledeće:

- Datasheet View otvara nedefinisanu tabelu sa univerzalnim poljima u koje se mogu unositi podaci, a Access će pokušati sam da postavi tip polja na osnovu unetih podataka;
- Design View predstavlja pogled na tabelu u prozoru se dizajn. U ovom prozoru zadajemo nazive kolone i adekvatne tipove podataka koji se mogu nalaziti u kolonama, odnosno definišemo broj kolona i tipove podataka;
- Table Wizard je čarobnjak koji će na osnovu vaših odgovora na postavljena pitanja kreirati tabelu;
- Import Table služi za unos podataka iz već postojeće tabele ili baze. Ovim se podaci kopiraju iz ulazne tabele u novokreiranu tabelu;
- Link Table postavlja prečicu koja pokazuje na drugu tabelu iz otvorene baze sa kojom trenutno radimo ili druge Access baze. Ovim se podaci ne kopiraju niti premeštaju, već se uspostavlja relacija na povezanu (linkovanu) tabelu.

Izborom druge opcije - Design View, i potvrdom na dugme OK, otvara se novi prozor za dizajniranje tabele (Sl.8).



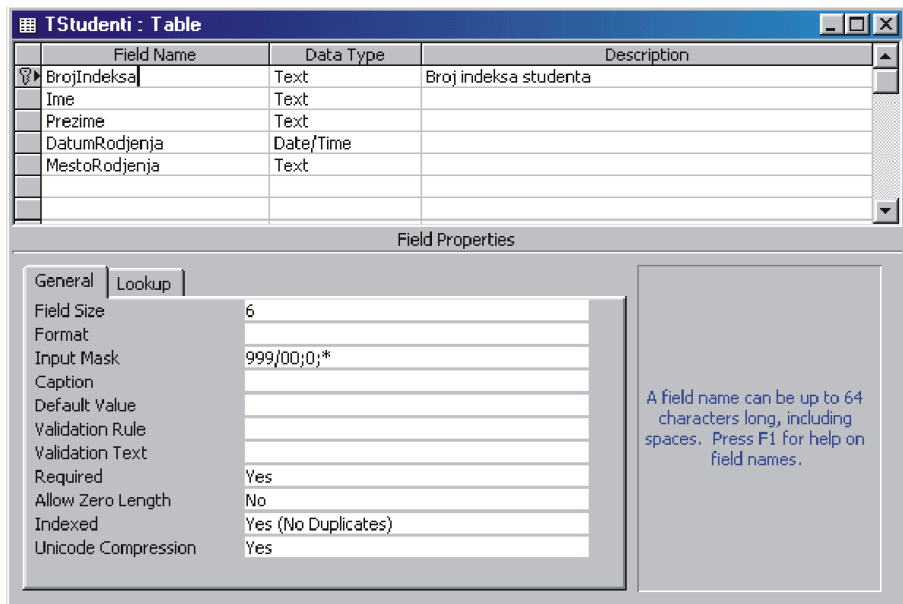
Sl.8 Prozor za definisanje polja nove tabele

U gornjem delu prozora unose se polja (kolone) nove tabele, dok se u donjem delu definišu svojstva svakog polja pojedinačno. U koloni 1 se unosi naziv polja. Taj naziv mora biti jedinstven za tabelu. U prvu ćeliju unosimo *BrojIndeksa*. U istom redu kolone 2 treba uneti tip podataka koji se čuva u datom polju. Iz padajuće liste biramo *Text*. U trećoj koloni možemo po želji uneti komentar koji bliže objašnava dato polje. Taj komentar se kasnije pojavljuje u statusnoj liniji Accessovog prozora i može biti od pomoći onome ko unosi podatke u tabelu. Sada na kartici *General* donjeg dela prozora (rb. 4) definišemo svojstva polja *BrojIndeksa*.

Značenje svojstava je sledeće:

- *Field Size* - veličina polja. Za tekstualne podatke, podrazumevana veličina je 50 karaktera, dok je za brojeve *Long Integer*.
- *Format* - definiše način na koji će sadržaj polja biti prikazan.
- *Input Mask* - definiše masku za unos podataka. Na ovaj način možemo obezbediti jednoobrazan unos podataka. Tako ako želimo da broj indeksa unosimo u formatu XXX/XX, gde X označava neku cifru 0-9, pri čemu prve tri cifre označavaju redni broj upisa, a druge dve (posle "/"), označavaju godinu upisa ulaznu masku ćemo postaviti na - 999/00;0;*. Devetke na početku omogućavaju unos ili blanko znaka (*space*) ili neke cifre, dok nule iza kose crte zahtevaju *obavezan unos* dve cifre za godinu upisa. Pri unosu broja indeksa ukoliko redni broj ima manje od tri cifre (npr. 15) prazna mesta popuniti blanko znakom (*space*) - znači išlo bi □15.
- *Caption* - Tekst koji se pojavljuje u zaglavlju kolone kada se tabela otvori u modu za editovanje (mi smo sada u dizajn modu).
- *Default Value* - podrazumevana vrednost polja. Pri popunjavanju novog reda u tabeli u polje se automatski unosi ova vrednost, koja se po potrebi može i menjati.
- *Validation Rule* - je izraz koji definiše zahteve koje moraju da ispune podaci uneti u određeno polje.
- *Validation Text* - tekst koji se pojavljuje ukoliko uslov iz *Validation Rule* nije ispunjen.
- *Required* - određuje da li je u tom polju unos podataka obavezan ili ne.
- *Allow Zero Length* - ovo je svojstvo specifično za tekstualne podatke: određuje da li je dozvoljeno postojanje stringa nulte dužine ("").
- *Indexed* - je opcija kojom se postavlja indeks na određeno polje radi bržeg pretraživanja ili sortiranja podataka. *No Duplicates* u zagradi znači da se ne mogu ukucati da ista broja indeksa. Ukoliko je indeksiranje uključeno, *update* podataka u tabeli će ići nešto sporije, ali će se pretraživanje po ovom polju značajno ubrzati.

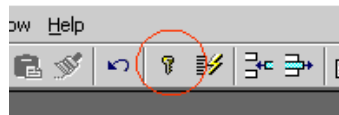
Uneta svojstva za polje *BrojIndeksa* se mogu videti na Slici Sl.8. Na sličan način kao ovo unose se i ostala polja u tabelu *TStudenti* (Sl.9).



Sl.9 Tabela TStudenti u dizajn prikazu

Ostalo nam je još da za ovu tabelu definišemo polje primarnog ključa. Primarni ključ je jedno ili više polja kojim se jednoznačno identifikuje svaki zapis u tabeli. Access će smatrati da su dva zapisa ista samo ako su podaci u polju (poljima), koje definiše primarni ključ, isti i neće nam dozvoliti unos dva takva zapisa. Kao što ne mogu postojati dve osobe sa istim matičnim brojem tako i u tabeli ne mogu postojati dva studenta sa istim brojem indeksa. Skoro za svaku tabelu je potrebno definisati primarni ključ. Polje primarnog ključa je indeksirano bez mogućnosti dupliranja podataka (Indexed: Yes(No Duplicates)).

Da bi smo polje primarnog ključa postavili na polje *BrojIndeksa*, prvo se pozicioniramo na to polje, a onda kliknemo na ikonicu ključa na paleti sa alatima (Sl.10). Posle toga će se u tabeli, levo od naziva polja *BrojIndeksa*, pojaviti ikonica ključa koja ukazuje da je to polje primarni ključ table (Sl.9).

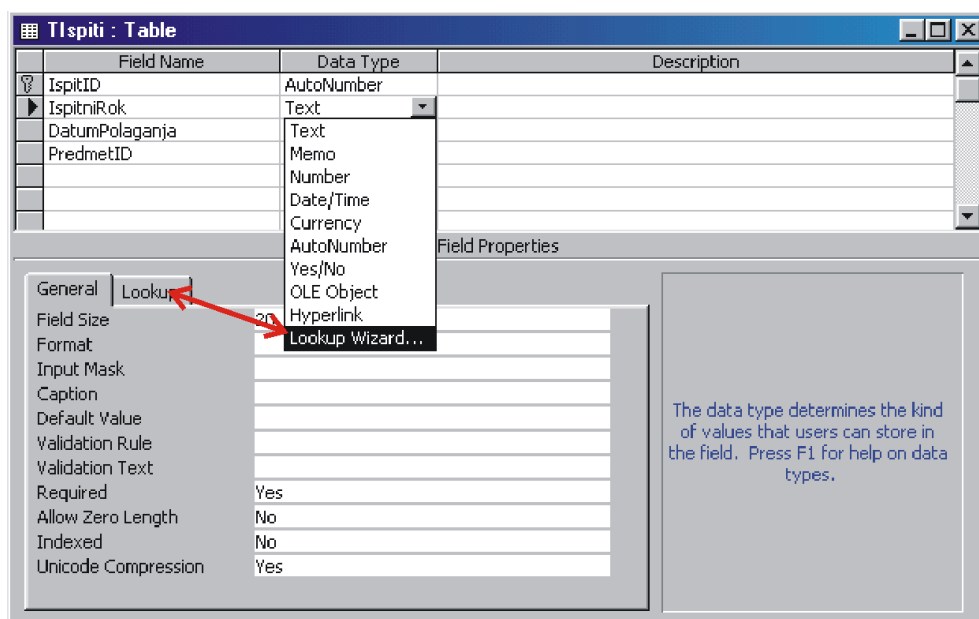


Sl.10 Alata kojom se formira primarni ključ

Ostaje nam da još snimimo strukturu ove table. Iz menija *File* odaberemo opciju *Save*. U okviru za dijalog koji će se pojaviti unesite za ime table *TStudenti*. I ovde važi pravilo *ne morate* tek na kraju da snimate to što radite, to možete raditi povremeno tokom rada od samog početka.

Ovim smo završili posao oko kreiranja prve table. Na sličan način se prave i ostale table*. Kod table *Tispiti* postoji jedna mala specifičnost. Ako u prozoru baze selektujemo ovu tabelu i pritisnemo na taster *Design* (Sl. 6b) otvara se njen dizajn prikaz (Sl. 11).

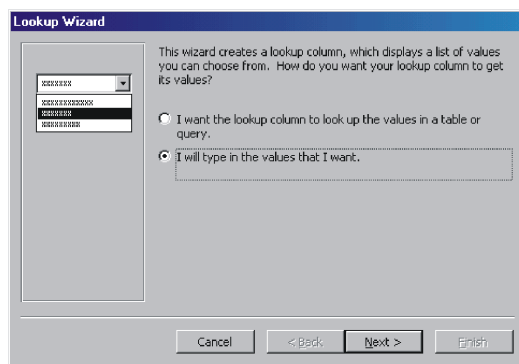
* Baza koja ide uz ovo uputstvo nalazi se u direktorijumu Ort pod nazivom EvidencijaIspita



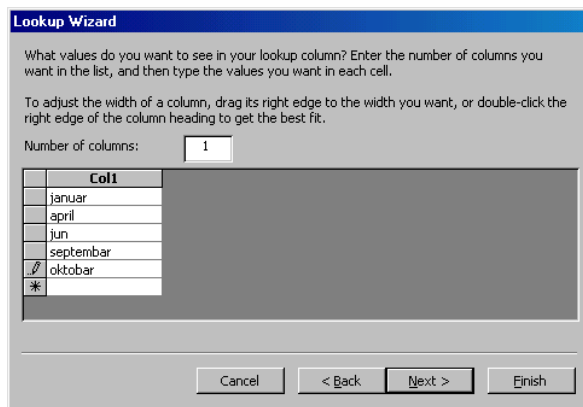
Sl.11 Dizajn prikaz tabele Ispiti

U polju *IspitniRok* unosimo naziv ispitnog roka (januarski, junjski, septembarski, oktobarski). Kako su ovi nazivi unapred poznati i retko se menjaju, da ih ne bi stalno kucali napravićemo padajuću listu iz koje ćemo jednostavno birati ispitni rok. To ćemo uraditi na sledeći način. Pri definisanju tipa podatka za ovo polje otvaramo padajuću listu (Sl.11) i iz nje biramo *Lookup Wizard...* nakon čega će se otvoriti okvir za dijalog kao na slici Sl.12. U njemu biramo šta će biti izvor podataka za padajuću listu. Mi ćemo izabrati drugu opciju (sami ukucavamo listu). Nakon toga, pritiskom na dugme *Next*, otvara se sledeći dijalog boks (Sl.13). U njemu treba da unesemo sadržaj padajuće liste kao što je to prikazano na slici. Pritiskom na dugme *Next* dolazimo do zadnjeg korak (Sl.14) gde nam se pruža mogućnost da promenimo naziv polja. Mi tu ništa nećemo menjati već ćemo pritiskom na dugme *Finish* završiti proceduru kreiranja liste.

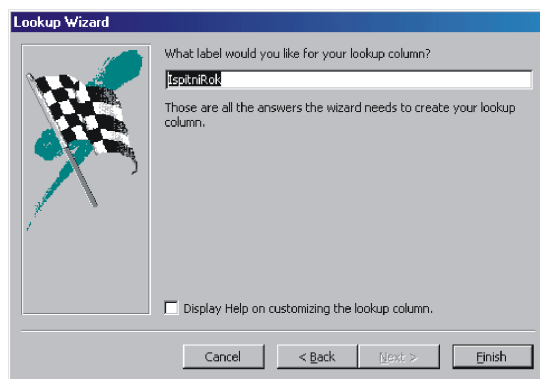
Ubuduće pri unosu podataka u polje *IspitniRok* dovoljan će biti izbor sa liste koju smo upravo napravili (Sl.15).



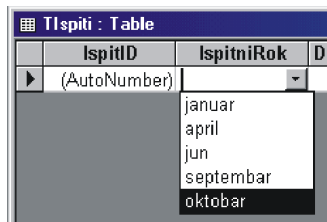
Sl. 12 Izbor opcije za način popune sadržaja padajuće liste



Sl. 13 Unos sadržaja padajuće liste



Sl. 14 Završni korak u formiranju padajuće liste

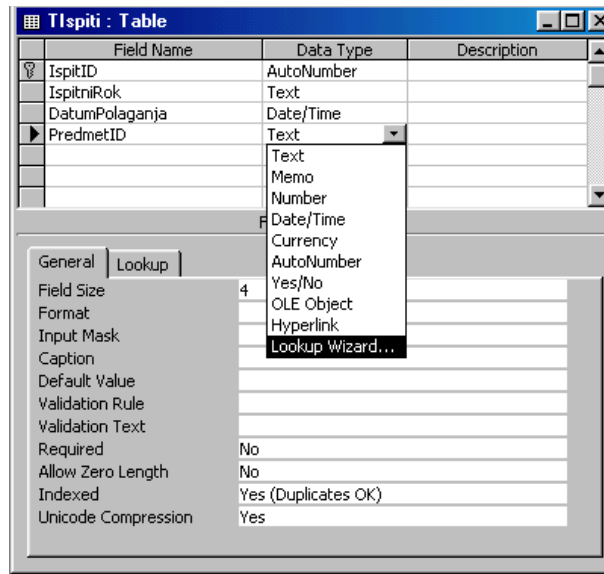


Sl. 15 Unos podataka u polje ispitni rok izborom iz padajuće liste

Povezivanje tabela

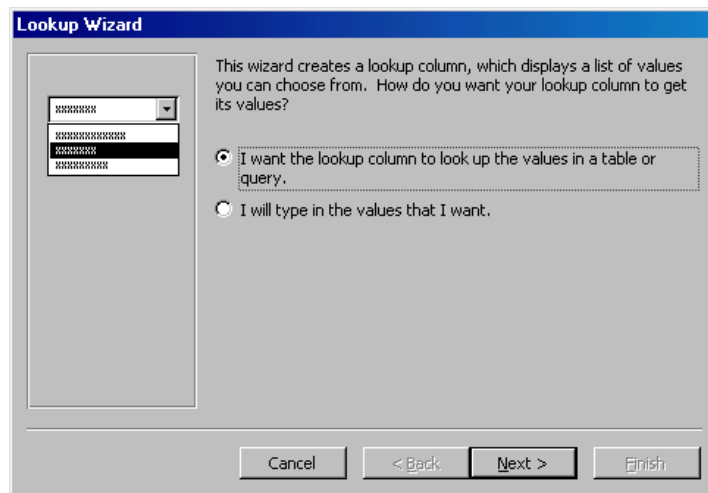
Ako je prvi korak u formiranju temelja buduće baze bio definisanje i pravljenje tabela, onda je drugi korak uspostavljanje veza između njih kao bi se dobila jedna celovita informacija o ispitu poput one prikazane na slici SI11. Tabele (informacije u njima) nisu nezavisne, između njih postoji jasna logička zavisnost. Npr. ne mogu se pojaviti podaci u tabeli *TRezultatiIspita* ako odgovarajući ispit ne postoji u tabeli *Tispiti*. Zahvaljujući relacionim tabelama Access ostvaruje punu snagu.

Kako se u polju *PredmetID* table *Tispiti* može uneti samo šifra onog predmeta koji postoji u tabeli *TPredmeti* prvu relaciju ćemo uspostaviti između ove dve tabela. Selektovanjem table *Tispiti* i pritiskom na dugme *Design* (SI.6) ulazimo u dizajn mod ove table (SI16).



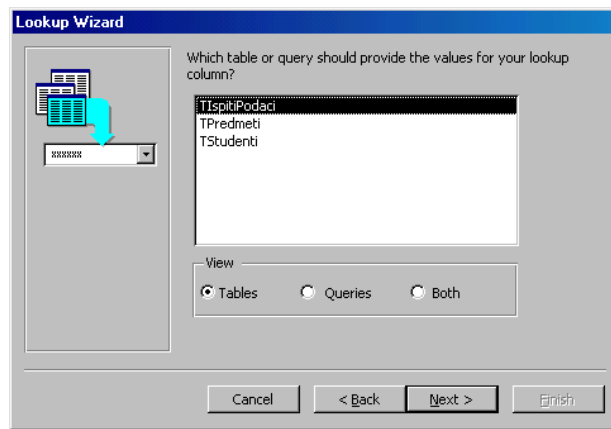
Sl. 16. Tabela Tispiti u dizajn modu

Otvaranjem padajuće liste, za tip podatka polja *PredmetID*, biramo opciju *Lookup Wizard* čime se pokreće procedura za uspostavljanje veze sa tabelom *TPredmeti*. Prvi korak je definisanje izvora podataka za to polje (Sl.17).



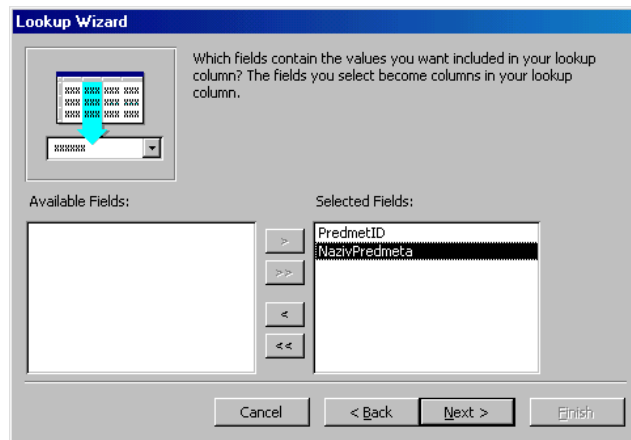
Sl. 17 Izbor načina popune polja *PredmetID*

U ovom slučaju treba izabrati prvu opciju jer se moguće vrednosti za polje *PredmetID* nalaze u tabeli. U drugom koraku biramo konkretno tabelu *TPredmeti* (Sl18).



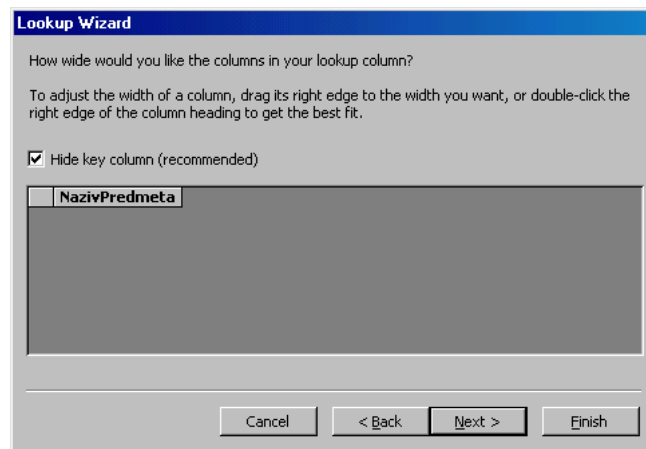
Sl. 18 Izbor tabele TPredmeti

Pritiskom na dugme *Next* otvara se prozor za izbor polja iz tabele *TPredmeti* (Sl.19).



Sl. 19 Izbor polja tabele TPredmeti

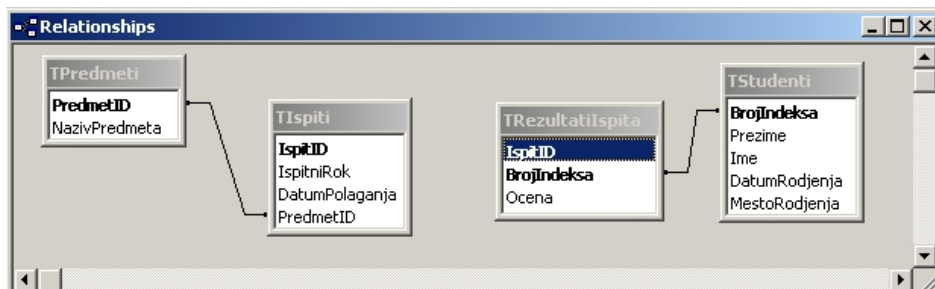
U posljednjem prozoru (Sl.20) podešavamo širinu kolone padajuće liste, nakon čega možete pritisnuti dugme *Finish*.



Sl. 20 Podešavanje kolone padajuće liste

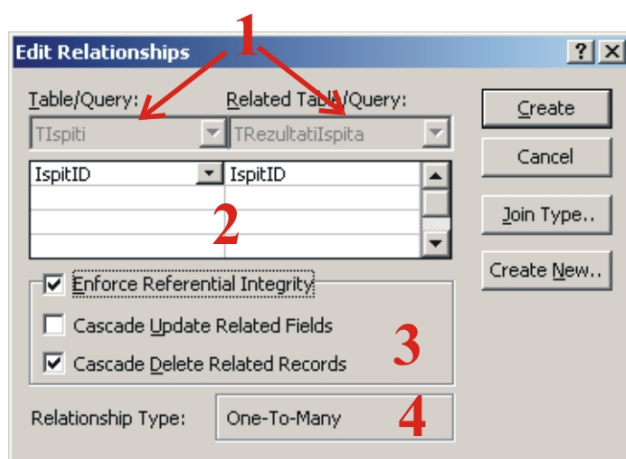
Time je napravljena relacija između tabela *TIspliti* i *TPredmeti*. Na sličan način treba napraviti relaciju između tabele *TRezultatiIspita* i *TStudenti* uz napomenu da u 3. koraku (Sl.19) treba izabrati polja *BrojIndeksa*, *Prezime* i *Ime* a u 4. koraku (Sl.20) treba isključiti opciju *Hide key column*. Ako u meniju *Tools* izaberemo komandu *Relationships* otvoriće

se prozor sa prikazom trenutno uspostavljenih relacija (Sl.21).



Sl. 21 Relacije između tabela uspostavljene uz pomoć Lookup Wizarda

Relaciju između tabele *TIspti* i tabele *TRezultatiIspita* uspostavićemo na sledeći način. U prozoru *Relationship* selektujemo polje *IspitID* u tabele *TIspti* i prevučemo ga na isto polje u tabeli *TRezultatiIspita*. Otvoriće se prozor kao na slici Sl.22.



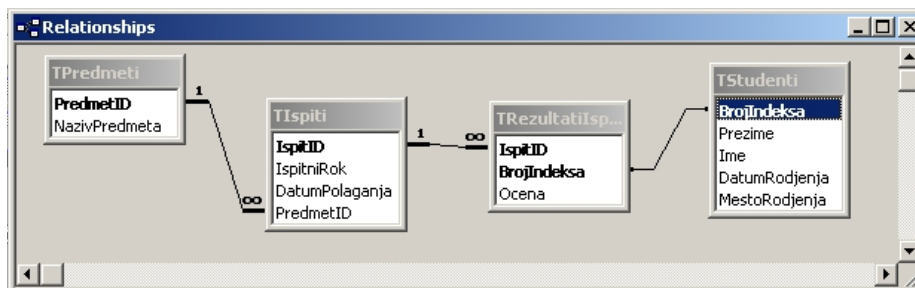
Sl. 22 Prozor za podešavanje svojstava relacije

Prozor koji se otvorio služi za dodatno podešavanje osobina relacije ove dve tabele. Sa 1 su označene tabele između kojih se ta relacija formira. Polja preko kojih se uspostavlja ove veza su označena sa 2. Opcijama 3 se obezbeđuje dodatni integritet podataka. Ukoliko bi postojala mogućnost da se šifra ispita (*IspitID*) u tabeli *TIspti* promeni tada bi smo morali, da bi smo sačuvali vezu između podataka, da prođemo kroz tabelu *TRezultatiIspita* i da i u njoj promenimo odgovarajuću šifru. U tabeli sa nekoliko hiljada zapisa to bi mogao biti veliki posao. Ako štikliramo prvu opciju (*Cascade Update Related Fields*) nalažemo Accessu da, pri promeni podatka u polju *IspitID* tabele *TIspti*, automatski ažurira odgovarajuće polje u tabeli *TRezultatiIspita*. U našem slučaju nismo uključili ovu prvu opciju jer vrednost podatka u polju *IspitID* popunjava sam Access (*AutoNumber*) i ne može se menjati. Druga opcija (*Cascade Delete Related Fields*) obezbeđuje vezano brisanje podataka iz ove dve tabele. Recimo, ako izbrišemo podatke o nekim ispitima iz tabele *TIspti*, ukoliko ova opcija nije uključena, ostali bi nam podaci o tim ispitima u tabeli *TRezultatiIspita*. Npr. imali bi smo situaciju da su studenti položili neke ispite koji ne postoje. Da bi se sprečila ova mogućnost uključujemo drugu opciju.

Pod rednim brojem 4 označen je tip relacije. U ovom primeru to je relacija "jedan prema više", tj. jednom zapisu u tabeli *TIspti* odgovara više zapisa (zavisno koliko je studenata polagalo taj ispit) u tabeli *TRezultatiIspita*.

Zbog mogućnosti da se kasnije izmeni šifra predmeta (*PredmetID*) u tabeli *TPredmeti*, a da bi smo sačuvali vezu sa podacima u tabeli *TIspti*, malo ćemo korigovaćemo relaciju između ove dve tabele. U prozoru *Relationships* desnim tasterom kliknite na liniju koja ih spaja i iz ponuđenog menija izaberite *Edit Relationship...*. Otvoriće se prozor kao na slici Sl.22. Uključite prvo opciju *Enforce Referential Integrity* a onad i opciju *Cascade Update Related Fields*. Time obezbeđujemo automatsku zamenu šifre predmeta u tabeli *TIspti* ukoliko to sami uradimo u tabeli *TPredmeti*.

Konačan izgled relacija u bazi dat je na slici Sl.23. Sada možemo da zatvorimo prozor *Relationships*.



Sl. 23 Relacije među tabelama u bazi EvidencijaIspita

Unos podataka u tabelle

Pre nego što kreiramo formulare za rad sa podacima, unecemo neke podatke direktno u tabelle. U prozoru baze na traci *Objects* selektujete *Tables* (Sl.6a) a onda izaberite tabelu *TPredmeti*. Pritiskom na dugme *Open* otvaramo ovu tabelu u edit modu (ažuriranje podataka u tabeli). Nakon unosa podataka tabela bi trebala da izgleda kao na slici Sl.24.

PredmetID	NazivPredmeta
0001	Matematika
0002	Fizika
0003	Statika
0004	Mašinski materijali
0005	Otpornost materijala

Sl. 24 Tabela TPredmeti sa unetim podacima

Unos podataka u tabelu prilično je intuitivan, nakon unetog podatka u nekom polju pritiskom na *Enter* (*Tab*, *strelice*) prelazite u novo polje ili red. Interesantno je primetiti, da pri prelasku na novi zapis nije potrebno da snimate tekući, Access će to sam uraditi. Ukoliko pri unosu podataka uočite neku grešku (npr. u nazivu predmeta) jednostavno se pozicionirajte na to polje, pritisnite taster *F2*, i korigujte uneti sadržaj. Da bi ste izbrisali neki zapis iz tabelle prvo je potrebno da taj zapis selektujete. To ćete uraditi tako što pomerite pokazivač miša na zaglavlje odgovarajućeg reda (r.b.1), pokazivač miša dobija oblik horizontalne crne strelice, i onda jednom kliknete. Nakon toga pritiskom na taster *Del* pruža Vam se mogućnost da zapis izbrišete. Naime, pre stvarnog brisanja Access će tražiti da potvrdite brisanje. Kako smo po polju *PredmetID* definisali primarni ključ tabelle, ako pokušate da unesete predmet sa šifrom koja već postoji, pojaviće se poruka upozorenja (Sl.25) da takva šifra postoji i Access Vam neće dozvoliti unos takvog zapisa. Ova poruka će se javiti u trenutku kada Access bude pokušao da apdejtuje zapis, odnosno u trenutku kada pokušate da pređete na drugi zapis. Moraćete ili da poništite unos (taster *Esc*) ili da promenite šifru predmeta.



Sl. 25 Poruka upozorenja koja se javlja kada pokušate da u tabelu unesete zapis sa istom vrednošću podatka u polju primarnog ključa

Znak "+" na levom kraju zapisa (rb.2) označava da je data tabela u relaciji sa nekom drugom (u ovom slučaju sa

TIspiti). Pritiskom na taj znak možete otvoriti vezanu tabelu (o ovome ćemo nešto kasnije).

Na sličan način popunite i tabelu *TStudenti* (ovde studenti mogu uneti svoje podatke). Izgled tabele sa unetim podacima dat je na slici Sl.26.

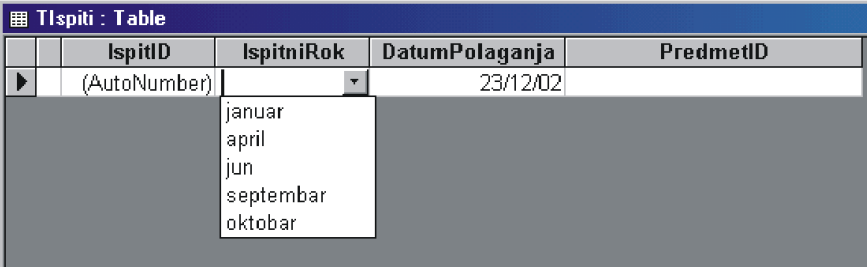


	BrojIndeksa	Prezime	Ime	DatumRodjenja	MestoRodjenja
+	11/80	Antić	Zoran	5/3/60	Požega
+	14/80	Jović	Jasmina	12/4/61	Trstenik
+	16/80	Marković	Drinka	24/12/60	Kraljevo
+	18/80	Petrović	Petar	1/7/60	Kraljevo
+	20/80	Tomić	Milan	15/6/60	Užice
▶	***/**				

Record: 6 of 6

Sl. 26 Tabela *TStudenti* sa unetim podacima

Sada kada smo uneli podatke u pomoćne tabele to možemo uraditi i u tabeli *TIspiti*. Otvaramo tabelu u edit modu (Sl.27).



	IspitID	IspitniRok	DatumPolaganja	PredmetID
▶	(AutoNumber)	januar april jun septembar oktobar	23/12/02	

Sl. 27 Unos podataka u tabelu *TIspiti*

U polju *IspitID* nemojte ništa upisivati, taj podatak unosi sam Access (*AutoNumber*). Podatak o ispitnom roku (*IspitniRok*) unosimo tako što otvaramo padajuću listu i iz nje biramo mesec. U polju *DatumPolaganja* već je upisan tekući datum jer smo pri kreiranju tabele za podrazumevanu vrednost ovog polja definisali (=Date()) odnosno tekući datum. Ukoliko datum odgovara jednostavno pritisnite *Enter*, a ako ne, ukucajte vrednost. Šifru predmeta (*PredmetID*) možemo uneti otvaranjem padajuće liste kao i kod ispitnog roka ili na sledeći način. Kada se kursor nađe u tom polju počnite sa kucanjem slova *m* Access će ponuditi da sam dovrši podatak (tj. matematika) i Vi jednostavno pritisnite *Enter*.

Ako sada kliknemo na znak "+" na levom kraju prvog zapisa otvara se odgovarajuća vezana tabela. U ovom slučaju to je tabela *TRezultatiIspita* u koju sada možemo uneti rezultate ispita kao što je to prikazano na slici Sl.28.

TIspiti : Table			
IspitID	IspitniRok	DatumPolaga	PredmetID
9	januar	24/1/82	Matematika
BrojIndeksa	Ocena		
11/80	7		
14/80	5		
16/80	8		
18/80	8		
20/80	9		
	0		
*	(A) 11/80	Antić	Zoran
	14/80	Jović	Jasmina
	16/80	Marković	Drinka
	18/80	Petrović	Petar
	20/80	Tomić	Milan

Sl.28 Prikaz vezanih tabela

Na ovaj način i vizuelno uspostavljamo vezu između podataka u ove dve tabele. Naravno, rezultate ispita možemo uneti i tako što zasebno otvorimo tabelu *TRezultatiIspita* (Sl.29).

TRezultatiIspita : Table			
IspitID	BrojIndeksa	Ocena	
9	11/80	7	
9	14/80	5	
9	16/80	8	
9	18/80	8	
9	20/80	9	
		0	

Record: 6 of 6

Sl. 29 Tabela *TRezultatiIspita* sa unetim podacima

Upiti

Osnovni zadatak jedne aplikacije zasnovane na bazi podataka je da nam omogući da brzo i lako dodjemo do određenih informacija na osnovu podataka koji se čuvaju u tabelama. Npr. interesuje nas kakva je prolaznost iz određenog predmeta ili koja je prosečna ocena studenta. Takve informacije se ne čuvaju eksplicitno u bazi ali se do njih može doći na osnovu podataka iz tabela. Upiti (queries) upravo služe za prikazivanje specifičnih informacija na osnovu unetih podataka u tabelama. Upiti predstavljaju onu alatku zbog kojeg se sistemi za rada sa bazama podataka (DBMS) najviše koriste. Recimo, podatke bi mogli da čuvamo u Excel-ovim radnim listovima ili Word-ovim dokumentima ali ovi programi nemaju alatke za manipulaciju tim podacima. Tu se misli na ažuriranje, filtriranje, grupisanje, pretraživanje, računanje sa podacima. Pored toga moguće je povezivanje podataka iz više tabela, brisanje postojećih ili kreiranje novih tabela. Neke osnovne obrade podataka možete uraditi i u Excelu ali su u Access-u mogućnosti znatno veće.

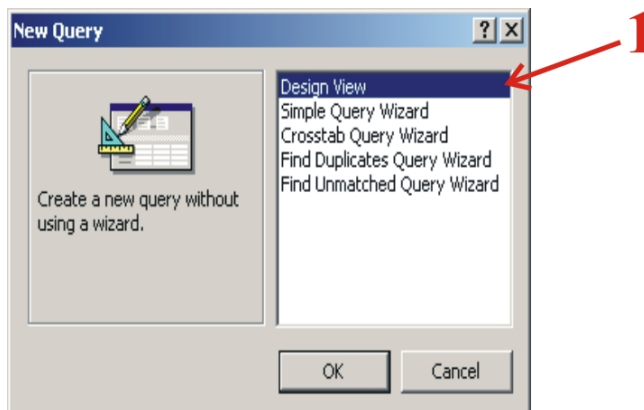
Upiti se najčešće koriste za kreiranje izvora podataka ("podloga") za obrasce, izveštaje, dijagrame ili druge upite. Obično su rezultati upita (skupovi zapisa - recordset) dinamički skupovi podataka - svaka izmena na podacima se manifestuje na izmenu podataka u tabelama. Osnovne vrste upita u Access-u su:

- Upiti za izbor (Select Query) - Ovo je najrasprostranjenija vrsta upita kojom se na neki način selektuju (radi pregleda i ažuriranja) podaci iz tabela. Podaci mogu da potiču iz jedne ili više tabela ili drugih upita.
- Unakrsni upiti (Crosstab Query) - Omogućavaju grupisanje podataka po određenim kategorijama u tabelarnom formatu. Pogodni su kao podloga za izveštaje i dijagrame.
- Akcioni upiti (Action Query) - Namenjeni su kreiranju akcija kojima se menjaju grupe podataka. Postoje četiri podgrupe ovih upita:
 - Upit za izradu tabela (Make Table Query) - Omogućava kreiranje novih tabela izborom polja iz postojećih.
 - Upit za ažuriranje (Update Query) - Služi za globalnu izmenu podataka u jednoj ili više tabela.
 - Upit za dodavanje (Append Query) - Dodaje zapise iz jedne ili više tabela na kraj druge (postojeće) tabele.
 - Upit za brisanje (Delete Query) - Briše podatke iz jedne ili više tabela u skladu sa navedenim uslovima.
- Kreiranje SQL upita. Ova vrsta upita se kreira u SQL dizajn modu (*SQL View*) koristeći standardan jezik za kreiranje upita - SQL (*Structured Query language*). To omogućava poznavacima SQL-a da novi upit kreiraju od početka ili da postojeći doraduju koristeći *SQL View*. U ovu grupu spadaju:
 - Upiti za kreiranje unija (Union Query). Kreiraju unije dva ili više skupa zapisa od srodnih polja.
 - Prolazni upiti (Pass-Through Query). Šalje komande direktno na SQL server. Time se sa tabelama radi

- direktno na serveru umesto da se previ link ka njima.
- Upit za definiciju podataka (data Definition Query). Slično akcionim upitima menja i kreira nove tabele u bazi.

Izrada, izvršavanje i izmena upita

Prvi upit koji ćemo napraviti treba da nam da informaciju o prosečnim ocenama studenata na osnovu podataka o položenim ispitima. U tu svrhu koristićemo grafičku alatku QBE (*query by example*) za lakše sastavljanje upita. U Access-u se upit može napraviti na dva načina: pomoću programskih modula poznatih pod imenom *Query Wizards*, koji nas korak po korak vode kroz postupak kreiranja upita, i "ručnim" pravljenjem od samog početka. Da bi ste napravili novi upit, u prozoru baze podataka na traci *Objects* izaberite *Queries* a onda na paleti sa alatima pritisnite na dugme *New* (kreiranje nove forme). Otvara se okvir za dijalog kao na slici Sl.30.



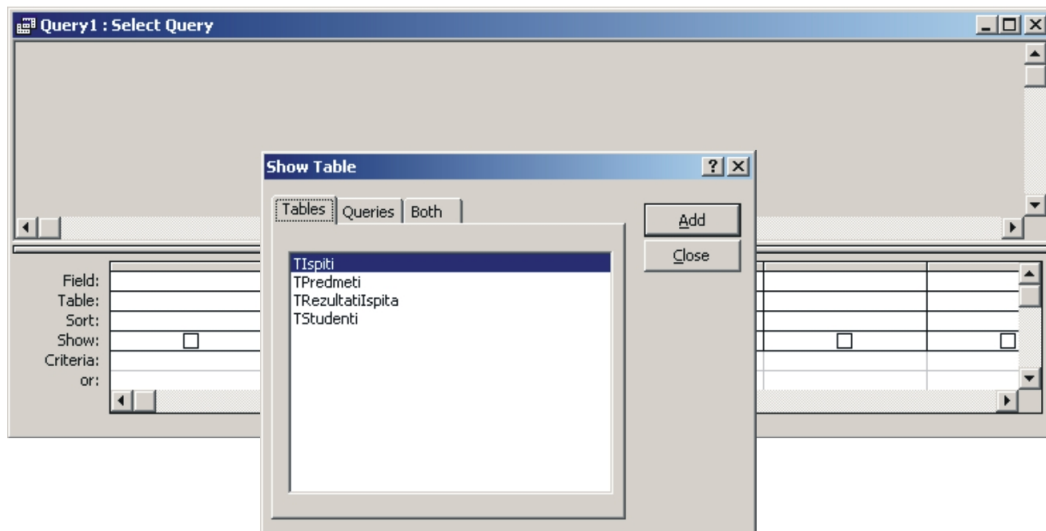
Sl.30 Izbor načina/vrste kreiranja upita

Značenje ponudjenih opcija je:

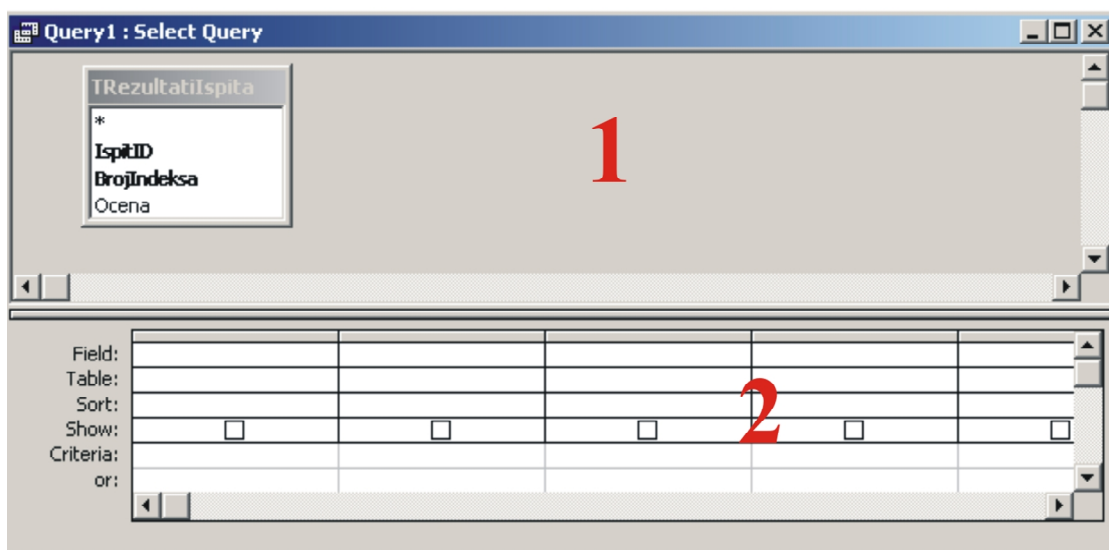
- *Design View* - Pokreće se ekran za ručno kreiranje upita.
- *Simple Query Wizard* - Ovo je standardan "čarobnjak" koji vas nizom pitanja vodi kroz postupak kreiranja upita.
- *Crosstab Query Wizard* - Služi za kreiranje istoimenih upita.
- *Find Duplicate Query Wizard* - Kreira upit koji pretražuje izabranu tabelu ili upit i nalazi zapise sa jednakim vrednostima u jednom ili više izabranih polja.
- *Find Unmatched Query* - Pronalazi zapise u jednoj tabeli kojima ne odgovara ni jedan zapis u drugoj tabeli. Npr. pomoću ovog Wizarda možete napraviti upit koji će naći sve studente koji nemaju ni jedan položen ispit.

Selektovanjem opcije *Design View* i pritiskom na dugme OK otvara se prozor za upit pod imenom *Select Query* (podrazumevani upit koji se pravi), a u njemu prozor *Show Table* (Sl.31).

U prozoru *Show Table* treba izabrati tabele/upite koje predstavljaju podlogu za novi upit koji kreiramo. Željeni objekat se selektuje a onda se pritiskom na dugme *Add* prebacuje u gornji deo prozora za upit. Pošto su nam za određivanje srednje ocene studenata potrebni podaci iz tabele *TRezultatiIspita* selektujemo tu tabelu a onda pritisnemo taster *Add*. Ako su Vam potrebne još neke tabele/upiti ponovite ovaj postupak. Isti rezultat se može postići i tako što se istovremeno selektuje više tabela/upita (tasterima *Shif* ili *Ctrl*) a onda pritisne dugme *Add*. Za sada, nisu nam potrebne druge tabele, pa ćemo postupak selektovanja tabela/upita završiti klikom na dugme *Close*. Nakon toga pojavice se prozor grafičke alatke za kreiranje upita kao na slici Sl.32.



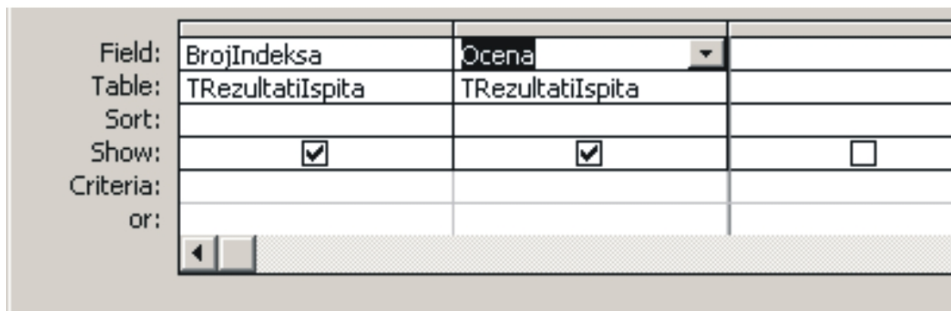
Sl. 31 Izbor tabela/upita koje predstavljaju podlogu za novi upit



Sl. 32 Prozor QBE alatke za kreiranje upita

Prozor se sastoji od dva dela. U gornjem delu (rb.1) prikazane su tabele koje čine podlogu upita. U donjem delu (rb.2) je tabela QBE u kojoj se pravi upit.

Sada je potrebno da odredimo koja polja (koje podatke) želimo da vidimo u rezultatu upita. U tabeli *TRezultatiIspita* selektujte polje *BrojIndeksa* i odvučite ga u prvu kolonu prvog reda (red *Field*) table QBE. Zatim i polje *Ocene* odvučite u drugu kolonu istog reda. Tabela QBE bi sada trebalo da izgleda kao na slici Sl. 33.

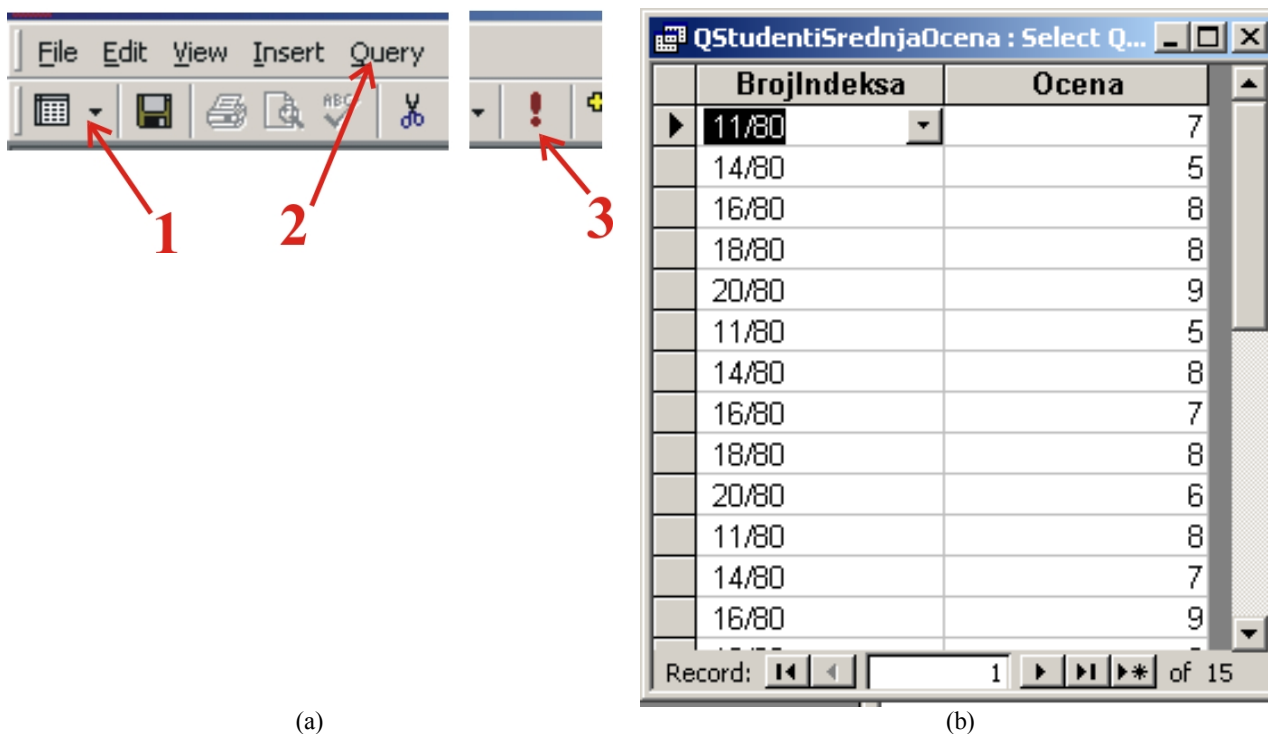


Sl.33 Definisane polja koja cine upit

Polja koja cine upit se mogu definisati i tako što se biraju iz liste koja se otvara kada se klikne u neko od polja reda *Field*.

Pre nego što pokrenemo (izvršimo) ovaj upit, snimite ga (meni *File->Save* ili pritiskom na taster *Save* na paleti sa alatima) pod imenom *QStudentiSrednjaOcena*.

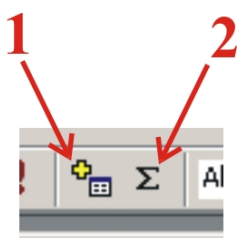
Pokretanje upita zavisi od stanja u kojem se on nalazi. Ako je zatvoren, u prozoru baze podataka treba prvo selektovati dati upit a onda ga pritiskom na dugme *Open* (Sl.6b) i otvoriti. Ako je upit u dizajn modu pokrece se tako što se na paleti sa alatima klikne na altku *View* ili alatku *Run* (Sl. 34 a) ili ako iz menija *Query* izabere komanda *Run*. Rezultati upita su prikazani na slici Sl. 34b (Rezultati vašeg upita se mogu razlikovati zavisno od podataka koji ste uneli u tabelu *TRezultatiIspita*).



Sl. 34 Pokretanje i rezultati upita *QStudentiSrednjaOcena*

Naravno, dobijeni rezultat nije ono što nama treba pa moramo gornji upit malo modifikovati. Prvo, informacije će biti potpunije ako pored broja indeksa piše ime i prezime studenta. Da bi smo izmenili upit treba ga prvo otvoriti u dizajn modu. Ako je upit zatvoren prvo ga u prozoru baze podataka selektujete pa pritisnite taster *Open* (Sl.6b). Ukoliko je upit otvoren u modu za prikaz (*View*) na paleti sa alatima klikne na altku *View* (Sl. 34 a).

Kako se podaci o studentima nalaze u tabeli *TStudenti* i tu tabelu moramo uključiti u definiciju upita. To ćemo uraditi tako što na paleti sa alatka kliknemo na taster *Show Table* (Sl. 35, rb.1) nakon čega se otvara istoimeni prozor (Sl.31)



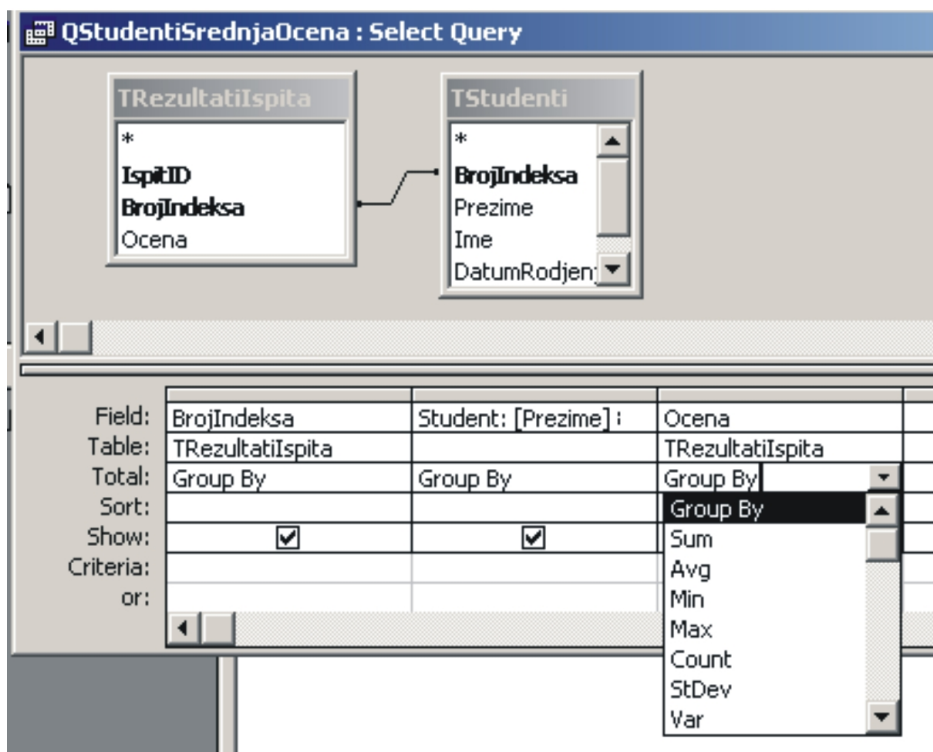
u kojem treba selektovati tabelu *TStudenti* i pritisnuti taster *Add*. Za sada je to jedina tabela koju dodajemo te zato pritisnite taster *Close*. Vraćamo se u prozor alatke QBE gde su sada prikazane dve tabele (*TRezultatiIspita* i *TStudenti*) sa ranije uspostavljenom relacijom izmedju njih. Da bi smo rezultat upita dopunili informacijom o imenu i prezimenu studenta mogli bi smo jednostavno da istoimena polja iz tabele *TStudenti* odvučemo u tabelu QBE. Medjutim, mi ćemo to ovde malo drugačije uraditi. Kliknite u polje *Ocena* (druga kolona tabele QBE) pa onda komandom *Columns* iz menija *Insert* ubacite novu kolonu izmedju kolona *BrojIndeksa* i *Ocena*. U prvom redu (*Field*) te nove kolone otkucajte sledeće:

Student: [Prezime] & " " & [Ime]

Sl. 35 Dodavanje novih tabela u upit

Na taj način mi samostalno definišemo polje u upitu koje se zove *Student* a koje se sastoji od polja *Prezime* i *Ime*. Da bi nam prezime studenta bilo odvojeno od imena izmedju ta dva polja umećemo blanko znak uokviren navodnicima.

Ako pokrenete ovaj upit rezultat će biti potpuniji nego u slučaju sa slike Sl. 34b, ali još uvek nemamo informaciju o srednjoj oceni. Da bi smo do nje došli, vratite se u dizajn mod upita i na paleti sa alatka kliknite na dugme *Totals* (Sl.35, rb2). Ova alatka nam omogućava da na neki način obradimo podatke i dodjemo do specifičnih informacija. Nakon njenog aktiviranja u tabeli QBE se pojavio novi red pod nazivom *Total*. U ovom redu, za svako polje upita, moguće je definisati načine obrade podataka. Podrazumevano se nudi grupisanje podataka (*Group by*). Rezultat takvog upita bi bio grupisan spisak studenata sa ocenama. Ako kliknete u odgovarajuće polje kolone sa ocenama (Sl.36) otvara se padajuća lista sa ponudjenim opcijama za obradu podataka. Pošto nam treba srednja ocena izaberite funkciju *Avg*. Ova funkcija izračunava srednju vrednost u polju *Ocena*, u odnosu na polja po kojima se izvršeno grupisanje. Drugim rečima, ovaj upit će uraditi sledeće: grupisaće sve zapise kod kojih su vrednosti u poljima *BrojIndeksa*, *Prezime* i *Ime* iste, a za svaku od tih grupa u polju *Ocena* izračunavaće srednju vrednost.



Sl. 36 Definisane načina grupisanja/obrade podataka u upitu

Rezultat ovog upita je spisak studenata sa njihovim prosečnim ocenama. Iako su to informacije do kojih smo želeli doći pomoću upita još malo ćemo korigovati ovaj upit.

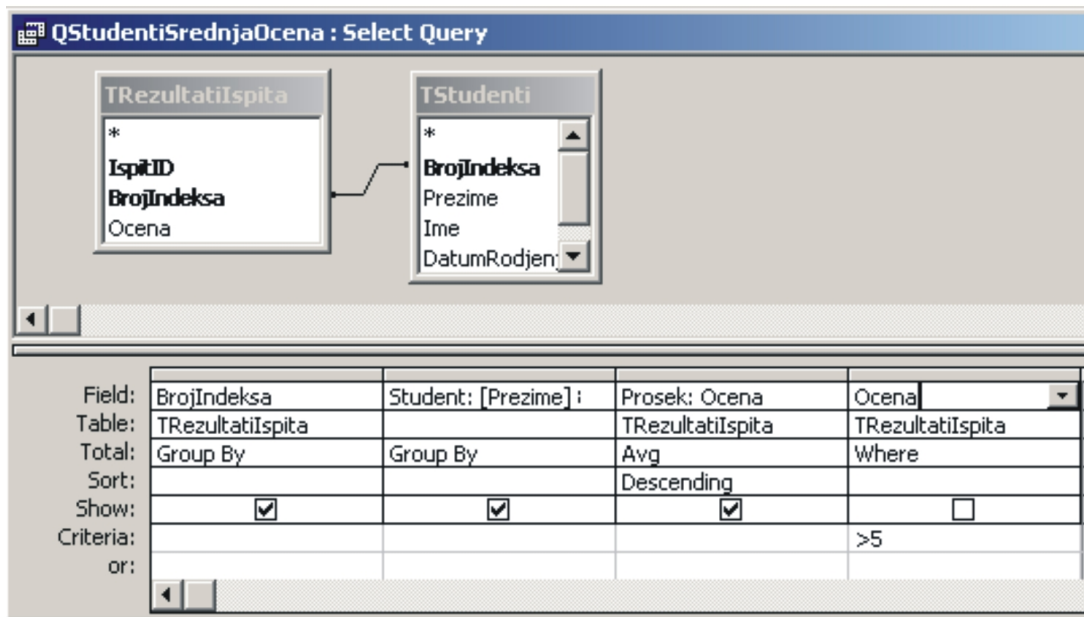
Kako će sada u koloni *Ocena* biti prosečne ocene za svakog studenta promenićemo naziv ovoj koloni u *Prosek*. Kliknite u to polje ispred teksta *Ocena* i dopišite *Prosek:*, tako da u njemu piše: **Prosek: Ocena**

U tabeli za definisanje upita postoji red sa labelom *Sort*. Pomoću ovog reda definišemo redosled pojavljivanja podataka u rezultatu upita. Ako želimo da nam spisak počne sa studentima koji imaju najveći prosek kliknemo u tu kolonu (u redu *Sort*) i iz padajuće liste biramo *Descending*. Slično, ako želimo sortiranje spiska u obrnutom redosledu (u odnosu na prosečnu ocenu) iz liste biramo *Descending*.

Red sa labelom *Show* u tabeli QBE nam omogućava da definišemo polja koja će se videti u rezultatu upita. Možda se pitate zašto uključivati polja koja ne želimo da se vide. Takva polja nam obično služe za definisanje negog kriterijuma u upitu. Ovo će Vam biti jasnije nakon objašnjenja reda *Criteria* iz tabele QBE.

Red sa ovom oznakom se koristi za definisanje kriterijuma pojavljivanja/nepojavljivanja zapisa u rezultatu upita. Recimo ako nas interesuje prosečna ocena određenog studenta čiji broj indeksa znamo u koloni *BrojIndeksa* u redu *Criteria* treba ukucati *tačno* taj broj indeksa. To znači ako ste pri unosu podataka u tabelu *TStudenti* za nekog studenta ukucali broj indeksa :14/03 (gde je : oznaka za blanko) to isto sada treba ukucati u polju za kriterijum. Drugim rečima ako u polje za kriterijum unosite redom brojeve indeksa :14/03, 1:4/03 ili 14:/03 rezultati upita će biti sasvim različiti (jer su za računar to različiti brojevi). Pri definisanju kriterijuma za selekciju možete koristiti i džoker znake kao što su: ?, *, #, !, -. Na primer da bi smo prikazali srednje ocene samo studenata koji su fakultet upisali 2003. godine u polju za kriterijum treba ukucati: */03. (Access će korigovati upis tako da će u polju pisati *Like "*/03"*). Ako želimo da u rezultatu upita vidimo samo studente čija je prosečna ocena veća ili jednaka od sedam, u koloni *Prosek* u redu *Criteria* treba ukucati ≥ 7 .

Razmotrimo sada sledeći problem. Korišćenjem funkcije *Avg* u redu *Total* Access računa srednju vrednost svih ocena koje je jedan student dobio. To jest, ako je student dobio 5 iz nekog predmeta i ta ocena će ući u prosek. Kako da mu kažemo da ne želimo da ocena 5 ulazi u prosek? To ćemo uraditi na sledeći način. U tabelu za definisanje upita još jednom odvućete polje *Ocena* iz tabele *TRezultatiIspita*. U redu *Total* te kolone umesto *Group by* iz liste izaberite *Where* (polje *Show* te kolone više nije štiklirano - ta kolona nam služi samo za definisanje kriterijuma), a u redu *Criteria* otkucajte >5 . Konačno upit *QStudentiSrednjaOcena*, u dizajn modu, bi trebalo da izgleda kao na slici Sl. 37.



Sl. 37 Upit *QStudentiSrednjaOcena* u dizajn modu

Ako sada pokrenete upit videćete da su pri sračunavanju srednje ocene uzete u obzir ocene koje su veće od 5. Snimite i zatvorite ovaj upit.

Još nešto ćemo reći oko sortiranja i definisanja kriterijuma kod upita. Ponavljajući postupak iz gornjeg upita napravite novi pod imenom *QStudenti*, na osnovu tabele *TStudenti*, prevlačeći redom sva njena polja u tabelu QBE. Pokretanjem upita dobija se spisak studenata onim redom kojim su unošeni u tabelu *TStudenti*. Ako želimo da nam spisak bude sortiran po mestu rođenja tada u redu *Sort* kolone *MestoRodjenja* sa liste izaberite *Ascending*. Ali šta ako želimo da nam spisak bude prvo sortiran po mestu rođenja, a ako imamo dva ili više studenata sa istim mestom rođenja, da nam se spisak onda sortira i po prezimenu. Ako bi smo jednostavno u redu *Sort* kolone *Prezime* izabrali *Ascending* ne bi smo dobili ono što želimo. Pošto je kolona *Prezime* levo od kolone *MestoRodjenja* u tabeli QBE, Access će izvršiti prvo sortiranje po prezimenu pa tek onda po mestu rođenja. Mi želimo upravo suprotno. Da bi smo to postigli treba zameniti

redosled kolona. Pomerite pokazivač miša iznad kolone *MestoRodjenja*. Kada se pojavi puna crna strelica levim klikom selektujete tu kolonu. Sada tu kolonu mišem odvučite ispred kolone *Prezime*. Naravno, željeni raspored kolona možete dobiti tako što izbrišete postojeće (nakon selektovanja) i ponovo ih prevučete u željenom redosledu. Tabela QBE ovog upita bi trebalo da izgleda kao na slici Sl.38.

Field:	BrojIndeksa	MestoRodjenja	Prezime	Ime	DatumRodjen
Table:	TStudenti	TStudenti	TStudenti	TStudenti	TStudenti
Sort:		Ascending	Ascending		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:					
or:					

Sl. 38 Definisane redosleda sortiranja u upitu

Pri definisanju kriterijuma za selektovanje zapisa u upitu možete koristiti logičke operatore I, ILI i NE za formiranje složenijih kriterijuma. Ako na primer želimo da prikazemo samo studente iz Kraljeva tada u koloni *mestoRodjenja* u redu *Criteria* treba otkucati *Kraljevo*. Da bi smo prikazali sve studente sem onih iz Kraljeva u istom polju treba otkucati *Not Kraljevo*. Ako želimo da prikazemo studente iz Kraljeva *ILI* iz Trstenika tada u redu *Criteria* otkucajte *Kraljevo*, a *Trstenik* otkucajte u polju odmah ispod Sl.39a. Da di smo prikazali studente koji su iz Kraljeva I koji se prezivaju Petrović u redu *Criteria* kolone *MestoRodjenja* otkucajte *Kraljevo* a koloni *Prezime* istog reda otkucajte *Petrovi?* (Sl.39b). Ovdje smo naše slovo *ć* sa kojim se upiti ne snalaze baš najbolje zamenili džokerom ?.

Field:	BrojIndeksa	MestoRodjenja	Prezime	Ime
Table:	TStudenti	TStudenti	TStudenti	TStudenti
Sort:		Ascending	Ascending	
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		"Kraljevo"	"Kraljevo"	Like "Petrovi?"
or:		"Trstenik"		

Sl. 39 Definisane kriterijuma za selektovanje zapisa u upitu

Znači kada definišete više kriterijuma tako da moraju biti svi ispunjeni (povezani logičkim I) kucajte ih u istom redu. Ukoliko je potrebno da bude ispunjen barem jedan od kriterijuma (povezani logičkim ILI) kucajte ih u narednim redovima (or).

Tako na primer za kriterijum prikazan na slici Sl.40 upit će nam dati studente iz Kraljeva koji se prezivaju Petrović i sve studente iz Trstenika bez obzira kako se prezivaju.

Field:	BrojIndeksa	MestoRodjenja	Prezime	I
Table:	TStudenti	TStudenti	TStudenti	T
Sort:		Ascending	Ascending	
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Criteria:		"Kraljevo"	Like "Petrovi?"	
or:		"Trstenik"		

Sl. 40 Definisane kombinovane kriterijume u upitu

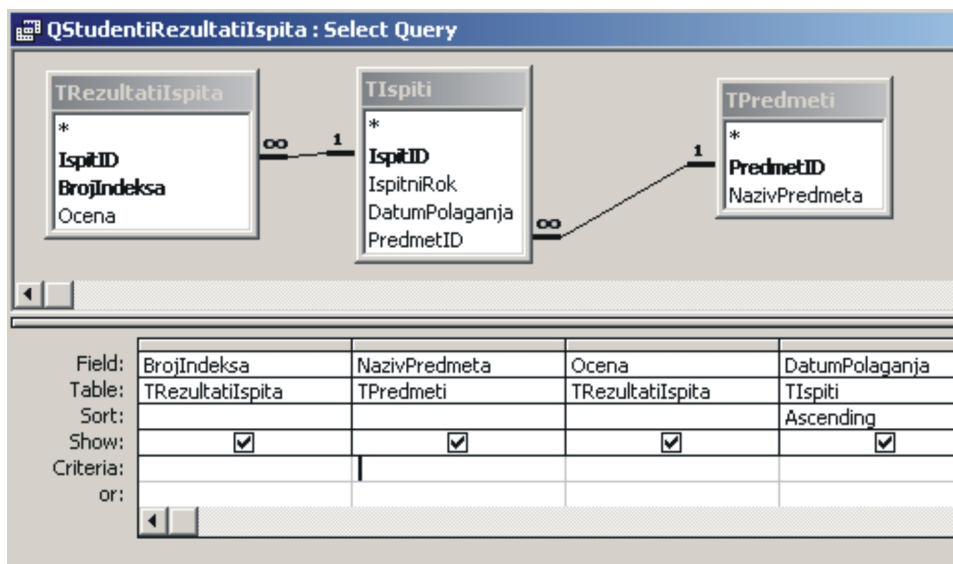
Novi upit koji nam je potreban treba da nam omogući lakše vođenje evidencije o položenim ispitima. Napravićemo ga tako što ćemo napraviti kopiju postojećeg, a onda tu kojiju uzmeniti. U prozoru baze podataka selektujte upit *QStudentiSrednjaOcena* i na paleti sa alatkama kliknite na taster *Copy*. Kada zatim pritisnete na taster *Paste* pojavice se okvir za dijalog gde je potrebno da unesete ime novog upita kojeg kreirate. Unesite ime *QRezultatiIspita* pa potvrdite unos klikom na dugme *OK*. Na listi upita u prozoru baze podataka pojavice se taj novi upit. Otvorite ga u dizajn modu. Pred Vama ce se otvoriti prozor kao na slici Sl. 37. Sada ćemo ga malo izmeniti da bi smo dobili definiciju novog upita. Prvo isključite alatku *Totals* (Sl.35 rb.2) a zatim izbrišite kolonu *Prosek* (prvo je selektujete pomeranjem pokazivača miša iznad nje). Iz kolone *Ocena* uklonite kriterijum (>5) i uključite njenu opciju *Show*. I na kraju iz tabele *TRezultatiIspita* u tabelu upita prevucite polje *IspitID*. Upit bi trebalo da izgleda kao na slici Sl.41.

QRRezultatiIspita : Select Query

Field:	BrojIndeksa	Student: [Prezime] :	Ocena	IspitID
Table:	TRezultatiIspita		TRezultatiIspita	TRezultatiIspita
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:				
or:				

Sl. 41 Definicija upita QRRezultatiIspita

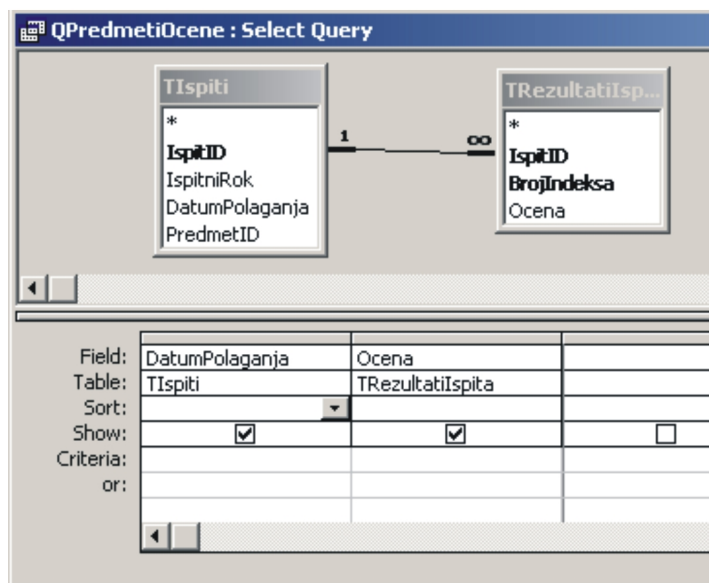
Pošto želimo da o svakom studentu imamo evidenciju o položenim ispitima napravićemo upit *QStudentiRezultatiIspita* čija je definicija data na slici Sl. 42. Koristeći znanje stečeno u kreiranju prethodnih upita pokušajte da napravite sami ovaj upit.



Sl. 42 Definicija upita QStudentiRezultatiIspita

Još ćemo nešto reći o jednoj vrsti upita koja se često sreće u aplikacijama vezanim za rad sa bazama podataka. To su takozvani parametarski upiti. Recimo da nas interesuju ocene iz pojedinih predmeta u toku prethodnih 365 dana. U tu svrhu napravićemo novi upit pod imenom *QPredmetiOcena*.

U prozoru baze podataka, na listi *Objects*, izaberite *Queries* a onda kliknite na dugme *New*. U prozoru za dijalog koji se nakon toga otvara izaberite tabele *TIspiti* i *TRezultatiIspita* kao podlogu za novi upit. Nakon toga, u prozoru za kreiranje upita, u tabelu QBE prevucite polje *DatumPolaganja* (iz tabele *TIspiti*) i polje *Ocena* (iz tabele *TRezultatiIspita*). Izgled upita je dat na slici Sl.43.



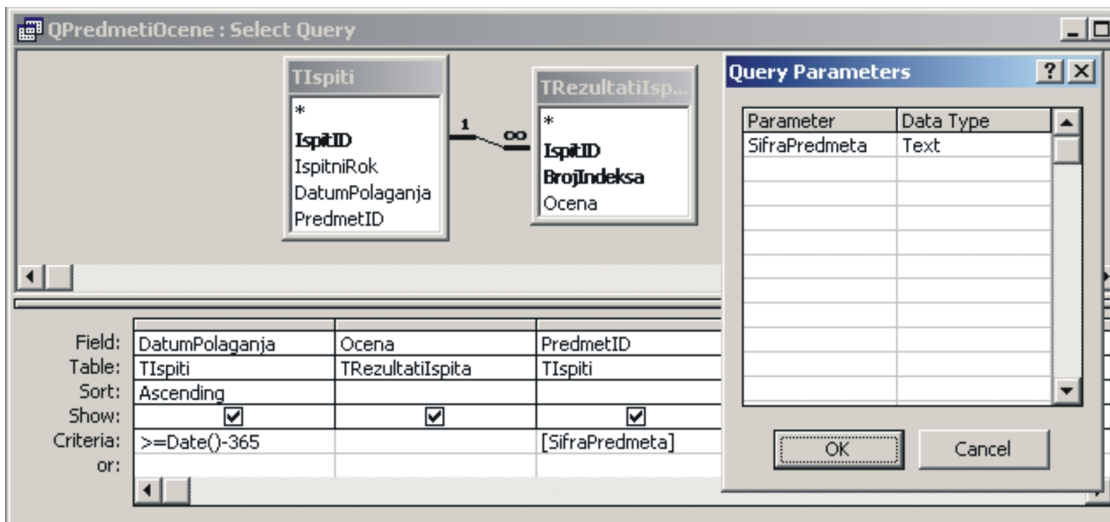
Sl. 43 Kreiranje upita QPredmetiSrednjaOcena

Pošto nas interesuju samo ocene dobivene tokom zadnjih 365 dana (u odnosu na tekući datum) u koloni *DatumPolaganja* u redu *Criteria* otkucajte sledeći uslov:

>=Date()-365

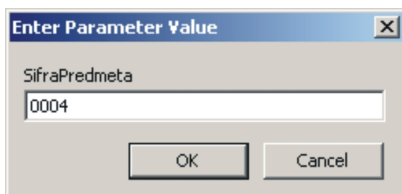
Za definisanje ovog kriterijuma koristimo ugradjenu funkciju *Date()* koja vraća tekući (današnji) datum. Znači od tekućeg datuma oduzimamo 365 dana i u rezultatu upita prikazujemo samo one ispite koji su jednaki ili mlađji od tog datuma. Da bi rezultati ispita bili sortirani u rastućem redosledu u polju *Sort* kolone *DatumPolaganja* izaberite *Ascending*.

Trenutna definicija ovog upita prikazaće nam rezultate polaganja svih ispita godinu dana unazad. Da bi smo mogli da biramo predmet iz kojeg želimo da vidimo rezultate u tabelu QBE odvućite i polje *PredmetID*. Sada u polju *Criteria* ove kolone otkucajte šifru predmeta (npr. "0004") za koji želite da vidite rezultate polaganja. Pokretanjem upita dobijaju se samo rezultati za željeni predmet. Ako želimo da vidimo rezultate za drugi predmet promenićemo sadržaj polja *Criteria* kolone *PredmetID*. Problem sa ovako definisanim upitom je što budući korisnik aplikacije mora da ulazi u dizajn mod ovog upita i tamo "ručno" menja šifru ispita, svaki put kada želi da promeni vrednost kriterijuma. Da bi smo olakšali korišćenje ovog upita, umesto fiksne vrednosti koristimo parametar. U polju *Criteria* kolone *PredmetID* otkucajte (u uglastim zagradama) [*SifraPredmeta*]. Zatim u meniju *Query* izaberite *Parameters* nakon čega se otvara okvir za dijalog *Query Parameters* u kojem je potrebno definisati parametre upita. U koloni *Parameter* otkucajte ponovo *SifraPredmeta* (sada bez uglastih zagrada) a u koloni *Data Type* sa liste izaberite *Text* kao tip podatka za ovo polje (tako smo ga definisali kada smo pravili tabelu *TPredmeti*). Izgled popunjenog okvira za dijalog dat je na slici Sl.44.



Sl. 44 Definisanje parametara upita

Svaki put kada pokrenete ovaj upit otvoriće se prozor kao na slici Sl.45. Kada unesete šifru predmeta čije rezultate želite da vidite i pritisnete taster *OK* u rezultatu upita pojavaće se samo ocene iz tog predmeta.



Sl. 45 Unošenje vrednosti za parametar upita

Ono što je bitno kod ovih upita je da se vrednost kriterijum za selektovanje definiše u trenutku pokretanja a ne u toku kreiranja upita.

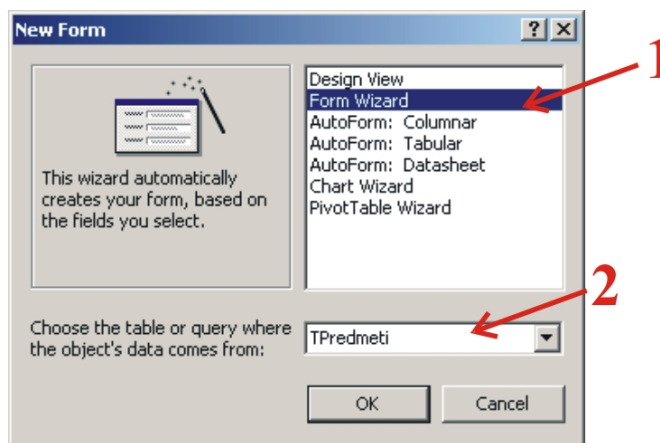
Kreiranje obrazaca

Neku bazu podataka možemo posmatrati iz barem dve perspektive. Iz perspektive onoga ko tu bazu kreira i iz perspektive onoga ko bazu koristi. Onaj ko razvija bazu mora da razmišlja o njenoj arhitekturi (tabelama, relacijama, integritetu podataka, pravima pristupa) i o njenoj tehničkoj realizaciji. Od njega se zahteva određeno poznavanje problematike i iskustvo. S druge strane, korisnik baze radi sa podacima iz baze (koristi bazu), bez potrebe da poznaje tehničke detalje baze. Drugim rečima, treba razdvojiti namenu baze (šta se želi postići bazom) od doga kako se to i postiže. Kada želite da pojačate jačinu zvuka na TV-u za Vas je bitno da znate kojim se tasterom na daljinskom upravljaču to postiže ali ne i kako se to postiže.

Obrasci (forme) nam upravo omogućavaju da razvijemo taj korisnički pogled na bazu - korisnički interfejs. Najčešće se koriste za ažuriranje (unos i izmenu) podataka koji se čuvaju u bazi. Zato podesćaju na papirne obrasce na kojima treba popuniti prazna mesta. Za razliku od tabelarnog prikaza gde su podaci prikazani u redovima i kolonama, na obrascu se podaci mogu prikazati u bilo kom formatu. Pored toga što omogućavaju korisniku da lakše pređe sa klasične na računarsku bazu, oni nam pružaju mogućnost da dodatno kontrolišemo unete podatke (npr. ocena mora da bude u intervalu 5-10), da za određena polja ponudimo podrazumevane vrednosti (npr. tekući datum), da se određeni podaci automatski unose (npr. da se na osnovu unetih ocena odredi srednja vrednost).

Osnovno je pravilo da elektronski obrazac treba što više da liči na papirne obrasce sa kojima je korisnik do sada radio. Obrasci treba da su skladno urađeni, bez mnogo upadljivih boja, slika, mnoštva fontova. Mada na prvi pogled to može biti efektno, posle dužeg rada to može delovati odbojno.

Prvi obrazac (forma) koji ćemo napraviti služi za unos podataka o predmetima koji se polažu na ispitima. U prozoru baze na traci *Objects* izaberite *Forms* a onda na paleti sa alatima pritisnite na *New* (kreiranje nove forme). Otvara se okvir za dijalog kao na slici Sl.46.

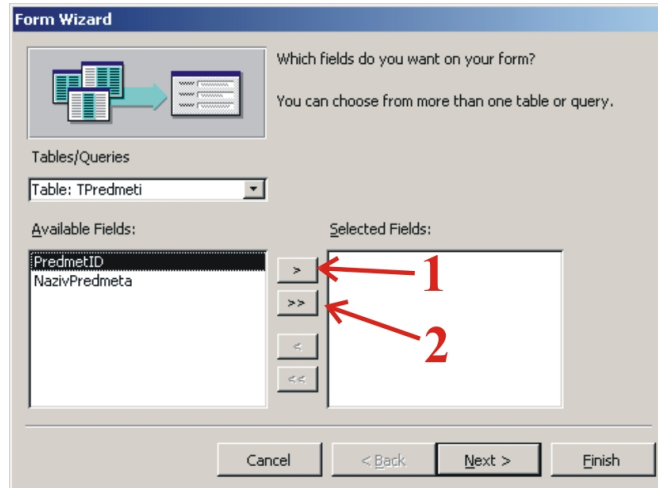


Sl.46 Izbor načina kreiranja forme

Prvo biramo način kreiranja forme. Ponudene su sledeće opcije:

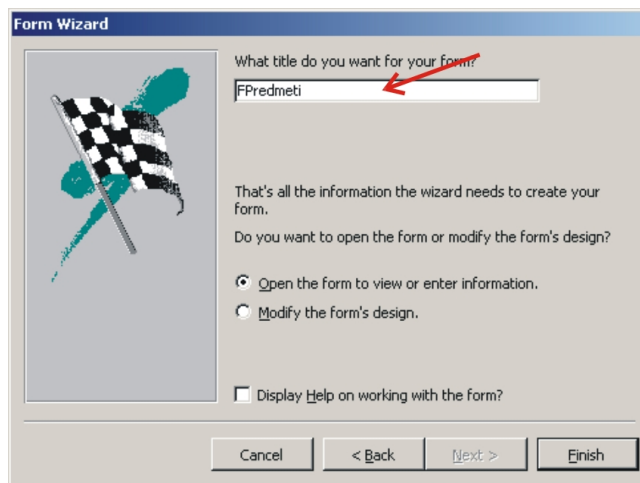
- Design View - dizajniranje forme "od nule" bez upotrebe čarobnjaka; sve opcije moramo samo podesiti;
- Form Wizard - čarobnjak koji na osnovu odgovora na postavljena pitanja deo opcija postavlja automatski, te sam raspoređuje polja za unos;
- AutoForm: (Columnar, Tabular, Datasheet) kreira formu bez ikakvih pitanja na osnovu tabele koju odaberemo; podopcije predstavljaju način na koji će polja za unos biti raspoređena na formu;
- Chart Wizard - čarobnjak koji kreira formu sa grafikonom na njoj (koristi se uslugama Microsoft Excela koji pri tome mora biti instaliran);
- PivotTable Wizard - čarobnjak koji iz podataka iz tabele kreira izvedenu tabelu (takođe zahteva Microsoft Excel);

Mi ćemo izabrati drugu opciju (*Form Wizard*). Druga stvar koju treba izabrati je tabela (upit) sa čijim podacima želimo da komuniciramo kroz datu formu. U donjem delu prozora, iz padajuće liste, izaberite *TPredmeti*. Pritiskom na dugme *OK* otvara se sledeći prozor kao na slici Sl.47.



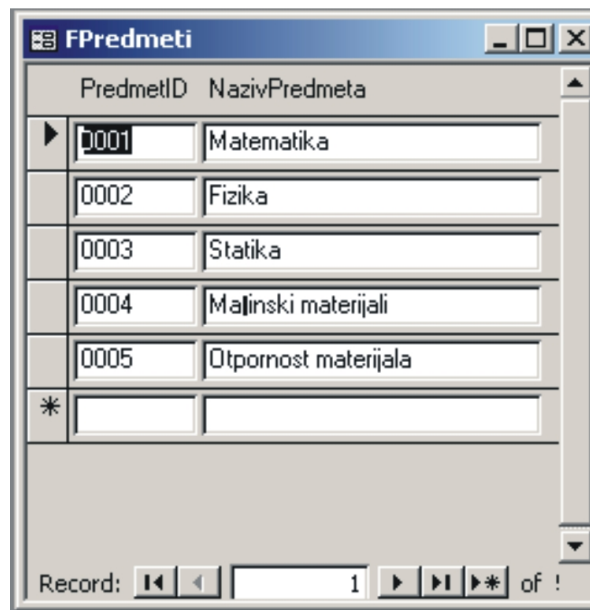
Sl. 47 Izbor polja tabele koja će se videti na formi

U njemu treba pojedinačno (rb.1) ili grupno (rb.2) izabrati polja iz tabele koja će se prikazati na formi. Pritiskom na dugme rb.2 prebacite sva polja. U sledećem koraku biramo raspored polja na formi. Izaberite drugu ponudjenu opciju (*Tabular*), pa pritiskom na dugme *Next* pređite na sledeći korak. Otvoriće se prozor gde Vam se pruža mogućnost da birate stil (format) forme koju želite na napravite. I tu ćemo prihvatiti ponudjenu opciju (*Standard*) nakon čega stižemo do zadnjeg koraka u kreiranju forme (Sl.48).



Sl. 48 Zadnji korak u kreiranju forme - izbor imena forme

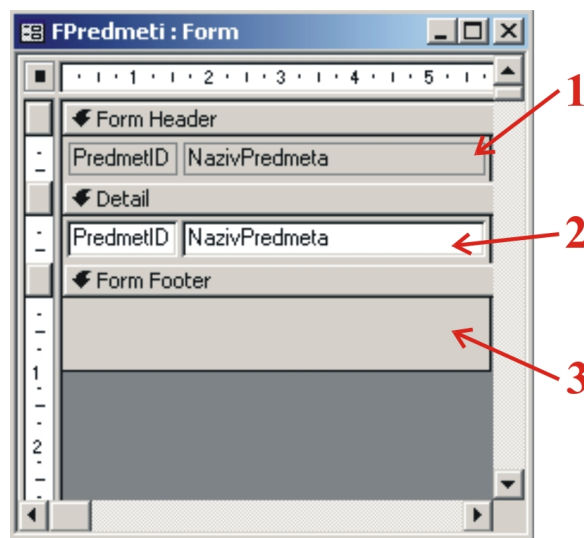
U polju na vrhu forme definišemo ime buduće forme. Ponudjenu opciju *TPredmeti* promenite u *FPredmeti* a ostale opcije neka ostanu kakve jesu. Pritiskom na dugme *Finish* završavamo proceduru kreiranja forme koja se posle izvesnog vremena otvara pred Vama (Sl.49).



Sl.49 Forma FPredmeti kreirana uz pomoć čarobnjaka

Upravo kreirana forma ima sve odlike Windows prozora tj., možete je pomerati, menjati veličinu, minimizirati, maksimizirati itd. Na formi su tabelarno prikazani ranije uneti podaci iz tabele *TPredmeti*. Zatvorite ovaj obrazac i proverite da li u prozoru baze među formama imate formu *FPredmeti*.

Kako forme koje kreirate pomoću čarobnjaka neće uvek ispuniti sva Vaša očekivanja, sada ćemo pogledati kako da formu naknadno doterate. U prozoru baze podataka selektujte formu *FPredmetii* a onda na paleti sa alatima pritisnite dugme *Design*. Izabrana forma se otvara u dizajn modu (Sl.50), modu koji nam omogućava njenu korekciju.



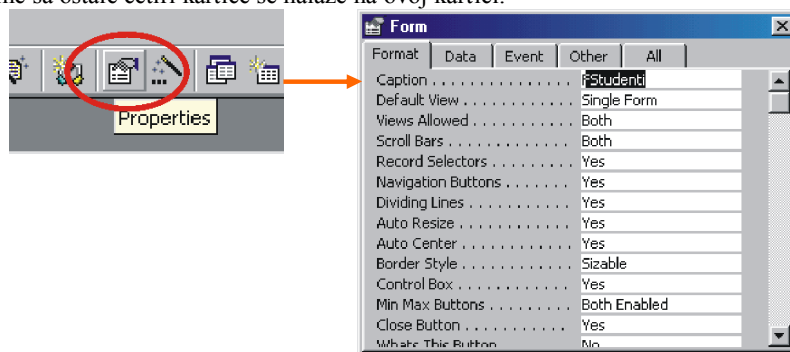
Sl.50 Forma FPredmetii u Design modu

Forma se sastoji iz tri dela: gornjeg (rb.1) i donjeg zaglavlja (rb.3) * (*Form Header*, *Form Footer*) i dela sa detaljima (*Detail*, rb.2). Visinu pojedinih delova forme možete menjati tako što pokazivač miša pomerite do podebljanih traka, koje te delove razdvajaju, sve dok pokazivač ne dobije oblik krstića. Držanjem levog tastera na mišu visinu podešavate po želji. U gornjem zaglavlju obično se daju labele koje bliže objašnjavaju funkciju forme. Deo sa detaljima najčešće sadrži sva polja iz tabele koja želite da vidite na formi. U donjem zaglavlju se mogu dati razna zbirna izračunavanja ili komandni tasteri.

* Ukoliko se u početnom prozoru ne vide gornje i donje zaglavlje uđite u Accessov meni *View* i uključite opciju *Form Header/Footer*.

Sve što vidite na jednoj formi u Accessu se označava terminom *kontrolni objekat* ili *kontrola*. Osobine kontrola sa forme možete pregledati i menjati uz pomoć *prozora sa svojstvima* (Sl. 51 b), koji se otvara tako što na Access-ovoj paleti sa alatima kliknete na alatku *Properties* (Sl. 51 a). Prozor sa svojstvima ima pet kartica na kojima su grupisane osobine objekta. Sadržaji kartica zavise od tipa izabranog objekta a namena im je sledeći:

- *Format* - podešavamo način na koji će na ekranu biti prikazani objekti (oblik, veličina, font, boje, itd;)
- *Data* - na ovoj kartici se podešava izvor podataka, mogućnost njegove izmene, kontrola unosa podataka u polje, osnovna vrednost i način ponašanja filtera nad poljem
- *Event* - na svakoj operaciji nad poljem aktivira se određeni događaj (Event) koji može pokrenuti program.
- *Other* - ostale osobine objekta koje ne spadaju niti u jednu od preostale četiri grupe.
- *All* - sve osobine sa ostale četiri kartice se nalaze na ovoj kartici.



(a)

(b)

Sl.51 Prozor sa svojstvima kontrola

Navođenje svih svojstava koja se mogu podešavati izlazi iz okvira ovog uputstva, te ćemo ovde navesti samo nekoliko njih. Da bi ste promenili svojstva neke kontrole (ili više njih) važi generalno pravilo: prvo je selektujte pa tek onda podešavajte njene osobine.

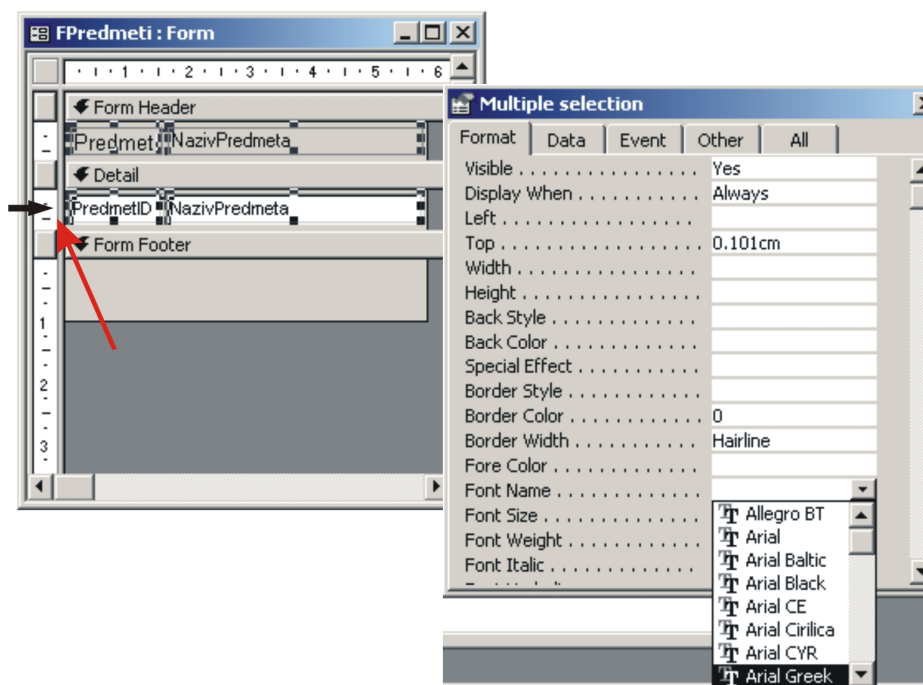
Svako polje iz tabele na formi je predstavljeno pomoću dva kontrolna objekta koji predstavljaju nedeljivu celinu: *Label*-e i *TextBox*-a. *Label* se obično koristi za ispis nekog statičnog teksta (opis, natpis, naslov itd.) na formi. *TextBox* služi za unos, ispravljanje i prikaz podataka. Najčešće odgovara jednom polju u tabeli nad kojom je forma napravljena. Razliku između ove dve kontrole najbolje ćete primetiti ako pogledate prikaze *Form View* (Sl.49) i *Design View* (Sl.50) jedne iste forme. Na primer, polju *BrojIndeksa* odgovara *Label*a i *TextBox* u kojima, u dizajn modu, piše *BrojIndeksa*. U *View* prikazu (meni *View* -> *Form View*) sadržaj *Label*e ostaje isti dok se sadržaj *TextBox*a popunjava odgovarajućim podatkom iz tabele *TPredmeti*.

Pre nego što promenimo neko od svojstava skrećemo pažnju na jedno svojstvo koje *svaka* kontrola *mora* da ima. To je svojstvo *Name* sa kartice *Other*. Tako npr. *Label*a sa tekстом *NazivPredmeta* nosi ime *NazivPredmeta_Label* dok je odgovarajući *TextBox* imenovan sa *NazivPredmeta*. Ime jednoznačno određuje svaku kontrolu tako da se unutar jedne forme ne mogu javiti dve kontrole sa istim imenom. Pri kreiranju forme Access sam automatski dodeljuje imena kontrolama tako da vi o tome ne morate posebno voditi računa. Uбудuće, sve kontrole ćemo referencirati pomoću njihovih imena.

Podešavanje osobina kontrola

Prvo ćemo promeniti tip slova kontrole *PredmetID_Label*. Nakon što je selektujete, u prozoru sa svojstvima (proverite da li na njegovoj naslovnoj traci piše *Label: PredmetID_Label*) potražite svojstvo *Font Name* (kartica *Format*) i iz padajuće liste izaberite *Arial*. Slično, pomoću svojstva *Font Size* promenite veličinu slova na 10. Postupak ponovite i za kontrolu *PredmetID* (*TextBox*).

I za ostale kontrole sa forme, koje odgovaraju poljima tabele, mogli bi smo da ponovimo iste korake. Međutim, posao možemo i da ubrzamo tako što prvo selektujemo sve kontrole a onda im grupno promenimo svojstvo *Font Name* i *Font Size* (Sl.52).

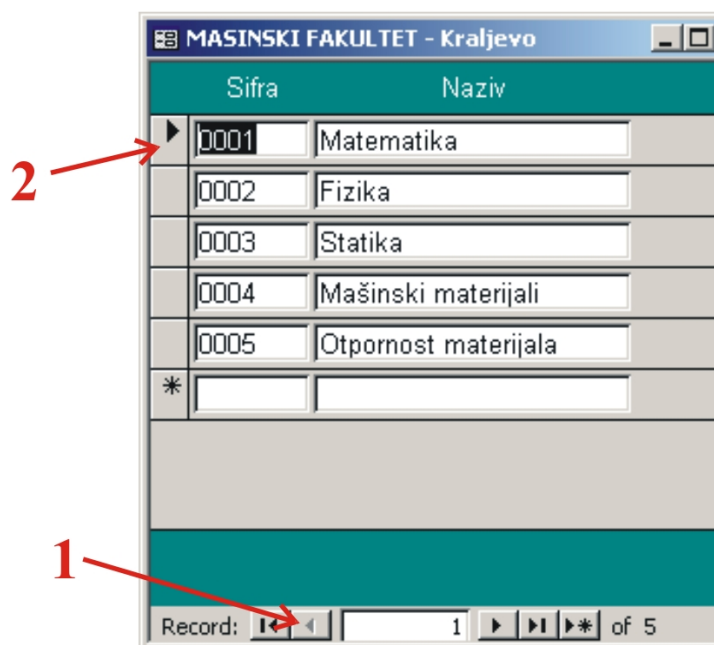


Sl.52 Grupna promena svojstva

Grupu kontrola možete da selektujete tako što ih selektujete jedan po jedan uz pritisnut taster *Shift*. Drugi način da ih selektujete je da pomerite pokazivač miša uz levu ivicu prozora, na vertikalni lenjir, pri čemu pokazivač dobija oblik horizontalne crne strelice (Sl.52), čijim se povlačenjem po vertikali selektuje grupa kontrola. Ostala svojstva promenite prema sledećoj tabeli.

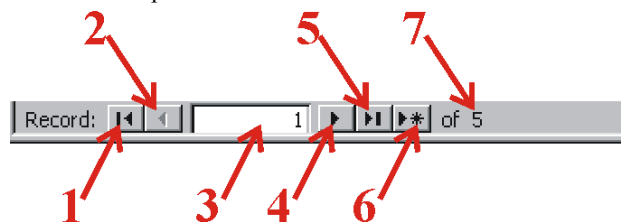
Kontrola	Ime	Svojstvo	Vrednost
Form	<i>FPredmeti</i>	<i>Caption</i>	MASINSKI FAKULTET - Kraljevo
FormHeader		<i>Back Color</i>	8421376
FormFooter		<i>Back Color</i>	8421376
Label	<i>PredmetID_Label</i>	<i>Caption</i>	Sifra
Label	<i>NazivPredmeta_Label</i>	<i>Caption</i>	Naziv
Label	<i>PredmetID_Label, NazivPredmeta_Label</i>	<i>Text Align</i>	Centar
Label	<i>PredmetID_Label, NazivPredmeta_Label</i>	<i>Back Color</i>	8421376
Label	<i>PredmetID_Label, NazivPredmeta_Label</i>	<i>Fore Color</i>	16777215
Label	<i>PredmetID_Label, NazivPredmeta_Label</i>	<i>Font Name</i>	Arial
Label	<i>PredmetID_Label, NazivPredmeta_Label</i>	<i>Font Size</i>	10
TextBox	<i>PredmetID, NazivPredmeta</i>	<i>Font Name</i>	Arial
TextBox	<i>PredmetID, NazivPredmeta</i>	<i>Font Size</i>	10

Posle ovih podešavanja snimite formu i zatvorite je. U prozoru baze pojavice se novoformirana forma *FPredmeti*. Ako je sada otvorite (taster *Open*) ona bi trebalo da izgleda kao na slici Sl.53.



Sl.53 Izgled forme FPredmetii

Pomoću ove forme sada možete pregledati, menjati, brisati i upisivati nove podatke u tabelu *TPredmeti*. Svaka novokreirana forma sadrži standardne kontrole: kontrole za navigaciju (r.b.1) i selektovanje (r.b.2) zapisa. Kontrolama za navigaciju (Sl.54) korisnik se kreće kroz zapise tabele.



Sl.54 Kontrola za navigaciju

Značenje pojedinih komandi na ovoj kontroli je sledeće:

1. Taster za skok na prvi zapis tabele
2. Taster za skok na prethodni zapis tabele
3. Polje sa relativnim pozicijom tekućeg zapisa tabele
4. Taster za skok na sledeći zapis tabele
5. Taster za skok na poslednji zapis tabele
6. Taster za skok na novi zapis
7. of 5 - ukupan broj zapisa

Druga kontrola (Sl.53 r.b. 2) služi za selektovanje tekućeg zapisa. Klikom levog tastera miša na bilo koji deo njene površine, Accessu nalažemo da selektuje ceo zapis nad kojim se mogu primeniti standardne operacije: Cut, Copy i Delete.

Da bi ste izmenili nešto u tekućem zapisu (neki podatak o predmetu) jednostavno se pozicionirajte u odgovarajuće polje i uradite izmenu. Za unos novih zapisa (novog predmeta) tasterima za navigaciju se pomerite na novi zapis (prazan red na kraju forme) i tu unesite podatke. Brisanje tekućeg zapisa će te obaviti tako što ga prvo selektujete (kontrola za selektovanje) a onda na tastaturi pritisnete taster *Delete*.

Za rad sa podacima na formi možete koristiti i tastere:

<i>Taster</i>	<i>Značenje</i>
TAB	sledeće polje
SHIFT+TAB	prethodno polje
END	poslednje polje sloga

HOME	prvo polje sloga
CTRL+END	poslednje polje poslednjeg sloga
CTRL+HOME	prvo polje prvog sloga
F2	editovanje sadržaja polja
ENTER	sledeće polje
ESC	poništanje unosa sloga
CTRL+F4	zatvara formu

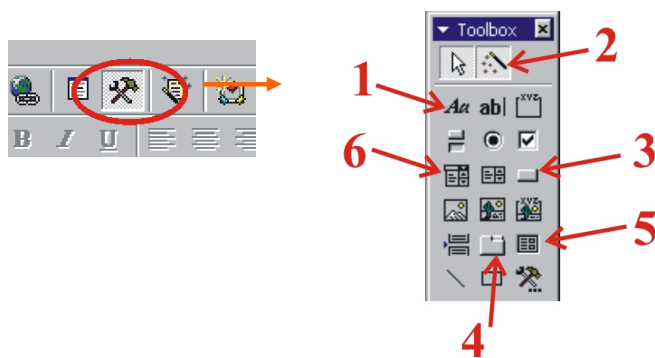
U modu editovanja polja mogu se koristiti sledeći tasteri:

<i>Taster</i>	<i>Značenje</i>
ENTER	Kraj editovanja skok na naredno polje
ESC	Poništavanje unosa u polje

Dodavanje novih kontrola na formu

Obično ćete poželeti da formu koju je napravio Wizard malo doterate i da joj ugradite dodatnu funkcionalnost. Otvorite obrazac *FPredmeti* u dizajn modu i povećajte visinu gornjeg zaglavlja. To ćete uraditi tako što pomerite pokazivač miša do sive trake koja razdvaja deo sa detaljima od gornjeg zaglavlja forme (ima natpis *Detail*), sve dok pokazivač ne dobije oblik crnog ktstića, a onda povlačenjem miša povećate visinu na oko 1.5 cm. Radi orijentacije, uz levu ivicu forme imate vertikalni lenjir. To isto možete uraditi i tako što prvo kliknete u gornje zaglavlje (prvo ga selektujete) a onda u prozoru *Properties*, na kartici *Format*, podesite vrednost svojstvu *Height* na 1.5 cm. Sada selektujte (jednu za drugom ili istovremeno) kontrole *PredmetID_Label* i *NazivPredmeta_Label* pa ih prevucite u donji deo zaglavlja. Time smo napravili mesta za dodavanje novih kontrola.

Na Accessovoj paleti *Form Design* pritisnite na taster *Toolbox* (Sl.55a) nakon čega će se na ekranu pojaviti kutija sa alatima pomoću koje možete dodavati nove kontrole na formu (Sl.55b).



(a)

(b)

Sl. 55 Paleta sa alatima

Na paleti izaberite alatku *Label* (rb.1 na slici Sl.55a) Pokazivač miša dobija oblik krstića, što znači da ste u režimu dodavanja (crtanja) nove kontrole. Kliknite u levi deo gornjeg zaglavlja i držeći pritisnut levi taster razvucite kontrolu. Nakon toga, pojaviće se pravougaonik sa kurzorom gde treba otkucati PREDMETI. Da bi ste završili sa editovanjem ove kontrole kliknite van nje. Ponovo je selektujte i podesite ostala svojstva prema sledećoj tabeli.

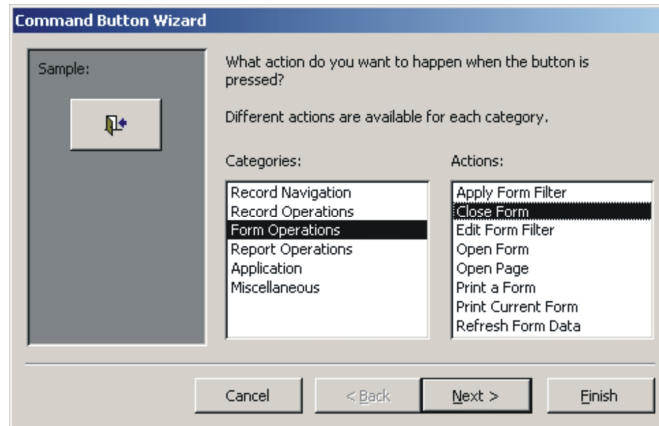
<i>Font Name</i>	MS Sans Serif
<i>Font Size</i>	18
<i>Fore Color*</i>	16777215
<i>Font Weight</i>	Bold

Kontrole iz zaglavlja možete dodatno korigovati (promeniti im veličinu i položaj).

Nova kontrola koju ćemo napraviti poslužiće nam za zatvaranje forme *FPredmet*. Pre nego što počnemo proverite da li je na paleti sa alatima uključeno dugme *Control Wizards* (Sl.55 rb.2) - ako nije uključite ga. Time nalažemo Accessu

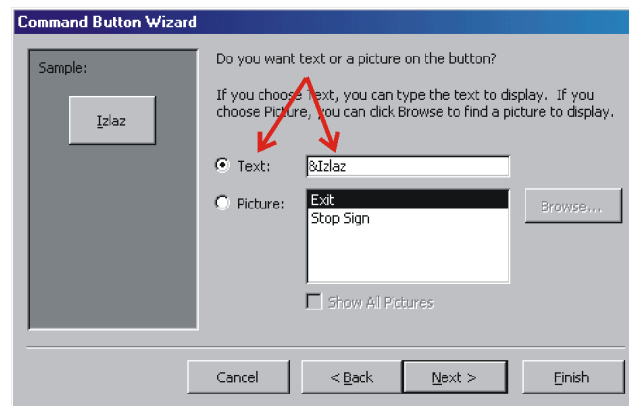
* Pri izboru boja za kontrole, naravno nije potrebno pamtiti kodove boja (npr. 16777215 u gornjem slučaju). Boje jednostavno birate iz palete koja se otvara kada kliknete na dugme sa oznakom "...". Ovo dugme se javlja kada u prozoru *Properties* kliknete u polje za podešavanje svojstva *Fore Color* ili *Back Color*.

da nam pri kreiranju kontrola pomaže sa svojim čarobnjacima. Pritisnite na alatku *Command Button* (Sl.55 rb.3) pa je nacrtajte na desnom kraju donjeg zaglavlja. Otvara se prozor čarobnjaka (Sl.56) tražeći od nas da mu kažemo koju aktivnost želimo da izvršimo sa komandnim dugmetom koje kreiramo.



Sl.56 Izbor aktivnosti za kontrolu *Command Button*

U prvom prozoru čarobnjaka sa liste levo izaberite *Form Operations* a na listi desno od ponuđenih opcija izaberite *Close Form*. U sledećem prozoru (Sl.57) biramo natpis/sliku za dugme. Izaberite opciju *Text* i u polju otkucajte *&Izlaz*.



Sl. 57 Izbor natpisa/slike za kontrolu *Command Button*

Pritiskom na dugme *Finish* završite proceduru kreiranja nove kontrole koja će se nakon toga pojaviti na formi. Naknadno joj podesite svojstva:

<i>Font Size</i>	10
<i>Width</i>	1.8
<i>Height</i>	0.8

Snimite formu i otvorite je u *Form View* prikazu. Trebalo bi da izgleda kao na slici Sl.58.



Sl. 58 Forma FPredmeti

Primitite da je slovo I u natpisu kontrole za zatvaranje forme podvučeno. To znači da aktivnost koja se krije u pozadini ovog dugmeta (zatvaranje forme) možemo aktivirati bilo klikom na dugme bilo kombinacijom tastera *Alt+I* na tastaturi. Ovo svojstvo smo podesili time što smo pri definisanju natpisa za dugme (Sl.57) ispred slova I otkucali znak &.

Da bi smo olakšali unos podataka o studentima kreiraćemo formu pod imenom *FStudenti*. Zatvorite sve prozore osim prozora baze podataka i na paleti sa alatima pritisnite na dugme *New* (kreiranje nove forme). Otvara se okvir za dijalog kao na slici Sl.46. U gornjoj listi ovog prozora izaberite *Design View* (obrazac pravimo bez pomoći Wizarda) a kao izvor podataka nove forme iz padajuće liste izaberite tabelu *TStudenti*. Nakon pritiska na dugme *OK* pred Vama se otvara prazan obrazac u dizaj modu. Snimite obrazac pod imenom *FStudenti* i podesite sledeća svojstva:

<i>Caption</i>	MASINSKI FAKULTET - Kraljevo
<i>Record Selectors</i>	No
<i>Dividing Lines</i>	No

Da bi ste na formi videli donje i gornje zaglavlje u meniju *View* uključite opciju *Form Header/Footer*. Zaglavljima podesite sledeća svojstva:

<i>Height</i>	1.2
<i>Back Color</i>	16737843

Ponavljajući postupak iz prethodne forme u gornjem zaglavlju ucrtajte labelu sa natpisom: STUDENTI.

Sledeća kontrola koju ćemo ugraditi na formu poslužiće nam da razdvojimo lične i podatke o ocenama studenta. Na paleti sa alatima selektujte kontrolu *Tab Control* (Sl.55 rb. 4) pa je razvucite po celoj površini dela za detalje (Detail) forme. Kontrolu podesite sledeća svojstva:

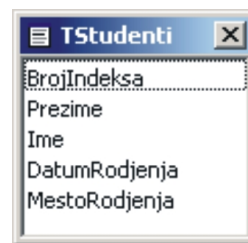
<i>Name</i>	TabStudenti
<i>Font Name</i>	MS Sans Serif
<i>Font Size</i>	8
<i>Width</i>	10
<i>Height</i>	6

Novoformirana kontrola (koja se sada zove TabStudenti) sastoji se od dva lista sa labelama *Page1* i *Page2*. Listove birate tako što kliknete na odgovarajući jezičak. Selektujte prvu stranu (proverite da li na naslovnoj traci prozora *Properties* piše *Page1*) i svojstvu *Caption* dodelite vrednost LICNI PODACI. Svojstvu *Caption* drugog lista (nakon što ga selektujete) dodelite vrednost OCENE.

Na paleti sa alatima kliknite na alatku *Field List* (Sl.59a) nakon čega se otvara prozor sa listom polja iz tabele *TStudenti* koju smo izabrali kao izvor podataka za ovu formu (Sl. 59b).



(a)



(b)

Sl. 59 Prikaz liste polja obrasca

Označite sva polja (pomoću tastera *Ctrl* ili *Shift*) sa liste i odvučite ih na prvi list kontrole *TabStudenti*. Na formi ćete dobiti deset novih kontrola (5 Label-a i 5 Text Box -ova). Svima njma podesite sledeća svojstva:

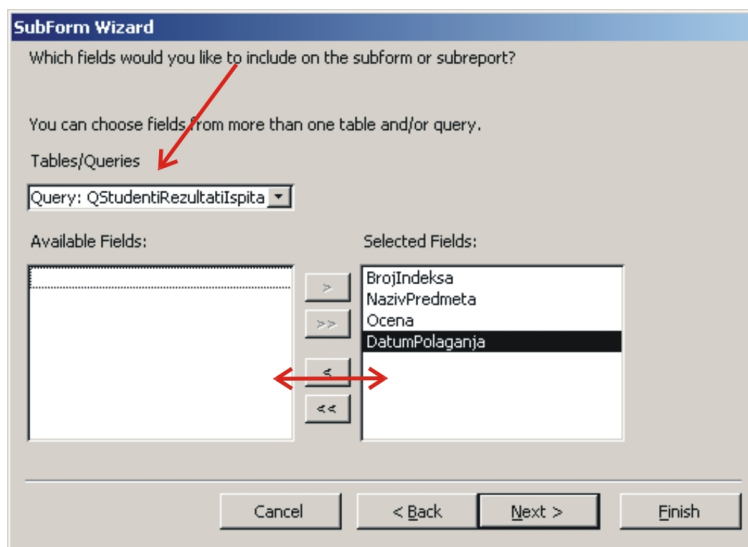
<i>Font Name</i>	Arial
<i>Font Size</i>	9

Labeli sa tekстом *BrojIndeksa* promenite svojstvo *Caption* u npr. *Br. indeksa*. Slično možete uraditi i sa drugim labelama. Izgled forme sa prvom stranom kontrole *TabStudenti* dat je na slici Sl.60.

Sl.60 Forma *FStudenti* sa prvom stranom kontrole *TabStudenti*

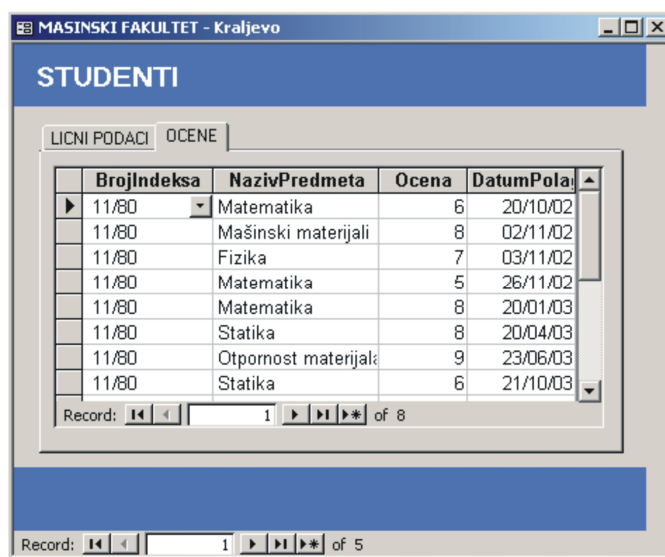
Sada ćemo na drugom listu kontrole *TabStudenti* prikazati sve ocene koje je student dobio. Kao izvor podataka poslužiće nam upit *QStudentiRezultatiIspita* koji smo ranije napravili. Ono što treba uočiti je da između zapisa tabele *TStudenti* (koja predstavlja podlogu ove forme) i upita *QStudentiRezultatiIspita* postoji relacija jedan prema više. Jednom studentu odgovara više rezultata ispita. U tom slučaju za prikaz podataka koristi se kontrola *Subform/Subreport*. (Sl.55 rb. 5) kojom se unutar forme kreira podforma. Dakle, osnovna namena ove kontrole je prikaz zapisa koji su u relaciji jedan prema više za zapisima iz osnovne forme.

U dizajnu modu forme *FStudenti* selektujte prvo kontrolu *Page2* (na naslovnoj traci prozora *Properties* treba da piše *Page:Page2*). Zatim na površini tog lista razvučite kontrolu *Subform/Subreport*. Pred Vama će se otvoriti prozor gde treba da izaberete način kreiranja podforme: od neke tabele/upita ili od neke postojeće forme. Podrazumevano se nudi prva mogućnost (*Use existing Tables and Queries*) tako da samo potvrdite izbor klikom na dugme *Next*. U sledećem prozoru potrebno je da selektujete jednu od postojećih tabela/upita kao i njihova polja koja želite prikazati na podformi (Sl.61).



Sl.61 Izbor izvora podataka za podformui

Mi ćemo izabrati upit *QStudentiRezultatiIspita* i iz njega sva polja. Sledeći korak predstavlja vrlo važnu etapu u kreiranju podforme jer u njemu treba definisati vezu između podataka na formi i podataka na podformi. U našem slučaju mi želimo da za određenog studenta prikazanog u formi, u podformi prikažu podaci o ocenama tačno tog studenta. Veza se definiše tako što se biraju polja preko kojih se ona ostvaruje. Naravno, ovde ćemo vezu uspostaviti preko polja pod imenom *BrojIndeksa*. Zato u prikazanom prozoru prihvatite ponudjenu opciju na listi (*Show QStudentiRezultatiIspita for each record in TStudenti using BrojIndeksa*) i pritisnite taster *Next*. U zadnjem koraku Wizard će Vas pitati za ime koje želite da date podformi. Umesto ponudjene opcije otkucajte *FStudentiRezultatiIspita subform*. Završite postupak kreiranja nove kontrole pritiskom na dugme *Finish*. Na površini drugog lista kontrole *TabStudenti* pojavuće se nova kontrola. Selektujte labelu ove kontrole i izbrišite je (vodite računa da to ne uradite sa samom kontrolom). Veličinu podforme podesite tako da zauzme celu površinu lista na koji ste je smestili. Ako sada formu otvorite u režimu za prikaz (*View*) i predjete na list sa ocenama forma će izgledati kao na slici Sl.62.



Sl.62 Forma FStudenti sa drugom stranom kontrole TabStudenti

Možete primetiti da se u svakom zapisu podforme ponavlja podatak o broju indeksa što je sasvim nepotrebno. Zato ćemo ovu kolonu (*BrojIndeksa*) sakriti tako što prvo kliknete u bilo koji zapis iz te kolone a onda iz menija *Format*

birate komandu *Hide Columns*. Ovo polje smo morali uključiti u polja podforme zbog uspostavljanja veze sa podacima sa glavne forme ali ga ne moramo prikazati. Svi podaci o studentu čije su ocene prikazane u podobrascu nalaze se na listu sa ličnim podacima.

Kako nam ova podforma pre svega služi za pregled ocena ali ne i za njihovu izmenu (ne možemo dozvoliti da student sam menja ocene), u dizajn modu, svojstvu *Locked* podforme dodelite vrednost *Yes*. Sada pomoću kontrola za navigaciju možete prelistavati podatke o studentima i pregledati ocene bez mogućnosti da se one izmene.

Da bi smo olakšali kretanje kroz zapise ugradićemo nove dugmiće za navigaciju. Prvo dugme koje ćemo kreirati je dugme za skok na prvi zapis tabele. Pritisnite na alatku *Command Button* i nacrtajte je na levom kraju donjeg zaglavlja. Otvara se prozor čarobnjaka (Sl.56) gde treba da izaberemo aktivnost koju želimo da izvršimo sa komandnim dugmetom koje kreiramo. U listi *Categories* izaberite *Record navigations* a u listi *Actions* opciju *Goto First Record*. Pošto ostali koraci procedure za kreiranje ove kontrole nisu toliko bitni pritiskom na taster *Finish* završite proceduru. Zatvara se prozor čarobnjaka a na formi se pojavljuje komandno dugme za skok na prvi zapis tabele (Sl.63).



Sl. 63 Komandno dugme za skok na prvi zapis tabele

Na sličan način dodajte tastere za:

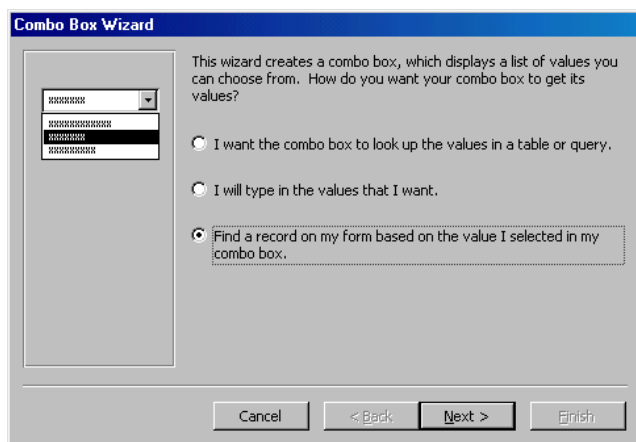
- skok na prethodni zapis (*Goto Previous Record*)
- skok na sledeći zapis (*Goto Next Record*)
- skok na poslednji zapis (*Goto last Record*)

Ponavljajući postupak primenjen kod prethodne forme, na desnom kraju futera ove, dodajte dugme za zatvaranje obrasca. Donje zaglavlje ove forme bi sada trebalo da izgleda kao na slici Sl.64.



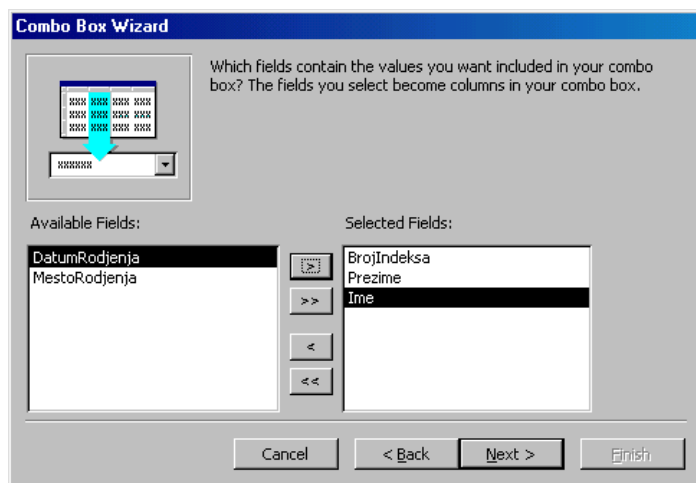
Sl.64 Futer forme FStudenti

Kako tokom korišćenja baze broj studenata može da naraste na nekoliko stotina, pronalaženje određenog studenta pomoću dugmića za navigaciju koje smo upravo kreirali, može biti naporno. Zato ćemo napraviti još jednu kontrolu čija je osnovna namena brže pronalaženje određenog studenta. U dizajn modu iz kutije sa alatima izaberite alatku *Combo Box* (padajuća lista, Sl.55 rb.6) pa je nacrtajte na desnom kraju gornjeg zaglavlja. U prvom koraku wizard će tražiti da definišemo šta je izvor podataka za padajuću listu (Sl.65).



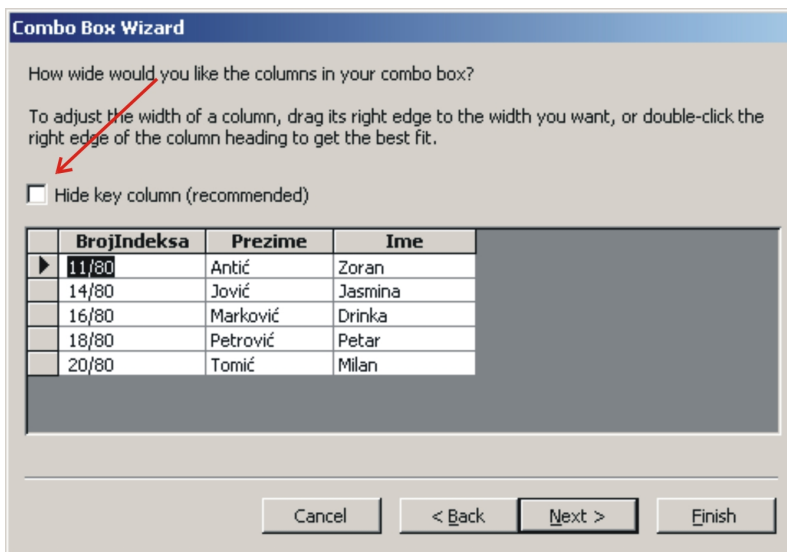
Sl.65 Definisavanje izvora podataka za padajuću listu

Izaberite treću opciju i pritisnite dugme *Next*. U sledećem prozoru (Sl.66) određujemo koja polja želimo da se pojave u padajućoj listi.



Sl.66 Određivanje polja padajuće liste

Selektujte polja *BrojIndeksa*, *Prezime* i *Ime*. U narednom prozoru koji se otvara isključite opciju *Hide key column*. (Sl.67) i pritiskom na dugme *Finish* završite proceduru.



Sl.67 Podešavanje širine kolona u padajućoj listi

Kada wizard završi posao na formi će se pojaviti nova kontrola kojoj podesite svojstva:

<i>Font Name</i>	Arial
<i>Font Size</i>	10
<i>Limit To List</i>	Yes
<i>Width</i>	2

Labeli te kontrole postavite svojstva:

<i>Caption</i>	Trazi
<i>Font Name</i>	Arial

Font Size	10
Font Weight	Bold
Text Align	Right
Fore Color	16777215

Snimite formu i otvorite je u *Form View* prikazu. Trebalo bi da izgleda kao na slici Sl.68.

Sl.68 Forma FStudenti sa kontrolom za pretraživanje

Ako sada želite na nađete nekog studenta u bazi otvorite padajuću listu i iz nje izaberite studenta. Nakon toga na formi će se pojaviti podaci o traženom studentu.

Sledeća forma koju ćemo napraviti poslužiće nam za ažuriranje podataka o ispitima. I ovde treba primetiti da se podaci o ispitima nalaze u dve tabele (*Tispiti* i *TRezultatiIspita*) koje su u međusobnoj relaciji jedan prema više. To znači da u okviru forme treba prikazati opšte podatke o ispitu (tabela *Tispiti*) a za podatke o studentima koji su polagali ispit (*TRezultatiIspita*) koristimo podformu.

Prvo ćemo napraviti podformu. U prozoru baze podataka kliknite na alatku *New* čime otvarate okvir za dijalog kao na slici Sl.46. U gornjoj listi ovog prozora izaberite *Design View* a kao izvor podataka nove forme iz padajuće liste izaberite upit *QRezultatiIspita*. Nakon pritiska na dugme *OK* pred Vama se otvara prazan obrazac u dizaj modu. Na paleti sa alatkama kliknite na alatku *Field List* (Sl.59a) i iz otvorene liste (Sl. 59b) prevucite sva polja iz upita *QRezultatiIspita* na formu. Zbog preglednosti podataka svojstvu *Default View* forme podesite vrednost na *Datasheet*. Snimite obrazac pod imenom *FRezultatiIspita* i zatvorite ga.

Sada ćemo napraviti glavnu formu za ispite. Ponovo kliknite na dugme *New*, izaberite opciju *Design View*, a kao podlogu za novu formu izaberite tabelu *Tispiti*. Na novootvorenu formu iz liste (Sl. 59b) prevucite sva polja. Ova polja se odnose na opšte podatke u vezi ispita: *IspitID*, *IspitniRok*, *DatumPolaganja*, *PredmetID*. Snimite formu pod nazivom *FIspiti* i podesite joj sledeća svojstva:

Caption	MASINSKI FAKULTET - Kraljevo
Record Selectors	No

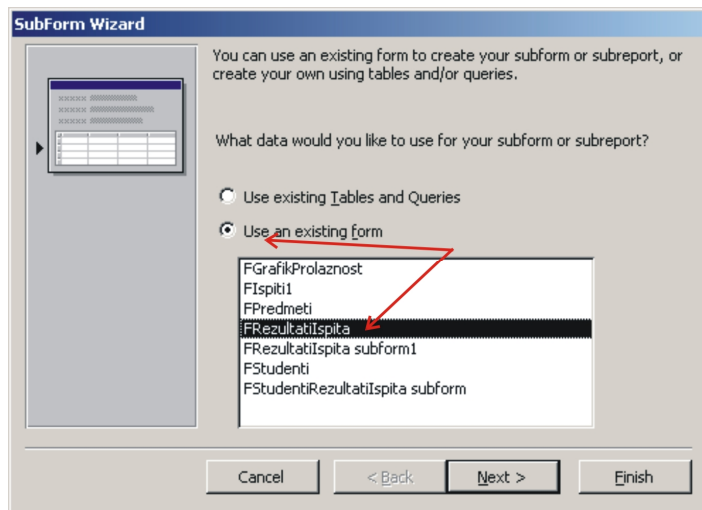
Svim kontrolama (labelama i poljima za tekst) podesite sledeća svojstva:

Font Name	Arial
Font Size	10
Height	0.5

Svojstvo *Caption* labela podesite po želji. Npr. labeli u kojoj piše *PredmetID* svojstvu *Caption* dodelite vrednost

Predmet. Pošto polje za tekst *IspitID* Access sam popunjava (korisnik ne može sam unositi vrednosti) svojstvu *Enabled* ovog polja dodelite vrednost *No*.

Da bi, pored opštih podataka o ispitu, preko forme unosili i podatke o studentima koji su izašli na ispit i njihovim ocenama sada ćemo u formu ugraditi prethodno napravljenu formu *FRezultatiIspita*. Na paleti sa alatka kliknite na alatku *Subform/Subreport* a onda kontrolu nacrtajte u delu sa detaljima glavne forme. U prvom prozoru koji se otvara treba da odgovorite šta je izvor podataka za podformu (Sl.69). Kako mi već imamo napravljenu podformu pod imenom *FRezultatiIspita* selektujte opciju "Use an existing form" i iz liste ispod izaberite odgovarajuću formu.



Sl.69 Izbor izvora podataka za podformu *FRezultatiIspita*

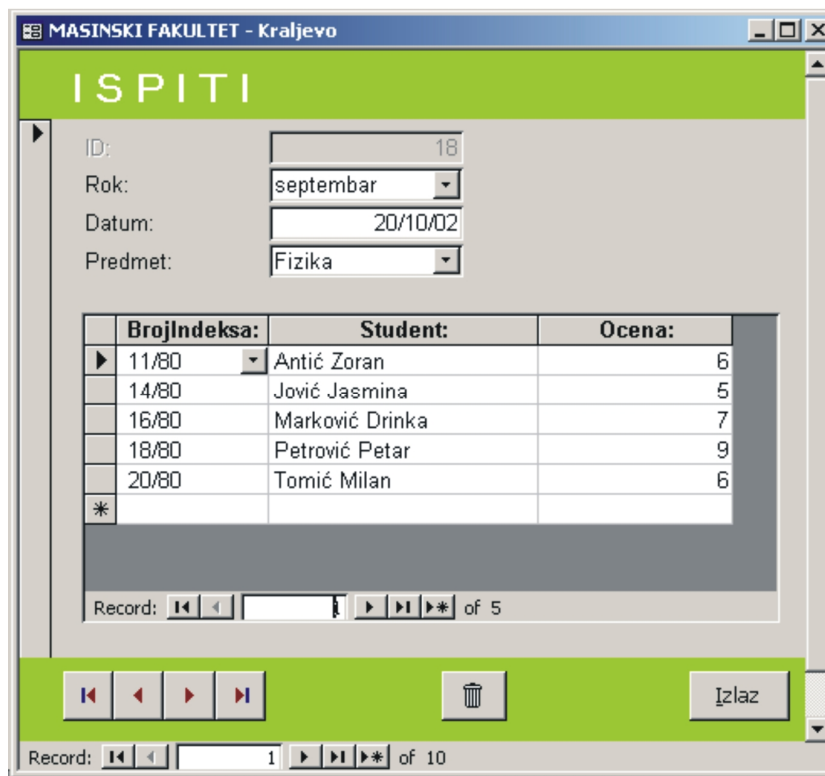
U sledećem prozoru wizarada treba definisati polja iz forme i podforme preko kojih se uspostavlja veza i omogućuje sinhronizacija prikaza. Access je već ponudio ispravnu vezu preko polja *IspitID* tako da Vam ostaje da to potvrdite pritiskom na dugme *Next*. Sledeći korak Vam pruža mogućnost da promenite ime podformi. Mi ćemo prihvatiti ponudjeno ime i pritiskom na dugme *Finish* završite proceduru.

Ako formu *FIspiti* otvorite u režimu za prikaz videćete da se u podformi javlja kolona *IspitID*. Pošto nam je ta kolona nepotrebna (polje nam je poslužilo za uspostavljanje veze) pomerite kurzor u tu kolonu i iz menija *Format* izaberite komandu *Hide Columns*.

Pri popunjavanju podataka o ispitu prvo treba uneti opšte podatke o ispitu. Nakon toga, podaci o studentima koji su polagali, unose se na sledeći način: u koloni *BrojIndeksa* iz padajuće liste birate studenta koji je polagao a program će sam upisati prezime i ime studenta. U koloni *Oцена* unosite ocenu koju je dobio student.

U gornjem i donjem zaglavlju forme možete dodati kontrole kao i u slučaju obrasca *FStudenti*. Pored njih, ugradićemo još jednu kontrolu koju do sada nismo koristili. Ona će nam poslužiti za brisanje tekućeg zapisa forme. Na paleti sa alatima izaberite alatku *Command Button* i nacrtajte je u futeru obrasca. U prvom prozoru wizarada u listi kategorija izaberite *Record Operations* a u listi aktivnosti *Delete record*. U sledećem prozoru birate izgled komandnog dugmeta. Možete izabrati tekst ili neku sličicu (sličice možete i sami praviti). Mi ćemo izabrati ugradjenu sličicu *Trash Can 2*. Posle toga možete pritisnuti dugme *Finish* i završiti sa kreiranjem kontrole. Da bi ste korisniku pružili dodatno objašnjenje funkcije ove alatke njenom svojstvu *ControlTip Text* dodelite vrednost *Brisanje podataka o ispitu*. Kada korisnik bude, u režimu za prikaz, mišem pokazao na ovu alatku, pojavice se prethodno uneti tekst sa ciljem objašnjenja funkcije alatke. Pritiskom na ovaj taster, nakon dodatne potvrde, brišu se podaci o tekućem ispitu (ispit koji trenutno vidite na formi). Pri tome se brišu i opšti podaci i podaci o studentima koju su polagali taj ispit. Prethodno izbrisani zapis ne možete vratiti pa zato budite pažljivi sa brisanjem.

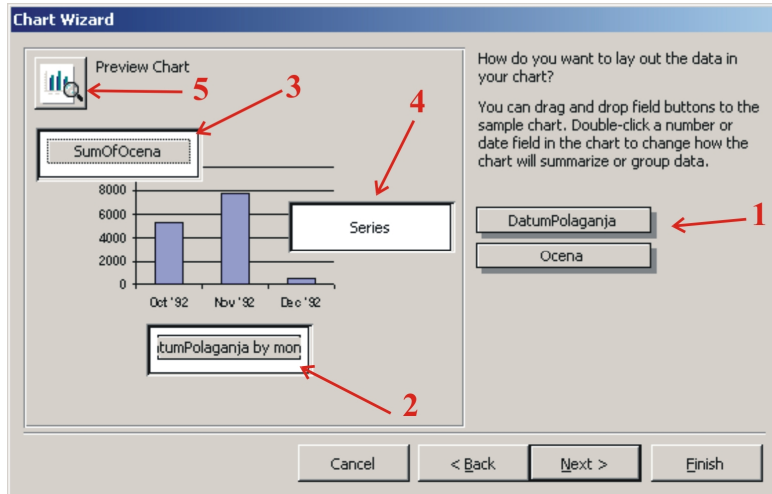
Konačan izgled forme *FIspiti* dat je na slici Sl.70.



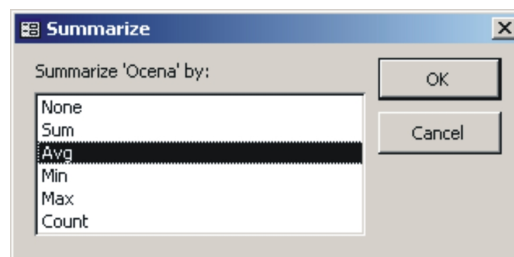
Sl.70 Forma FIspiti

Da bi smo ilustrovali mogućnosti grafičkog prikazivanja podataka napravićemo formu *FGrafSrednjaOcena*. Ona će nam poslužiti za vizuelno predstavljanje prosečne ocene, za određeni predmeta, po ispitnim rokovima u toku proteklih godinu dana.

Pritiskom na dugme *New*, u prozoru baze podataka, pokretnite proceduru za kreiranje nove forme. U prozoru za izbor načina kreiranja forme (Sl.46) izaberite *Chart Wizard* a za izvor podataka selektujte upit *QPredmetiOcene*. U sledećem prozoru za polja koja sadrže podatke za dijagram izaberite *datumPolaganja* i *Ocena*. Nakon toga pruža Vam se mogućnost da izaberete tip dijagrama. Prihvatite ponudjenu opciju (prvi dijagram) i kliknite na taster *Next*. Sada treba definisati način prikaza podataka na dijagramu (Sl.71). Pošto želimo da prikažemo prosečne ocene po ispitnim rokovima, duž horizontalne ose treba prikazati datume polaganja a duž vertikalne srednje ocene. Izbor podataka koji će se prikazati duž jedne od osa vrši se tako što se odgovarajuće polje iz liste (Sl.71, rb.1) prevuče u odgovarajući pravougaonik (Sl.71 rb2, rb3). Polje sa labelom *Series* (Sl.71, rb4) omogućava dodatno grupisanje podataka po nekom od polja. Vizard je sam ponudio da duž horizontalne ose prikaže datume polaganja grupisane po ispitnim rokovima tako da tu ne treba ništa menjati. Medjutim, duž vertikalne ose ne želimo sumiranje ocena već njihov prosek. Zato dva puta kliknite na dugme sa labelom *SumOfOcena* i iz ponudjene liste (Sl.72) izaberite *Avg (Average)*. Nakon toga u polju duž vertikalne ose trebalo bi da piše *AvgOfOcena*. Ako želite da vidite kako će Vaš dijagram izgledati pritisnite na dugme *Preview Chart* (Sl.71, rb.5). Polje sa labelom *Series* (Sl.71, rb4) ostaje prazno jer ne želimo dodatno grupisanje podataka. Preostaje nam da još u zadnjem koraku vizarda odredimo naslov dijagrama. Ukucajte *Prosečne ocene* i finiirajte proceduru.



Sl.71 Definisane načina prikazivanja podataka na dijagramu forme FGrafSrednjaOcena

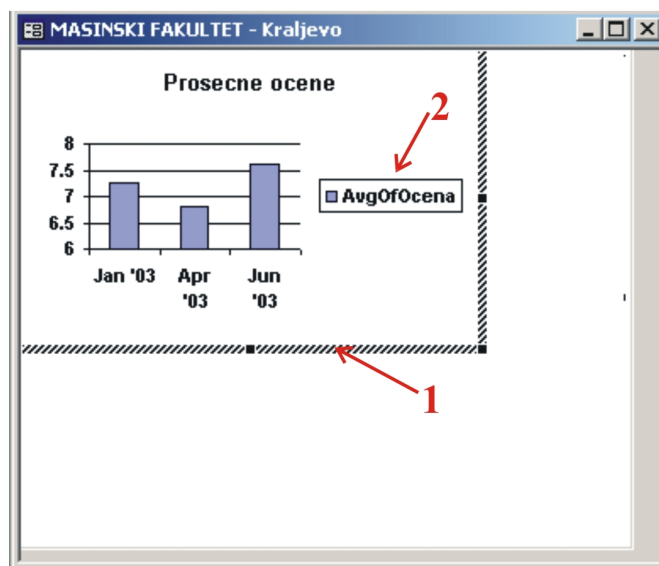


Sl.72 Definisane načina grupisanja podataka na dijagramu forme FGrafSrednjaOcena

Nakon toga, pre nego što se otvori nova forma, parametarski upit *QPredmetiOcene* (koji predstavlja podlogu dijagrama) tražiće da unesemo vrednost parametra *SifraPredmeta* (Sl.45). Otkucajte recimo *0001* i pred Vama će se otvoriti dijagram koji grafički prikazuje srednju ocenu, po ispitnim rokovima, za predmet čija je šifra *0001*. Otvorite sada ovaj obrazac u dizajn modu i podesite mu sledeća svojstva:

<i>Caption</i>	MASINSKI FAKULTET - Kraljevo
<i>Scroll Bars</i>	Neither
<i>Record Selectors</i>	No
<i>Navigation Buttons</i>	No
<i>Dividing Lines</i>	No
<i>Border Style</i>	Thin

Ako pogledate naslovnu traku prozora *Properties* videćete da je grafikon prikazan unutar kontrole *Unbound Object Frame*. Povećanjem dimenzija ove kontrole povećava se veličina okvira ali ne i veličina samog dijagrama. Da bi ste promenili veličinu dijagrama dva puta kliknite na dijagram nakon čega se otvara prozor za editovanje dijagrama (Sl.73).



Sl.73 Prozor za uređivanje dijagrama

Povećajte veličinu dijagrama (pomoću osenčenog okvira (Sl.73, rb.1)) po celoj veličini okvira. Legendu (Sl.73, rb.2) selektujte i izbrišite. Da bi ste izašli iz moda za editovanje dijagrama kliknite na formu izvan ove kontrole. Ono što malo smeta kada se pokreće ova forma je potreba da se prvo izabere šifra predmeta za koji želimo prikaz. Da bi smo malo poboljšali funkcionalnost, na formu ćemo ugraditi padajuću listu gde će korisnik da bira predmet za koji želi grafički prikaz. Zato oslobodite gornji deo forme pomerajući okvir sa dijagramom na dole i desno. U gornjem levom uglu forme ucrtajte kontrolu *Combo Box* sa tabelom *TPredmeti* kao izvor podataka. Kada završite sa kreiranjem ove alatke podesite sledeća svojstva:

Kontrola	Ime	Svojstvo	Vrednost
Label	<i>NazivPredmeta Label</i>	Caption	Predmet
Label	<i>NazivPredmeta Label</i>	Font Name	Arial
Label	<i>NazivPredmeta Label</i>	Font Size	10
Combo Box	<i>Comb1</i>	Font Name	Arial
Combo Box	<i>Comb1</i>	Font Size	10
Combo Box	<i>Comb1</i>	Name	cmbPredmet
Combo Box	<i>Comb1</i>	Default Value	"0001"

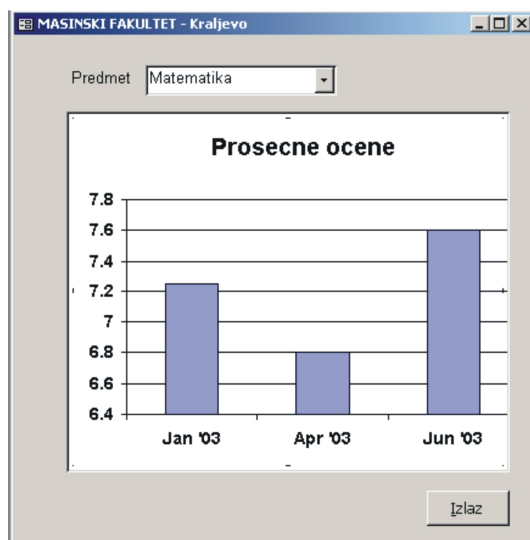
Zadnjim svojstvom smo podesili da je podrazumevana vrednost za padajuću listu predmet sa šifrom "0001".

Sada je potrebno da parametarskom upitu *QPredmetiOcene* kažemo gde da traži podatak za parametar. Otvorite upit u dizaj modu i u koloni *PredmetID* reda *Criteria* umesto *SifraPredmeta* otkucajte:

```
= [Forms]![FGrafSrednjaOcena]![cmbPredmet]
```

Takodje otvorite prozor *Query Parameters* (Sl.44) i izbrišite parametar *SifraPredmeta*. Time smo upitu rekli da umesto parametra kao kriterijum za polje *PredmetID* uzima podatak sa adrese *[Forms]![FGrafSrednjaOcena]![cmbPredmet]*. Ako sada pokrenete obrazac *FGrafSrednjaOcena* podrazumevano će se pojaviti grafikom sa srednjim ocenama za predmet sa šifrom "0001". Grafički prikaz srednjih ocena za drugi predmet dobićete tako što ga selektujete u padajućoj listi a onda na tastaturi pritisnete taster F9 (naredba za ažuriranje podataka).

I na kraju u donjem delu forme ugradite dugme za zatvaranje obrasca. Forma bi trebalo da izgleda kao na slici Sl 74.



Sl. 74 Forma FGrafSrednjaOcena

Izrada izveštaja

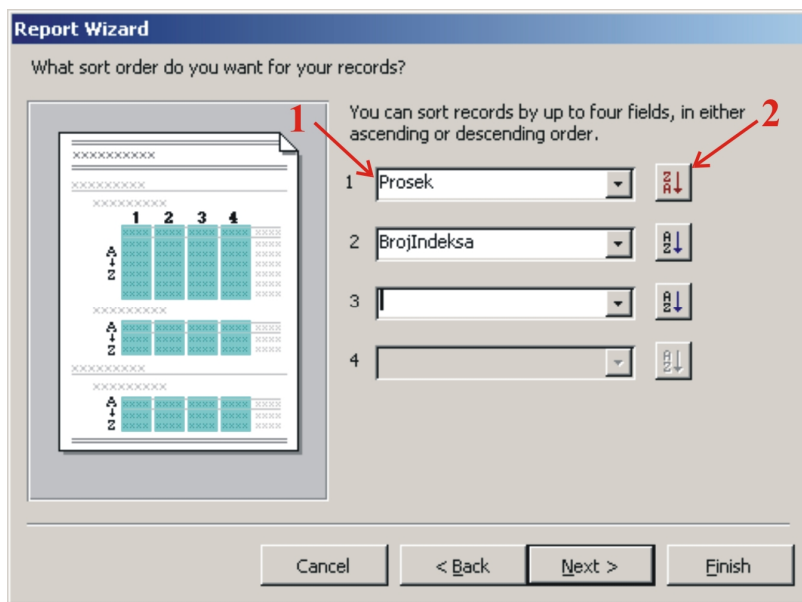
Jedan od osnovnih razloga korišćenja aplikacija za rad sa bazama podataka je dobijanje željenih izveštaja na osnovu podataka prikupljenih u bazi. Izveštaji (Reports) su upravo objekti baze kojima se to postiže. Ono što ih razlikuje od obrazaca je to što su prilagodjeni za štampu bilo na lokalnom štampaču bilo na udaljenoj lokaciji (slanje faksom, e-poštom itd.).

Kreiranje izveštaja je slično kreiranju formi pa se i tehnike koje smo tamo naučili mogu primeniti i ovde. Prvi izveštaj koji ćemo napraviti je spisak studenata rangiranih po uspehu. U prozoru baze na traci *Objects* izaberite *Reports* a onda na paleti sa alatima pritisnite na *New* (kreiranje novog izveštaja). Otvara se okvir za dijalog sličan onom na slici Sl.46. Pored izbora tabele/upita kao izvora podataka za izveštaj, iz liste treba izabrati i način kreiranja izveštaja. Ponudjene su opcije:

- *Design View*
- *Report Wizard*
- *AutoReport: Columnar*
- *Autoreport: Tabular*
- *Chart Wizard*
- *Label Wizard*

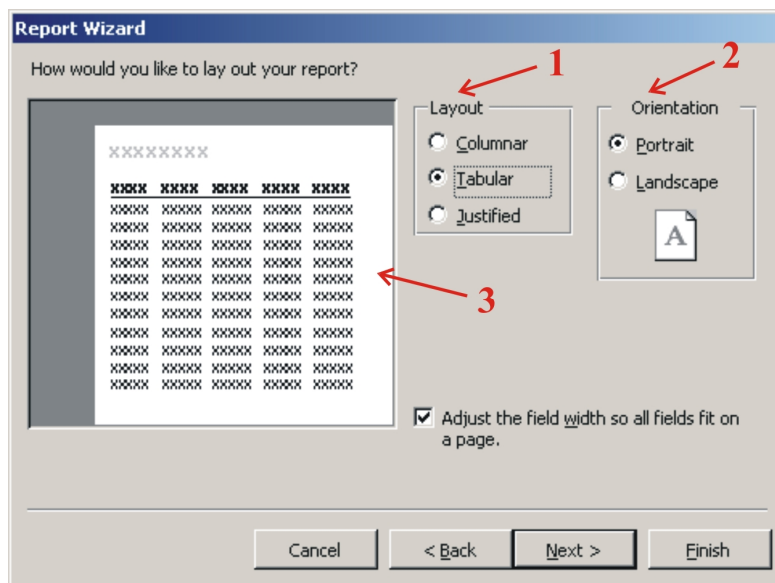
Značenje ovih opcija je slično kao i kod formi osim što *Report Wizard* nudi dodatne mogućnosti grupisanja podataka, višestrukog sortiranja, različite vrste sumiranja. Obično se osnovni kostur izveštaja napravi pomoću wizarđa a onda se ručnim doterivanjem, u dizajn modu, dovede do finalne verzije. Novi wizarđ u ovoj listi je *Label Wizard* namenjen izveštajima koji se štampaju na labelama ili nalepticama. Recimo kada želite da iz nekog adresara odštampate adrese direktno na kovertama koristite ovaj wizarđ.

Na listi izaberite *Report Wizard* a kao podlogu označite upit *QStudentiSrednjaOcena*. U sledećem prozoru birate polja iz upita koja želite da se pojave u izveštaju. Izaberite sva tri polja. Sledeći korak pruža Vam mogućnost da izvršite grupisanje podataka po poljima kao i prioritet grupisanja. Na primer, ako pravite neki izveštaj o prodaji pa želite da kupce grupišete po mestu to možete uraditi u ovom dijalogu. U našem slučaju nema potrebe za grupisanjem podataka pa predjite na sledeći korak. U prozoru koji se otvara možete odrediti način sortiranja podataka (Sl.75).



Sl.75 Definisane načina sortiranja u izveštaju

Iz padajuće liste (Sl.75 rb.1) birate polja po kojima želite sortiranje i to redosledom kako se ona navode. Imate mogućnost da podatke sortirate po četiri polja. Za svako polje možete izabrati sortiranje u rastućem ili opadajućem redosledu (Sl.75 rb.2). Mi smo izabrali sortiranje prvo po prosečnoj oceni (*Prosek*) i to u opadajućem redosledu a onda po broju indeksa (*BrojIndeksa*) u rastućem redosledu. To znači ako postoje dva studenta sa istim prosekom prvo će biti prikazan onaj sa manjim brojem indeksa. Narednim dijalogom birate raspored podataka na izveštaju (Sl.76, rb.1) i orijentaciju papira (Sl.76, rb.2).



Sl.76 Izbor rasporeda podataka na izveštaju

Efekte promene rasporeda možete pratiti u prozoru (Sl.76, rb.3). Prihvatite ponudjene opcije i predjite na sledeći korak. U njemu treba izabrati stil (izgled) budućeg izveštaja. Izaberite *Formal*. Poslednji prozor sadrži polje u kojem treba otkucati naslov izveštaja. Otkucajte *RRangListaStudenata* i pritiskom na dugme *Finish* zatvorite wizard. Ovde smo izabrali malo čudan naslov (reči su spojene sa početnim slovom R) jer wizard kreira izveštaj upravo pod tim imenom. Pa da bi smo ispoštovali konvenciju o imenovanju objekata, o kojoj smo nešto rekli na početku ovog uputstva, zato smo dali ovakav naslov. Pred Vama se otvara izveštaj kao na slici Sl.77.

Prosek BrojIndeksa	Student
8.33333333333333 20/80	Tornić Milan
7.90909090909091 16/80	Marković Drinka
7.625 18/80	Petrović Petar
7.44444444444444 14/80	Jović Jasmina
7.33333333333333 11/80	Antić Zoran

Sl.77 Izveštaj *RRangListaStudenata* kreiran pomoću wizarda

Ako izveštaj otvorite u dizajn modu primetićete na njemu pet sekcija: gornje i donje zaglavlje izveštaja i strane i deo za prikaz detalja. Donje zaglavlje izveštaja se ne vidi (visina mu je jednaka nuli) ali ga možete povećati pomeranjem pokazivača miša do sive trake na kojoj piše *Report Footer* (pokazivač dobija oblik krstića) i povlačenjem na dole. U sekciji sa detaljima prikazuju se podaci koje želimo da vidimo na izveštaju. Ukoliko se izveštaj sastoji od više strana na svakoj od njih će se prikazati sadržaj iz hedera i futera strane. U našem slučaju na vrhu svake strane pojaviće se

zaglavlja kolona (*Prosek, BrojIndeksa, Student*) a na dnu svake strane pišaće datum štampanja izveštaja (to se postiže funkcijom *Now*) i broj strane u formatu *tekuća strana/ukupno strana*. Sadržaj gornjeg i donjeg zaglavlja izveštaja se prikazuje samo jednom: na početku i kraju izveštaja.

Formirani obrazac još nema izgled kakav želimo pa ćemo ga dodatno doraditi. Labeli ku kojoj piše *RRangListaStudenata* svojstvu *Caption* dodelite vrednost *Rang lista studenata*. Sva polja za tekst (*Text Box*) i odgovarajuće labele pomerite udesno za oko 1 cm i pregrupišite (prevlačući mišem) tako da idu sledećim redosledom: *BrojIndeksa, Student, Prosek*. Da bi smo uz svakog studenta imali njegov redni broj na listi ugradićemo još jednu kontrolu. Iz kutije sa alatima uzmite alatku *Text Box* i ucrtajte je na levom kraju sekcije sa detaljima. Kako ova alatka ide u kombinaciji sa labelom, a ona nam netreba u delu sa detaljima, selektujte tu labelu (proverite u prozoru *Properties* da li ste stvarno nju selektovali) i izbrišite je. Novom polju za tekst podesite sledeća svojstva:

<i>Name</i>	Rb
<i>Control Source</i>	=1
<i>Running Sum</i>	Over All
<i>Input Mask</i>	0.

Na ovaj način Access-u smo rekli da numeraciju studenata počne od 1 i da kod svakog narednog ovo polje poveća za 1 (*Running Sum*). Pomoću svojstva *Input Mask* podešavamo format prikaza rednog broja - iza svakog broja stajaće tačka. Da bi i ova kolona imala svoje zaglavlje na levom kraju hedera strane (*Page Header*) ugradite labelu sa natpisom: *Rb*. Labeli sa tekстом *BrojIndeksa* svojstvu *Caption* dodelite vrednost *Indeks*. Svim labelama iz hedera strane i poljima za tekst iz sekcije sa detaljima svojstvu *Font Size* dodelite vrednost 11. Polju za tekst pod imenom *Prosek* svojstvu *Format* dodelite vrednost *Standard**

Da bi podaci o studentima bili čitljiviji, posebno ako je lista dugačka, zapise iz tabele vizuelno ćemo razdvojiti pomoću linija. Na paleti sa alatima selektujte alatku *Line* pa je nacrtajte u delu sa detaljima ispod polja za tekst. Izjednačite dužinu ove linije sa linijom u hederu strane.

Radi ilustracije dodatnih mogućnosti sumiranja podataka ugradićemo kontrolu koja će prikazati srednju ocenu svih studenata sa rang liste. U donje zaglavle izveštaja (*Report Footer*) nacrtajte polje za tekst sa sledećim svojstvima:

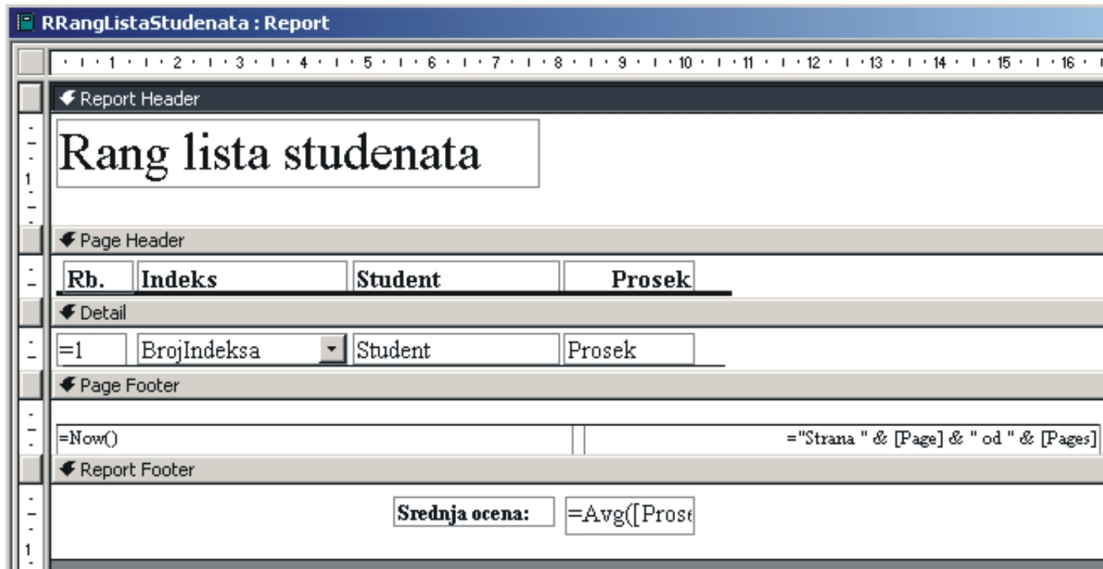
<i>Font Size</i>	11
<i>Control Source</i>	=Avg(Prosek)
<i>Format</i>	Standard

Svojstvu *Caption* labele ovog polja za tekst dodelite vrednost *Srednja ocena*:

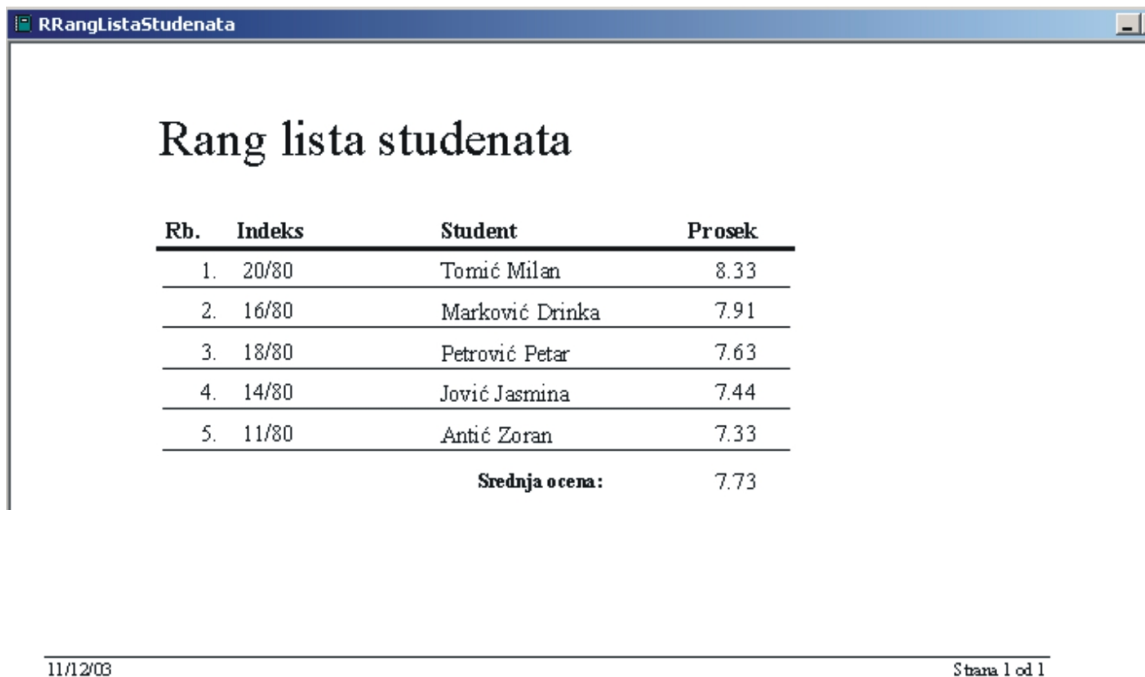
Na kraju dizajniranja ovog izveštaja promenićemo neka svojstva kontrola koje se nalaze u donjem zaglavlju strane. Svojstvu *Format* polja za tekst u kojem piše datum štampanja izveštaja, umesto *Long Date* dodelite vrednost *Short Date*. Time izbegavamo naziv dana u formatu datuma. Kod polja za tekst u kojem se ispisuje numeracija strane svojstvo *Control Source* ima vrednost: *= "Page " & [Page] & " of " & [Pages]*. Umesto toga otkucajte *= "Strana " & [Page] & " od " & [Pages]*. Zahvaljujući tome u dnu strane umesto *Page 1 of 1* pišaće *Strana 1 od 1*.

Izgled ovog izveštaja u dizajn modu dat je na slici Sl.78 a u režimu za prikaz na slici Sl.79.

* Efekte svake od promena možete odmah pogledati ako predjete u režim za prikaz (*Print Preview*) izveštaja.



Sl.78 Izveštaj RRangListaStudenata kreiran u dizajn modu

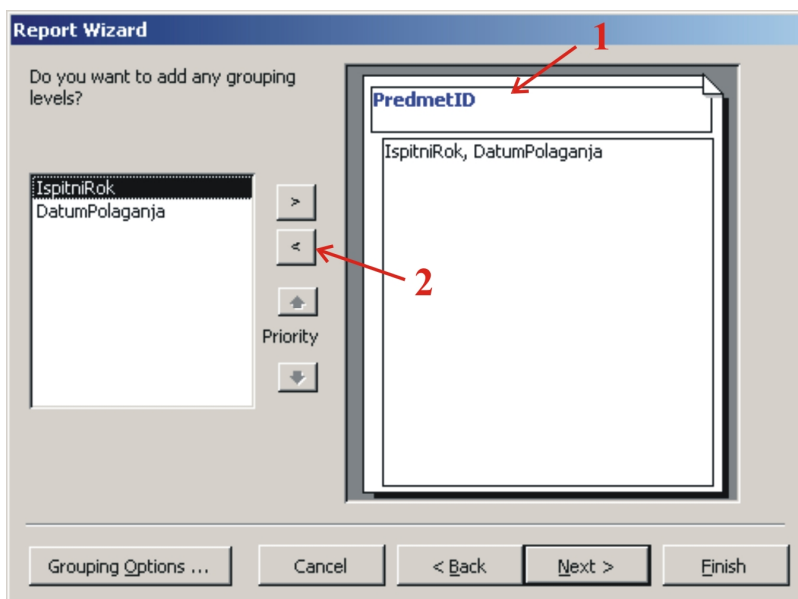


Sl.79 Izveštaj RRangListaStudenatakreiran u režimu za prikaz

Dok ste u režimu za prikaz izveštaj možete odštampati.

Sledeći izveštaj koji ćemo napraviti treba da nam omogući da na papiru dobijemo popunjen obrazac sa podacima o ispitu poput onog na slici Sl.1. Pritiskom na dugme *New* u prozoru baze podataka, pokrenite proceduru za kreiranje novog izveštaja. U prvom prozoru na listi izaberite kreiranje uz pomoću vizarda (*Report Wizard*) a kao podlogu

izveštaja izaberite tabelu *TIspiti*. U sledećem prozoru, od raspoloživih polja tabele, izaberite sva sem polja *IspitID* (nije potrebno da se vidi ovaj podatak o ispitu). Sledeći okvir za dijalog ponudiće Vam grupisanje podataka po predmetima (SI.80 rb.1). U ovom slučaju ne želimo grupisanje podataka pa zato pritiskom na dugme sa oznakom "<" (SI.80 rb.2) uklonite polje za grupisanje.



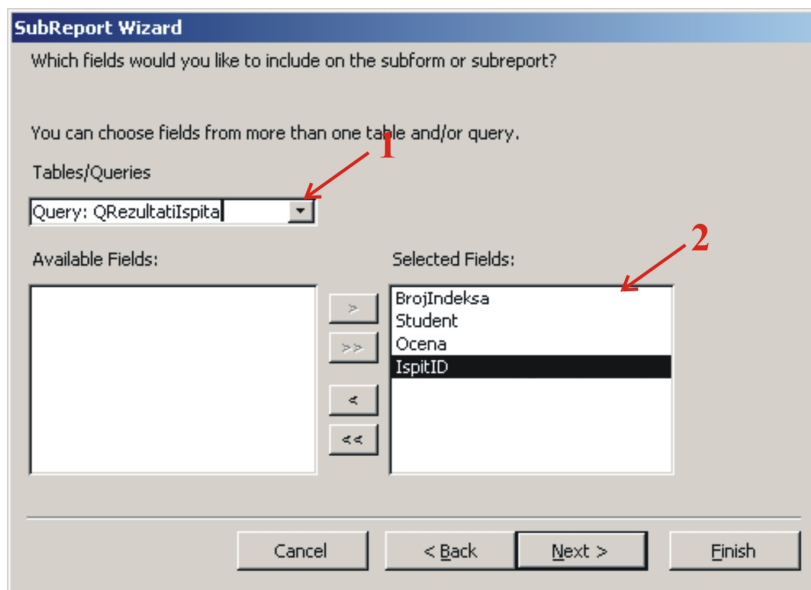
SI.80 Grupisanje podataka u izveštaju RIspiti

U sledećem koraku procedure za kreiranje izveštaja podesite sortiranje po polju *DatumPolaganja*. Na listi *Layout* narednog prozora izaberite opciju *Columnar*. Kao i ranije za stil izveštaja izaberite *Formal* a u zadnjem koraku u polju za naslov (naziv) izveštaja otkucajte *RIspiti*. Pokušajte sada da novonastali izveštaj formatirate tako da što više liči na obrazac sa slike SI.1. (Promenite naslov u *EVIDENCIJA ISPITA*, zamenite redosled polja za tekst *DatumIspita* i *IspitniRok*, labelama kontrola u delu sa detaljima promenite svojstvo *Caption* redom u: *Ispitni rok:*, *Datum:*, *Predmet* (SI. 81)).

SI.81 Deo izveštaja RIspitisa opštim podacima o ispitu

Pored opštih podataka o ispitu na izveštaju moramo prikazati i podatke o rezultatima ispita (studentima koji su polagali i njihovim ocenama). Kako se podaci o rezultatima ispita nalaze u tabeli *TRezultatiIspita* koja sa izvornom tabelom (*TIspiti*) stoji u relaciji jedan prema više, kao i kod formi, te podatke ćemo prikazati pomoću kontrole *Subreport*. U delu sa detaljima, ispod opštih podataka o ispitu, nacrtajte kontrolu za podizveštaj. U prvom koraku wizarda prihvatite ponudjenu opciju *Use existing Tablesand Queries*. To znači da podizveštaj kreiramo od početka koristeći postojeću

tabelu/upit. Sledeća pitanja na koja moramo dati odgovor su: koja je to tabela/upit od koje pravimo podizveštaj i koja se polja iz njih koriste (Sl.82)?



Sl.82 Izbor podataka koje želimo uključiti u podizveštaj RRezultatiIspita

Kao podlogu za ovaj podizveštaj izaberite upit *QRezultatiIspita* (Sl.82 rb.1) i iz njega sva polja (Sl.82, rb.2). Na pitanje kako se sinhronizuju podaci u izveštaju i podizveštaju prihvatite ponudjeni odgovor (veza se uspostavlja preko polja *IspitID*) i predjite u sledeći korak vizarda. Ostaje nam da podizveštaju damo ime. U polju za tekst otkucajte *RRezultatiIspita* i završite proceduru. Labeli ove nove kontrole promenite svojstva:

<i>Caption</i>	Rezultati ispita
<i>Font Size</i>	11
<i>Font Weight</i>	Bold

Ako izveštaj *RIspiti* otvorite u režimu za prikaz videćete da se podaci o ispitima kontinualno nastavljaju jedan za drugim. Tako se može desiti da na jednom listu papira imate odštampane podatke o više ispita. Mi ipak želimo da na jednom papiru imamo prikazane podatke o samo jednom ispitu. Zato, u dizajn modu ovog izveštaja ispod kontrole *Subreport:RRezultatiIspita* ucrtajte kontrolu *Page Break*. Ova kontrola, vizuelno prikazana kratkom tačkastom linijom, obaveštava Access da posle prikaza rezultata jednog ispita opšte podatke o sledećem ispitu počne na novoj strani.

Sada snimite i zatvorite izveštaj *RIspiti* a u dizajn modu otvorite *RRezultatiIspita*. Na ovom izveštaju postoji padajuća lista (*Combo Box*) *IspitID*. Kako nema potrebe da se taj podatak vidi u izveštaju selektujte ovu kontrolu i izbrišite je. (Ovo polje i dalje čini podlogu izveštaja ali se jednostavno više ne vidi.) Takođe, izbrišite i labelu u kojoj piše *IspitID*. Sada, po ugledu na izveštaja *RRangListaStudenata* formatirajte kontrole na ovom izveštaju (*Font Site, Caption*), ucrtajte liniju izmedju studenata, ugradite polje za tekst (i odgovarajuću labelu) u kojem će se prikazivati redni broj studenta. Snimite i zatvorite ovaj podizveštaj, a u režimu za prikaz otvorite *RIspiti*. Izveštaj bi trebalo da izgleda kao na slici Sl.83.

RIspti

EVIDENCIJA ISPITA

Ispitni rok:

Datum:

Predmet:

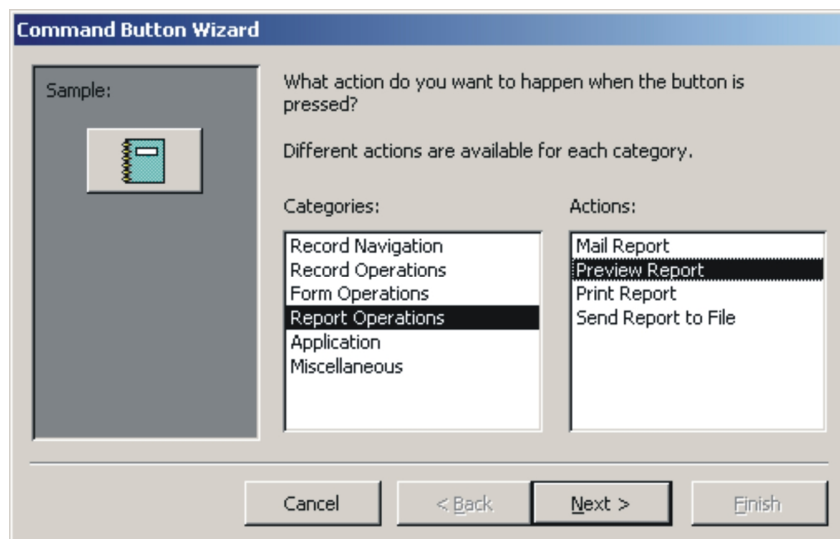
Rezultati ispita

Rb.	Indeks	Student	Ocena
1	11/80	Antić Zoran	6
2	14/80	Jović Jasmina	5
3	16/80	Marković Drinka	7
4	18/80	Petrović Petar	9
5	20/80	Tomić Milan	6

Sl.83 Izveštaj RIspti

Kako se ovaj izveštaj može sastojati od nekoliko strana (za svaki ispit po jedna ili više) u dnu prozora se nalaze kontrole (tasteri) za navigaciju (kao i na Sl.54) pomoću kojih možete prelistavati izveštaj. Pri štampi, kao i kod Word-ovog dokumenta, možete izabrati jednu, više uzastopnih ili sve strane.

Da bi smo dodatno unapredili funkcionalnost baze omogućićemo korisniku da nakon što pomoću forme *FIspti* unese podatke o ispitu te iste podatke i odštampa direktno sa forme. Otvorite formu *FIspti* i u donjem zaglavlju ucrtaite kontrolu *Command Button*. Na pitanje koju aktivnost želite pokrenuti pomoću ovog dugmeta u listi kategorija izaberite *Report Operations* a u listi akcija *Preview Report* (Sl.84).



Sl.84 Kreiranje komandnog dugmeta za pregled izveštajai

U sledećem prozoru treba izabrati izveštaj koji želite otvoriti pritiskom na dugme. Selektujte izveštaj *RIspti*. U narednom koraku pritiskom na dugme *Finish* završite proceduru kreiranja komandnog dugmeta. Ako sada u režimu za prikaz forme *FIspti* kliknemo na upravo formiran taster (Sl.85) otvoriće se izveštaj *RIspti*.



Sl.85 Izgled donjeg zaglavlja forme FIspti

Medjutim, u izveštaju će biti podaci o svim ispitima a mi želimo samo podatke o tekućem ispitu, o onome koji se upravo vidi na formi. Da bi smo to postigli, otvorite izveštaj *RIspti* u dizajn modu i podesite sledeća svojstva:

<i>Filter</i>	IspitID=Forms!FIspti!IspitID
<i>Filter On</i>	Yes

Posle ovoga, kada pritiskom na dugme sa slike Sl.85 otvorite izveštaj, biće prikazani samo podaci o tekućem ispitu.