

დ. აგთანდილაშვილი

კომპიუტერული დანერგვები

II ნაწილი

(მონაცემთა ბაზასთან მუშაობა)

თბილისი 2006

საქართველოს ტექნიკური უნივერსიტეტი

დ. ავთანდილაშვილი

კომპიუტერული დავრობება

II ნაწილი

(მონაცემთა ბაზასთან მუშაობა)

დამტკიცებულია სახელმძღვანელოდ სტუ-ს
სარედაქციო-საგამომცემლო საბჭოს მიერ

თბილისი 2006

წიგნში განხილულია საქართველოს ტექნიკური უნივერსიტეტის 1201 სპეციალობის - "მანქანათმშენებლობის ტექნოლოგია" და 1201.06, 1201.05 სპეციალიზაციების შესაბამისად "მანქანათსაშენი წარმოების ტექნოლოგიური პროცესების ოპტიმიზაცია და კომპიუტერული დაპროექტება", "სამედიცინო და პრეციზიული მიკროიარაღების კონსტრუირება და წარმოება" სტუდენტთათვის. იგი აგრეთვე დიდ დახმარებას გაუწევს მაგისტრანტებსა და ასპირანტებს.

რეცენზენტები: პროფ. თ. გეგეჭკორი

პროფ. თ. მჭედლიშვილი

შესავალი

თანამედროვე ტექნოლოგიების მეშვეობით მარტივდება პროექტის დამუშავების პროცედურა. მონაცემთა ბაზაში შესაძლებელია დიდი ინფორმაციის შენახვა, მოდიფიკაცია და საჭირო ინფორმაციის მოძიება. Ms Access-ი და Visual Basic-ი საკმაოდ სრულყოფილი საშუალებებია მონაცემთა ბაზაში მოხერხებულად მუშაობისათვის.

წიგნში თავების მიხედვით განხილულია როგორც Ms Access-ის ობიექტების შექმნა და ურთიერთკავშირი, ისე Visual Basic-ის ფორმების დამუშავება, თვისებების დაყენება და მარტივი პროგრამული კოდის საშუალებით მათი ამოქმედება. გარდა ამისა, წიგნში განხილულია მაგალითები მართვის ცალკეული ელემენტების შესასწავლად.

I თავში გაეცნობით მონაცემთა ბაზის რაობას, მისი ობიექტების სტრუქტურის შედგენას, Ms Access-ში ცხრილების სხვადასხვაგვარად შექმნას, ცხრილის ველებთან მუშაობას, ცხრილის სტრუქტურის მოდიფიკაციას და ცხრილებს შორის კავშირს.

II და III თავებში განხილულია ცხრილების შექმნის სხვადასხვა გზა, მონაცემების დათვალიერება და რედაქტირება, ძებნა და დახარისხება.

IV თავში განხილულია ფორმების დაგეგმვა და შექმნა როგორც კონსტრუქტორის, ისე ოსტატის რეჟიმში, მართვის ელემენტებთან მუშაობა.

V თავში განხილულია მოთხოვნები, მათი შექმნა, გამოსათვლელი ველი და არჩევის კრიტერიუმის გამოთვლა.

VI თავში აღწერილია უწყისების შექმნა უწყისის ოსტატის მეშვეობით, რედაქტირება და ბეჭდვა.

VII თავში განხილულია Visual Basic, დამუშავების ინტეგრირებული საშუალება – IDE, ფორმის დიზაინერი, მართვის ელემენტები და მათი გამოყენება, ობიექტების თვისებების დაყენება და სათანადო პროგრამული კოდის ჩაწერა.

VIII თავში განხილულია მონაცემთა ბაზის მართვის ელემენტი, მისი თვისებების დაყენება, ფორმაზე ბრძანებითი ღილაკების დამატება და სათანადო პროცედურაში პროგრამული კოდის ჩაწერა.

IX თავში განხილულია რელაციური მონაცემთა ბაზა, ველის ტიპები, პროექტი ველის ტიპების ტესტირებისათვის.

იგულისხმება რომ ამ წიგნით სარგებლობისას მომხმარებელს შეუძლია კლავიატურისა და თავის გამოყენება, Windows-ის ნებისმიერი ვერსიით სარგებლობა და ერკვევა მის ინტერფეისში იმდენად მაინც, რომ ჩატვირთოს პროექტი.

გარდა ამისა, დაინსტალებული უნდა გქონდეთ მაგალითები Visual Basic-ის კომპაქტდისკოდან.

წიგნი განკუთვნილია მექანიკა-მანქანათმშენებლობის ფაკულტეტის სტუდენტებისათვის, მაგისტრანტებისა და მათთვის, ვისაც სურს დაეუფლოს მონაცემთა ბაზაში პროფესიულ მუშაობას.

ავტორი დიდ მადლობას უხდის მექანიკა-მანქანათმშენებლობის ფაკულტეტის სასწავლო-კომპიუტერული ცენტრის თანამშრომლებს ო. პიტავასა და ე. ნინოშვილს, რომლებმაც დახმარება გაუწიეს მას სახელმძღვანელოს კომპიუტერულ მომზადებაში.

ავტორი მადლიერებით მიიღებს ყველა შენიშვნასა და სურვილს და გაითვალისწინებს სახელმძღვანელოს შემდგომი დახვეწისა და სრულყოფისას.

I ო ა ზ ი

მონაცემთა ბაზის შექმნა

მონაცემთა ბაზა არის ინფორმაციის მოწესრიგებული ერთობლიობა. მასზე სრულ წარმოდგენას გვიქმნის სატელეფონო ცნობარი, ვინაიდან მასში მოცემულია ყოველი აბონენტის გვარი, სახელი, მისამართი, ტელეფონის ნომერი და ეს მონაცემები გარკვეული წესითაა დალაგებული.

მონაცემთა ბაზის მართვის სისტემა (მზმს) ეწოდება კომპიუტერულ პროგრამას, რომლების მეშვეობითაც მომხმარებელს მონაცემთა ბაზაში შეაქვს არსებული ინფორმაცია, ინახავს, პერიოდულად იძახებს მას, ახარისხებს, აანალიზებს და ბეჭდავს. კომპიუტერულ მონაცემთა ბაზაში ინფორმაცია ინახება ცხრილებში, რომლებიც ჰგვანან ელექტრონულ ცხრილებს. ცხრილში სვეტის სათაურს ეწოდება ველის სახელი, თვით სვეტს ველი, ხოლო სტრიქონს - ჩანაწერი.

მაშასადამე, მონაცემთა ბაზა შეიცავს ცხრილებს, რომლებიც შედგება სტრიქონების ანუ ჩანაწერებისაგან და სვეტების ანუ ველებისაგან.

მაგალითად

სახელი	გვარი	ეროვნება	წლოვანება	ტელეფონი
გიორგი	ჯგუშია	ქართველი	23	22-40-56
ივანე	სამხარაძე	ქართველი	30	53-55-70
ტიგრან	პეტროსიანი	სომეხი	35	92-45-58
ალისა	ნეჩიპურენკო	უკრაინელი	45	99-34-54
ალა	პუგანოვა	რუსი	55	62-48-44

სურ. 1. 1. მონაცემთა ბაზის ცხრილი

არსებობს მონაცემთა ბაზის მართვის ორგვარი სისტემა

1. ფაილების მართვის სისტემა ინფორმაციას ინახავს ფაილებში და ხდება მონაცემების მიმდევრობით დამუშავება. ფაილების მართვის სისტემის ნაკლია ის, რომ ერთი და იმავე მონაცემის ჩანაწერი ერთდროულად რამდენიმე ადგილასაა განთავსებული და ამდენად მოუქნელია;

2. რელაციურ მონაცემთა ბაზის მართვის სისტემა შესაძლებლობას იძლევა უფრო მოხერხებულად შევძლოთ მონაცემთა მართვა და მონაცემებით მანიპულირება, რაც განაპირობებულია მონაცემების ურთიერთდამოკიდებულებით. შეიძლება ნაკრებში შენახული მონაცემების განახლება და გამოტანა სხვა მასთან დაკავშირებულ ნაკრებში შენახული ინფორმაციის საფუძველზე.

რელაციური მონაცემთა ბაზა განისაზღვრება როგორც მონაცემთა ერთობლიობა მის ელემენტებს შორის კავშირის მითითებით. მარტივად, მონაცემთა ბაზა არის ერთმანეთთან დაკავშირებული მონაცემების ერთობლიობა.

განსხვავება რელაციურ მონაცემთა ბაზასა და ფაილების მართვის სისტემებს შორის უფრო გასაგები რომ გახდეს განვიხილოთ მაგალითი. შევადგინოთ მონაცემთა ბაზა ვიდეოფირების გამქირავებელი ფირმისათვის. ამ შემთხვევაში უნდა გვქონდეს ინფორმაცია დამქირავებლების არსებული ვიდეოფირებისა და ამ ფირების გადასახადის შესახებ. მონაცემთა ფაილების მართვის სისტემაში, დამქირავებლის სახელი და ტელეფონის ნომერი ყოველთვის უნდა იქნეს შეტანილი ვიდეოფირის დასახელებასთან ერთად, ხოლო მონაცემთა რელაციურ სისტემაში დამქირავებლის სახელი და მისი ტელეფონის ნომერი ავტომატურად იქნება გამოძახებული დამქირავებლისა და ვიდეოფირის დასახელება დამაკავშირებელი სიიდან და ქვითარში შევა ამ ინფორმაციის ხელახლა შეტანის გარეშე.

ყოველდღიურ ყოფა-ცხოვრებაში პერსონალური კომპიუტერი და კომპიუტერული პროგრამები გამოიყენება სამუშაოს გასაადვილებლად, რისთვისაც Microsoft Office გთავაზობთ იარაღების საშუალებებს. მთავარია იცოდეთ როდის უნდა გამოიყენოთ მონაცემთა ბაზის მართვის საშუალება – Access-ი.

მონაცემთა ბაზაში ხდება ინფორმაციის შეგროვება და შენახვა. ქვემოთ ჩამოთვლილია იმის მაგალითები, თუ რა სახის ინფორმაციის შენახვაა შესაძლებელი მონაცემთა ბაზაში:

1. მონაცემები რომელიმე დაწესებულების თანამშრომელთა შესახებ;
2. მონაცემები რომელიმე წარმოების პროდუქტის ჩამონათვალზე;
3. კლიენტების საანკეტო მონაცემები;
4. დაზღვევის ამოცანებისათვის ოჯახის ქონების აღწერა;
5. მონაცემები სამუშაო დროისა და დასაქმების შესახებ;
6. მონაცემები გაყიდვების კონტრაქტების შესახებ;
7. მონაცემები მომწოდებლების შესახებ;
8. მონაცემები სტუდენტებსა და აუდიტორიების შესახებ;
9. მონაცემები ვიდეოფირების შესახებ კოლექციაში;
10. ცნობები ინვესტიციების შესახებ და ა. შ.

ყურადღება მიაქციეთ იმას რომ ეს ამოცანები ეხება ინფორმაციის შეგროვებას და არა გამოთვლას. მონაცემთა ბაზაში შეიძლება შესრულდეს სტატისტიკური და ფინანსური გამოთვლები, მაგრამ გამოთვლები მონაცემთა ბაზის მართვის სისტემებით ისე სწრაფად ვერ ხდება, როგორც ელექტრონულ ცხრილებში. ერთჯერად შესასრულებელი გამოთვლებისათვის უფრო მოსახერხებელია ელექტრონული ცხრილები, ვიდრე მონაცემთა ბაზა. თუ საჭიროა დიდი ინფორმაციის ფენებად დალაგება, ანალიზი და ამ ინფორმაციის რამდენიმე წლის განმავლობაში შეგროვება, შენახვა და გაახლება, მოხერხებულია მონაცემთა ბაზის მართვის სისტემის გამოყენება. მონაცემთა ბაზის მართვის სისტემები: Access, Delfin, Foxpro, Visual Basic და ა. შ.

1.1. ზოგადად Access-ის შესახებ

Microsoft Access არის რელაციურ მონაცემთა ბაზის მართვის სისტემა, რომელიც დამუშავებულია სპეციალურად Windows (ფანჯრების) გარემოსათვის.

Access-ის მეშვეობით შესაძლებელია შემდეგი ამოცანების შესრულება:

1. მონაცემების ორგანიზება ადვილად მართვად შეკავშირებულ ელემენტებში;
2. მონაცემების შეტანა, მათი მოდიფიკაცია და ძებნა;
3. საბაზისო კრიტერიუმების საფუძველზე მონაცემის მოძიება;
4. საკუთარი ფორმებისა და უწყისების შექმნა;
5. იმ ამოცანების ავტომატიზება, რომლებსაც ხშირად იყენებთ მონაცემთა მართვისას;
6. გრაფიკებისა და დიაგრამების შექმნა;
7. უწყისებსა და ფორმებზე გამოსახულების დამატება;
8. მონაცემთა ბაზის მართვის სისტემისათვის საკუთარი პროექტის შექმნა, რომელიც აღჭურვილი იქნება დიალოგური ფანჯრებით, მმართველი კლავიშებითა და ბრძანებების ჩამონათვალით (მენიუთი).

1.2. მონაცემთა ბაზის ობიექტები

მონაცემთა ბაზის შექმნამდე უნდა გაირკვეს მონაცემთა ბაზის კომპონენტები. Access-ში მონაცემთა ბაზა არის ფაილი, რომელიც შეიცავს ნაკრებ ინფორმაციას. აქ ყოველი მონაცემთა ბაზის შემადგენელია ძირითადი ობიექტები, ცხრილები, მოთხოვნები, ფორმები, უწყისები, მაკრობრძანებები და მოდულები. Access-ის ძირითადი ობიექტები აღწერილია ცხრილში 1.1.

ობიექტი	აღწერა
ცხრილი	მონაცემები ინახება ცხრილში, რომლებიც მოგაგონებს ელექტრონულ ცხრილს და შედგება სტრიქონებისა და სვეტებისაგან.
მოთხოვნა	მონაცემთა ბაზიდან, მომხმარებლის მიერ მითითებული კრიტერიუმის მიხედვით გამოიძახებს მონაცემებს. მოთხოვნების მეშვეობით შესაძლებელია ველების დათვალიერება ერთზე მეტი მონაცემთა ბაზიდან.
ფორმა	ცხრილიდან ან მოთხოვნიდან ეკრანზე გამოიტანს ინფორმაციას, მომხმარებლის მიერ მითითებული ფორმატით. ფორმის საშუალებით შესაძლებელია მონაცემების დათვალიერება, შესწორება და დაბეჭდვა. მას შეუძლია გამოიტანოს მონაცემი ერთზე მეტი მონაცემთა ბაზიდან.
უწყისი	ეკრანზე გამოიტანს და ბეჭდავს მომხმარებლის მიერ მითითებული ფორმატის ცხრილს, ფორმას ან მოთხოვნას.
მაკრობრძანება	მონაცემთა ბაზასთან მუშაობისას ხშირად შესასრულებელი მოქმედებების ავტომატიზებას ახდენს, მომხმარებლის მიერ მითითებული ბრძანებებით.
მოდული	ახდენს კომპლექსური ოპერაციების ავტომატიზებას და პროგრამისტს აძლევს უფრო სრული კონტროლის საშუალებას, ვიდრე მაკრობრძანება. მოდული არის რამდენიმე პროცედურა, რომელიც დაწერილია პროგრამირების ენაზე – Visual Basic.

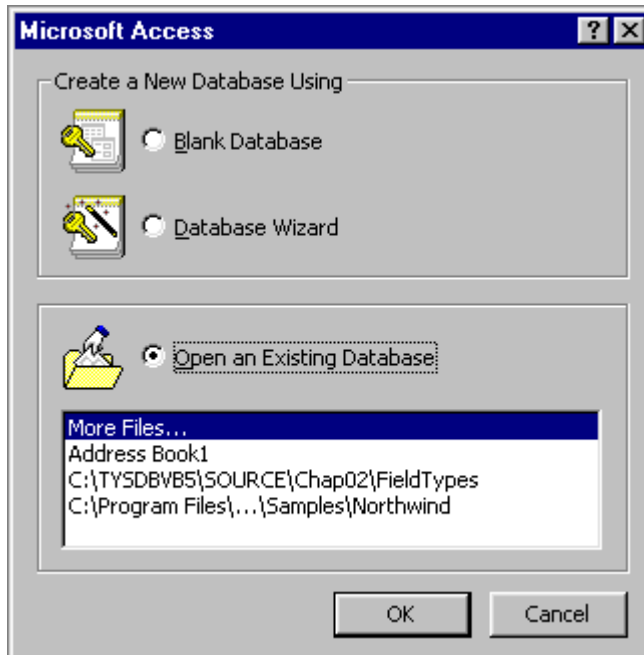
Access-ში ობიექტი არის ის, რაც შეგიძლიათ აირჩიოთ და რით მანიპულირებაც გსურთ. ცხრილი, ველი ცხრილში, ფორმა და ლიდაკი – ყველა ობიექტებია.

Access-ის გაშვება სხვადასხვანაირად შეგიძლიათ. ერთ-ერთი გზაა მონახოთ Access-ის ჩამტვირთავი ფაილი და გაუშვათ შესრულებაზე. მაგალითად, აირჩიეთ Start→Programs→Microsoft Access და დაარტყით კლავიშს Enter (შეტანა).

1.3. მონაცემთა ბაზის შექმნა

Access ჩატვირთვის შემდეგ, ეკრანზე გამოჩნდება დიალოგური ფანჯარა Microsoft Access. სათანადო პარამეტრის არჩევის შემდეგ, შეგიძლიათ შექმნათ ახალი მონაცემთა ბაზა ან გახსნათ უკვე არსებული. ახალი ბაზის შესაქმნელად დიალოგურ ფანჯარაში აირჩიეთ გადამრთველი Blank Database (მონაცემთა ბაზის სტრუქტურა), შეიქმნება ცარიელი ბაზა, რომელიც არ შეიცავს ობიექტებს ცხრილების, ფორმების, მოთხოვნების და სხვათა სახით. თუ აირჩევთ გადამრთველს Database Wizard (მონაცემთა ბაზის ოსტატი), მაშინ გამოჩნდება მონაცემთა ბაზის შაბლონების სია, რომელთაგან აირჩევთ თქვენთვის სასურველს. შეგიძლიათ „ოსტატს“ მოსთხოვოთ ნიმუში, რომელიც მოგეხმარებათ მუშაობის დაწყებაში. გადამრთველით Open an Existing Database (მონაცემთა ბაზის გახსნა) გამოიტანება ბოლო დროს გამოყენებული მონაცემთა ბაზის სიის ჩამონათვალი. განყოფილება „ფაილები კიდევ“ გთავაზობთ უკვე არსებულ მონაცემთა ბაზაში შედგენას, იხ. სურ. 1.1.

თუ დიალოგურ ფანჯარაში არ აირჩევთ არცერთ გადამრთველს და ისე დახურავთ, ეკრანზე ჩანს Access-ის მენიუ, სადაც ბრძანებების უმრავლესობა აქტიური არ არის (ვერცხლისფერია). იარაღების სტრიქონში მოთავსებულია კლავიშები New (ახალი), Open (გახსნა), რომლებიც მოგეხმარებათ შესაბამისად ახალი ბაზის შექმნასა და არსებული ბაზის გახსნაში.

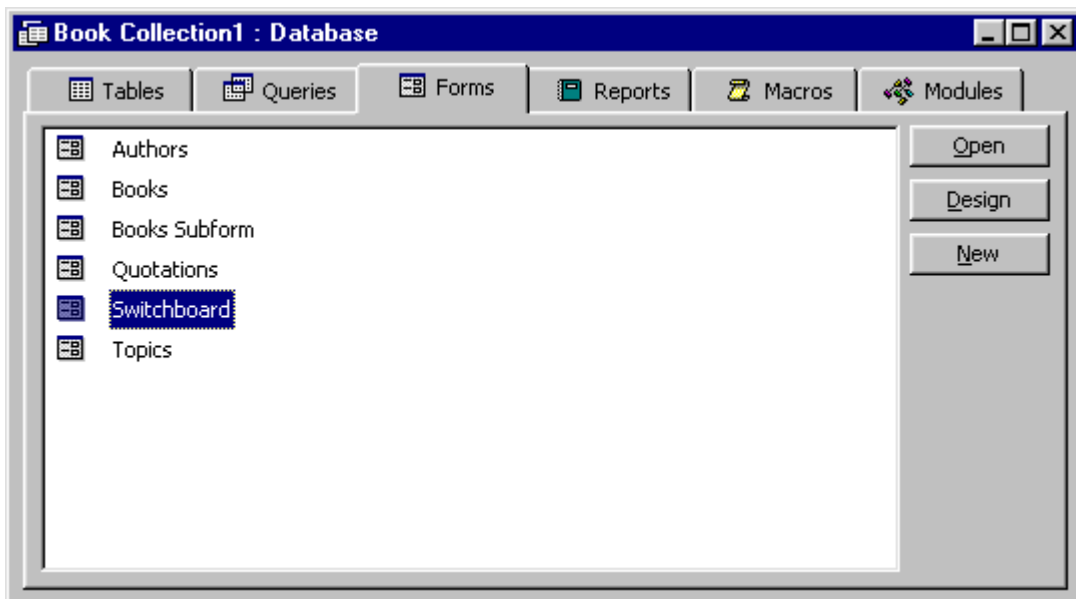


სურ. 1.1. დიალოგური ფანჯარა - Microsoft Access

☒ არსებული მონაცემთა ბაზის გასახსნელად აირჩიეთ ბაზის სახელი და თავგით მასზე ორჯერ დააწკაპუნეთ.

არსებული მონაცემთა ბაზის გასახსნელად იქცევით შემდეგნაირად:

1. ირჩევთ მენიუდან ბრძანებას File→Open Database (ფაილი→ბაზის გახსნა), გამოვა დიალოგური ფანჯარა;
2. ტექსტურ ველში Look In მიუთითეთ გასახსნელი ბაზის სახელი ან მისამართი.



სურ. 1. 2. მონაცემთა ბაზის დიალოგური ფანჯარა

3. ველში File Name (ფაილის სახელი) მიუთითეთ ბაზის სახელი;
4. დააწკაპუნეთ ღილაკზე Open (გახსნა). გაიხსნება ბაზა. იხილეთ სურ. 1. 2.

იმისათვის რომ დახუროთ მონაცემთა ბაზა, აირჩიეთ მენიუდან ბრძანება File→Close (ფაილი→დახურვა) ან დააწკაპუნეთ დახურვის ღილაკზე.



ერთდროულად რამდენიმე მონაცემთა ბაზის გახსნა შეუძლებელია.

მონაცემთა ბაზის ფანჯარა შეიცავს ობიექტებისათვის ცალ-ცალკე ველებს. აირჩიეთ ის ობიექტი რომელთანაც გინდათ მუშაობა. მონიშნეთ ამ ობიექტის სახელი და დაარტყით ღილაკს Open (გახსნა) ან ამ სახელზე მანველებელით ერთჯერ დააწკაპუნეთ. ღილაკი Open (გახსნა) იცვლის სახელს უწყისებისათვის გახდება Preview (წინასწარ დათვალიერებით), ხოლო მაკრობრძანებებისა და მოდულებისათვის - Run (გაშვებით). ახალი მონაცემთა ბაზის შესაქმნელად ან არსებული ბაზის მოდიფიცირებისათვის ისარგებლეთ ღილაკებით New (ახლის შექმნა), Design (კონსტრუქტორი).

14. მონაცემთა ბაზის სტრუქტურის დაგეგმვა და შექმნა

ცხრილების, ფორმებისა და უწყისების შექმნამდე, დაიწყეთ მონაცემთა ბაზის სტრუქტურის დაგეგმვა. მონაცემთა ბაზის სტრუქტურის დაგეგმვისას უნდა გაითვალისწინოთ შემდეგი:

- ❑ დაიწყეთ არსებული ბაზის ანალიზით. დაათვალიერეთ არსებული ფორმები და უწყისები. განსაზღვრეთ მონაცემების მიღების წყარო (თუ კომპიუტერულია, შეიძლება თუ არა მათი იმპორტირება ან დაკავშირება მონაცემთა ბაზასთან). შეხედით იმ პირებს რომლებიც ისარგებლებენ მონაცემთა ბაზით და გაითვალისწინეთ მათი მოთხოვნები. შეაფასეთ ის ამოცანები, რომელთა გადასაწყვეტად საჭიროა მონაცემთა ბაზა მაგალითად, ყოველკვირეული უწყისის შედგენა, მონაცემების გადაცემა, დახარისხება და ანალიზი;

- ❑ მას შემდეგ, რაც გაარკვევთ, თუ სად იქნება მონაცემები შენახული და როგორ მოხდება მათი გამოძახება ანალიზისათვის, დააჯგუფეთ ისინი რაიმე ნიშან-თვისების მიხედვით (მაგალითად, გამოყავით მყიდველის შესახებ მონაცემები გადასახადის ცნობებისაგან). ეს ჯგუფები შემდგომ გადიქცევა ცხრილებად;

- ❑ განსაზღვრეთ იმ ინფორმაციის ტიპი, რომლის შენახვაც გსურთ ყოველ ცხრილში (მაგალითად, მყიდველების ცხრილი შეიძლება შეიცავდეს მყიდველის გვარს, მისამართს და ტელეფონს). ამ კატეგორიის ინფორმაციას ყოველ ცხრილში ჰქვია ველი;

- ❑ მონახეთ ცხრილების საერთო ელემენტები (მაგალითად, მყიდველის გვარი შეიძლება იყოს საერთო ელემენტი მყიდველის ცხრილისთვისაც და გადახდის შესახებ შემცველ ცნობების ცხრილისთვისაც). ამ საერთო ელემენტს ეწოდება საგასაღებო ველი;

- ❑ განსაზღვრეთ მოთხოვნებისათვის კრიტერიუმები და კითხვები, რომლებიც შეიძლება დაუსვას.

- ❑ განსაზღვრეთ თუ რა სახის უნდა იყოს ფორმები და უწყისები;

- ❑ იფიქრეთ ხშირად შესასრულებელი ოპერაციების, როგორიცაა ფორმების გამოძახება, მოთხოვნების შესრულება და უწყისების ბეჭდვა, მეტნაკლებად ავტომატიზებაზე;

- ❑ გაითვალისწინეთ მონაცემთა უსაფრთხოების უზრუნველყოფის საკითხი მაგალითად, სარეზერვო დუბლების შექმნა, ფაილების ერთობლივი გამოყენება და ამ ფაილებზე ქსელიდან შეღწევა.

პ რ ო ბ ლ ე მ ა და მ ი ხ ი გ ა დ ა წ ყ ვ ე ტ ა

Access-ის ჩატვირთვისას არ ჩანს თქვენი მონაცემთა ბაზა. როგორ მოიქცევით?

დიალოგური ფანჯარა Access-ი ავტომატურად არ ხსნის მონაცემთა ბაზას, რადგან მიზანს რომ მიაღწიოთ, საჭიროა მიუთითოთ სრული კატალოგი, საქაღალდე და ფაილის სახელი.

მონაცემთა ბაზის სტრუქტურის შექმნისას გაირკვა რომ ცხრილი დიდია - შეიცავს უამრავ ველს. როგორ იქცევით?

ნახეთ, რა სახის ინფორმაციას შეიცავს ეს ველები, არის თუ არა სხვადასხვა ველში შენახული ინფორმაცია ერთმანეთის მსგავსი შინაარსის. მოიფიქრეთ, როგორ დაყოთ დიდი ცხრილი პატარ-პატარა ცხრილებად თემის მიხედვით. გახსოვდეთ, რომ საჭიროების შემთხვევაში შეგიძლიათ დაყოფის შედეგად მიღებული ცხრილების ერთმანეთთან დააკავშირება. „დაყავი და იბატონე“ არის ძირითადი წესი რელაციურ მონაცემთა ბაზის მართვაში.

მონაცემთა ბაზის დამუშავებული სტრუქტურა შეიცავს საკმაოდ ბევრ ცხრილს, ძნელი იქნება თუ არა მასთან მუშაობა.

ნახეთ, ხომ არ მეორდება სხვადასხვა ცხრილში ისეთი ველები, რომლებიც არ გჭირდებათ. გარდა ამისა, შეგიძლიათ იფიქროთ ერთზე მეტი ველების შემცველ რამდენიმე ცხრილის გაერთიანებაზე. მაგალითად, თუ ცალ-ცალკე გაქვთ ყოველკვირეული გაყიდვების ცხრილები, შეგიძლიათ შექმნათ ერთი მოზრდილი ცხრილი, რომელიც შეიცავს ველებს – “კვირის დასაწყისი” და “კვირის ბოლო” შემდეგ კი ისარგებლოთ მოთხოვნით, რომ გამოეყოს კონკრეტული კვირის მონაცემები.

ახალი მონაცემთა ბაზის შექმნა

მას შემდეგ, რაც მონაცემთა ბაზის სტრუქტურა შექმნილია, შეგიძლიათ შექმნათ ახალი ბაზა. მონაცემთა ბაზაში ობიექტები შეგიძლიათ შექმნათ როგორც თქვენ თვითონ, ისე ავტომატური ოსტატის საშუალებით.

☒ თუ მონაცემთა ბაზას ქმნის დამწყები მომხმარებელი, სასურველია მონაცემთა ბაზის ოსტატის გამოყენება. მოდიფიცირება ყოველთვის შეიძლება.

1.5. მონაცემთა ბაზის ოსტატის გამოყენება

ოსტატი შეიცავს 20-ზე მეტ სრულ ბაზას, ცხრილების, მათ შორის განსაზღვრული კავშირებით, ფორმების, მოთხოვნებისა და უწყისების ჩათვლით. ოსტატის უპირატესობა ისაა, რომ შეგიძლიათ არჩეული ცხრილის ნიმუშის სტრუქტურით შექმნათ ცარიელი მონაცემთა ბაზა, მოახდინოთ ცხრილის სტრუქტურის მოდიფიცირება და ჩაწეროთ საჭირო მონაცემები.

ოსტატს შეუძლია შემდეგი შაბლონების შემოთავაზება:

სამისამართო წიგნი	პირადი ქონება	რესურსები
ბიბლიოთეკა	ღონისძიებები	რეცეპტები
ღვინის სარდაფი	მუსიკალური კოლექცია	საწყობი
ვიდეოჩანაწერების	ძირითადი ფონდები	სტუდენტები და
კატალოგი		მეცადინეობები
სამუშაოზე შეკვეთები	შეკვეთების მიღება	ვარჯიშები
მთავარი წიგნი	პროექტები	მონაწილენი
კონტაქტები	გასაგალი	ფოტოალბომები

ოსტატის საშუალებით მონაცემთა ბაზის შესაქმნელად შეასრულეთ შემდეგი სეგლები:

1. დიალოგურ ფანჯარაში Microsoft Access-ი აირჩიეთ Database Wizard (მონაცემთა ბაზის ოსტატი) ან თუ ეს ფანჯარა უკვე დახურეთ, მენიუდან აირჩიეთ ბრძანება File→New Database (ფაილი→ახალი ბაზის შექმნა), გამოჩნდება დიალოგური ფანჯარა Create (შექმნა), გახსენით ველი Database (მონაცემთა ბაზა);

2. აირჩიეთ რომელიმე პიქტოგრამა და ორჯერ დააწკაპუნეთ მასზე;

3. გამოვა დიალოგური ფანჯარა – ახალი მონაცემთა ბაზის ფაილი, მიუთითეთ სახელი და ადგილი, სადაც მისი ჩაწერა გასურთ;

4. დააწკაპუნეთ ღილაკს Create (შექმნა);

5. ადევნეთ თვალი ოსტატის მიერ შემოთავაზებულ კითხვებს და გაეცით სათანადო პასუხები სასურველი მონაცემთა ბაზის შესაქმნელად.

ცარიელი მონაცემთა ბაზის შექმნა

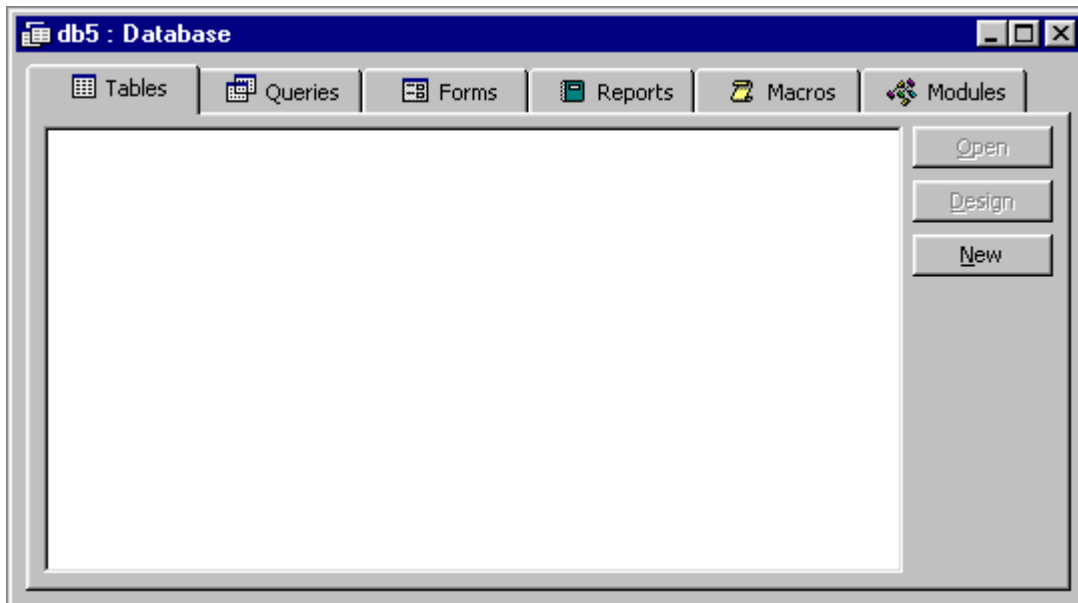
თუ შაბლონებში არ არის თქვენი სამუშაოს შესაბამისი ნიმუში, შეგიძლიათ შექმნათ ცარიელი მონაცემთა ბაზა, რომელიც არ შეიცავს ობიექტებს, კავშირებს ან ნიმუშებს. ამისათვის გააკეთეთ შემდეგი სვლები:

1. აირჩიეთ მენიუდან ბრძანება File→New Database (ფაილი→ახალი მონაცემთა ბაზა), ან იარაღების სტრიქონიდან დააწკაპუნეთ კლავიშზე New (ახალი), გამოჩნდება დიალოგური ფანჯარა New Database (ახალი ბაზა), გახსენით ველი General (საერთო), შემდეგ ორჯერ დააწკაპუნეთ პიქტოგრამაზე Blank Database (მონაცემთა ბაზის სტრუქტურა) და ერთჯერ დააწკაპუნეთ ღილაკზე Create (შექმნა);

2. ველში Save in (შენახვა) აირჩიეთ ის დისკი და საქაღალდე, სადაც გინდათ რომ ახალი ბაზა ჩაწეროთ, ტექსტურ ველში File Name (ფაილის სახელი) შეიტანეთ ის სახელი, რასაც არქმევთ თქვენს ბაზას;

3. დააწკაპუნეთ ღილაკზე Create (შექმნა).

მას შემდეგ, რაც Access-ი შექმნის ახალ ბაზას, გაიხსნება მონაცემთა ბაზის ცარიელი ფანჯარა, რომელიც მზადაა ნებისმიერი ახალი ობიექტის შესაქმნელად, იხილე სურ. 1.3.



სურ. 1. 3. ცარიელი მონაცემთა ბაზის დიალოგური ფანჯარა

II თ ა ვ ი

ცხრილების შექმნა

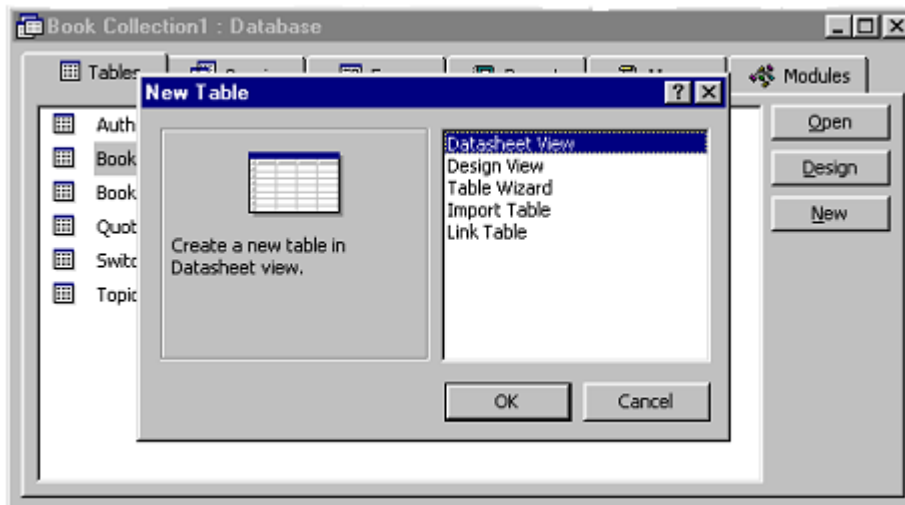
2. 1. ახალი ცხრილის შექმნა

მონაცემთა ბაზაში ახალი ცხრილის შესაქმნელად, გახსენით მონაცემთა ბაზის ფანჯარა, აირჩიეთ ველი Table (ცხრილი) და დააწკაპუნეთ ღილაკზე Great (შექმნა). ეკრანზე გამოჩნდება დიალოგური ფანჯარა New Table (ახალი ცხრილი). Access-ი გთავაზობთ ცხრილის შექმნის შემდეგ ვარიანტებს:

- ☐ მონაცემის ფურცლის ნახვა (ცხრილის რეჟიმი);
- ☐ ცხრილის სტრუქტურის შედგენა (კონსტრუქტორი);
- ☐ ცხრილის ოსტატი;
- ☐ ცხრილების შემოტანა (იმპორტი);
- ☐ ცხრილებს შორის კავშირი (ქსელი).

ცხრილის რეჟიმი შესაძლებლობას იძლევა შექმნათ ცარიელი ცხრილი და მასში მონაცემები შეიტანოთ ისევე, როგორც ელექტრონულ ცხრილებში შეგაქვთ. Access-ი გააანალიზებს შეტანილ მონაცემებს და მიანიჭებს ველებს საჭირო ფორმატსა და ტიპს. იხილეთ სურ. 2.1. დამოყვები მომხმარებლისათვის, უმჯობესია, მონაცემების შეტანის ცხრილის რეჟიმის არჩევა.

კონსტრუქტორი შესაძლებლობას იძლევა, ცხრილი შექმნათ „ნულიდან“. ამ რეჟიმში იქმნება ცხრილის სტრუქტურა. კერძოდ, შეგვაქვს ველის სახელი, ვირჩევთ ველის ტიპს, სახელს და ფორმატს.



სურ. 2. 1. ახალი ცხრილის შექმნის დიალოგური ფანჯარა

ცხრილის ოსტატი გთავაზობთ ყველაზე მეტად გავრცელებულ ცხრილის ტიპებს, საიდანაც შეგიძლიათ აირჩიოთ და გამოიყენოთ თქვენთვის საჭირო ვარიანტი. ოსტატის მეშვეობით შეგიძლიათ ცხრილების უამრავი შაბლონიდან სწრაფად აირჩიოთ და შეადგინოთ სასურველი ცხრილი.

მონაცემთა ბაზის იმპორტისას (შემოტანისას) ხდება მონაცემების გადმოწერა სხვა ბაზიდან და ავტომატურად იქმნება ცხრილი.

ცხრილების დაკავშირებისას, ისინი არ გადმოიქანება მუშა ბაზაში და შეიძლება მათი დათვალიერებაც და რედაქტირებაც.

2.2. ცხრილის შექმნა ცხრილის რეჟიმში

Access-ი შესაძლებლობას იძლევა ცხრილის შექმნის რეჟიმში შექმნილ ცხრილში მაშინვე დაიწყოთ მონაცემების შეტანა – Datasheet-ში (მონაცემის ფურცელზე). ეს ფურცელი ჰგავს Excel-ის სამუშაო ფურცელს. როდესაც მონაცემების შეტანას დაამთავრებთ, შეინახეთ ფურცელი, Access-ი კი ავტომატურად შექმნის ცხრილს. ახალ ცხრილში ველის სახელებია: ველი1, ველი2 და ა.შ., რომლებიც შეგიძლიათ შეცვალოთ უფრო გასაგები და სამუშაოსათვის შესაფერი სახელით.

Table1 : Table						
	num	gvari	dabadeba	Xelfasi	misamarti	telefoni
	1	chitaia	10/17/80	5000000	gurdiani 12/14	34-70-95
	2	grishashvi	2/5/83	1000	mukhiani 2/5	63-84-96
	3	vepkhavadze	7/28/78	1000000	marjanishvili 32	94-24-67
	4	bolashvili	12/24/82	10	vaja pshavela18	39-55-12
	5	Samkharadz	8/18/83	999999999	chavchavadze 5	22-28-98
	6	beridze	6/15/84	15000	kazbegi 45	39-56-87
	(AutoNumber)			0		

სურ. 2. 2. ცხრილის ფურცლის დიალოგური ფანჯარა.

Access-ი ცხრილის შექმნისას იწვევს პირველადი გასაღების ძებნას. პირველადი გასაღები არის ჩანაწერის უნიკალური მონიშვნა, როგორც საიდენტიფიკაციო ნომერი. თუ Access-ი ვერ მონახავს პირველადი გასაღების ველს, ეკრანზე გამოაქვს შეტყობინება „გინდათ თუ არა, რომ შექმნათ პირველადი გასაღების ველი?“ თუ უპასუხებთ „დიახ“-ს, მაშინ Access-ი ცხრილის დასაწყისში დაუმატებს ველს და იქნება ველის ტიპი Autonumber (ავტომატური დანომვრა). იგი შეიცავს ყველა ველისათვის უნიკალურ საიდენტიფიკაციო ნომერს, რომელსაც Access-ი უწოდებს პირველად გასაღებს.

ცხრილის შექმნისა და შენახვის შემდეგ, თუ საჭიროა, შეგიძლიათ შეცვალოთ ცხრილის სტრუქტურა.

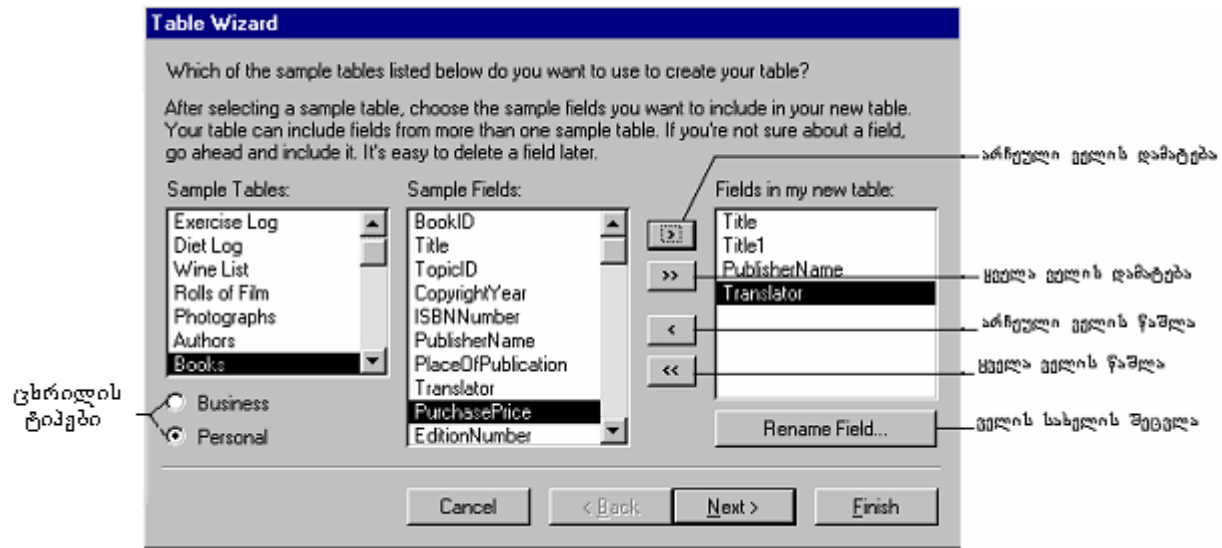
ცხრილის რეჟიმში ცხრილის შესაქმნელად და მონაცემების შესატანად შეასრულეთ შემდეგი სელები:

1. გახსენით მონაცემთა ბაზის ფანჯარა (დაარტყით კლავიშს F11), აირჩიეთ დიალოგური ფანჯრიდან პირველი რეჟიმი Datasheet View (ცხრილის ფურცლის ნახვა);
2. გახსენით Table (ცხრილი), რისთვისაც ერთჯერ დააწკაპუნეთ ღილაკზე New (ახალი ცხრილი), გამოჩნდება ცხრილის ფურცელი;
3. შეავსეთ ერთი სტრიქონი, ანუ გააკეთეთ ერთი ჩანაწერი, რისთვისაც შეიტანეთ მონაცემები თითოეულ ველში;
4. შეავსეთ კიდევ რამდენიმე სტრიქონი;
5. შეინახეთ ცხრილი;
6. როდესაც გამოვა შეკითხვა პირველადი გასაღების ველის შექმნის შესახებ, მიეცით ბრძანება „დიახ“;
7. დაათვალიერეთ მონაცემები და დარწმუნდით, რომ ველის ტიპები, რომლებიც აირჩია Access-მა შეესაბამება თქვენთვის სასურველს;
8. თუ საჭიროა, ჩაიხედეთ ცხრილის სტრუქტურის შექმნის რეჟიმში და შეცვალოთ ველის ტიპები ან საერთოდ შეცვალოთ ცხრილის სტრუქტურა;
9. ხელახლა შეინახეთ ცხრილი;
10. დაბრუნდით ცხრილის რეჟიმში, რომ გააგრძელოთ მონაცემების შეტანა.

❌ ობიექტის სახელი შეიძლება შედგებოდეს ასოებისა და ციფრებისაგან - სულ 64 სიმბოლოსაგან, მაგრამ მოერიდეთ დიდი სახელის დარქმევას, სახელი არ უნდა შეიცავდეს წერტილს, დახილის ნიშანსა და კვადრატულ ფრჩხილებს. ეცადეთ, სახელი აზრობრივად შეესაბამებოდეს თემას.

2.3. ცხრილის შექმნა ცხრილის ოსტატის მეშვეობით

დიალოგურ ფანჯარაში New (ახალი ცხრილი), როდესაც ირჩევთ ცხრილის ოსტატს, გამოჩნდება ახალი დიალოგური ფანჯარა Table Wizard (ცხრილის ოსტატი).



სურ. 2. 3. ცხრილის ოსტატის დიალოგური ფანჯარა

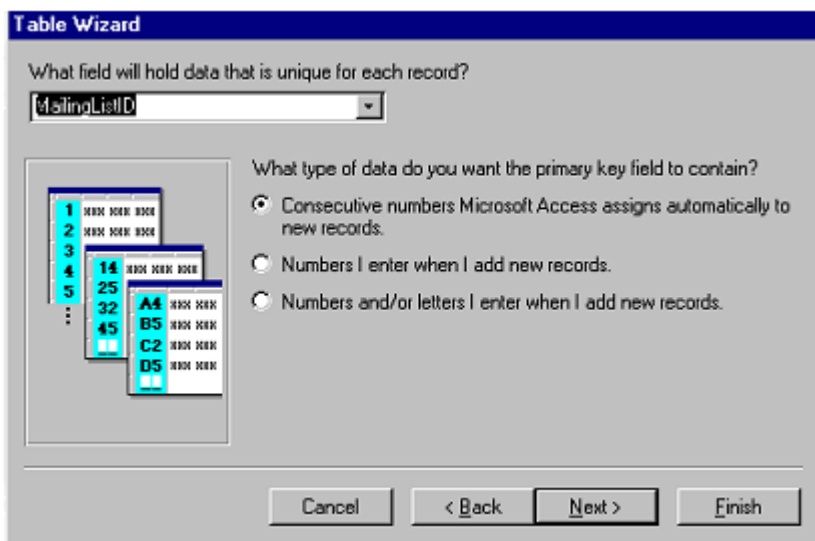
ცხრილის ნიმუშების ჩამონათვალი ისეთივე იქნება რა ტიპის ცხრილსაც აირჩევთ: კერძო, თუ საქმიანი მოხმარებისათვის. ცხრილის ნიმუშის არჩევის შემდეგ გვერდით ველში Sample Field (ველების ნიმუშები), გამოჩნდება ამ ცხრილის ველების ჩამონათვალი. აირჩიეთ თქვენთვის საჭირო ველები, გამოიყენეთ ველის დამატებისა და წაშლის ღილაკები, გადმოიტანეთ არჩეული ველები ახალ ცხრილში და დააწკაპუნეთ ღილაკზე Next (შემდეგ),



სურ. 2. 4. ცხრილის ოსტატის დიალოგური ფანჯარა ცხრილის სახელის დასარქმევად

ცხრილის შესაქმნელად გაჩნდება დიალოგური ფანჯარა Table Wizard-ი, ტექსტურ ველში ჩაწერეთ ცხრილის სახელი. ეს დიალოგური ფანჯარა გახსენებთ, რომ ცხრილში თითოეულ ჩანაწერს უნდა ჰქონდეს უნიკალური იდენტიფიკატორი ე. წ. პირველადი გასაღები, რომლის საშუალებითაც მოიძებნება ჩანაწერი და დაუკავშირდება ცხრილს, იხ. სურ. 2. 4.

ცხრილის ოსტატი გთავაზობთ თვითონ აირჩიოთ და მიუთითოთ ველი - პირველადი გასაღების ან ნება დართოთ Access-ს, რომ ეს თვითონ გააკეთოს. თუ არა ხართ დარწმუნებული რომელი ველი გამოგადგებათ პირველად გასაღებად, მაშინ დაეთანხმეთ ოსტატს, თავად მონახოს იგი. თუ მოგვიანებით არ დაგაკმაყოფილებთ შემოთავაზებული ვარიანტი, შეგიძლიათ მისი შეცვლა.



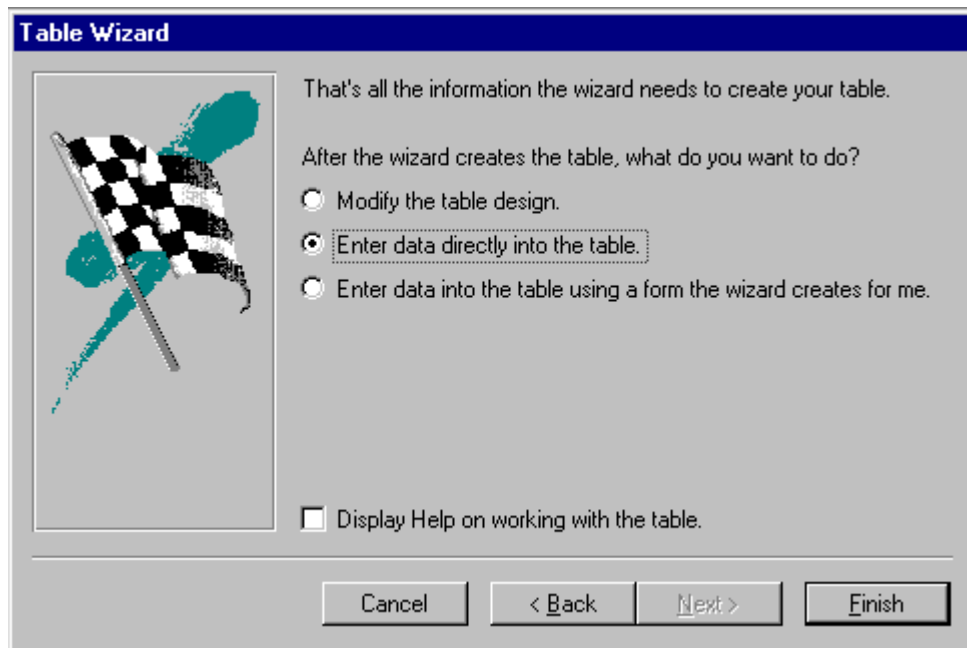
სურ. 2. 5. ცხრილის ოსტატის დიალოგური ფანჯარა გასაღების ველის დასაყენებლად

თუ გადაწყვეტთ, პირველადი გასაღების ველის დამოუკიდებლად შექმნას, გამოჩნდება ცხრილის ოსტატის დიალოგური ფანჯარა იხილეთ სურ. 2. 5.

გახსენით ტექსტური ველი „აირჩიეთ თითოეული ჩანაწერისათვის უნიკალური ველი“, გამოჩნდება სია, ამ სიიდან უნდა აირჩიოთ საჭირო ველი და დააწკაპუნოთ ღილაკზე Next (შემდეგ), ეკრანზე გამოვა კითხვა - დაკავშირებულია თუ არა ცხრილი სხვა ცხრილებთან.

ერთმანეთთან დაკავშირებულ ცხრილში არის ისეთი ჩანაწერები, რომლებიც ერთ-ერთი მათგანის პირველადი გასაღებით შეესაბამება რომელიმე სხვა ველს Foreign key (მეორადი გასაღები). მაგალითად, ცხრილში კლიენტები, კლიენტის ნომერი არის პირველადი გასაღები. ამ ცხრილის კავშირის დასამყარებლად ცხრილთან ანგარიშები, მყიდველის ნომერი ამ ცხრილისათვის იქნება მეორადი გასაღები. მეორე ველი - ანგარიშის ნომერი, ცხრილისათვის ანგარიშები იქნება პირველადი გასაღები. Access-ს გამოაქვს ახალ ცხრილსა და უკვე არსებულ ცხრილებს შორის კავშირების სია. იმისათვის, რომ განსაზღვროთ ცხრილებს შორის კავშირები დაარტყით ღილაკს Relationships (კავშირი), შემდეგ დააწკაპუნეთ ღილაკზე Next (მომდევნო), გამოჩნდება ცხრილის ოსტატის საბოლოო დიალოგური ფანჯარა, აირჩიეთ სათანადო პარამეტრი და დააწკაპუნეთ ღილაკზე Finish (მზადაა). ცხრილი შეიქმნება და გამოჩნდება ეკრანზე, იხ. სურ. 2. 6.

ცხრილი შეიძლება დაათვალიეროთ როგორც ცხრილის რეჟიმში, ისე ცხრილის ოსტატის რეჟიმში. ცხრილის რეჟიმში მონაცემები ჩანს სვეტებიან ცხრილში, რომელიც მზადაა მონაცემების შეტანისა და რედაქტირებისათვის. ცხრილის ოსტატის რეჟიმში კი შეგიძლიათ შეცვალოთ ცხრილის სტრუქტურა და სახე. ამ რეჟიმში მონაცემების შეტანა შეუძლებელია.



სურ. 2. 6. ცხრილის ოსტატის საბოლოო დიალოგური ფანჯარა

2.4. ცხრილის შექმნა კავშირის დაყენების საშუალებით

ხშირად მონაცემები გვაქვს სხვა ფორმატით და საჭიროა მათი ნახვა და რედაქტირება Access-ის საშუალებით, მიუხედავად იმისა რომ ინფორმაცია ინახება საწყის პროექტში, მაინც შეიძლება გამოვიყენოთ ოპცია “ცხრილების კავშირი“. ცხრილებს შორის კავშირის დასამყარებლად Access-ი შეიცავს შემდეგ ფორმატებს:

- ☐ Access (MDB)
- ☐ ტექსტი (TXT, CSV, TAB, ASC)
- ☐ Excel (XLS)
- ☐ HTML (ფაილები Access 97); ეს ფაილები მოთავსებული უნდა იყოს Web-სერვერში
- ☐ Paradox (DB)
- ☐ DBASE III, IV, V (DBF)
- ☐ FoxPro (DBF)
- ☐ მონაცემთა ბაზის ODBC

 შეუძლებელია FoxPro 3.0 (DBC) და Lotus 123 (WK*) ფორმატის ფაილების დაკავშირება Access-ის ცხრილებთან.

დაკავშირების საშუალებით ახალი ცხრილის შესაქმნელად შეასრულეთ შემდეგი:

1. გახსენით მონაცემთა ბაზა, აირჩიეთ Table (ცხრილი) და დააწკაპუნეთ ღილაკზე Create (შექმნა);
2. აირჩიეთ პუნქტი “ცხრილების კავშირი“ და მიეცით დასტური. გამოვა დიალოგური ფანჯარა “კავშირი“;
3. მიუთითეთ იმ ფაილის ფორმატი , რომლის დაკავშირებაც გინდათ მოცემულ ცხრილთან;
4. მიუთითეთ ფაილის განლაგების ადგილი;
5. მიუთითეთ ფაილის სახელი;
6. დააწკაპუნეთ ღილაკზე “კავშირი“.

თუ დაკავშირების პროცესი წარმატებით მიმდინარეობს, მაშინ ახალი ცხრილი გამოჩნდება მონაცემთა ბაზის ფანჯარაში. თუ რაიმე პრობლემა გაჩნდება გამოვა დიალოგური ფანჯარა, რომელიც გვატყობინებს ამის შესახებ და გვიხსნის მის მიზეზს.

2.5. ცხრილის შექმნა კონსტრუქტორის რეჟიმში

ცარიელი ცხრილის შესაქმნელად, ეკრანზე გამოიტანეთ მონაცემთა ბაზის ფანჯარა, გახსენით ველი Table - ცხრილი და დააწკაპუნეთ ღილაკზე New - ახალი. აირჩიეთ პუნქტი Design View (ცხრილის შექმნა კონსტრუქტორის რეჟიმით). Access-ი შექმნის ცარიელ ცხრილს და გამოიტანს მას კონსტრუქტორის რეჟიმში. იხ. სურ. 2.7. ზედა ფანჯარაში ჩაწერეთ ველის სახელი, მის გვერდით - ველის ტიპი, შემდეგ კი - მასი აღწერა. როგორც კი მაჩვენებელი ჩადგება ველის ტიპის შესაბამის სვეტში, ავტომატურად აქტიურდება ქვედა ფანჯარა, რომელიც გეხმარებათ დააყენოთ ან შეცვალოთ ველის შემდეგი პარამეტრები: ფორმატი, ველის სიგანე, მნიშვნელობა გაჩუმებით და შეტანის დროს შემოწმების წესი.

Field Name	Data Type	Description
id	AutoNumber	
dasaxebeba	Text	
html	Hyperlink	
done	Text	
sheqmni tarigi	Date/Time	

Field Properties	
General	Lookup
Field Size	50
Format	
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	No
Indexed	No

A field name can be up to 64 characters long, including spaces. Press F1 for help on field names.

სურ. 2. 7. ცხრილის სტრუქტურის მოდიფიცირების დიალოგური ფანჯარა.

ცხრილის სტრუქტურის მოდიფიცირება

უკვე შექმნილი ცხრილის მოდიფიცირებისათვის უნდა ჩაიხედოთ ცხრილის სტრუქტურის შექმნის ანუ კონსტრუქტორის რეჟიმში, რისთვისაც საჭიროა:

1. გახსნათ მონაცემთა ბაზა;
2. მონაცემთა ბაზის ფანჯარაში მონიშნოთ თქვენთვის საჭირო ველი;
3. დააწკაპუნოთ ღილაკზე Design - კონსტრუქტორი.

თუ ათვალიერებთ ცხრილს, ცხრილის რეჟიმში, მენიუდან ბრძანება აირჩიეთ View→Design View – ნახვა→სტრუქტურის ნახვა (კონსტრუქტორი).

სტრუქტურის ნახვის რეჟიმში მისაწვდომია შემდეგი კომპონენტები:

- ❑ „კონსტრუქტორის იარაღების სტრიქონი“, რომელიც შეიცავს სხვადასხვა იარაღის საშუალებას ცხრილის სტრუქტურის შესაქმნელად და მასთან სამუშაოდ;
- ❑ „ველების ცხრილი“, რომელიც შეიცავს სვეტებს, სადაც შეგიძლიათ განსაზღვროთ ველების სახელები, მათში მოთავსებულ მონაცემთა ტიპები და მათი აღწერა;
- ❑ „ველების თვისებების ფანჯარა“, რომელიც საშუალებას იძლევა ყოველ კონკრეტულ ველს შეცვალოთ თვისებები.

პრობლემები და მათი გადაწყვეტა

თქვენი მიზანია შექმნათ მონაცემთა ბაზა გაყიდვის კონტრაქტების რეგისტრირებისათვის.

მონაცემთა ბაზის საფუძვლად გამოიყენეთ ცხრილის შექმნის ოსტატის შაბლონი - კონტრაქტები. იგი ისე გარდაქმენით, რომ აკმაყოფილებდეს თქვენს მოთხოვნებს.

თუ გსურთ ცხრილის ოსტატის მიერ შექმნილი ველების სახელების შეცვლა, ჯერ უნდა გახსნათ შექმნილი მონაცემთა ბაზა, შემდეგ – ცხრილები, მონიშნოთ იმ ცხრილის სახელი, რომელშიც გსურთ ველების სახელების შეცვლა და დააწკაპუნოთ ღილაკზე Design (ცხრილის სტრუქტურა), გამოჩნდება ფანჯარა, რომელიც შეიცავს ცხრილის სტრუქტურას. თქვენ შეგიძლიათ შეცვალოთ ველის სახელი, წაშალოთ რომელიმე ველი ან დაამატოთ ახალი, სათანადო პარამეტრებიანი.

კონსტრუქტორის იარაღების სტრიქონთან მუშაობა

კონსტრუქტორის რეჟიმის იარაღების ჯგუფის ღილაკების აღწერა მოცემულია 2. 1 ცხრილში.

კონსტრუქტორის იარაღების ჯგუფის ღილაკები

ცხრილი 2. 1

ღილაკები	ღილაკების დასახელება	აღწერა
1	2	3
	ნახვა	ცხრილი გამოდის ცხრილის რეჟიმში
	შენახვა	შენახავს ცხრილს ან სტრუქტურას
	ამოჭრა	კონსტრუქტორის ფანჯარიდან ამოჭრის ტექსტის მონიშნულ ფრაგმენტს ან ობიექტს და მოათავსებს ბუფერზე
	ასლი	გადიდებს კონსტრუქტორის ფანჯარაში მონიშნული ტექსტის ან ობიექტის ასლს და მოათავსებს ბუფერზე
	ჩასმა (ჩადგმა)	ბუფერზე არსებულ ასლს მოათავსებს კურსორით მითითებულ ადგილას

1	2	3
	პირველადი გასაღები (გასაღების ველი)	მომხმარებელს აძლევს საშუალებას გამოეცნოს ერთი ან რამდენიმე ველი პირველად გასაღებად. ჩართავს/გამორთავს პირველად გასაღებს.
	ინდექსები	გამოიძახებს ინდექსების ფანჯარას მონიშნული ობიექტისათვის.
	სტრიქონის ჩამატება	მონიშნული სტრიქონის წინ ჩამატებს ცარიელ სტრიქონს
	სტრიქონების წაშლა	წაშლის მონიშნულ სტრიქონებს
	თვისებები	გახსნის ან დახურავს მონიშნული ობიექტის თვისებების ფანჯარას
	აგება	საშუალებას იძლევა შექმნათ ისეთი ელემენტი როგორცაა ველი ან შეტანის შაბლონი
	მონაცემთა ბაზის ფანჯარა	გამოიძახებს მონაცემთა ბაზის ფანჯარას
	ახალი ობიექტი	გამოიძახებს იმ ახალი ობიექტების სია, რომელთა შექმნაც შესაძლებელია, ესაა: ცხრილი, ფორმა, უწყისი, მოთხოვნა, მაკრობრძანება და მოდული
	დამხმარე	დამხმარე, ოფისის საცნობარო სისტემაში ძებნისათვის ან რჩევისათვის.

2.6. ცხრილის ველებთან მუშაობა

კონსტრუქტორის რეჟიმში ცხრილის შექმნის ფანჯარა გამოდის ცხრილის სახით და სვეტების სახელებია: Field Name (ველის სახელი), Data Type (მონაცემთა ტიპი) და Description (აღწერა).

„ველის სახელი“, ისე როგორც Access-ის სხვა ობიექტები შედგება ციფრებისა და სიმბოლოებისაგან, ოღონდ ნიშანთა რაოდენობა უნდა იყოს არა უმეტეს 64-ისა.

„მონაცემთა ტიპი“ განსაზღვრავს, თუ რა ინფორმაცია შეიძლება ინახებოდეს მოცემულ ველში. მაგალითად, თუ მონაცემთა ველის ტიპი განისაზღვრება, როგორც Data (თარიღი), მაშინ Access-ი არ მოგცემთ იმის საშუალებას, რომ მასში შეიტანოთ ტექსტი. მონაცემს მიენიჭება ტიპი. მონაცემთა ტიპის საშუალებით განისაზღვრება გამოთვლებისა და სხვა ოპერაციების ტიპები, რომლებიც შეიძლება შესრულდეს მოცემული ველის მონაცემებზე. იხილეთ სურ. 2. 7.

Access-ი გთავაზობთ მონაცემთა შემდეგ ტიპებს:

- ❑ Text (ტექსტური) - ციფრები და ასოები 255 ბაიტამდე (სიმბოლოების ერთი ბაიტი);
- ❑ Memo (მოზრდილი ტექსტისთვის) - ციფრები და ასოები 65535 სიმბოლომდე;
- ❑ Numb (რიცხვითი) – ნებისმიერი რიცხვი დასაშვები ზომითა და საჭირო ადგილით;
- ❑ Data/Time (თარიღი) - მრგვალი რიცხვები, რომლებიც უზრუნველყოფენ ზუსტ შედეგს ათობითი მძიმედან მარცხნივ 15 თანრიგამდე და მარჯვნივ 4 თანრიგამდე;
- ❑ AutoNumber (მთვლელი) – უნიკალური ნომერი, რომელსაც Access-ი ანიჭებს მიმდევრობით (ერთის მიმატებით) ან შემთხვევით ყოველ შეტანილ ჩანაწერს;

- ❑ Yes/No(ლოგიკური) – ლოგიკური მონაცემები (დიახ/არა, ჯეშმარიტი/მცდარი, ჩართვა/გამორთვა);
- ❑ OLE Object (OLE ობიექტი) – გრაფიკული ან სხვა რომელიმე ბინარული ცვლადები;
- ❑ Hyperlink (გზავნილები) – ტექსტი და ციფრები, რომლებიც განსაზღვრავენ საბუთისაკენ გზას;

ცხრილი 2. 2 რიცხვითი ტიპის ველისათვის დასაშვები რიცხვითი მნიშვნელობები.

ტიპი	ზომა	დიაპაზონი
ბაიტი	1 ბაიტი	0-დან 255-მდე, არ მონაწილეობს წილადი ნაწილი
მთელი	2 ბაიტი	-32 768-დან 32 767-მდე, არ მონაწილეობს წილადი ნაწილი
გრძივი მთელი	4 ბაიტი	-2147483648-დან 2147483648-მდე, არ მონაწილეობს წილადი ნაწილი
მცოცავმძიმისანი ორდინალური სიზუსტე	4 ბაიტი	დიაპაზონში -3.402823E38-დან 3.402823E38-მდე შეიძლება რიცხვი.
მცოცავმძიმისანი ორმაგი სიზუსტე	8 ბაიტი	დიაპაზონში -1.7976933486231E308-დან 1.7976933486231 E308-მდე 15-ნიშნისანი რიცხვი.
რეპლიკაციის კოდი	16 ბაიტი	გლობალური უნიკალური იდენტიფიკატორი, რომელიც გამოიყენება მონაცემთა ბაზის რეპლიკაციისათვის.

ყოველ ახალ ველს Access-ი თვითონ ანიჭებს ნაგულისხმევ ტექსტურ ტიპს. იმისათვის, რომ მიანიჭოთ სხვა თქვენთვის სასურველი ტიპი, დააწკაპუნეთ პატარა შავ სამკუთხედზე და სიიდან აირჩიეთ საჭირო ტიპი.

 როდესაც რიცხვი შეიცავს დევისს, მაგალითად, სოციალური დაზღვევის ნომერი ან ტელეფონის ნომერი, უნდა ისარგებლოთ ტექსტური ტიპით - შეცვალთ ფორმატი ან შეტანის შაბლონი, ვინაიდან რიცხვით ტიპში ეს სიმბოლო არ არის.

 ველის ტიპის შერჩევისას მონაცემები, რომლებსაც აქვთ ფუალადი ტიპი დამრგვალდება, ხოლო რიცხვითი ველი იყენებს მცოცავ მძიმეს (ათწილადში), რომელიც საჭიროების მიხედვით გადაადგილდება. ფუალადი ტიპი იყენებს გამოთვლების უფრო სწრაფ ხერხს. ამ შემთხვევაში მძიმე ფიქსირებულია და თავიდანვე განსაზღვრული მძიმის შემდეგ ათწილადი ნიშნების რაოდენობა, რაც ამრგვალებისას გვაძლევს ცდომილებას.

ველის აღწერა.

ეს სვეტი განკუთვნილია ველის შესახებ დამატებითი ინფორმაციისათვის. მაგრამ იგი აუცილებელი არ არის რადგან ველთან დაკავშირებული დამატებითი ინფორმაცია ყოველთვის ჩნდება მდგომარეობის სტრიქონში, როდესაც კურსორი არის მოცემულ ველში.

პირველადი გასაღების დაყენება

Access-ში ეფექტურად მუშაობისათვის ყოველ ცხრილს უნდა ჰქონდეს გასაღების ველი. პირველადი გასაღები ჩანაწერს ხდის უნიკალურს. მაგალითად, მონაცემთა ბაზაში თითოეულ თანამშრომელს აქვს სოციალური დაზღვევის უნიკალური ნომერი. ამიტომ სოციალური დაზღვევის ნომრის ველი გამოიყენება როგორც პირველადი გასაღები.

პირველადი გასაღების შექმნის უპირატესობები:

- ❑ *სისწრაფე.* Access-ი გასაღების ველის საფუძველზე ქმნის ინდექსს, რომლის საშუალებითაც სწრაფად ხდება მოთხოვნების დამუშავება და სხვა ფუნქციების შესრულება;

□ *მიმდევრობა*. Access პირველადი გასაღების მქონე მონაცემთა ბაზას ავტომატურად ახარისხებს;

□ *ჩანაწერი არ მეორდება*. Access მომხმარებელს უფლებას არ აძლევს შეიტანოს მონაცემები პირველადი გასაღების იმავე მნიშვნელობით, როგორიც აქვს არსებულ ჩანაწერს.

□ *კავშირები*. ერთ ცხრილში პირველადი გასაღებისა და მეორე ცხრილის შესაბამისი მეორადი გასაღების ურთიერთდამოკიდებულების საფუძველზე, Access ინარჩუნებს ერთმანეთთან დაკავშირებული ცხრილების კავშირებს.

პირველადი გასაღების შესაქმნელად უნდა აირჩიოთ კონსტრუქტორის რეჟიმი, გახსნათ ცხრილი, რისთვისაც იქცევით შემდეგნაირად:

1. მონაცემთა ბაზის ფანჯრიდან აირჩიეთ თქვენი მონაცემთა ბაზა;
2. გახსენით Table (ცხრილები), მონიშნეთ თქვენთვის საჭირო ცხრილი და დააწკაპუნეთ ღილაკზე Design (კონსტრუქტორი);
3. ველებიდან აირჩიეთ ის, რომლისთვისაც გსურთ პირველადი გასაღების შექმნა და მასზე დააწკაპუნეთ ერთჯერ;
4. აირჩიეთ მენიუდან ბრძანება Edit→Primary Key (რედაქტირება→პირველადი გასაღები) და დააწკაპუნეთ ერთჯერ. ველის მარცხენა ნაპირას გაჩნდება გასაღები, რაც იმას ნიშნავს, რომ ეს ველი დანიშნულია გასაღების ველად.

იგივე შეგიძლიათ შეასრულოთ იარაღების სტრიქონიდან Primary Key ღილაკის მეშვეობით.

ველისათვის ზოგიერთი თვისებების მიკუთვნება

ველისათვის თვისებების მიკუთვნება ხდება იმის მიხედვით, თუ რა სახით გსურთ მონაცემების შენახვა ან ეკრანზე გამოტანა. ველის თვისებების მიკუთვნებით შეგიძლიათ განსაზღვროთ:

- სათაური;
- სისტემის მიერ ნაგულისხმევი მნიშვნელობა;
- ფორმატი (სტრუქტურა) მონაცემებისათვის;
- შემოწმება მონაცემების შეტანისას;
- ინდექსი (ინდექსირებული ველებისათვის);
- ველების გამოსატანი პარამეტრები - ფორმატი და ზომა.

ცხრილების ველების თვისებები ავტომატურად გამოიყენება მონაცემთა ბაზის იმ ობიექტებისათვის (ფორმები, მოთხოვნები და უწყისები), რომლებიც იყენებენ ამ ცხრილებს. ველების თვისებები წარმოდგენილია ფურცლის ორ ჩანაცვლებაში General (საერთო) და Lookup (ჩასმა).

ჩანაცვლებაში General (საერთო) მოცემულია შემდეგი თვისებები:

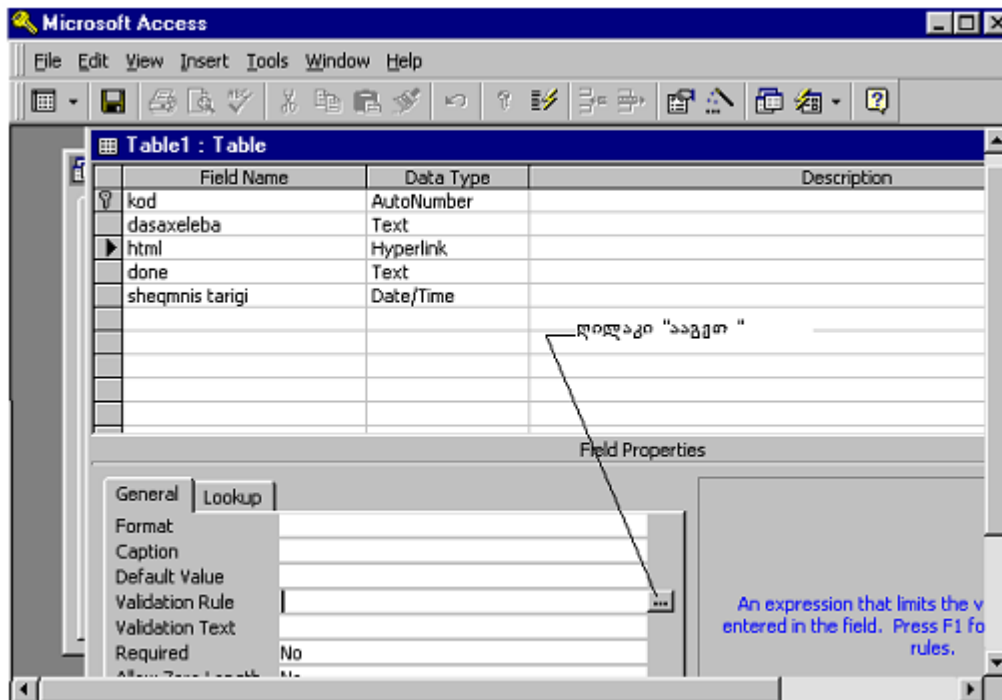
- Field Size (ველის ზომა) განსაზღვრავს ტექსტურ ველში სიმბოლოების მაქსიმალურ რაოდენობას და რიცხვითი ველის ტიპს;
- Format (ფორმატი), მიუთითებს ფორმატს თარიღისა და რიცხვებისათვის;
- Decimal Places (ათწილადი რიცხვისათვის), განსაზღვრავს ათწილადი ნიშნების რაოდენობას ვალუტისა და რიცხვითი ველებისათვის;
- Input Mask (შაბლონის შეტანა), (მხოლოდ ტექსტური და თარიღის შემცველი ველებისათვის) ისეთი სიმბოლოების ფორმატის სპეციფიკაციისათვის, როგორიცაა ტირე - ტელეფონის ნომრის ველში, რომელიც მონაცემების შეტანისას ავტომატურად შეივსება;
- Caption (სათაური/წარწერა), არის ტექსტი, რომელიც ფორმებსა და უწყისებში, ან ცხრილების დათვალიერების რეჟიმში გამოიყენება სვეტის სათაურად, ველის სახელის ნაცვლად. მაგალითად, ვიდეოფირის ნომერი - videoID-ის მაგივრად.

ცხრილის სტრუქტურის რეჟიმში ველების თვისებების დაყენება:

1. მონიშნეთ ის ველი, რომლის თვისებების შეცვლაც გსურთ. ფანჯრის ქვედა ნაწილში გამოჩნდება ამ ველის ზოგადი თვისებები;

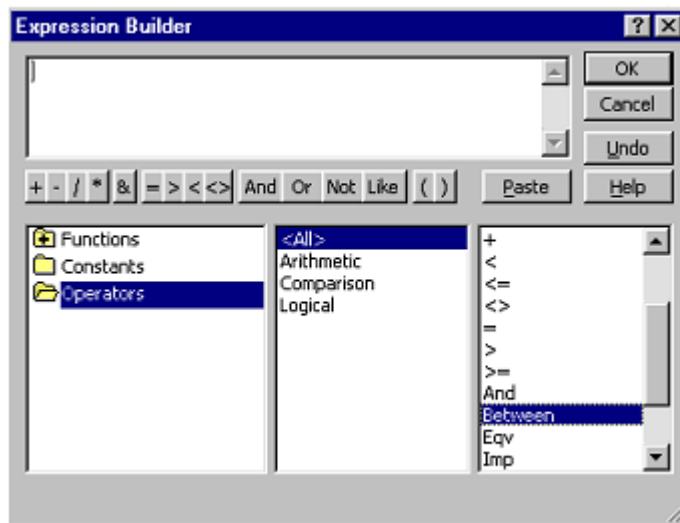
2. General-დან (ზოგადი) აირჩიეთ თვისება, რომლის შეცვლაც გსურთ ან დააჭირეთ ხელი კლავიშს <F6> ფანჯრის ქვედა ნაწილში გადასასვლელად, და გამოიყენეთ კლავიში <Tab> ველის თვისებებს შორის გადასადგილებლად;
3. შეიტანეთ მოცემული თვისების მნიშვნელობა, ან აირჩიეთ გასხნილი სიიდან (თუ ეს სია აქვს);
4. გააგრძელეთ მოცემული ველისათვის თვისებების შეცვლა;
5. გახსენით Lookup და იქაც მიაკუთვნეთ თვისება;
6. თუ საჭიროა ანალოგიურად მიაკუთვნეთ თვისებები სხვა ველებსაც;
7. შეინახეთ ცხრილი. იხილეთ სურ. 2.8.

თუ კურსორი დგას ველების თვისებებში, მარჯვენა კლავიშზე ერთჯერ დაწკაპუნებით გამოვა მენიუ, რომელიც შეიცავს ბრძანებებს: Build (ამგები), Zoom (მასშტაბი), Cut (ამოჭრა), Paste (ჩასმა) და ცხრილის თვისებების დასაყენებელ ბრძანებებს. (ამ ბრძანებების აქტიურობა დამოკიდებულია იმაზე, თუ რომელი თვისება იცვლება).



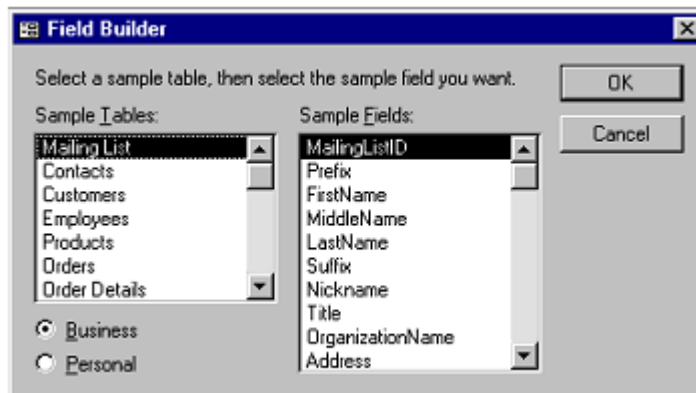
სურ. 2. 8. ცხრილის სტრუქტურის რეჟიმში ველებზე თვისებების მიკუთვნების დიალოგური ფანჯარა

ლილაკი “ააგეთ” მდებარეობს კონსტრუქტორის იარაღების ჯგუფში, “ცხრილების თვისებების” მარჯვნივ და საშუალებას იძლევა დაინახოთ შედარებით ფუნქციები ან მაგალითები. სურათზე 2.9 ნაჩვენებია გამოსახულების ამგების დიალოგური ფანჯარა. ეს ფანჯარა შეგიძლიათ გამოიძახოთ ასევე ლილაკით “ააგეთ”. იმისათვის რომ ნახოთ ვარიანტების სია, საჭადალდეზე დააწკაპუნეთ ორჯერ. შუა სვეტიდან შეგიძლიათ აირჩიოთ ოპერატორის კატეგორია, ხოლო მის გვერდით სვეტში, ოპერატორის მნიშვნელობაზე დააწკაპუნეთ ორჯერ. ახლა უკვე შეგიძლიათ მოახდინოთ გამოსახულების რედაქტირება იხ. სურ. 2. 9.



სურ. 2. 9. გამოსახულების ამგების დიალოგური ფანჯარა

თუ მუშაობა გიწევთ ველის სათაურთან და დააწკაპუნებთ ღილაკზე “აგეთ”, გაიხსნება ველის ამგების დიალოგური ფანჯარა და სხვადასხვა ნიმუშიდან შეგიძლიათ აირჩიოთ ფაილის სახელი. იხ. სურ. 2. 10.



სურ. 2. 10. ველის ამგების დიალოგური ფანჯარა

2.7. ინდექსის თვისებების დაყენება

ინდექსი ეხმარება Access-ს მონაცემის მნიშვნელობის მოძებნაში. პირველადი ველისათვის Access ავტომატურად ქმნის ინდექსს.

თუ რომელიმე ველით ხშირად სრულდება ძებნა ან დახარისხება, მონაცემთა დამუშავების სისწრაფის გაზრდის მიზნით, შეიძლება ამ ველისათვის ინდექსის შექმნა. შეიძლება ნებისმიერი ველის ინდექსაცია, გარდა ველებისა OLE , Memo და გზავნილებისა.

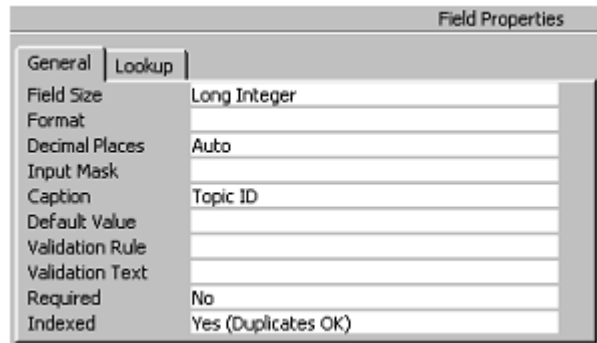
ინდექსებისათვის შეიძლება შემდეგი თვისებების დაყენება:

- ❑ დიახ (დამთხვევა დასაშვებია). იქმნება ინდექსი, რომელიც შეიცავს ველების ერთნაირ მნიშვნელობებს (მნიშვნელობების გამეორება დასაშვებია);
- ❑ კი (დამთხვევა დაუშვებელია). იქმნება ინდექსი, რომელიც ემყარება ველის უნიკალურ მნიშვნელობას;
- ❑ არა. ინდექსი არ იქმნება.

ინდექსების თვისებების დასაყენებლად:

1. ცხრილში აირჩიეთ ის ველი, რომლის ინდექსაციაც გსურთ;
2. ველის თვისებების ფანჯარაში მონიშნეთ თვისება – ინდექსირებული ველი;
3. სიიდან აირჩიეთ ინდექსის ტიპი.

☒ შექმენით ინდექსები მხოლოდ იმ ველებისათვის, რომლის მნიშვნელობებიც ერთმანეთისაგან განსხვავდება. იმ ველების ინდექსირება, რომლის მნიშვნელობებიც მეორდება, ძეგნისა და დახარისხების სისწრაფეს არ ზრდის.



Field Properties	
General	Lookup
Field Size	Long Integer
Format	
Decimal Places	Auto
Input Mask	
Caption	Topic ID
Default Value	
Validation Rule	
Validation Text	
Required	No
Indexed	Yes (Duplicates OK)

სურ. 2. 11. ველის თვისებების ფანჯარა

გ ა ს ა ლ ე ბ ი დ ა ი ნ დ ე ქ ს ი

რელაციურ მონაცემთა ბაზის საშუალებით შეგიძლიათ სწრაფად მონახოთ და მოთხოვნების, ფორმებისა და უწყისების საშუალებით დააკავშიროთ სხვადასხვა ცხრილის მონაცემები. ამისათვის ყოველი ცხრილი უნდა შეიცავდეს ერთ ან რამდენიმე ველს, რომელიც ცალსახად განსაზღვრავს ცხრილში თითოეულ ჩანაწერს. ასეთ ველს ეწოდება ცხრილის გასაღები ველი. არსებობს სამგვარი გასაღები ველი: მთაველი, მარტივი გასაღები და შედგენილი გასაღები.

მთაველი შეიძლება ისეთი იყოს, რომ ცხრილში ყოველი ჩანაწერის დამატებისას ამ ველში რიგითი ნომერი შეიცვალოს. ასეთი ველის მითითება ველის გასაღებად უმარტივესი შემთხვევაა. თუ ცხრილის შენახვამდე არ იყო განსაზღვრული ველი გასაღებად, მაშინ შენახვისას გამოვა ასეთი ველის შექმნის შეტყობინება და დათანხმების შემთხვევაში, ავტომატურად შეიქმნება იგი.

მარტივი გასაღები. თუ ეს ველი შეიცავს უნიკალურ მნიშვნელობებს, ისეთებს, როგორიცაა კოდები ან ინვენტარული ნომრები, მაშინ იგი შეიძლება განისაზღვროს გასაღებ ველად. თუ არჩეული ველი შეიცავს განმეორებად ან ნულოვან ჩანაწერებს, მაშინ ეს ველი გასაღებ ველად ვერ ჩაითვლება. განმეორებითი ჩანაწერების განსაზღვრისათვის უნდა შეასრულოთ მოთხოვნა მათ მოძებნის შესახებ. თუ მათი შეცვლა არ ხერხდება, მაშინ ცხრილში უნდა დაამატოთ მთაველის ველი და ის დააყენოთ გასაღებ ველად, ან განსაზღვროთ შედგენილი გასაღები.

შედგენილი გასაღები. როდესაც არ ხერხდება, რომ ველის ჩანაწერი იყოს უნიკალური (ე. ი. ხდება ჩანაწერების გამეორება), უნდა შექმნათ გასაღები, რომელიც შედგება რამდენიმე ველისაგან. ხშირად ასეთი სიტუაცია ისეთი ცხრილებისათვის, რომლებიც გამოიყენებიან ორი ცხრილის ერთმანეთთან დასაკავშირებლად, რომელთა შორის თანაფარდობაა მრავალი – მრავალთან. მაგალითად, მონაცემთა ბაზის – კარტოთეკა – ცხრილი Books (წიგნები), რომელიც უკავშირდება ცხრილებს Authors ავტორები და გამომცემლები. ამ ცხრილში შეგიძლიათ გამოიყენოთ შედგენილი გასაღები, რომელიც ორი ველისაგან AuthorID და PublisherID-საგან შედგება. ცხრილში Books (წიგნები) შეიძლება შედიოდეს მრავალი ავტორი და მრავალი გამომცემელი, მაგრამ კარტოთეკაში ყოველი წიგნი გვხვდება მხოლოდ ერთჯერ, ამიტომ ამ ველების მნიშვნელობების კომბინაცია საკმარისია გასაღების შესაქმნელად.

ინდექსი. მონაცემთა ბაზის ობიექტია, რომელიც უზრუნველყოფს ცხრილის ცალკეულ სტრიქონზე სწრაფად გადასვლას. ინდექსი იქმნება ცხრილის მონაცემების დახარისხებისა და მოთხოვნათა ოპერაციების სწრაფმოქმედების ასამაღლებლად.

ინდექსი ინახავს ცხრილის ერთ ან რამდენიმე სვეტის მნიშვნელოვნად ოპტიმიზებულ ვერსიას. ინდექსის შექმნა ან წაშლა არ მოქმედებს ინდექსირებული ცხრილის მონაცემებსა და მათ გამოთვლაზე.

ველის გასაღების ინდექსაცია ავტომატურად ხდება. არ შეიძლება Memo-ს ან OLE-ს ტიპის ველის ინდექსაცია. ველებისათვის (ტექსტური, რიცხვითი, ფულადი ან თარიღის ტიპის), რომლებისთვისაც ხდება მნიშვნელობების დახარისხება ან ძებნა, ინდექსაცია გამოიყენება.

~~მა~~ მართალია ინდექსი ზრდის ძებნისა და დახარისხების სისწრაფეს, მაგრამ მონაცემების დამუშავება ნელდება, რადგან ყოველი ჩანაწერის დამატების, შეცვლისა და წაშლისას ხდება ყველა ინდექსის განახლება (მიერთება).

ცხრილის თვისებების დაყენება

ველების მსგავსად, ცხრილებსაც აქვთ თვისებები. ცხრილების თვისებების შეცვლა შეიძლება როგორც მთელ ცხრილში, ისე ცხრილის ყოველ ჩანაწერში. ცხრილისათვის შეიძლება შემდეგი თვისებების დაყენება:

1. *აღწერა.* შეიტანეთ ცხრილის აღწერა და მისი ამოცანა. მაგალითად, არის თუ არა ცხრილში, რომლის სახელია კინო გაყიდვული ფილმების ჩამონათვალი. (შეგიძლიათ გამოიყენოთ ფანჯარა Zoom (მასშტაბი), რომლის გამოძახება ხდება <Shift>+<F2> კომბინაციით). თუ მონაცემთა ბაზის ფანჯარაში, იარაღების სტრიქონზე დააწკაპუნებთ Design (კონსტრუქტორი) კლავიშზე გამოჩნდება ეს აღწერა და ბაზის შექმნის თარიღი.

2. *მნიშვნელობებზე პირობები.* ზღუდავს ყველა ველში მონაცემების მნიშვნელობების შეტანას, მაგალითად, მოითხოვს, რომ კატეგორია და გაყიდვის თარიღი შეივსოს ყოველ ახალ ჩანაწერში.

3. *შეცდომის შეტყობინება.* თუ დარღვეულია მნიშვნელობაზე პირობა გამოდის შეტყობინება მაგალითად, რატომღაც საჭირო მონაცემების შეტანა საქონლის კატეგორიაზე და გაყიდვის თარიღზე.

4. *ფილტრი.* მიუთითებს ფილტრს, რომელიც ჩატვირთული უნდა იყოს ცხრილთან ერთად. მაგალითად, ფილტრი, რომელსაც გამოაქვს საწყობში ამჟამად არსებული ფილმები.

5. *დახარისხების მიმდევრობა.* მიუთითებს ცხრილის ელემენტების დახარისხების მიმდევრობას. მაგალითად, ფილმები უნდა დახარისხდეს გაყიდვის თარიღის კლების მიხედვით.

2.8. ცხრილების მოდიფიკაცია

ცხრილის სტრუქტურის შეცვლის ოპერაციებს Access ამარტივებს. შეგიძლიათ ცხრილში ველების დამატება, მათი სახელის შეცვლა, წაშლა და გადაადგილება. სანამ დაიწყებდეთ ცხრილის სტრუქტურის შეცვლას, ჯერ უნდა შექმნათ ამ ცხრილის ასლი. ასევე უნდა იფიქროთ იმაზე, თუ რას გამოიწვევს ცხრილის სტრუქტურის შეცვლა, განსაკუთრებით მონაცემთა ბაზის ისეთი დამოუკიდებელი ობიექტებისათვის როგორიცაა ფორმები, მოთხოვნები და უწყისები.

□ ველების წაშლა. ცხრილიდან წაშლილი ველები ასევე უნდა წაიშალოს ფორმებიდან, მოთხოვნებიდან და უწყისებიდან.

□ ველის სახელის შეცვლა. ველს რომელსაც სახელი შეუცვალეთ ცხრილში, ასევე უნდა შეუცვალოთ სახელი ფორმებში, მოთხოვნებსა და უწყისებში. გარდა ამისა, ამ ველზე ყველა მითითება უნდა შეიცვალოს როგორც გამოთვლებში, ისე გამოსახულებებში, მაკრობრძანებებსა და მოდულებში.

□ მონაცემთა ტიპის შეცვლა. მონაცემთა ზოგიერთი ტიპის შეცვლა დაუშვებელია. მაგალითად, დაუშვებელია მონაცემთა მთელელში ნებისმიერი ტიპის გარდაქმნა. სხვა შემთხვევაში, თუ ხდება მონაცემთა უფრო დიდი ტიპის გარდაქმნა შედარებით მცირე ტიპად, მონაცემები შეიძლება დაიკარგოს. მაგალითად, რიცხვითი ფორმატის შეცვლით სავალუტო ფორმატით დაიკარგება დიაპაზონის გარეთ დარჩენილი მონაცემები.

□ ველის ზომის შეცვლამ შეიძლება გამოიწვიოს დამრგვალებული რიცხვების დაკარგვა. თუ ცვლილება ისეთია, რომ სიდიდე გაცილებით მეტია, ვიდრე მოცემული ველისთვისაა დასაშვები, მაშინ მონაცემები იკარგება (წინასწარ გამოვა შეტყობინება შეცდომის შესახებ).

ახალი ველის ჩასმატებლად:

1. მოათავსეთ კურსორი იმ სტრიქონში, რომლის წინაც ამატებთ ახალ ველს;
2. აირჩიეთ მენიუდან ბრძანება Insert→Rows (ჩასმა→სტრიქონი), ან დააწკაპუნეთ იმ ველის სელექტორზე, რომლის წინაც ამატებთ სტრიქონს, დაარტყით კლავიშს Insert (ჩასმა).

 სელექტორი ჰქვია მარცხნივ პირველ ნაცრისფერ სვეტს და იმ სტრიქონს, რომელშიც ველების სათაურები წერია.

ველის სახელის შესაცვლელად:

1. მონიშნეთ ის უჯრა, რომელიც შეიცავს ველის სახელს;
2. ჩაწერეთ ველის ახალი სახელი.

ველის წასაშლელად:

1. მონიშნეთ წასაშლელი ველი;
2. აირჩიეთ მენიუდან ბრძანება Edit→Delete Rows (რედაქტირება→სტრიქონის წაშლა), ან დააწკაპუნეთ იარაღების სტრიქონში მოთავსებულ კლავიშზე Delete Rows (სტრიქონის წაშლა), ან ველის სელექტორით მონიშნეთ სტრიქონი და დაარტყით კლავიშს Delete (წაშლა).

ველის გადასადგილებლად:

1. კურსორი გააჩერეთ გადასატანი ველის სათაურზე, მაჩვენებელი მიიღებს მარჯვნივ მიმართული შავი ისრის ფორმას, მარცხენა კლავიშით დააწკაპუნეთ სელექტორზე. მონიშნება გადასატანი ველი;
2. მაჩვენებელი დააყენეთ ველის სათაურზე, ამ დროს იგი თეთრი ფერის იქნება და გადაიტანეთ ველი ახალ ადგილას.

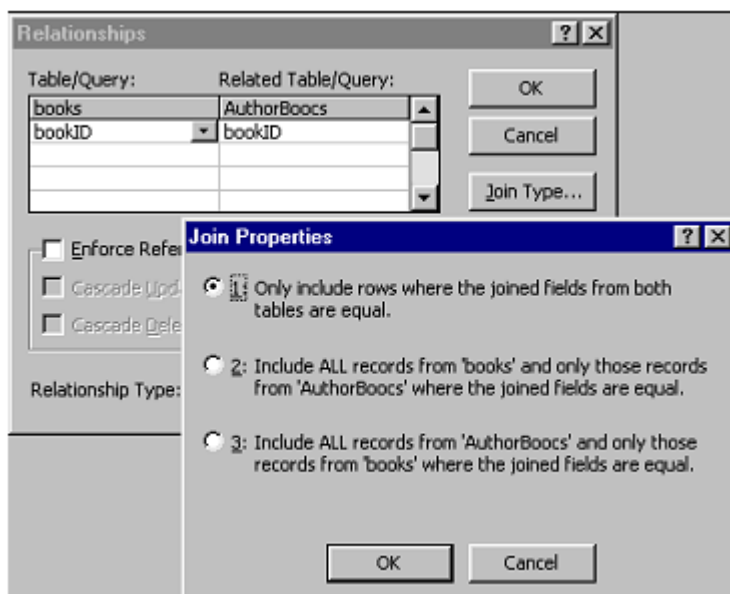
2.9. ცხრილებს შორის კავშირის დამყარება

ცხრილის ოსტატის მეშვეობით შექმნილი ცხრილის შემთხვევაში, Access-ი გეკითხებათ, თუ როგორ დაუკავშირდება იგი მონაცემთა ბაზის სხვა ცხრილებს. Access-ს თავადაც შეუძლია შექმნას ცხრილებს შორის კავშირები. თუ გსურთ ამ კავშირების დათვალიერება, რედაქტირება ან ახალი კავშირის შექმნა, მენიუდან აირჩიეთ ბრძანება Tools→Relationships (სერვისი→კავშირის სქემა), ან იარაღების სტრიქონიდან ისარგებლეთ კლავიშით Relationships. თუ რაიმე მიზეზით ვერ დამყარდა რომელიმე კავშირი, მაშინ გამოვა დიალოგური ფანჯარა Add Table (ცხრილის დამატება). მონაცემთა სქემის ფანჯარას დაამატეთ ნებისმიერი ცხრილი.

ხშირად საჭიროა კავშირის შექმნა ცხრილებს, ფორმებსა და ქვეფორმებს შორის ან საბოლოო და შუალედურ უწყისებს შორის.

კავშირების უმრავლესობა არის „ერთი მრავალთან“ ტიპის - ერთი ჩანაწერი ცხრილიდან “მონაცემები მყიდველის შესახებ” შეიძლება დაუკავშიროთ “ანგარიშების” ცხრილის რამდენიმე ჩანაწერს. პირველადი გასაღები უკავშირდება სხვა ცხრილის იმ ველს, რომელიც შეიცავს ზოგად ინფორმაციას. დაკავშირებული ცხრილის ამ ველს ეწოდება მეორადი გასაღები და შეიძლება თავისთავად იყოს ან არ იყოს პირველადი გასაღები ამ მონაცემთა ბაზისათვის. იხილეთ სურ. 2.12. როგორც ვიცით, ერთი მეცნიერი შეიძლება რამდენიმე წიგნის

ავტორი იყოს. ამიტომ, ცხრილში books bookID ველი არის პირველადი გასაღები და დაკავშირებულია bookID მეორად გასაღებთან ცხრილში AuthorBooks.



სურ. 2. 12. დიალოგური ფანჯარა Relationships (კავშირის სქემა), რომელსაც ახლავს ფანჯარა Join Properties (მიერთების პარამეტრები).

ცხრილებს შორის კავშირის შესაქმნელად:

1. აირჩიეთ მენიუდან Tools→Relationships (სერვისი→კავშირის სქემა) ბრძანება;
2. Show Table (ცხრილის ჩვენება) დიალოგური ფანჯრიდან დაამატეთ საჭირო ცხრილი, რისთვისაც მაჩვენებლის მარცხენა ღილაკით დააწკაპუნეთ Add Table (ცხრილის დამატება) კლავიშზე. გაჩნდება დამატებითი ცხრილი;
3. უკვე არსებული კავშირის სქემის ფანჯარაში დაკავშირებული ველების სახელების პირველადი გასაღების ველის სახელი გადაიტანეთ ველის ერთი სიიდან მეორე სიაში. ავტომატურად გაჩნდება კავშირის ფანჯარა სურ. 2. 12.

4. ამ ფანჯარაში დააყენეთ საჭირო პარამეტრები და ბოლოს დააწკაპუნეთ ღილაკზე Create (შექმენი). კავშირი საერთო ველების შემცველ ცხრილებს შორის ნაჩვენები იქნება ხაზებით.

დიალოგურ ფანჯარაში - Join Properties (მიერთების პარამეტრები) არის შემდეგი პარამეტრები:

❑ *მონაცემთა მთლიანობის უზრუნველყოფა.* თუ არჩეულია ეს პარამეტრი, მაშინ პირველადი გასაღების შეცვლის ან წაშლის შედეგად დაკავშირებულ ცხრილში ან მონაცემები შეიცვლება, ან მიიღებთ შეტყობინებას, რომ ეს მოქმედება არ შესრულდება.

❑ *დაკავშირებულ ველებთან ჩანჩქერული მიერთება.* თუ არჩეულია ეს პარამეტრი, მაშინ პირველადი გასაღების შეცვლის შედეგად ავტომატურად იცვლება მაკავშირებელი მეორადი გასაღები. თუ ეს პარამეტრი არ არის არჩეული, მაშინ პირველადი გასაღების შეცვლის მცდელობისას ეკრანზე გამოვა გაფრთხილება, რომ ცვლილების შეტანა არ შეიძლება.

❑ *დაკავშირებული ველების ჩანჩქერული მოშლა.* თუ არჩეულია ეს პარამეტრი, მაშინ ჩანაწერის წაშლისას, პირველადი გასაღებით დაკავშირებულ მონაცემთა ყველა ჩანაწერი წაიშლება. თუ ეს პარამეტრი არ არის არჩეული, მაშინ სისტემა გაფრთხილებთ, რომ მოცემულ ჩანაწერებთან დაკავშირებული ჩანაწერების წაშლა არ შეიძლება.

❑ *დაკავშირებული ტიპი.* მოთხოვნებისათვის ნაგულისხმევი სახე***783გვ, შეიცავს მითითებას რომ, ეკრანზე გამოიტანოს მხოლოდ ის ჩანაწერები, რომლებსაც ორივე ცხრილში საერთო გასაღები აქვთ, ან გამოიტანოს ერთი ცხრილი მთლიანად, მეორე ცხრილიდან კი მასთან დაკავშირებული ყველა ჩანაწერი.

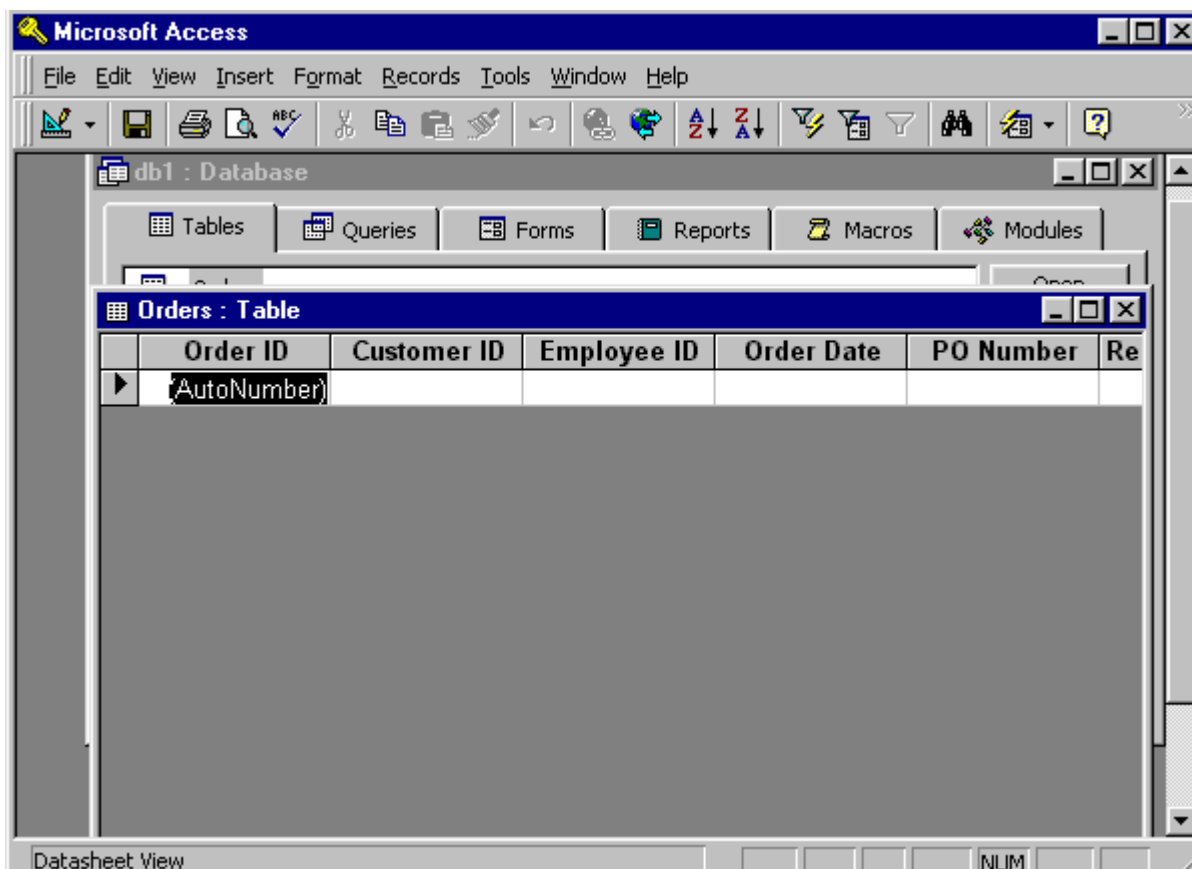
III თავი

მონაცემთა დათვალიერება და რედაქტირება

მონაცემთა ცარიელი ბაზისა და ცხრილების შექმნის შემდეგ, შეიძლება მონაცემების შეტანა და დათვალიერება. მონაცემების შეტანისა და დათვალიერების მოქმედებები შეგიძლიათ შეასრულოთ როგორც ცხრილის ფურცლების, ისე ფორმების დათვალიერების რეჟიმში. ცხრილის რეჟიმში სტრიქონებისა და სვეტების ფორმატში მუშაობის საშუალებას იძლევა. ფორმების დათვალიერების რეჟიმის დროს კი საქმე გვაქვს თითოეულ ჩანაწერთან.

3.1. ცხრილის დათვალიერება ცხრილურ ფორმატში

ცხრილის შექმნის შემდეგ, შეგიძლიათ დაიწყოთ მონაცემების შეტანა. იმისათვის, რომ კონსტრუქტორის რეჟიმში გადართოთ ცხრილის რეჟიმში, მენიუდან აირჩიეთ View→Datasheet (ნახვა→მონაცემების ფურცელი) ბრძანება ან მონაცემთა ბაზის ფანჯარაში დააწკაპუნეთ Open (გახსნა) ღილაკზე. ცხრილის რეჟიმში საშუალებას იძლევა მონაცემები შეიტანოთ და დაათვალიეროთ ცხრილურ ფორმატში. ცხრილის რეჟიმში, ახალი ცხრილი შეიცავს ერთ ცარიელ სტრიქონს, იხ. სურ. 3.1.



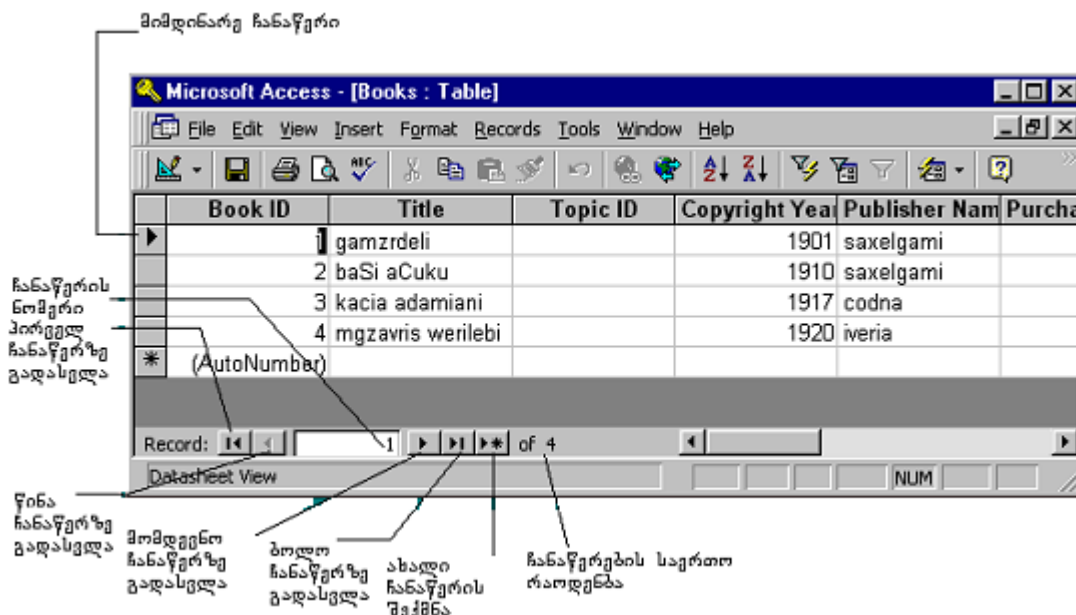
სურ. 3. 1. ცხრილის დათვალიერების რეჟიმში

Access-ში მონაცემების შეტანა ძალიან ჰგავს Word-ისა და Excel-ის ცხრილებში მონაცემების შეტანას. მონაცემი შევაქვთ უჯრედში და შემდეგ <Tab> ან <Enter> კლავიშების საშუალებით გადადიხართ მომდევნო უჯრედზე. როდესაც ბოლო ჩანაწერი შევაქვთ ამ კლავიშების მეშვეობით ავტომატურად გადაადგილდება კურსორი მომდევნო ჩანაწერზე.

ცხრილში ველების მონიშვნა, რედაქტირება და გადაადგილება შესაძლებელია როგორც კლავიშების, ისე თაგვის მეშვეობით. კლავიშების ფუნქციები მოცემულია Access-ის ნებისმიერ სახელმძღვანელოში.

გავეცნოთ ამ ოპერაციების შესრულებას თაგვის საშუალებით.

ცხრილის პირველ, მეორე, ბოლო ან მითითებულ ჩანაწერზე გადასასვლელად მაჩვენებლით დააწკაპუნეთ გადაადგილების კლავიშებზე, რომლებიც მოთავსებულია ცხრილის ბოლო ჩანაწერის ქვეშ მდგომარეობის სტრიქონში და გიჩვენებთ მიმდინარე ჩანაწერს. ეს ველი შეგიძლიათ გამოიყენოთ კონკრეტულ (მითითებულ) ჩანაწერზე გადასასვლელად, რისთვისაც უნდა დააწკაპუნოთ მასზე, შეიტანოთ იმ ჩანაწერის ნომერი, რომელზედაც გსურთ გადასვლა და დაარტყათ <Enter> კლავიშს. იმისათვის რომ მიხვიდეთ იმ სტრიქონამდე ან ველამდე, რომლებიც არ ჩანს, შეგიძლიათ გამოიყენოთ ლიფტი.



სურ. 3. 2. ცხრილის რედაქტირების საშუალებები

☒ მონაცემების რედაქტირების შემდეგ Word-ისა და Excel-საგან განსხვავებით Access-ში არ გჭირდებათ შენახვა, რადგან ცვლილება ავტომატურად ინახება.

ზოგიერთი კლავიშში, იმის მიხედვით, თუ რომელ რეჟიმში მუშაობთ სხვადასხვა ფუნქციას ასრულებს. მაგალითად, რედაქტირების რეჟიმში <Home> კლავიშში კურსორს გადაადგილებს მოცემული ველის დასაწყისში, გადაადგილების რეჟიმში იგივე კლავიშში კურსორს გადაადგილებს ჩანაწერის დასაწყისში. ერთი რეჟიმიდან მეორეში გადართვა ხდება ფუნქციური <F2> კლავიშით.

ჩანაწერის წასაშლელად, ასლის გასაკეთებლად ან ბუფერზე გადასატანად, საჭიროა მისი მონიშვნა. მონიშვნა შეგიძლიათ როგორც კლავიშით, ისე მონიშვნის სელექტორით. ჩანაწერის მონიშვნის შემდეგ, კლავიშზე დარტყმით, ეს ჩანაწერი წაიშლება. ასლის გასაკეთებლად ან გადასაადგილებლად, ბუფერზე გადატანა ხდება შესაბამისად კლავიშების კომბინაციით, <Ctrl>+<X> ან <Ctrl>+<C> ბუფერიდან გადმოწერა კი - <Ctrl>+<V> კლავიშების კომბინაციით.

თაგვის საშუალებით რომ მონიშნოთ, საჭიროა დააწკაპუნოთ ველის სელექტორზე და გადაათრიოთ მთელი სვეტი.

ახალი ჩანაწერის დამატება

ჩანაწერის დასამატებლად Access-ი გთავაზობთ ორ რეჟიმს: რედაქტირებისა და მონაცემთა შეტანის რეჟიმებს. რედაქტირების რეჟიმში ახალი ჩანაწერის დამატება ხდება ცხრილის ბოლოში. როგორც კი დაიწყოთ ჩანაწერის შეცვლას ან დამატებას, Access-ი ავტომატურად გადადის ამ რეჟიმში. მონაცემების შეტანის რეჟიმში ცხრილის ჩანაწერები არ არის, გამოჩნდება ცარიელი ცხრილი, რომელიც მზადაა ინფორმაციის მისაღებად. მონაცემთა შეტანის რეჟიმის გასააქტიურებლად მენიუდან აირჩიეთ Record→Data Entry (ჩანაწერი→მონაცემების შეტანა), ხოლო ამ რეჟიმიდან გამოსასვლელად კი Record→Remove Filter/Sort (ჩანაწერი→ფილტრის მოშლა) ბრძანება.

ცხრილში ახალი ჩანაწერის შესატანად, ნებისმიერ შემთხვევაში შეგიძლიათ გამოიყენოთ ცხრილის რეჟიმში, ბოლოში, ჩანაწერის დამატების შესაძლებლობა. ახალი ჩანაწერის შესაქმნელად ასევე შეგიძლიათ გამოიყენოთ იარაღების სტრიქონიდან New Record (ახალი ჩანაწერი) ღილაკი.

books : Table			
bookid	title	publishyear	page
1	phizika	9/10/85	500
2	ewgeni onegin	8/5/17	50
3	Matanaliz	7/7/92	30
4	zadachnik po mat	4/3/95	100
5	tri mushketera	6/5/10	500
6	grafinia demonsor	8/12/17	300
(AutoNumber)			0

სურ. 3. 3. ცარიელი სტრიქონი ახალი ჩანაწერისათვის

ჩანაწერის სელექტორის სვეტის საშუალებით Access მასში გამოიტანს შემდეგ სიმბოლოებს:

- სამკუთხედი (ისრის თავი) მიუთითებს მიმდინარე ჩანაწერს.
- ვარსკვლავი მონიშნავს ცხრილის ბოლოში მოთავსებულ ცარიელ ჩანაწერს. როდესაც ამ ჩანაწერზე გადადიხართ ვარსკვლავი იცვლება სამკუთხედით, რომელიც აღნიშნავს მიმდინარე ჩანაწერს.
- ფანქარი მიუთითებს, რომ მოცემული ჩანაწერი შეიცავს ისეთ მონაცემებს ან შესწორებულ ჩანაწერს, რომელიც ჯერ კიდევ შენახული არ არის.
- დახურული შეღწევის ნიშანი (გადახაზული წრე) მიუთითებს, რომ მოცემული ჩანაწერის რედაქტირება მიმდინარეობს სხვა ობიექტში (ფორმაში ან მოთხოვნაში) ან სხვა მომხმარებლის მიერ, თუ ქსელში ხართ ჩართული. მანამ, სანამ მომხმარებელი მასთან მუშაობას არ დაამთავრებს ასეთი ჩანაწერის მხოლოდ წაკითხვაა შესაძლებელი.

მონაცემების შეტანისას Access ამოწმებს, რამდენად სწორადაა შეტანილი ველის ტიპი და სხვა პარამეტრები. მონაცემების არასწორად შეტანის შემთხვევაში Access არ გადავა მომდევნო ველზე ან ჩანაწერზე, მანამ, სანამ შეცდომას არ გაასწორებთ.

ცხრილში ახალი ჩანაწერის დასამატებლად უნდა შეასრულოთ შემდეგი მიმდევრობა:

1. ცხრილში ჩანაწერის დასამატებლად არსებული ჩანაწერების დათვალიერების გარეშე, აირჩიეთ მენიუდან Record→Data Entry (ჩანაწერი→მონაცემების შეტანა), ან კურსორი გადაადგილეთ ცხრილის ბოლოში ცარიელ ჩანაწერზე;

2. დაიწყო ჩანაწერის შეტანა. ჩანაწერის ინდიკატორი მიიღებს ფანქრის ფორმას, ხოლო Access დაამატებს ახალ ცარიელ ჩანაწერს. მომდევნო ველზე გადასასვლელად გამოიყენეთ <Enter> ან <Tab> კლავიშები.

3. გააგრძელეთ მონაცემების შეტანა მიმდინარე ჩანაწერის დანარჩენ ველებში. Access-ს ავტომატურად შეაქვს მოვლელის ველში საჭირო ინფორმაცია. ველიდან ველში გადასვლა შეგიძლიათ <Tab> კლავიშით რის შემდეგაც შესაძლებელია Access-ის მიერ ნაგულისხმევი პარამეტრების შეცვლა.

4. იმისათვის, რომ დაინახოთ ყველა ჩანაწერი, აირჩიეთ მენიუდან ბრძანება Record→Remove Filter/Sort(ჩანაწერი→ფილტრის მოშლა/დახარისხება).

მონაცემთა რედაქტირება

ცხრილის მონაცემთა რედაქტირებისათვის <F2> კლავიშზე დარტყმით ან ამ უჯრედზე მაჩვენებლის დაწკაპუნებით დააყენეთ ტექსტური კურსორი იმ უჯრედში, რომლის შეცვლაც გსურთ, წაშალეთ არსებული ტექსტი და ჩაწერეთ ახალი.

 მაჩვენებელი დააყენეთ ჩანაწერის დასაწყისში ან ველის თავში და ერთჯერ დააწკაპუნეთ, მოინიშნება შესაბამისად ჩანაწერი ან ველი, ხოლო თუ მაჩვენებელს ერთჯერ დააწკაპუნებთ უჯრედზე, ტექსტური კურსორი უჯრედში მოთავსდება.

რედაქტირების შედეგის შეცვლა

ბრძანებების მეშვეობით Access-ს შეუძლია მონაცემებში შეტანილი ცვლილებების მოშლა და პირვანდელი სახის აღდგენა. ამისათვის იხმარება იარაღების სტრიქონში არსებული Undo (წინა დონის აღდგენა) ღილაკი. იმავეს შესრულება შესაძლებელია მენიუდან Edit→Undo (რედაქტირება→წინა დონის აღდგენა) ბრძანებით.



მას შემდეგ, რაც დაიწყოთ სხვა ჩანაწერის რედაქტირება, დააყენეთ ან მოხსენით ფილტრი, ან გადახედით სხვა ფანჯარაში, შეცვლას ვეღარ შეძლებთ. ე. ი. წინა ვარიანტს ვეღარ აღადგენთ.

შესაძლებელია აღადგინოთ:

- ☐ ბოლოს შენახული ჩანაწერი;
- ☐ ბოლოს წაშლილი ნებისმიერი ჩანაწერი;
- ☐ ბოლო გადაადგილება ან ჩასმა;
- ☐ ბოლოს შეტანილი სიმბოლო.

მონაცემების შეცვლა

Word-ისა და Excel-ის მსგავსად, ჩანაწერის რედაქტირებისათვის Access-საც აქვს ჩასწორების ფუნქცია. მენიუდან აირჩიეთ Edit→Replace (რედაქტირება→ჩასწორება) ბრძანება. გამოვა დიალოგური ფანჯარა Replace (ჩასწორება). ტექსტურ ველში Sample (ნიმუში) შეიტანეთ ის ტექსტი, რომლის გამოცვლაც გსურთ, მეორე ტექსტურ ველში კი ის - რომლითაც უნდა ჩაანაცვლოთ. თუ გამოიყენებთ Replace (ჩასწორება) ღილაკს, შეიცვლება მოცემულ მომენტში მონიშნული ელემენტი, Replace All (ყველას ჩასწორება) ღილაკით კი - ყველა ელემენტი, რომელიც ემთხვევა ნიმუშს.

ჩანაწერის წაშლა

თუ გსურთ მოელი ჩანაწერის წაშლა, სელექტორით მონიშნეთ იგი, მენიუდან აირჩიეთ Edit→Delete Record (რედაქტირება→ჩანაწერის წაშლა) ბრძანება, მონიშნული ჩანაწერი წაიშლება და ეკრანზე გამოვა შეტყობინება, რომელიც გთხოვთ მისცეთ დასტური ან უარყოთ ოქვენი გადაწყვეტილება. დასტურის შემდეგ ჩანაწერი ცხრილში აღარ იქნება.

მონაცემის ამოჭრა, ასლის გაკეთება და ჩასმა

Windows-ის სტანდარტული მოქმედებები, მონაცემის ამოჭრა, ასლის გაკეთება და ჩასმა Access-შიც ისევე მუშაობს, როგორც Word-სა და Excel-ში. მონაცემები შეგიძლიათ ამოჭრათ, გააკეთოთ ასლი ან გადაიტანოთ ერთი უჯრედიდან მეორეში ან ერთი ცხრილიდან მეორეში.

იმისათვის, რომ ამოჭრათ მთელი ჩანაწერი ან გააკეთოთ ასლი, ჯერ უნდა მონიშნოთ იგი, რისთვისაც მენიუდან აირჩიეთ Edit→Cut/Copy (რედაქტირება→ამოჭრა/ასლის გაკეთება) ბრძანება. ჩანაწერი მოთავსდება ბუფერში, შემდეგ იმ ცხრილში, რომელშიც გსურთ მისი გადმოწერა მონიშნეთ ის ჩანაწერი, რომლის შეცვლაც გსურთ და მენიუდან აირჩიეთ Edit→Paste (რედაქტირება→ბუფერიდან გადმოწერა) ან Edit→Paste Append (რედაქტირება→ბოლოში ჩასმა) ბრძანება.

 ჩანაწერის ერთი ცხრილიდან მეორეში გადასატანად, ორივე ცხრილში ჩანაწერის ველების ტიპები ერთნაირი უნდა იყოს.

ბუფერიდან გადმოწერის ოპერაციის შესრულება შეუძლებელია ისეთი ველებისათვის რომელთა ასლის გაკეთებაც არ შეიძლება, მაგალითად, პირველადი გასაღების ველისათვის.

იმისათვის, რომ გააკეთოთ ასლი ან გადაიტანოთ ჩანაწერი სხვა ცხრილში, შეასრულეთ შემდეგი:

1. პირველ ცხრილში მონიშნეთ ის ჩანაწერი, რომლის ასლის გაკეთება ან გადატანა გსურთ;
2. მენიუდან აირჩიეთ Edit→Cut/Copy (რედაქტირება→ამოჭრა/ასლის გაკეთება) ბრძანება;
3. გახსენით ის ცხრილი რომელშიც გსურთ მონაცემების მოთავსება. თუ საჭიროა, შეცვალეთ ველების მიმდევრობა და მათი თვისებები პირველი ცხრილის შესაბამისად;
4. მენიუდან აირჩიეთ Edit→Paste (რედაქტირება→ბუფერიდან გადმოწერა) ბრძანება;
5. ცხრილში მონაცემების დასამატებლად მენიუდან აირჩიეთ Edit→Paste (რედაქტირება→ბუფერიდან გადმოწერა) ბრძანება.

ცხრილის გადაადგილება

Access-ში შექმნილი ცხრილის გადასატანად Word-სა ან Excel-ში, ჯერ უნდა გახსნათ Word-ისა ან Excel-ის საჭირო დოკუმენტი, რისთვისაც დააპატარავეთ ფანჯარები და ისე განალაგეთ, რომ ეკრანზე ორივეს ხედავდეთ. გადადით მონაცემთა ბაზის ფანჯარაში, ცხრილის ან მოთხოვნის ნიშნაკი გადააცოცეთ Word-ის ან Excel-ის ფანჯარაში, რამდენიმე უჯრედის გადასატანად კი ჯერ გადადით ცხრილის რეჟიმში, ხოლო შემდეგ თავიდან მონიშნეთ და გადააცოცეთ სასურველი უჯრედები. ანალოგიურად შეგიძლიათ ობიექტების გადატანა სამუშაო მაგიდაზე.

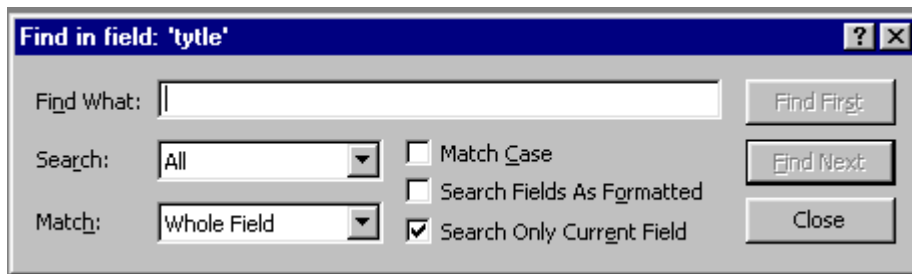
3.2. მონაცემის ძებნა

ბაზასთან მუშაობისას ხშირად საჭიროა ჩანაწერის მოძებნა. დაუშვათ, დამკვეთს აინტერესებს არის თუ არა საწყობში რომელიმე ვიდეოფირი. Access-ს შეუძლია ვიდეოფირის სახელის მიხედვით მისი მოძებნა.

იმისათვის რომ მონახოთ ჩანაწერი:

1. მენიუდან აირჩიეთ Edit→Find (რედაქტირება→ძებნა), გამოვა დიალოგის ფანჯარა Find (ძებნა), იხ. სურ. 3. 4;
2. Sample (ნიმუში) ტექსტურ ველში აკრიფეთ ის ტექსტი, რომლის მოძებნაც გსურთ;
3. მიუთითეთ ძებნის მიმართულება Search (ძებნა, All-ყველგან, Up-ზემოთ, Down-ქვემოთ) ველში;
4. აირჩიეთ შესაბამისი კრიტერიუმი Match (თანამთხვევა) ველში, აირჩიეთ ერთ-ერთი: Whole Field (მთელი ველი) ან Start of Field (ველის დასაწყისი) ან Any Part of Field (ველის ნებისმიერი ნაწილი);
5. სურვილის მიხედვით შეგიძლიათ მიუთითოთ რეგისტრის აღრიცხვა;

6. სურვილის მიხედვით, შეიძლება აირჩიოთ Search Fields As Formated (ველის ფორმატის აღრიცხვა) პარამეტრი, რომლის მეშვეობითაც ხდება თარიღის ფორმატის შენარჩუნება იმ სახით, რა სახითაც შეიტანეთ;
7. თუ ძებნა გსურთ მხოლოდ მოცემულ ველში, ალამი დააყენეთ Serch only current Fild (მხოლოდ მიმდინარე ველში) პარამეტრზე;
8. დააწკაპუნეთ Find (ძებნა) ღილაკზე, Access გამოიძახებს პირველ ჩანაწერს, რომელიც აკმაყოფილებს თქვენს კრიტერიუმებს;
9. იმისათვის რომ გააგრძელოს ძებნა დააწკაპუნეთ Find Next (შემდეგი ძებნა) ღილაკზე;
10. ძებნის დასამთავრებლად დააწკაპუნეთ Close (დახურვა) ღილაკზე.



სურ. 3. 4. მონაცემის ძებნის დიალოგური ფანჯარა

3.3. მონაცემების დახარისხება და გაფილტვრა

Access-ს შეუძლია ცხრილის რეკომში მონაცემების დახარისხება და გაფილტვრა. ეს ფუნქციები ძალიან გამოსადეგია მაშინ, როდესაც მონაცემების სიას ქმნით, რომელიმე კრიტერიუმით დახარისხების ან გაფილტვრის საფუძველზე. ძებნა იყენებს მხოლოდ ერთ კრიტერიუმს, ფილტრის საშუალებით კი ერთდროულად რამდენიმე ველისათვის გამოიყენება სხვადასვა კრიტერიუმი.

Access გთავაზობთ მონაცემთა გაფილტვრისა და დახარისხების შემდეგ ფუნქციებს:

❑ **ზრდის მიხედვით დახარისხება.** მონიშნულ ველს ან ველებს დაახარისხებს ზრდის მიხედვით. ცხრილის შენახვის შემდეგ დახარისხებული ველიც ინახება. მაგალითად, კინოფილმების საიდენტიფიკაციო ნომრის ზრდის მიხედვით დახარისხებისას ჩანაწერები დალაგდება ნომრის -1, 2, 3 და ა. შ. ზრდის მიხედვით ;

❑ **კლების მიხედვით დახარისხება.** მონიშნულ ველს ან ველებს დაახარისხებს კლების მიხედვით. ცხრილის შენახვის შემდეგ დახარისხებული ველიც ინახება. მაგალითად კინოფილმების საიდენტიფიკაციო ნომრის კლების მიხედვით დახარისხებისას ჩანაწერები დალაგდება ნომრის -9, 8, 7 და ა. შ. კლების მიხედვით ;

❑ **გაფილტვრა მონიშნულის მიხედვით.** ეძებს რომელიმე კრიტერიუმის შესაბამის მონაცემს. ძებნა შესაძლებელია რამდენიმე კრიტერიუმით (ძებნა გადაბმის საშუალებად იყენებს AND (და) ლოგიკურ კავშირს. არ გამოიყენება OR (ან), ცალ-ცალკე მიუთითება ახალი კრიტერიუმი. კრიტერიუმად არ შეიძლება გამოსახულების არჩევა.

❑ **გაფილტვრა ფორმის მიხედვით.** ჩანაწერის ძებნა ხდება რამდენიმე პირობის შესაბამისად. პირობები შეიძლება ერთმანეთთან დააკავშიროთ როგორც AND (და), ისე OR (ან) -ით. კრიტერიუმად შეიძლება გამოსახულების გამოყენებაც;

❑ **ფილტრი/დახარისხება.** არის დახარისხებისა და გაფილტვრის კომბინირებული საშუალება. შეგიძლიათ რამდენიმე კრიტერიუმის მითითება, შეასრულოთ ძებნა AND-ით და OR-ით, კრიტერიუმად შეიძლება გამოსახულების გამოყენება და დახარისხება როგორც ზრდის, ისე კლების მიხედვით.

დახარისხება ცხრილის რეჟიმში

ცხრილის რეჟიმში ჩანაწერების დახარისხება შეიძლება როგორც ზრდის, ისე კლების მიხედვით. ყოველთვის შეგიძლიათ დახარისხებული ველის მიმდევრობის შეცვლა, ანუ სხვა კრიტერიუმით ხელახლა დახარისხება. იმისათვის, რომ შეცვალთ დახარისხება, აირჩიეთ მენიუდან ბრძანება **Record→Remove Filter/Sort** (ჩანაწერი→მოხსნა ფილტრის/დახარისხების). თუ თქვენ გამოიყენეთ ფილტრი, აღნიშნული ბრძანება მასაც მოხსნის.

ცხრილის რეჟიმში მონაცემების დასახარისხებლად შეასრულეთ შემდეგი:

1. მონიშნეთ დასახარისხებელი ველი ან ველები;
2. ზრდის მიხედვით დასახარისხებლად აირჩიეთ მენიუდან **Records→Sort→Sort Ascending** (ჩანაწერი→დახარისხება→ზრდის მიხედვით) ბრძანება ან იარაღების სტრიქონზე დაარტყით შესაბამის კლავიშს;
3. კლების მიხედვით დასახარისხებლად აირჩიეთ მენიუდან **Records→Sort→Sort Descending** (ჩანაწერი→დახარისხება→კლების მიხედვით) ბრძანება ან იარაღების სტრიქონზე დაარტყით შესაბამის კლავიშს.

 ცხრილის რეჟიმში შეიძლება ერთმანეთის მომდევნო ორი ან მეტი სვეტის მონიშვნა, შემდეგ კი დახარისხება ზრდის ან კლების მიხედვით. Access-ი დახარისხებას იწყებს ყველაზე მარცხენა ველიდან. ფორმების რეჟიმში კი შეიძლება თითო ველის დახარისხება.

გაფილტვრა მონიშნულის მიხედვით

მონიშნულის მიხედვით გაფილტვრის ფუნქცია საშუალებას იძლევა აირჩიოთ საძიებელი მონაცემები და დაათვალიეროთ იმ ჩანაწერების სია, რომლებიც აკმაყოფილებენ რაიმე პირობას. მაგალითად, თუ გინდათ ნახოთ სია პროგრამებისა, რომლებიც უზრუნველყოფენ საფოსტო გზავნილების დაცვას, ინფორმაციის აღწერის პროგრამაში, აირჩიეთ სიტყვა „საფოსტო“, შემდეგ მენიუდან აირჩიეთ **Records→Filter→Filter By Selection** (ჩანაწერი→ფილტრი→ფილტრი მონიშნულით) ბრძანება ან ისარგებლოთ იარაღების სტრიქონიდან შესაბამისი ღილაკით. თუ მოისურვებთ მთელ ცხრილში ჩანაწერების დათვალიერებას, მოხსენით ფილტრი და ისე დაათვალიერეთ. ყოველთვის შეგიძლიათ ფილტრის ხელახლა დაყენება ან მოხსნა.

გაფილტვრა ფორმის მიხედვით

ფორმის მიხედვით გაფილტვრის ფუნქცია საშუალებას იძლევა აირჩიოთ მონაცემები და დაათვალიეროთ იმ ჩანაწერების სია, რომლებიც აკმაყოფილებენ არჩეულ კრიტერიუმებს. მენიუს ბრძანებით, **Records→Filter→Filter By Selection** (ჩანაწერი→ფილტრი→ფილტრის შეცვლა), ხდება ცარიელი ცხრილის გამოძახება ე.ი. ცხრილი არ შეიცავს ჩანაწერებს. ეკრანზე სხვა ცვლილებებიც შეიმჩნევა, კერძოდ დაფორმატებისა და ჩანაწერების მენიუ შეიცვლება ახალი მენიუთი, რომელიც შეიცავს სხვა ქვეპუნქტებს. მაგალითად ფილტრის შეცვლა, გაფართოებული ფილტრი და სხვა. იარაღების სტრიქონზეც ჩნდება შესაბამისი კლავიშები. ასე მაგალითად ჩატვირთეთ მოთხოვნიდან, შეინახეთ როგორც მოთხოვნა, დახურეთ, გაწმინდეთ ფორმულარი, გამოიყენეთ ფილტრი. მენიუში **File** (ფაილი) გაჩნდება ორი ახალი პუნქტი:

1. **Save As Query** (შეინახე როგორც მოთხოვნა), საშუალებას გაძლევთ ფილტრი შეინახოთ, როგორც მოთხოვნა;
2. **Load is Query** (ჩატვირთე მოთხოვნიდან), საშუალებას გაძლევთ ადრე შენახული მოთხოვნა გამოიყენოთ როგორც ფილტრი.

გარდა ამისა გაჩნდება ფურცლის ორი ჩანაცვლება : **Find** (ძებნა) და **Or** (ან). **Find** (ძებნა). ეს ის არის სადაც იწყებთ თქვენი ფილტრის შექმნას (ძებნის კრიტერიუმები დაკავშირებულია AND-ით), ხოლო **Or** (ან), გამოიყენება ალტერნატიული მნიშვნელობის მისათითებლად, რომლებიც უნდა აღმოჩნდნენ საბოლოო სიაში.

ფილტრით სარგებლობის შემდეგ ცხრილის დათვალიერების რეჟიმში ბრუნდებით ფილტრის მოხსნით.

ფილტრის შესაქმნელად შეასრულეთ შემდეგი:

1. აირჩიეთ მენიუდან Records→Filtr→Filtr By Selectioun (ჩანაწერი→ფილტრი→ფილტრის შეცვლა). გამოვა დიალოგური ფანჯარა ფილტრის შესაცვლელად;
2. გახსენით ჩანაცვლება Find (ძებნა) და დააწკაპუნეთ საჭირო ველზე;
3. გაშალეთ სია და აირჩიეთ მნიშვნელობა ან თვითონ შეიტანეთ;
4. გააგრძელეთ კრიტერიუმის მითითება სხვა ველებისათვის;
5. თუ საჭიროა ალტერნატიული კრიტერიუმის მითითება დაარტყით Or (ან)-ს და შეავსეთ იგი სხვა მნიშვნელობების კომბინაციით;
6. შეინახეთ ფილტრი, როგორც მოთხოვნა;
7. შედეგის დასათვალიერებლად აირჩიეთ მენიუდან Records→Aply Filter/Sort (ჩანაწერები→ისარგებლე ფილტრით/დახარისხებით)
8. ყველა შესაძლო ჩანაწერის დასათვალიერებლად ისარგებლეთ ბრძანებით მენიუდან Records→Remove Filter/Sort (ჩანაწერები→ფილტრის მოხსნა).

☒ ფილტრიდან ყველა კრიტერიუმის მოსახსნელად აირჩიეთ მენიუდან ბრძანება Edit→Clear Grid (რედაქტირება→ფორმულების გაწმენდა), ან დაარტყით შესაბამის ღილაკზე.

IV თავი

ფორმის შექმნა

საინფორმაციო სისტემაში უმნიშვნელოვანესია ფორმა, რომელიც გამოიყენება ინფორმაციის შეგროვებისა და შენახვისათვის. ზოგი ფორმა, მაგალითად, კადრების აღრიცხვის ფურცელი უბრალოდ აგროვებს ინფორმაციას, ამავე დროს სხვა სახის ფორმები, როგორიცაა საგადასახადო დეკლარაციის ფორმა, ასრულებს გამოთვლებს. ზოგიერთი ფორმა ასრულებს უწყისის ფუნქციას (მაგალითად მაღაზიის ანგარიში ან ქვითარი).

4. 1. ფორმის დაგეგმვა

ფორმა გვაძლევს ცხრილის მონაცემების სხვა სახით დანახვის საშუალებას. ამიტომ ფორმის შექმნის პირველი ნაბიჯი ცხრილის შექმნა ან ამ ცხრილის სტრუქტურის გარკვევაა. ფორმების პრობლემა ხშირად დაკავშირებულია არასწორად დამუშავებულ ცხრილთან. გახსოვდეთ, რომ ცხრილის ველისა და თვით ცხრილის ისეთი თვისებები, როგორიცაა მონაცემებისა და მონაცემთა ტიპების შემოწმება, საშუალებას იძლევა გააუმჯობესოთ მონაცემების ხარისხი.

მას შემდეგ, რაც ცხრილის სტრუქტურა შექმნილია, შეგიძლიათ დაიწყოთ ფორმების შექმნა. ცხრილთან შედარებით ფორმით მონაცემების დათვალიერება უფრო მოსახერხებელია.

□ ფორმის საშუალებით ხდება ერთი ჩანაწერის მთლიანად, ჩვეულებრივ ვერტიკალური ფორმატით, გამოჩენა ეკრანზე;

□ ფორმას შეუძლია მონაცემების გარეგნულ სახეზე აქტიური ზეგავლენა მოახდინოს სხვადასხვა შრიფტის, ფერისა და გრაფიკის სახით;

□ ფორმა შეიძლება შეიცავდეს ისეთ ველებს, რომელთა რედაქტირებაც შეუძლებელია და ისეთებსაც, რომელთა რედაქტირება შესაძლებელია;

□ ფორმზე შეიძლება ველების ადგილების შეცვლა;

□ ფორმა შეიძლება შეიცავდეს სხვადასხვა ცხრილის ველებს;

□ ფორმას აქვს სპეციალური ფუნქციები - სიები;

□ ფორმა შეიძლება შეიცავდეს გრაფიკს;

□ ფორმა ზოგიერთი მოქმედების ავტომატიზების საშუალებას იძლევა და გამოაქვს სპეციალურად შექმნილი მენიუ.

კარგად დამუშავებული ფორმა ადვილად იკითხება. საჭიროა მისი ისე დამუშავება, რომ ადვილი იყოს მასში მონაცემების შეტანა.

ფორმის დამუშავებისას უნდა გაითვალისწინოთ რომ:

□ ფორმა უნდა იყოს მარტივი. უნდა გამოიყენოთ ისეთი შრიფტები და ფერები, რომლებიც ადვილად იკითხება და კარგად აღიქმება. ფორმის გარეგნული სახის გასაუმჯობესებლად უნდა გამოიყენოთ გრაფიკები და სხვა ობიექტები, მაგრამ არ უნდა გადატვირთოთ იგი;

□ თუ ფორმა შავ-თეთრ საბეჭდო მოწყობილობაზე იბეჭდება, მოახდინეთ ფერებისა და ელემენტების ისეთი კორექტირება, რომ ნაბეჭდი გასაგები იყოს. მართალია, ფორმის ბეჭდვა შეიძლება, მაგრამ ბეჭდვისთვის განკუთვნილია უწყისი და ჯობს ის დაიბეჭდოს;

□ გასაგებად მიუთითეთ რომელი მონაცემი სად უნდა იყოს შეტანილი.

4. 2. ფორმის შექმნა

ახალი ფორმის შექმნა შეიძლება როგორც იარაღების სტრიქონიდან, ისე დიალოგური ფანჯრის მენიუდან New Form (ახალი ფორმა) ბრძანების მეშვეობით. იარაღების სტრიქონის New Object (ახალი ობიექტი) ღილაკი ქმნის მარტივ ფორმას არსებული ცხრილის ან მოთხოვნის საფუძველზე.

ყველა ახალი ფორმა იქმნება დიალოგური ფანჯრის New Form (ახალი ფორმა) ბრძანების გამოყენებით. მის გასახსნელად ჯერ დააწკაპუნეთ იარაღების სტრიქონში არსებულ ღილაკზე (ახალი ობიექტი) New Object, შემდეგ კი აირჩიეთ Form (ფორმა) ღილაკი ან მონაცემთა ბაზის ფანჯარაში გახსენით ჩანაცვლება - ფორმა და დააწკაპუნეთ

New (ახალი) დილაკზე. ნებისმიერ შემთხვევაში გამოვა New Form (ახალი ფორმა) დიალოგური ფანჯარა.



სურ. 4. 1. დიალოგური ფანჯარა ახალი ფორმის შესაქმნელად

New Form (ახალი ფორმის) დიალოგური ფანჯრის გაცნობა

ეს ფანჯარა საშუალებას იძლევა აირჩიოთ ცხრილი ან მოთხოვნა, რომლისთვისაც გსურთ ფორმის შექმნა და გთავაზობთ შემდეგ პარამეტრებს:

- ❑ *Design View (კონსტრუქტორი)* დაგეხმარებათ საკუთარი ფორმის თავდაპირველ შექმნაში.
- ❑ *Form Wizard (ფორმის ოსტატი)* დაგეხმარებათ ფორმის შექმნაში, გაძღვეთ კითხვებს და იყენებს წინასწარ განსაზღვრულ ფორმის შაბლონს;
- ❑ *AutoForm:Columnar (ავტოფორმა, სვეტებიანი)* - გამოაქვს ეკრანზე ერთი ჩანაწერი ვერტიკალური ფორმატით (ყოველი ველის სათაური იწყება ახალი სტრიქონით);
- ❑ *AutoForm:Tabular (ავტოფორმა ლენტისებრი)* - გამოაქვს რამდენიმე ჩანაწერი სტრიქონისა და სვეტის ფორმატში;
- ❑ *AutoForm:Datasheet (ავტოფორმა ცხრილური)* - ამ ფორმის სახე პრაქტიკულად არ განსხვავდება ცხრილური ფურცლის ფორმატისაგან. ეს ფორმები ხშირად გამოიყენება სხვა ფორმაში;
- ❑ *Chart Wizard (დიაგრამის ოსტატი)* - ქმნის მონაცემთა დიაგრამის ან გრაფიკის ფორმას;
- ❑ *PivotTable Wizard (ნაკრები ფორმა ოსტატი)*. ნაკრები ცხრილის საფუძველზე შექმნილი ფორმა. იხ. სურ. 4. 1.

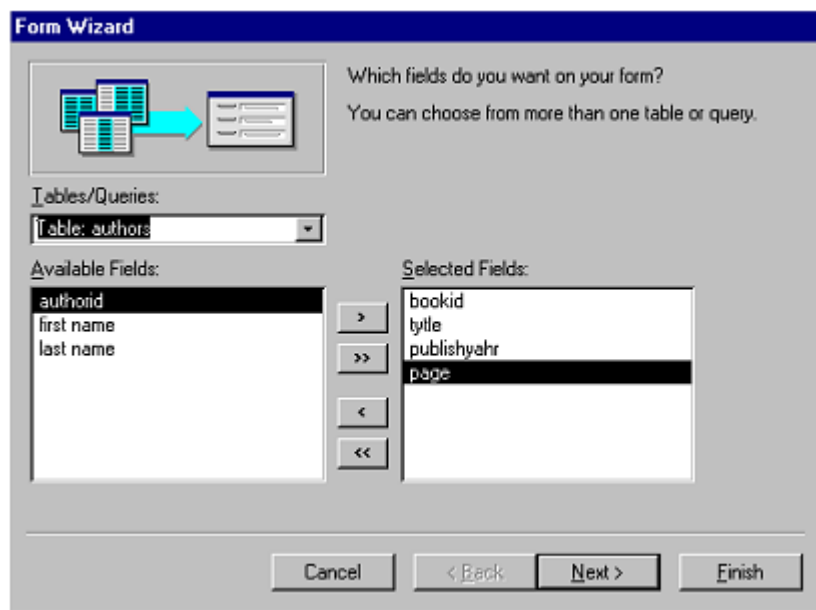
ფორმის შექმნა ფორმის ოსტატის მეშვეობით

ფორმის ოსტატი ქმნის ფორმის სტრუქტურას მითითებული სპეციფიკაციის საფუძველზე. ჩნდება დიალოგური ფანჯარა, რომლის მეშვეობითაც მივუთითებთ, თუ რომელი ცხრილის ან მოთხოვნის საფუძველზე იქმნება ფორმა და რომელი ტიპის ფორმა უნდა გამოიყენოს.

ფორმის ოსტატის მეშვეობით ახალი ფორმის შესაქმნელად:

1. იარაღების სტანდარტულ სტრიქონზე დააწკაპუნეთ New Object (ახალი ობიექტი) დილაკზე, შემდეგ კი დილაკების სიიდან აირჩიეთ Form (ფორმა). გამოვა New Form (ახალი ფორმა) დიალოგური ფანჯარა;
2. ტექსტურ ველში შეიტანეთ ან სიიდან აირჩიეთ იმ ცხრილის (მოთხოვნის) სახელი, რომლის საფუძველზეც ქმნით ფორმას;
3. იმავე ფანჯარაში აირჩიეთ ფორმის ოსტატი და მიეცით დასტური. გამოვა Form Wizard (ფორმის ოსტატი) დიალოგური ფანჯარა;
4. ტექსტური სიიდან Table/Query (ცხრილები/მოთხოვნები) აირჩიეთ ცხრილი ან მოთხოვნა, რის საფუძველზეც ქმნით ფორმას;

5. იმავე ფანჯრის Available Field (არსებული ველები) სიიდან, მონიშნეთ ის ველები რომლის ჩართვაც გსურთ ფორმაში, ისარგებლეთ >, <, >>, << ღილაკებით. სამუშაოს გასაგრძელებლად დააწკაპუნეთ Next (მომდევნო) ღილაკზე;
6. მომდევნო დიალოგურ ფანჯარაში აირჩიეთ სტრუქტურის ტიპი (სვეტებიანი, ლენტისებრი, ცხრილური) და დააწკაპუნეთ Next (მომდევნო) ღილაკზე ღილაკზე;
7. მომდევნო დიალოგურ ფანჯარაში აირჩიეთ ფორმის სტილი და დააწკაპუნეთ Next (მომდევნო) ღილაკზე;
8. Form Name (ფორმის სახელი) დიალოგური ფანჯრის ტექსტურ ველში შეიტანეთ ფორმის სახელი და აირჩიეთ შესაბამისად ერთ-ერთი პარამეტრი ან ფორმის დათვალიერება ან გახსნა მონაცემების შესატანად;
9. ფორმასთან მუშაობისას საცნობარო ინფორმაციის გამოსატანად შეგიძლიათ დააყენოთ შესაბამისი ალამი;
10. დააწკაპუნეთ Finish (შესრულება) ღილაკზე.



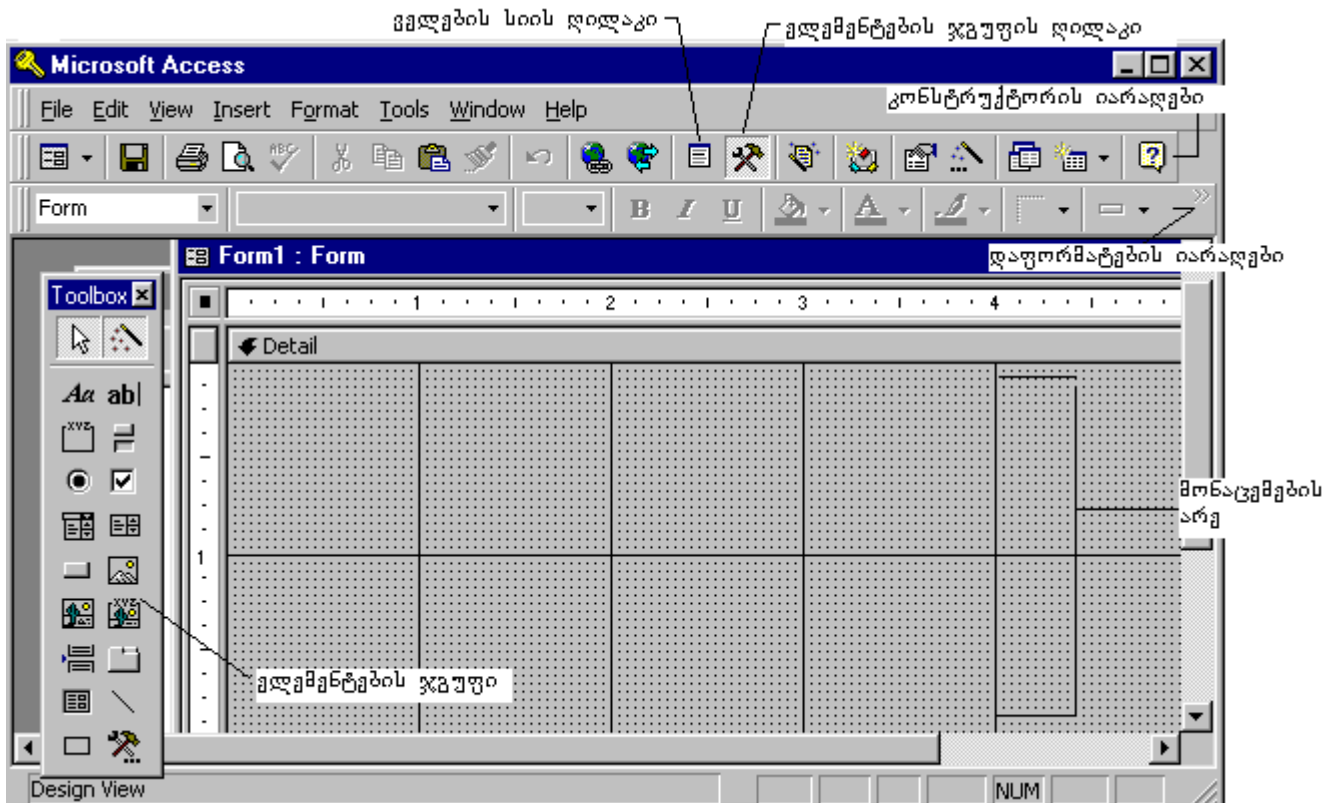
სურ. 4. 2. ფორმის შექმნა ფორმის ოსტატით

ფორმების შექმნა კონსტრუქტორის საშუალებით

ახალი ფორმის შესაქმნელად შეგიძლიათ ისარგებლოთ კონსტრუქტორის რეჟიმით, რომლის დროსაც მონაცემები გამოდის გარკვეულ ადგილას, რასაც ვერ განახორციელებთ ფორმის ოსტატის წინასწარ განსაზღვრული შაბლონით. მაგალითად, შეიძლება შექმნათ რომელიმე სახელმწიფო ფორმის მსგავსი ფორმა. შეიძლება შექმნათ ფორმა, რომელიც შეიცავს მხოლოდ მმართველ კლავიშებსა და მათზე წარწერებს. ასეთი ფორმა, მენიუს სახით, გამოიყენება სხვა ფორმებისა და უწყისებისათვის.

კონსტრუქტორის რეჟიმში ფორმის შესაქმნელად:

1. სტანდარტულ იარაღებს შორის მონახეთ და დააწკაპუნეთ New Object (ახალი ობიექტი) ღილაკზე, ჯგუფიდან აირჩიეთ Form (ფორმა) ღილაკი, გამოვა დიალოგური ფანჯარა New Form (ახალი ფორმა);
2. სიიდან აირჩიეთ იმ ცხრილის ან მოთხოვნის სახელი, რომლის საფუძველზეც ქმნით ფორმას;
3. სხვა სიაში აირჩიეთ სიტყვა კონსტრუქტორი და მიეცით დასტური. Access გამოიტანს ცარიელ ფორმას და გადაგიყვანთ კონსტრუქტორის რეჟიმში. იხ. სურ. 4. 3.



სურ. 4. 3. ახალი ფორმის შექმნა კონსტრუქტორის რეჟიმში

4.3. ფორმების მოდიფიკაცია და უწყისის სტრუქტურის დამუშავება

ფორმების კონსტრუქტორისა და უწყისის კონსტრუქტორის რეჟიმები ერთნაირია. ორივე ობიექტში (ფორმა, უწყისი) არის კონსტრუქტორის იარაღებისა და დაფორმების იარაღების ელემენტთა ჯგუფები, მონაცემთა არე, ფორმებისა და უწყისების გვერდების თავი და ბოლო. ორივე რეჟიმი ერთმანეთის მსგავსია.

ფორმებისა და უწყისების სხვადასხვა ნაწილთან მუშაობა

ფორმებისა ან უწყისების შექმნა და მოდიფიცირება შეიძლება როგორც უწყისის კონსტრუქტორის, ისე ფორმის კონსტრუქტორის რეჟიმში. მართალია, კონსტრუქტორის რეჟიმში ახალი ფორმები შეიცავს მხოლოდ მონაცემების არეს, მაგრამ შესაძლებელია სხვა ნაწილების დამატებაც. მოცემულ გვერდზე ან ფორმებისა და უწყისებისათვის სათაურისა და შენიშვნის დასამატებლად მენიუდან აირჩიეთ ბრძანება View→Page Header/Foter (ნახვა→გვერდის თავი/ბოლო) ან View→Form Header/Foter (ნახვა→ფორმის თავი/ბოლო). საჭიროების შემთხვევაში მათი წაშლა შეგიძლიათ ამავე ბრძანებებით.

უწყისების ან ფორმების ნაწილების აღწერა

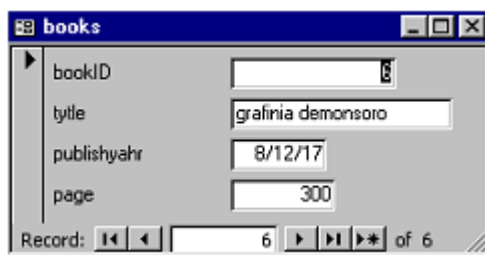
- ❑ ფორმის ან უწყისის სათაური გამოდის ეკრანის ზედა ნაწილში და იბეჭდება პირველი გვერდის თავში;
- ❑ ფორმის თავი გამოდის მხოლოდ ბეჭდვისას ყოველი გვერდის თავში;
- ❑ მონაცემების არე შეიცავს მონაცემებს;
- ❑ ფორმის ბოლო გამოდის მხოლოდ ბეჭდვისას ყოველი გვერდის ბოლოს;
- ❑ ფორმების ან უწყისების შენიშვნები - გამოდის ეკრანის ქვემოთ. ბეჭდვისას გამოდის ბოლო გვერდის ქვედა ნაწილში;
- ❑ უწყისების ჯგუფში სათაურები და შენიშვნები გამოდის მხოლოდ უწყისებში და ეოფს მას კატეგორიებად.

ღილაკი	ღილაკის სახელი	შინაარსი
	ნახვა	გამოიძახებს ფორმების, კონსტრუქტორის ან ცხრილის დათვალიერების რეჟიმს.
	ნიმუშის მიხედვით დაფორმატება	იღებს მონიშნული ობიექტის ფორმატს და გამოიყენებს მომდევნო მონიშნულ ობიექტზე
	ველების სია	გამოიძახებს ფორმებთან ან უწყისებთან დაკავშირებულ ველების სიას
	ელემენტების ჯგუფი	გამოიძახებს ან დახურავს ელემენტების ჯგუფს, რომელიც შეიცავს ობიექტებს, ფორმების ან უწყისების ასაგებად.
	ავტომატური დაფორმატება	იღებს წინასწარ განსაზღვრულ სტილს
	პროგრამა	გამოიძახებს მოცემული ფორმის ან უწყისისათვის მოდულის ფანჯარას (პროგრამულ ოპერატორებს)
	პარამეტრები	გამოიძახებს პარამეტრების ფანჯარას, სადაც შეგიძლიათ მართვის ელემენტის მახასიათებლების შეცვლა
	ააგეთ	გამოიძახებს მგეგმავს ან ამგებს

4.4. მართვის ელემენტებთან მუშაობა

არსებული ფორმებისა და უწყისების სტრუქტურის შეცვლა არის საკმაოდ მარტივად ხდება. ხშირად საკმარისია მხოლოდ ობიექტის გადართევა, საჭირო ადგილას მოთავსება, შემდეგ კი მისი ფორმის გაუმჯობესება, ფერის შეცვლა ან რაიმე თვისების დაყენება.

ობიექტებს, რომლებსაც ათავსებთ ფორმებზე ან უწყისებზე მართვის ელემენტები ეწოდება. იმისათვის, რომ შეცვალოთ როგორც შექმნილი ფორმის, ისე უწყისის გარეგნული სახე, შეგიძლიათ ამ ელემენტების გადაადგილება, წაშლა ან მათი მახასიათებლების შეცვლა. ამ მიზნით შეგიძლიათ გამოიყენოთ დაფორმატების იარაღების ჯგუფის ღილაკები.



სურ. 4. 4. ფორმის სახე, რომელიც შექმნილია Wizard-ით (ოსტატი)

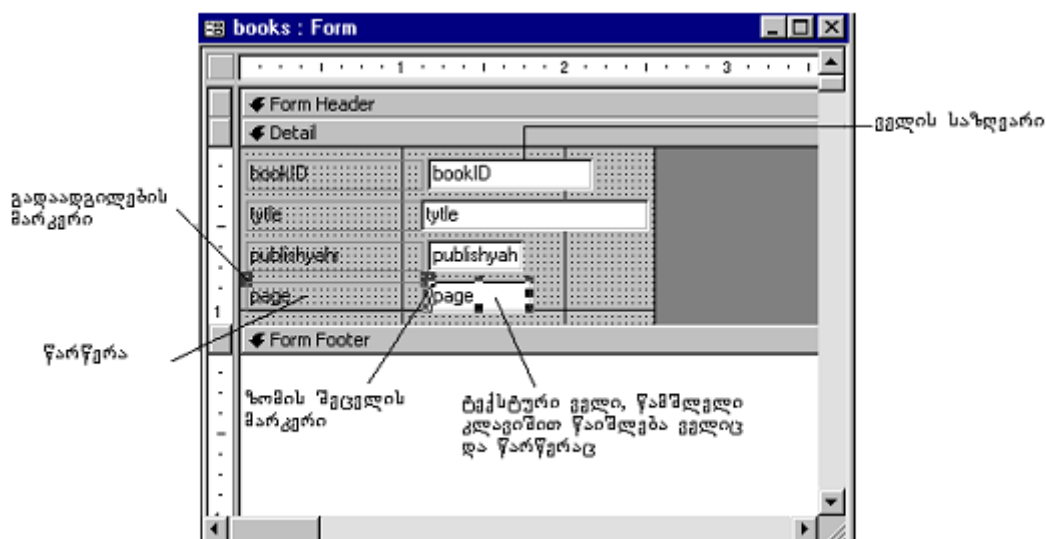
რედაქტირება (შესწორება)

მართვის ტექსტურ ელემენტზე დაწკაპუნებით იგი მონიშნება. როდესაც მართვის ელემენტი მონიშნულია, შეგიძლიათ მისი გადაადგილება, ზომისა და გარეგნული სახის შეცვლა. ამ ელემენტზე მეორედ დაწკაპუნების შემდეგ თქვენ აღმოჩნდებით რედაქტირების რეჟიმში. სიმბოლოების წასაშლელად და ახალი სიმბოლოების შესატანად შეგიძლიათ ისარგებლოთ Backspace (უკუშლა) და Delete (წაშლა) კლავიშებით. „წარწერის“ ტიპის მართვის ელემენტის შესწორება ხშირად ხდება საჭირო, ხოლო ტექსტური ველისა კი - იშვიათად.

მონიშვნა და კორექტირება

მანამ, სანამ დაიწყებდეთ ფორმაზე მართვის ელემენტების მოდიფიცირებას და მათი თვისებების დაყენებას, უნდა იცოდეთ, როგორ მოინიშნება მართვის ელემენტი და როგორ მოინიშნება ფორმა.

მართვის ელემენტის მოსანიშნად დააწკაპუნეთ მასზე ან ისარგებლეთ იარაღების ჯგუფში არსებული Select Object (ობიექტის მონიშვნა) ღილაკით და აირჩიეთ მისი სახელი. Access მონიშნავს არჩეულ ობიექტს პატარა შავი კვადრატებით. ამ პატარა შავ კვადრატებს ეწოდება ზომების შეცვლის მარკერები, ხოლო ზემოთ არსებულ უფრო დიდ კვადრატებს - გადაადგილების მარკერები. თუ მაჩვენებელს დააყენებთ პატარა შავ კვადრატზე, მაშინ იგი მიიღებს ორმხრივი ისრის ფორმას, მასზე დაჭერით და გადართევით შეგიძლიათ ელემენტის ზომის შეცვლა. თუ მაჩვენებელს დააყენებთ ელემენტის ზედა მარცხენა კუთხეში მოთავსებულ მოზრდილ კვადრატზე, მაჩვენებელი მიიღებს საჩვენებელი თითის (შემდეგში ვიხმართ მხოლოდ თითს) ფორმას, მასზე დაჭერით და გადართევით შეგიძლიათ მხოლოდ ელემენტის (მაგალითად წარწერის ან ტექსტური ველის) გადაადგილება სასურველ ადგილას. თუ მაჩვენებელს დააყენებთ არა კვადრატზე, არამედ ელემენტის საზღვარზე, მაჩვენებელი მიიღებს ხელის მტევნის ფორმას, მასზე დაჭერით და გადართევით ტექსტურ ველთან ერთად გადაადგილდება წარწერაც. ერთდროულად რამდენიმე ელემენტის მოსანიშნად უნდა გამოიყენოთ <Shift> კლავიშისა და მაჩვენებლის დაწკაპუნების კომბინაცია.



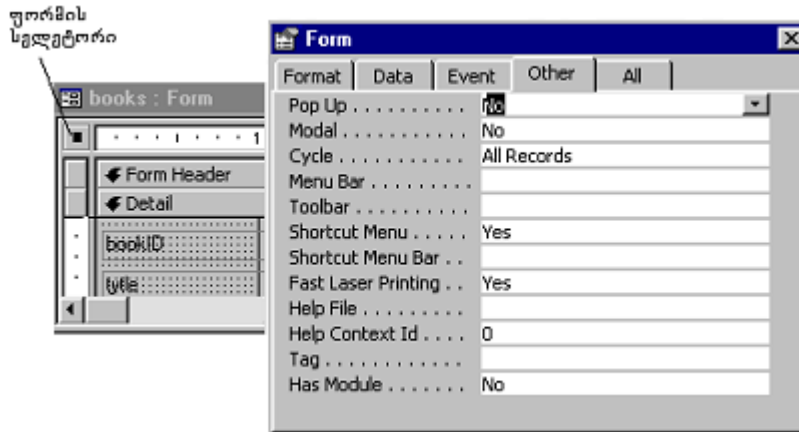
სურ. 4. 5. ობიექტისა და მასთან დაკავშირებული ველის მონიშვნა

მთელი ფორმის ან უწყისის მოსანიშნად, დააწკაპუნეთ ზედა მარცხენა კუთხეში არსებული ფორმის (უწყისის) მონიშვნის სელექტორზე. იხ. სურ. 4. 5.

მონიშნული ელემენტის ამოტრა და გადატანა ბუფერზე, ბუფერიდან გადმოწერა, წაშლა და შემთხვევით წაშლილი ელემენტის აღდგენა და დაფორმატება ზუსტად ისევე ხდება, როგორც Word-სა და Excel-ში.

ფორმებისა და მართვის ელემენტების თვისებების დაყენება

Access-ის ყოველ ობიექტს აქვს თვისება, რომელიც განსაზღვრავს ობიექტის გარეგნულ სახეს და მის ყოფაქცევას. მოცემული მართვის ელემენტის თვისებების სანახავად საჭიროა, ორჯერ დააწკაპუნოთ ელემენტზე, ან მონიშნოთ იგი და მენიუდან აირჩიოთ View→Properties (ნახვა→თვისებები) ბრძანება. იმისათვის, რომ ნახოთ ფორმის ან უწყისის თვისებები ორჯერ დააწკაპუნეთ ფორმის ან უწყისის მონიშვნის სელექტორზე, რომელიც მოთავსებულია ფორმის ან უწყისის ზედა მარცხენა კუთხეში. თვისებების ფანჯრის გახსნის შემდეგ, შეგიძლიათ მონიშნოთ სხვადასხვა ობიექტი. იმის მიხედვით, თუ რომელი ობიექტია არჩეული, ფანჯარაში შესაბამისად იცვლება თვისებები.



სურ. 4. 6. ფორმის თვისებების ფანჯარა

თვისებების ფანჯარას აქვს ხუთი შეცვლის საშუალება:

- **Format** (ფორმატი), ფორმატისა და სტრუქტურის შესაცვლელად;
- **Data** (მონაცემები), მონაცემთა ორგანიზაციისა და შეტანის წესისათვის;
- **Event** (პირობა), შეგიძლიათ რომელიმე პირობის შესრულების დაპროგრამება, (მაგალითად თავის მანქენებით დაწკაპუნება, ან მონაცემის შეცვლა);
- **Other** (სხვადასხვა), მომხმარებლის კარნახისათვის;
- **All** (ყველა), თუ არ იცით სად არის მოცემული თვისება და გინდათ ნახოთ თვისებების სია.

მრავალი თვისება გადაეცემა ერთმანეთთან დაკავშირებული ცხრილებიდან და მოთხოვნებიდან. ზოგი თვისება შეგიძლიათ დააყენოთ დაფორმატების იარაღების ჯგუფის ღილაკებით, ზოგი – ელემენტების ჯგუფის ღილაკებით, ზოგისთვის კი თვისების დაყენება არ შეიძლება.

მოცემულ თვისებაზე სრულყოფილი ინფორმაციის მისაღებად ისარგებლეთ კარნახით, რისთვისაც კურსორი დააყენეთ მის ველზე და დაარტყით <F1> კლავიშს.

თვისების შესაცვლელად:

დააყენეთ კურსორი ობიექტზე (ელემენტზე, ფორმა/უწყისზე) და დააწკაპუნეთ თავის მარჯვენა კლავიშზე. გამოვა მენიუ, საიდანაც აირჩიეთ **Properties** (თვისებები);

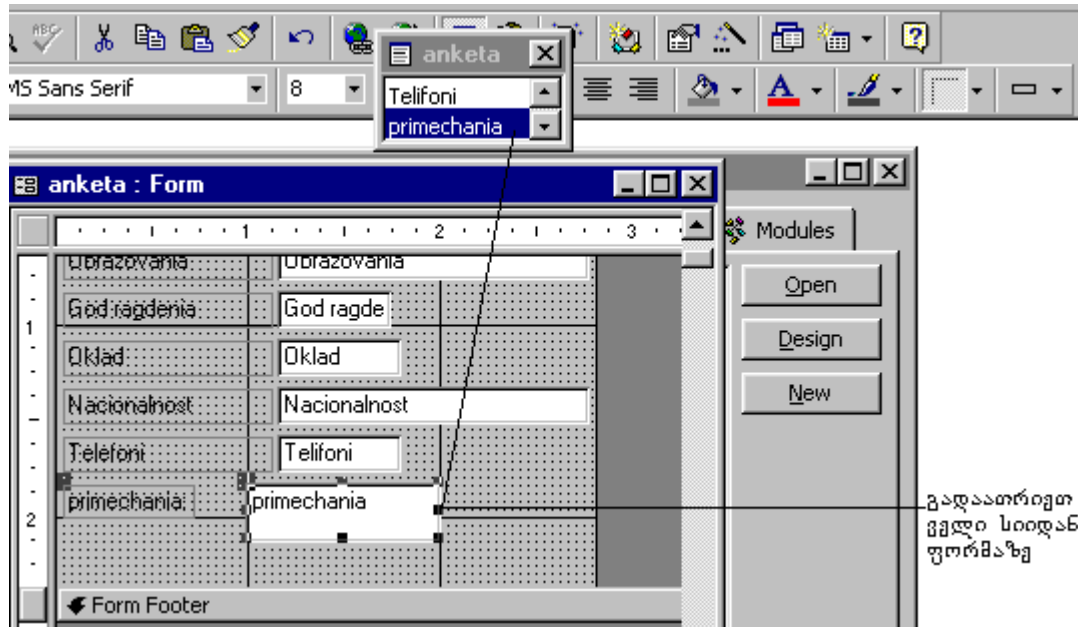
1. დააწკაპუნეთ შესაცვლელი თვისების უჯრაზე;
2. ქვემოთ ჩამოთვლილიდან შეასრულეთ ერთ-ერთი.

შეიტანეთ ამ თვისების ახალი მნიშვნელობა, აირჩიეთ იგი სიიდან (თუ არის სია), იმისათვის რომ ნახოთ შესაძლო პარამეტრები, დააწკაპუნეთ ორჯერ, დაარტყით ღილაკს **Build** (ამგები), რომელიც მოთავსებულია უჯრიდან მარჯვნივ, ან სხვა ადგილიდან გააკეთეთ ასლი.

თუ შემთხვევით წაშალეთ ელემენტი ან თუ ფორმას (უწყისს) ქმნით, შეიძლება დაგჭირდეთ მართვის დამატებითი ელემენტები. მათი დამატების უმარტივესი ხერხია ველების სიით სარგებლობა. როდესაც ველს გადაათრევთ ფორმაზე, Access ავტომატურად ქმნის მის შესაბამის მართვის ელემენტს. თუ სხვა ტიპის მართვის ელემენტის მიღება გსურთ, გამოიყენეთ ელემენტების ჯგუფი, ან აირჩიეთ მენიუდან ბრძანება **Format→Change To** (ფორმატი→ელემენტის გარდაქმნა). იმისათვის რომ გამოიძახოთ ან დახუროთ ველების სია, კონსტრუქტორის იარაღების ჯგუფში დააწკაპუნეთ შესაბამის ღილაკზე ან აირჩიეთ მენიუდან ბრძანება **View→Field List** (ნახვა→ველების სია).

მართვის ახალი ელემენტის შესაქმნელად ელემენტების ჯგუფიდან აირჩიეთ და მოათავსეთ იგი ფორმაზე. გარდა ამისა, შეგიძლიათ ჯერ ჯგუფიდან აირჩიოთ საჭირო ელემენტი, შემდეგ კი სახელების სიიდან - სახელი. იხ. სურათი 4. 7. არსებობს მართვის ელემენტების სამი კატეგორია:

- ❑ დაკავშირებული ელემენტები, რომლებიც მიბმულია ცხრილის ან მოთხოვნის რომელიმე ველთან;
- ❑ თავისუფალი ელემენტები, რომლებიც ცხრილში ან მოთხოვნაში არ არიან მიბმული რომელიმე ველთან;
- ❑ გამოსათვლელი ელემენტები, რომლებიც თავისუფალნი არიან, მაგრამ გამოიყენებიან მოცემულ ველში, გამოთვლების შესასრულებლად. გამოთვლების შედეგები არ ინახება ცხრილში ან მოთხოვნაში.



სურ. 4. 7. ველების სიიდან ფორმაზე ველის გადათრევა

ფორმაზე მართვის ელემენტის დასამატებლად შესრულებული შემდეგი:

1. გამოიძახეთ ელემენტების ჯგუფი, რისთვისაც მენიუდან აირჩიეთ ბრძანება View→ToolBox (ნახვა→ელემენტების ჯგუფი);

2. დააწკაპუნეთ ამ ჯგუფში თქვენთვის საჭირო ელემენტზე;

3. თავისუფალი ან გამოსათვლელი ელემენტები თავიდან გადაათრეოთ ფორმაზე და მოათავსეთ თქვენთვის საჭირო ადგილას. იმისათვის რომ შეიქმნას ნაგულისხმევი ელემენტის ზომები დააწკაპუნეთ მასზე. თუ გსურთ ზომების შეცვლა, ხელახლა დააწკაპუნეთ იმავე ელემენტზე და როცა მაჩვენებელი მიიღებს ორმხრივი ისრის ფორმას დაარეგულირეთ ზომა;

4. დაკავშირებული ელემენტებისათვის გამოიძახეთ ველების სია. ამისათვის მენიუდან აირჩიეთ ბრძანება View→Field List (ნახვა→ველების სია), შემდეგ აირჩიეთ თქვენთვის საჭირო ველი, გადაათრეოთ ფორმაზე და მოათავსეთ საჭირო ადგილას.

გამოსათვლელი ელემენტის შესაქმნელად:

1. შექმენით ველის ტიპის თავისუფალი ელემენტი (არ იყოს დაკავშირებული ცხრილის ველებთან);

2. დაბეჭდეთ ელემენტში გამოსახულება ან დაწერეთ გამოსახულება ამ ელემენტისათვის მონაცემების თვისების მნიშვნელობის სახით, რომელსაც ავსხნით ქვემოთ. მაგ. = $[ფასი]*[რაოდენობა]$ სადაც ფასი და რაოდენობა შესაბამისი ცხრილის ველებია (ან ელემენტების სახელებია). მიაქციეთ ყურადღება იმას, რომ ველების სახელები კვადრატულ ფრჩხილებშია მოთავსებული.

4. 5. ფორმის კონსტრუქტორის ელემენტების ჯგუფის გამოყენება

კონსტრუქტორის ელემენტების ჯგუფი შეიცავს ობიექტებს (როგორიცაა ველები, წარწერები და სიები), რომელთა მოთავსებაც შეგიძლიათ როგორც ფორმაზე, ისე უწყისზე. ამ ელემენტებიდან ფორმაში ყველა გამოიყენება, უწყისში კი – მხოლოდ ზოგიერთი. ისეთი ელემენტები, როგორიცაა სია და გადამრთველი, რა თქმა უნდა, უწყისში არ გამოიყენება, იხ. სურ. 4. 8.

შეგიძლიათ ელემენტების ეს ჯგუფი გადაადგილოთ ნებისმიერ ადგილას და მისი ზომები ჩვეულებრივ შეცვალოთ. ეკრანზე ამ ჯგუფის გამოსატანად უნდა გახსნათ ფორმის კონსტრუქტორი.

ფორმასა ან უწყისში ახალი ელემენტის შესაქმნელად, ელემენტების ჯგუფიდან სასურველზე დააყენეთ მაჩვენებელი და ერთჯერ დააწკაპუნეთ. მაჩვენებელი მიიღებს ჯვრის ფორმას და გამოჩნდება ამ ელემენტის ნიშანი (ნახატი).

ელემენტების მოკლე აღწერა

- *ობიექტის მონიშვნა*, მონიშნავს ობიექტს.
- *ოსტატი*, ჩართავს და გამორთავს ელემენტების ოსტატს (ელემენტებისათვის კარნახს). სასურველია, რომ იყოს ჩართული.

სურ. 4. 8. ელემენტები, რომლებიც გამოიყენება ფორმის შესაქმნელად

□ *წარწერა*, ქმნის ტექსტურ ელემენტს. ეს ელემენტი გამოიყენება ფორმაში მომხმარებლის ტექსტის ჩასაწერად. ამ ელემენტის დაყენების შემდეგ, იწყებთ ფორმაში ტექსტის ჩაწერას. თუ ტექსტი დიდია, მისი ზომაც იზრდება ისე, რომ ტექსტი სრულად გამოჩნდეს. გაიტანეთ მაჩვენებელი ტექსტური ველის გარეთ და დააწკაპუნეთ. თუ გინდათ ჩაწერილი ტექსტის ან ფორმატის შეცვლა, დააყენეთ მაჩვენებელი ტექსტურ ველზე და დააწკაპუნეთ.

□ *ველი (ტექსტური ველი)*, გამოიყენება ცხრილის ველებიდან მონაცემების გამოსატანად (აქვს სახელი და ტექსტური ველი). ეს პარამეტრი ხშირად გამოიყენება ვინაიდან შესაძლებელია ველების სიიდან ველის გადმოთრევა, გამონაკლისია ის შემთხვევა, როდესაც იქმნება გამოსათვლელი ველი. გამოსახულების შესაქმნელად ჯერ დაბეჭდეთ ტოლობის ნიშანი, შემდეგ კი - გამოსახულება. გარდა ამისა შეგიძლიათ გამოიძახოთ ჯგუფის თვისებები, მონახოთ თვისება „მონაცემები“ და ისარგებლოთ ღილაკით Buildig (ამგები).

ტექსტური ველების უმრავლესობისათვის შეიძლება “Yes“-ზე დააყენოთ თვისების მნიშვნელობა „გაფართოება და შეკუმშვა“, დააყენოთ ვერტიკალური ლიფტი და Enter (შეტანა) კლავიშზე ხელის დაჭერით შეცვალოთ კურსორის ქცევა.

□ *გადამრთველების ჯგუფი.* ამ ელემენტის საშუალებით შეგიძლიათ შექმნათ ერთმანეთზე დამოკიდებული კლავიშების ჯგუფი, საიდანაც შესაძლებელია მხოლოდ ერთი ვარიანტის არჩევა. გადამრთველების ჯგუფის ფორმაზე მოვათავსებთ, შევძლებთ ოსტატის გამოძახებას, რომელიც გეხმარებათ ამ ჯგუფის განსაზღვრაში და საჭირო პარამეტრის დაყენებაში. ამ ველის თითოეული რიცხვი წარმოადგენს გარკვეულ არჩევას (მაგალითად, 1=მებრძოლი, 2=კომედია, 3=საბავშო ფილმი და ა. შ.).

□ *გამომრთველი* ქმნის ჩართვა/გამორთვის ღილაკს. ეს ღილაკი შეიძლება იყოს გადამრთველის ნაწილი ან დამოუკიდებელი. თუ თქვენ გსურთ გამომრთველის გამოყენება, უნდა დარწმუნდეთ, რომ თვისება “ხელმოწერა” დაყენებული გაქვთ ისე რომ შეგატყობინოთ ღილაკის ფუნქცია.

□ *გადამრთველი*, ჩვეულებრივ, გამოიყენება ჯგუფში. თუ თქვენ გამოიყენებთ გადამრთველების ჯგუფის შექმნის იარაღი, მაშინ გადამრთველი ავტომატურად იქნება მასში, ხოლო თუ ჯგუფს იგი უნდა დაამატოთ, მაშინ საჭიროა ჯერ შექმნათ გადამრთველი, შემდეგ აირჩიოთ ელემენტი გადამრთველი და დააწკაპუნოთ ჯგუფზე (ჯგუფი მონიშნება). გადამრთველთან დაკავშირებული წარწერა შეცვალეთ ისე, რომ ასახავდეს მის მნიშვნელობას. გადამრთველის შესაცვლელი თვისება არის რიცხვი, რომლის მნიშვნელობა ასახავს ჯგუფით განსაზღვრული ველის შესაძლო ვარიანტებიდან ერთ-ერთს.

□ *ალმის მეშვეობით შესაძლებელია* ფანჯრების შემოწმება. იგი ქმნის Yes/No (დიახ/არა) ტიპის მართვის ლოგიკურ ელემენტებს. ალმის შესაქმნელად შეგიძლიათ ველების სიიდან ლოგიკური ველის გადათრევა. ალამი ნიშნავს დიახ-ს, თუ ალამი არ არის - არა-ს. თუ ახალი ჩანაწერი ავტომატურად უნდა იძლეოდეს პასუხს “კი” ან “არა”, დაბეჭდეთ ეს პასუხი ველის ნაგულისხმევ მნიშვნელობაში.

□ *ველი სიით* - ქმნის წინასწარ განსაზღვრული პარამეტრების სიას, რომელიც იხსნება და მომხმარებელს აძლევს მონაცემების შეტანის საშუალებას. ველი სიით და სია ხშირად გამოიყენება, როდესაც გჭირდებათ მონაცემების დათვალიერება სხვა ცხრილიდან ან მოთხოვნიდან. მაგ. ცხრილში “დაკვეთები” შეგიძლიათ შესთავაზოთ ამ დაკვეთაზე პასუხისმგებელი მუშაკის იდენტიფიცირების შესაძლებლობა. იმის ნაცვლად რომ მუშაკის სახელი მოლიანად დაბეჭდოთ, შეგიძლიათ შექმნათ ველი სიით მუშაკის იდენტიფიკაციური ნომრის საფუძველზე.

როდესაც თქვენ მიიღებთ გადაწყვეტილებას რომ გამოიყენოთ ველი სიით, მგებავი შეგეკითხებათ, რომელი ცხრილით უნდა ისარგებლოთ და რომელი ველები გინდათ რომ ნახოთ. პირველი ველი, რომელსაც აირჩევთ უნდა შეიცავდეს იმ მნიშვნელობებს, რომლის დანახვაც გინდათ ველი სიით დახურვის შემდეგ.

შეიძლება გამოიყენოთ ნებისმიერი სხვა ველი, მუშაკის იდენტიფიცირებისათვის. მაგალითად, შეიძლება გამოიყენოთ მუშაკის გვარი, სახელი და მისი საიდენტიფიკაციო ნომერი, რომელიც გამოიყენება ველში სიით.

თუ ველი სიით არ მუშაობს, უნდა ნახოთ შემდეგი თვისებების მნიშვნელობები:

□ *მონაცემები*, იმ ცხრილების ველები, სადაც ინახება პასუხები;

□ *სტრიქონის წყარო*, ჩასმისათვის გამოყენებული ცხრილის ან მოთხოვნის სახელი. შეგიძლიათ გამოიყენოთ კლავიში Buildig (ამგები) და გამოიძახოთ მოთხოვნების ამგები, საიდანაც აირჩევთ იმ ველებსა და ელემენტების დახარისხების მიმდევრობას, რომლებიც სიის გახსნის დროს გამოჩნდებიან.

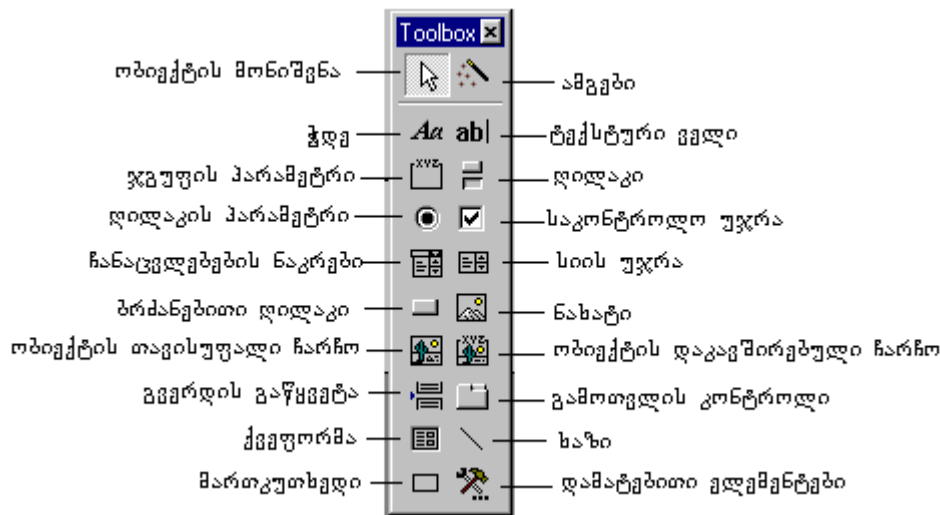
□ *სვეტების რაოდენობა*, სიის ფორმირებისათვის გამოყენებული ჩანაწერების წყაროდან;

□ *სვეტების სიგანე*, სიაში თითოეული სვეტის სიგანე. ნულოვანი სიგანის სვეტი, სიაში არ გამოჩნდება;

□ *დამაკავშირებელი სვეტის სიგანე*, ჩანაწერის წყაროდან ის სვეტი, რომელიც მოთავსებული იქნება ფორმების ველში;

□ *სიის სიგანე*, სიის საერთო სიგანე;

□ *სიის შეზღუდვა*, ეს თვისება განსაზღვრავს, გინდათ თუ არა, რომ შეზღუდოთ სიაში ჩამოთვლილ ველში შეტანილი მონაცემების მნიშვნელობები და, რომ მომხმარებელს შეეძლოს მონაცემებს თვითონ შეიტანს.



სურ. 4. 9 ფორმის კონსტრუქტორის ელემენტების ჯგუფი

სია, წინასწარ განსაზღვრული ელემენტებით ქმნის სიას, რომელიც იხსნება, მაგრამ სიის შემცველი ველისაგან განსხვავებით, მომხმარებელს არ აძლევს საშუალებას, რომ თვითონ შეიტანოს მონაცემები. გარდა ამისა, სხვა ველებთან მუშაობის დროს ჩანს მთელი ცხრილი და სტრიქონები, რის გამოც სია იკავებს დიდ ადგილს და იშვიათად გამოიყენება.

ლილაკი, ქმნის ლილაკს, რომელსაც შეუძლია მიკრობრძანების შესრულებაზე გაშვება, ან Access-ის პროცედურის გამოძახება. როდესაც ამ ელემენტს მოათავსებთ ფორმაზე, გაიშვება ფართო შესაძლებლობების პარამეტრების მქონე ოსტატი, რომლითაც შეგიძლიათ სხვა ჩანაწერზე გადასვლა, ჩანაწერების დამატება ან მოშლა, ფორმის გახსნა და უწყისის შექმნა. Access ავტომატურად ქმნის Visual Basic-ის პროცედურას, რომელიც ლილაკთან ერთად მუშაობს. შეგიძლიათ გადააკეთოთ პროცედურა რედაქტირებით და დააკავშიროთ თვისებასთან ამ ლილაკზე დაწკაპუნებით.

ნახატი, ქმნის ნახატის გამოსატან ჩარჩოს. ამ ელემენტით სარგებლობისას Access გამოიძახებს დიალოგურ ფანჯარას „გამოსახულების ჩასმა“, რომლითაც მონახავთ გამოსახულებებიან ფაილს. ზოგჯერ საჭიროა შეცვალოთ თვისება „ზომების დაყენება“, რომ სურათი სწორად აისახოს.

ობიექტის თავისუფალი ჩარჩო, ქმნის ჩარჩოს იმ გამოსახულების, გრაფიკის და OLE – ობიექტის გამოსატანად, რომლებიც მონაცემთა ბაზაში არ არიან.

გვერდის გაწვევა, ფორმაში ან უწყისში გვერდის ბოლოს აყენებს ნიშანს, რომლის ქვემოთ არსებული მონაცემები გამოიტანება შემდეგ გვერდზე.

ჩანაცვლებების ერთობლიობა, ქმნის დიალოგური ფანჯრის ეფექტს ფორმაზე ჩასადებებით. მოათავსეთ ეს ელემენტი სულ თავში, შემდეგ კი შექმენით მართვის ელემენტი.

დაქვემდებარებული ფორმა/უწყისი (ქვეფორმა), ეს არის ერთმანეთთან დაკავშირებული ცხრილების, ერთი მრავალთან კავშირის ტიპის თანაფარდობის, დემონსტრირება, მაგალითად, იმისათვის რომ ნახოთ მყიდველის მიერ დაქირავებული ფილმები. როდესაც ერთი ფორმის აგება ხდება მეორე ფორმაზე, გამოიყენეთ ელემენტი მთავარი ფორმის ქვეფორმა. ფორმების ოსტატის გამოყენებით დაზოგავთ დროს როგორც მთავარი ისე ქვეფორმის შესადგენად. ქვეფორმის მთავარი თვისებაა „ობიექტის წყარო“ (ქვეფორმის სახელი) და ქვეფორმები (ქვეფორმაზე ველების სახელები), რომელთა მნიშვნელობები ემთხვევა მთავარი ფორმის ძირითად ველებს, რომლებიც იდენტიფიცირებულია თვისებასთან „ძირითადი ველები“.

წრფე, ხაზავს ხაზს. ეს ელემენტი გამოიყენება ხაზისა და ჩარჩოსათვის მართკუთხედის დასახაზად, რომლებიც ყურადღების მისაქცევად აჯგუფებენ ელემენტებს.

მართკუთხედი, ხაზავს მართკუთხედს. თუ გინდათ ელემენტების დაჯგუფება გამოიყენეთ **ელემენტი**, გადამრთველების ჯგუფი.

დამატებითი ელემენტები, გამოიძახებს რეგისტრირებულ მართვის ელემენტებს, რომლებსაც მართვის Activex ელემენტებსაც უწოდებენ.

V თ ა ვ ი

მონაცემთა ბაზის მოთხოვნების შექმნა

თქვენ უკვე იცით მონაცემთა ბაზაში ინფორმაციის შეტანა, ახლა ვისწავლოთ ბაზიდან ინფორმაციის მიღება. თანამედროვე მონაცემთა ბაზის ერთ-ერთი სასარგებლო ფუნქცია არის მოთხოვნები, რომლის მეშვეობითაც შესაძლებელია კითხვის დასმა. შემდეგ კი ამ კითხვების პასუხების ეკრანზე გამოიტანა. ახლა გავეცნოთ ამ ფუნქციას.

5. 1. მოთხოვნა

მოთხოვნა არის გამოსახულება, რომელიც შეატყობინებს Access-ს, თუ რა ტიპის ინფორმაციის მიღება გვსურს ერთი ან რამდენიმე ცხრილიდან. მოთხოვნის მეშვეობით შესაძლებელია ცხრილის მონაცემებზე სხვადასხვა მოქმედების შესრულება და შედეგების შეჯამება.

მოთხოვნებს შეუძლია შემდეგი ფუნქციების შესრულება:

- გარკვეულ რაიონში მცხოვრებ თანამშრომელთა სიის შექმნა;
- ერთ უწყისში დამკვეთების სახელების, მათი ანკეტები მონაცემებისა და ნაყიდ საქონელზე ინფორმაციის შეტანა;
- მოცემული ფილმის იჯარით აღების პერიოდულობის განსაზღვრა;
- ფილმების კატეგორიების მიხედვით ჯამური ღირებულების განსაზღვრა;
- მონაცემთა ბაზიდან იმ მომხმარებლების ამოღება, რომელთაც მიმდინარე წელს არაფერი შეუძენიათ;
- მონაცემთა ბაზის არქივში მყიდველების შესახებ მონაცემების დამატება.

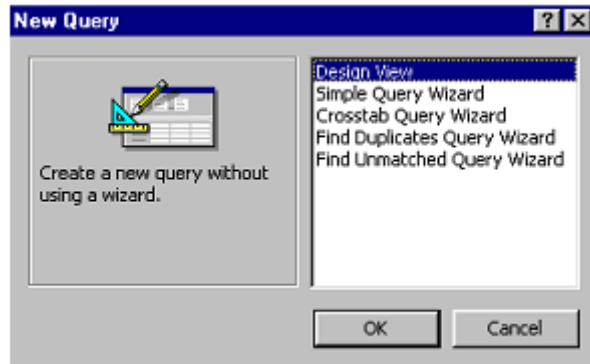
Access-ის საშუალებით შეგიძლიათ შექმნათ შემდეგი ტიპის მოთხოვნები:

- *არჩევის მოთხოვნა*, გამოიყენება მონაცემთა ბაზის ცხრილებიდან მოთხოვნების შესაბამისი ინფორმაციის მისაღებად;
- *აქტიური მოთხოვნები*, გამოიყენება ჩანაწერებზე იმ მოქმედებების შესასრულებლად, რომლებიც აკმაყოფილებენ მოთხოვნებში მითითებულ პირობებს, ამ ტიპის მოთხოვნებით შეგიძლიათ მონაცემების შეცვლა და გადაადგილება, ახალი ცხრილების შექმნა და ცხრილში ჩანაწერის წაშლა;
- *შემაჯამებელი მოთხოვნები* გამოიყენება შუალედური გამოთვლებისათვის, ჯამური მონაცემებისათვის და სხვა ჯამური ანგარიშებისათვის;
- *ჯვარედინა მოთხოვნები* გამოიყენება მოთხოვნებში მითითებული კრიტერიუმების საფუძველზე, ცხრილურ ფორმებში მონაცემების შესაჯამებლად;
- *გამაერთიანებელი მოთხოვნები* გამოიყენება სხვადასხვა ცხრილისათვის საერთო ველების ჩანაწერების გასაერთიანებლად;
- *გასახველელი მოთხოვნები* ენის საფუძველზე, მონაცემთა ბაზაზე ბრძანებების გადასაგზავნად;
- *მონაცემთა განსზღვრის მოთხოვნები*, გამოიყენება SQL-ის გამოსახულების დახმარებით, Access-ის მონაცემთა ბაზაზე მოქმედებების შესასრულებლად.

თითოეული მონაცემის ტიპისათვის შეგიძლიათ მიუთითოთ მოთხოვნის პარამეტრები. Access მოთხოვნების (ფილტრის) შედეგებს მოათავსებს Dinaset-ში (დინამიურ ცხრილებში). მას ცხრილის ფორმა აქვს, მაგრამ დინამიურია და მისი დანიშნულებაა ერთი ან რამდენიმე ცხრილის მონაცემების დათვალიერება. დინამიურ ცხრილში შეგიძლიათ მონაცემების შეტანა და მათი რედაქტირება. ამის შემდეგ Access ავტომატურად განაახლებს დაკავშირებული ცხრილების მონაცემებს. დინამიური ცხრილი ჰგავს საშედეგო ცხრილებს, რომლებიც, თავის მხრივ, ჩვეულებრივი ცხრილების მსგავსია, მაგრამ მათი მოდიფიცირება არ შეიძლება. შედეგის ცხრილების მაგალითია ჯვარედინა და შემაჯამებელი მოთხოვნები.

5. 2. ახალი მოთხოვნის შექმნა

ახალი მოთხოვნის შესაქმნელად იარაღების სტანდარტულ სტრიქონში მონახეთ New Object (ახალი ობიექტი) ღილაკი და დააწკაპუნეთ მასზე, ელემენტების ჩამონათვალიდან აირჩიეთ Query (მოთხოვნა). იგივე შეგიძლიათ შეასრულოთ თუ მონაცემთა ბაზის ფანჯრიდან აირჩევთ ჩანაცვლებას Queries (მოთხოვნები) და დააწკაპუნებთ New (ახალი) ღილაკზე. გამოვა დიალოგური ფანჯარა New Query (ახალი მოთხოვნა), იხ. სურ. 5.1.



სურ. 5. 1. ახალი მოთხოვნის შექმნის დიალოგური ფანჯარა

Access გთავაზობთ მოთხოვნების შექმნის რამდენიმე ხერხს. ჯერ მოიფიქრეთ მოთხოვნის სტრუქტურა და გადწყვეტილების მიღებისას გაითვალისწინეთ შემდეგი:

- ☐ რომელი ცხრილი (ცხრილები) შეიცავს საჭირო ინფორმაციას;
- ☐ ცხრილების ურთიერთდამოკიდებულება (სწორად არის თუ არა გასაღები ველები მითითებული);
- ☐ რა ტიპის მოთხოვნას ქმნით;
- ☐ პირობები და კრიტერიუმები თქვენთვის საჭირო ჩანაწერებისათვის;
- ☐ არის თუ არა საჭირო გაანგარიშება;
- ☐ დახარისხების მიმდევრობა;
- ☐ მოთხოვნის სახელი.

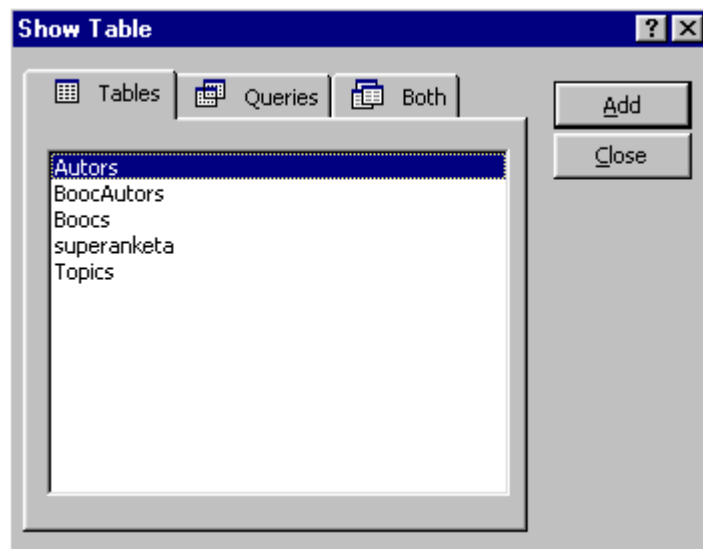
დიალოგური ფანჯრის გამოყენება მოთხოვნის შესაქმნელად

დიალოგურ ფანჯარაში New Query (ახალი მოთხოვნა), მოთხოვნების შესაქმნელად გთავაზობთ ხუთნაირ არჩევის მეთოდს.

- ☐ *კონსტრუქტორი.* ეს პუნქტი გამოიძახებს მოთხოვნის ცარიელ ბლანკს და არჩეულ ცხრილს, რომლის საფუძველზეც იქმნება მოთხოვნა;
- ☐ *მარტივი მოთხოვნა.* იგი ქმნის მოთხოვნას არჩეული ველებისათვის;
- ☐ *ჯვარედინი მოთხოვნა.* მონაცემებს შეაჯამებს ცხრილურ ფორმაში;
- ☐ *გამეორებითი ჩანაწერი.* აღმოაჩენს ცხრილში გამეორებით ჩანაწერებს;
- ☐ *ჩანაწერები დაქვემდებარების გარეშე.* აღმოაჩენს ისეთ ჩანაწერს, რომლებსაც დაკავშირებულ ცხრილში არა აქვს შესაბამისი ჩანაწერი.

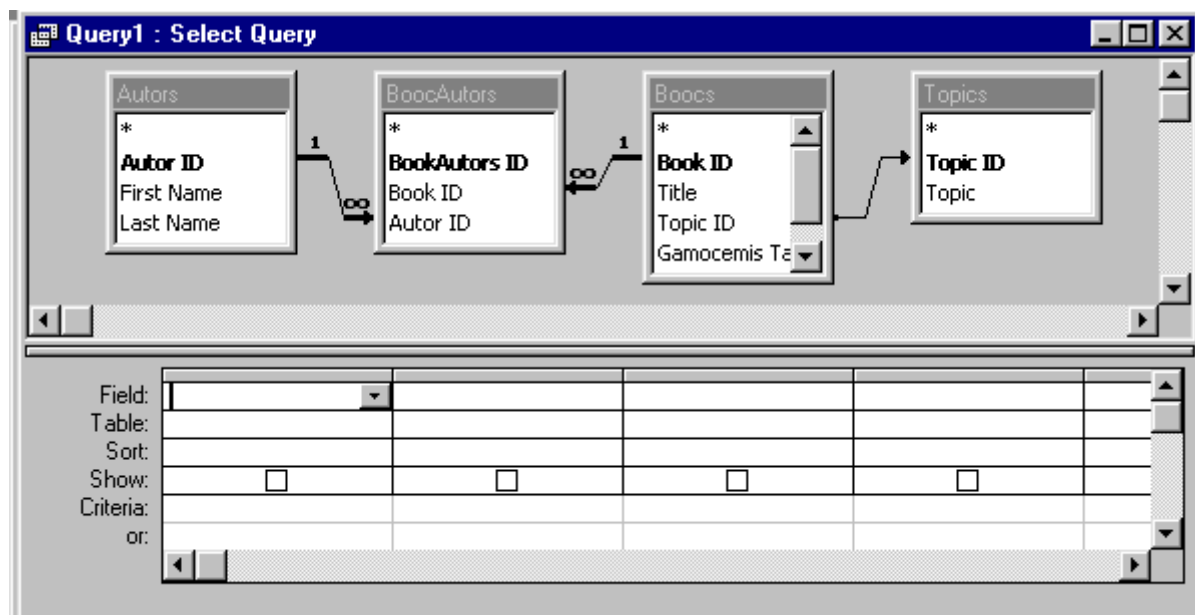
ყოველი ტიპის მოთხოვნა აიგება ოსტატით, რომელიც ითხოვს თითოეული ტიპის მოთხოვნის ასაგებად საჭირო ინფორმაციას. ყოველთვის უნდა მიუთითოთ ის ცხრილი ან მოთხოვნა, რომლის საფუძველზეც აიგება ახალი მოთხოვნა.

ცხრილის ან მოთხოვნის დასამატებლად სიის ჩამონათვალიდან დააწკაპუნეთ მის სახელზე, ან მონიშნეთ ცხრილი/მოთხოვნა და დაარტყით Add (დაამატება) ღილაკს, გაჩნდება ახალი ცხრილი. Access ავტომატურად პოულობს ცხრილებს შორის არსებულ კავშირს და გამოსახავს, დაკავშირებულ ველებს შორის ხაზებით. იხ. სურ. 5. 2 და 5. 3.



სურ. 5. 2. ცხრილის დამატების დიალოგური ფანჯარა

☒ მოთხოვნის აგების დაწყებამდე, წინასწარ მონიშნეთ ცხრილი ან მოთხოვნა, რომლის საფუძველზეც აგებთ ახალ მოთხოვნას. შემდეგ გამოიყენეთ ჩანაცვლებაში – Queries (მოთხოვნები) ღილაკი და დააწკაპუნეთ New (ახალი) ღილაკზე, Access-ი მოათავსებს მონიშნული ცხრილიდან ველების სიას. ყველა სხვა შემთხვევაში გამოვა დიალოგური ფანჯარა Add Table (ცხრილის დამატება), რომელიც მოგთხოვთ დასამატებელი ცხრილის სახელს.



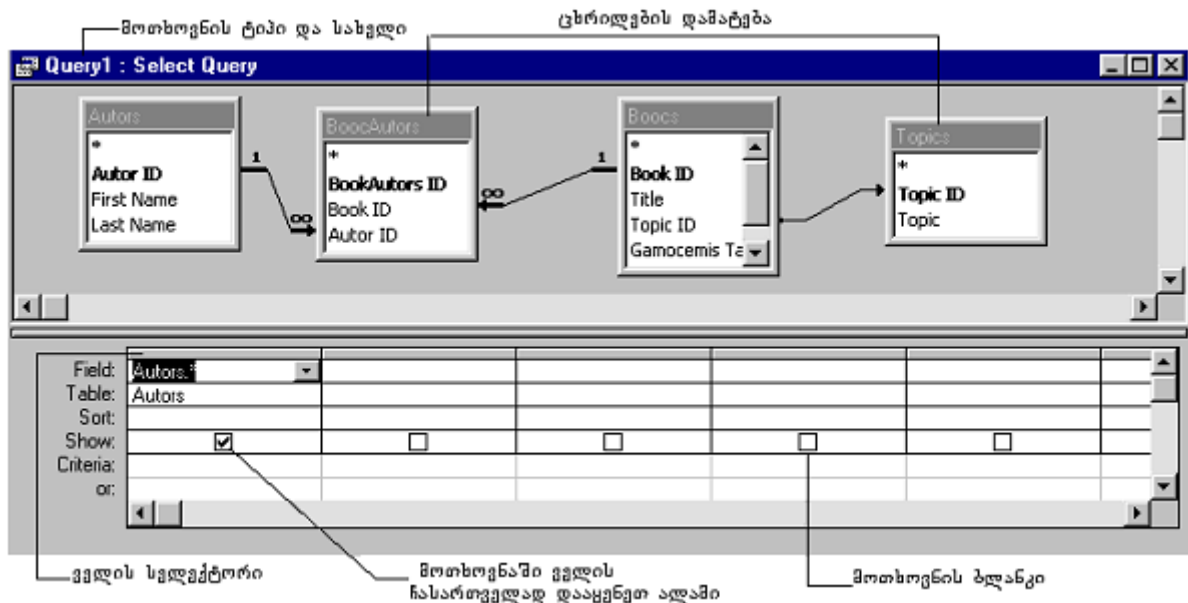
სურ. 5. 3. კონსტრუქტორის ფანჯრის ზემოთ დამატებული ცხრილები

Access ავტომატურად პოულობს ველებს შორის კავშირს და დაკავშირებულ გასაღებიან ველებს შორის ავლებს ხაზებს (ერთი და იმავე ტიპისა და სახელის მქონე ველები). ამ ხაზებს ეწოდება დამაკავშირებელი ხაზები. შეგიძლიათ თქვენ თვითონ შექმნათ დამაკავშირებელი ხაზები გასაღები ველის ერთი ცხრილიდან მეორეში გადათრევით.

კონსტრუქტორის ფანჯრის ქვედა ნახევარი შეიცავს ბლანკს Queryby Example (მოთხოვნა ნიმუშით), რომელშიც თქვენ განსაზღვრავთ მოთხოვნის კრიტერიუმს. Queryby Example (მოთხოვნა ნიმუშით), საშუალებას გაძლევთ განსაზღვროთ ძებნისათვის საჭირო კრიტერიუმი.

ცხრილების დამატების შემდეგ დააწკაპუნეთ Close (დახურვა) ღილაკზე.

ღილაკი	ღილაკის სახელი	აღწერა
	სახე	იძახებს დათვალიერების შესაძლო რეჟიმების სიას: კონსტრუქტორი, ცხრილების რეჟიმი და SQL-ის რეჟიმი.
	შენახვა	ინახავს მოთხოვნის სტრუქტურას.
	ბეჭდვა	ამობეჭდავს მოთხოვნის შედეგებს (ცხრილების რეჟიმიდან)
	წინასწარი დათვალიერება	ეკრანზე გვიჩვენებს, თუ როგორი იქნება მოთხოვნა დაბეჭდილი სახით.
	ორთოგრაფიის შემოწმება	მოთხოვნის შედეგებში ამოწმებს ორთოგრაფიას (აქტიურია მხოლოდ ცხრილების რეჟიმში)
	ამოჭრა	ამოჭრის მონიშნულ მონაცემებს ან ობიექტს და გადაწერს ბუფერზე
	ასლი	მონიშნული მონაცემების ან ობიექტის ასლს გადაწერს ბუფერზე
	ჩასმა	ბუფერიდან გადმოწერა კურსორით მითითებულ ადგილას
	აღდგენა	გააუქმებს ბოლო ცვლილებას (აღადგენს წინა ბრძანებას)
	მოთხოვნის ტიპი	გამოიძახებს მოთხოვნების ტიპების სიას: არჩევა, ჯვარედინი, ცხრილების შექმნა, განახლება, დამატება, წაშლა.
	გაშვება	შესრულებაზე გაუშვებს მოთხოვნას
	ცხრილის დამატება	გამოიძახებს დიალოგურ ფანჯარას – ცხრილის დამატება, სადაც მითითებულია ის ცხრილები და მოთხოვნები, რომლებიც გამოიყენება მოთხოვნის შექმნისას.
	ჯგუფური მოქმედებები	გამოიძახებს ან დახურავს მოთხოვნის ბლანკზე შესაჯამებელი მონაცემების სტრიქონებს
	მნიშვნელობათა ნაკრები	იძლევა მნიშვნელობათა არჩევისათვის სიას (პირველი 5, 25, 100, 5% ან 25%)
	თვისებები	გამოიძახებს თვისებების ფანჯარას
	ააგეთ	გამოიძახებს ამგებს ან ოსტატს
	მონაცემთა ბაზის ფანჯარა	გამოიძახებს მონაცემთა ბაზის ფანჯარას
	ახალი ობიექტი	გამოიძახებს სიას: ფორმები, მოთხოვნები და უწყისები მონაცემთა ბაზის ახალი ობიექტების შესაქმნელად.
	დამხმარე	გამოიძახებს საოფისე დახმარებას, რომელიც ეხმარება მომხმარებელს კარნახის ძებნაში ან იძლევა რჩევას, შეტყობინების საშუალებით.



სურ. 5. 4. მოთხოვნის ბლანკზე მოთხოვნის განსაზღვრა, ველების დამატება და მოთხოვნის პარამეტრის დაყენება

5. 3. მოთხოვნის ფორმულარის აღწერა

მას შემდეგ, რაც შექმნილია ახალი მოთხოვნა ან მონაცემთა ბაზის ფანჯარაში უკვე არსებული მოთხოვნისათვის დააწკაპუნეთ Design (კონსტრუქტორი) ლილაკზე, გაიხსნება მოთხოვნის ფანჯარა კონსტრუქტორის რეჟიმში. ეს ფანჯარა დაყოფილია ორ ძირითად ნაწილად, იხ. სურ. 5. 4. ფანჯრის ზედა ნაწილი შეიცავს თითოეული ცხრილის იმ ველების სიას, რომლებიც გამოიყენება მოთხოვნის ასაგებად.

ქვედა ნაწილი შეიცავს მოთხოვნის ფორმულარის ნიმუშს – Queryby Example (QBE), სადაც შეიძლება მოთხოვნის განსაზღვრა. ბლანკის თითოეული სვეტი თავის მხრივ არის ველი. თითოეული ველისათვის შეიძლება მოთხოვნის ისეთი პარამეტრების განსაზღვრა, როგორიცაა არჩევის კრიტერიუმი და დახარისხების ტიპი.

ველების არჩევა

მოთხოვნის ფორმულარის ნიმუშში (QBE) შედგება სტრიქონებისა და სვეტებისაგან. თითოეული სვეტი წარმოადგენს მოთხოვნაში გამოყენებულ ერთ-ერთ ველს. იმისათვის რომ ფორმულარში დაამატოთ ველი, ორჯერ დააწკაპუნეთ ველების სიაში ველის სახელზე, ან გადმოათრეით ველის სახელი. Access შეიტანს ველის სახელს და სტრიქონში Show (ეკრანზე გამოტანა) დააყენებს ალამს. თუ მოცემულ ველს იყენებთ მონაცემების დასახარისხებლად, მაგრამ მისი გამოჩენა დინამიურ ცხრილში არ არის სასურველი. მაშინ მოხსენით ახლახან თქვენს მიერ დაყენებული ალამი. ცხრილში ყველა ველის ერთდროულად მონიშვნისათვის, ორჯერ დააწკაპუნეთ ცხრილის სახელზე და მონიშნული არე გადაათრეით QBE-ს ფორმულარზე. ველის წასაშლელად დააწკაპუნეთ ველის სელექტორზე და დაარტყით Del (წაშლა) კლავიშს, ან მოათავსეთ კურსორი სვეტში ამ ველის სახელზე და აირჩიეთ მენიუდან Edit→Delete Column (რედაქტირება→სვეტის წაშლა) ბრძანება.

ფორმულარი შეიცავს ჯგუფური ოპერაციების სტრიქონს, რომელიც არ ჩანს. მის გამოსაჩენად მენიუდან აირჩიეთ View→Totals (ნახვა→ჯგუფური ოპერაციები) ბრძანება, რის შემდეგაც ფორმულარში გაჩნდება ჯგუფური ოპერაციები.

QBE-ფორმულარის სტრიქონებია: Field (ველი), Table (ცხრილი), Show (დანახვა), Criteria (კრიტერიუმები), or (ან).

5. 4. გამოსათვლელი ველების დამატება

QBE - ფორმულარში შეგიძლიათ დაამატოთ გამოსათვლელი ველებიც. გამოსათვლელი ველები დროებითი ველებია, რომლებიც იქმნება დინამიურ ცხრილებში მოთხოვნის გაშვებით, ისინი ინახავენ ცხრილის ველებზე მოქმედებების შედეგებს. გამოთვლითი ველები შეგიძლიათ გამოიყენოთ საქონელზე გადასახადების ანგარიშისათვის და შესაჯამებელი ტექსტური ველების შესადგენად.

☒ ფორმულარის სვეტების სიგანის შესაცვლელად დააყენეთ კურსორი ველის თავში, ველების გამყოფ ხაზზე, კურსორი მიიღებს ორმხრივი ისრის ფორმას და ხაზი შეგიძლიათ გადაათრიოთ სასურველ ადგილას.

გამოსათვლელი ველის შესაქმნელად, QBE – ფორმულარზე აირჩიეთ ცარიელი უჯრედი ველის სახელისათვის. შეგიძლიათ პირდაპირ ამ უჯრედში აკრიფოთ გამოსახულება. შეგიძლიათ დავალების დაწყება სახელით, შემდეგ უნდა დასვათ ორწერტილი (:), დაწეროთ არითმეტიკული გამოსახულება ან სვეტების შეკრების მოქმედება. ველის სახელები უნდა მოაქციოთ კვადრატულ ფრჩხილებში ([]), სტრიქონები უნდა ჩასვათ ბრჭყალებში, არითმეტიკული მოქმედებები და ციფრები არ საჭიროებენ ჩაწერის სპეციალურ სისტემას. ახალი ველის სახელი არ უნდა იყოს ბრჭყალებში, იხ. სურ. 5. 5.

გამოსათვლელი ველის შესაქმნელად შეასრულეთ შემდეგი მიმდევრობა:

1. გადადით QBE – ფორმულარის ცარიელ უჯრედზე;
2. შეიტანეთ ველის ახალი სახელი Field Name (ველის სახელი) უჯრედში, შემდეგ დასვით ორწერტილი;
3. უჯრედში მთლიანად შეიტანეთ გამოსახულება;
4. შეინახეთ მოთხოვნა და გაუშვით შესრულებაზე.

Field:	qvari: [last name] & ", " & [first name]	fasi: [cena]*1.1		
Table:				
Sort:				
Show:	☒	☐	☒	☐
Criteria:				
or:				

სურ. 5. 5. მოთხოვნებში გამოსათვლელი ველის საშუალებით გამოთვლების შესრულება და ორი ველის ამოღება ერთ ველში (ოპერაცია კონკატენაცია).

5. 5. ჯგუფური გამოთვლების დამატება

ზოგჯერ საჭიროა გამოთვლების ჩატარება არა თითოეულ ჩანაწერზე, არამედ ჯგუფურ ჩანაწერებზე. Access-ის საშუალებით შეგიძლიათ ჩაატაროთ საკმაოდ რთული გამოთვლები ჯგუფური ჩანაწერებისათვის. მაგალითად, შეიძლება განსაზღვროთ განყოფილებების მიხედვით საშუალო ხელფასი ან, სამუშაო საათების მაქსიმალური რაოდენობა სამუშაოს ხასიათის მიხედვით.

ჯგუფები განისაზღვრება “ჯგუფური ოპერაციების” სტრიქონის მეშვეობით. მის გამოსაძახებლად მენიუდან აირჩიეთ View→Totals (ნახვა→ჯგუფური ოპერაციები) ბრძანება.

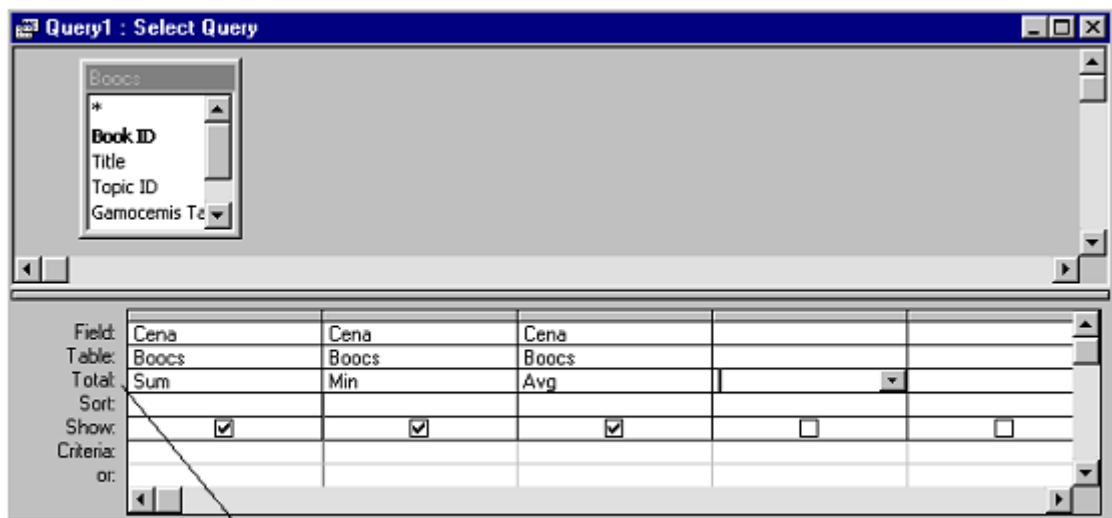
ფორმულარში ამ სტრიქონის დამატების შემდეგ Access ავტომატურად ამატებს ყოველ ველში Totals by (ჯგუფური ოპერაციები) სიტყვებს. გამოიყენეთ ეს მნიშვნელობა იმისათვის, რომ მიუთითოთ ჯგუფში შემავალი ველები, რომლებისთვისაც უნდა შესრულდეს გამოთვლები.

არჩეული ველებისათვის შესაძლებელია შემდეგი ტიპის გამოთვლების შესრულება:

1. Sum (ჯამი);
2. Avg (საშუალო არითმეტიკული);
3. Min (მინიმალური მნიშვნელობის პოვნა);
4. Max (მაქსიმალური მნიშვნელობის პოვნა);
5. Count (ველში ჩანაწერების რაოდენობის დათვლა, ნულოვანი მნიშვნელობების გარდა);
6. Std Dev (სტანდარული გადახრის დათვლა);
7. Var (დისპერსიის მნიშვნელობის გამოთვლა);
8. First (დააბრუნებს ველის თავდაპირველ მნიშვნელობას);
9. Last (დააბრუნებს ველის საბოლოო მნიშვნელობას);

ჯგუფური გამოთვლების შესასრულებლად:

1. შექმენით ან აირჩიეთ მოთხოვნა;
2. კონსტრუქტორის რეჟიმში გამოიძახეთ ფორმულარზე ჯგუფური ოპერაციების სტრიქონი, რისთვისაც მენიუდან აირჩიეთ View→Totals (ნახვა→ჯგუფური ოპერაციები) ბრძანება;
3. თითოეული ველისათვის ჯგუფური ოპერაციების უჯრაში აირჩიეთ ოპერაციის ტიპი;
4. თუ საჭიროა, შეასრულეთ შემაჯამებელი გამოთვლები ყველა ველისათვის, ჯგუფური ოპერაციების უჯრედში, არც ერთი მათგანი არ უნდა შეიცავდეს მნიშვნელობას „დაჯგუფება“;
5. თუ საჭიროა ჯგუფების მიხედვით მონაცემების ჯამის გამოთვლა, მაშინ დასაჯგუფებელი ველებისათვის უჯრედში „ჯგუფური ოპერაციები“ დააყენეთ მნიშვნელობა დაჯგუფება;
6. შეინახეთ მოთხოვნა;
7. გაუშეით შესრულებაზე ან გადადით ცხრილის რეჟიმში.



დაჯგუფების სტრიქონი

სურ. 5. 6. ჯგუფური ოპერაციების სტრიქონის გამოყენება

5. 6. არჩევის კრიტერიუმების გამოთვლა

QBE ფორმულარის კრიტერიუმის სტრიქონი საშუალებას გაძლევთ, რომ მოთხოვნის შედეგში ჩართოთ მხოლოდ ის ჩანაწერები, რომლებიც აკმაყოფილებენ განსაზღვრულ პირობებს. კრიტერიუმი - არის პირობა, რომელიც გამოიყენება ჩანაწერების არჩევისას. მოთხოვნების ამ თვისებით გაცილებით ხშირად სარგებლობენ.

ჩანაწერების არჩევა შეიძლება შემდეგი პირობების გამოყენებით:

□ **ზუსტი შესაბამისობა.** გამოიყენება შემოკლება, მაგალითად, ტექსტური სტრიქონი NJ (ნიუ-იორკის შემოკლებული აღნიშვნა) ან ფულის რაოდენობა - 1000, რომელსაც უნდა შეესაბამებოდეს ველში მოცემული მნიშვნელობა;

□ **ნაწილობრივი შესაბამისობა.** ციფრების, ასოებისა და ნიშნაკის კომბინაცია, მაგალითად, N* ან Like “###“, რომელსაც უნდა შეესაბამებოდეს ველის მნიშვნელობა, იხ. ცხრილი 5. 2;

□ **გამორიცხვის კრიტერიუმები.** მოცემული კრიტერიუმის შესაბამისი ჩანაწერის გამოსარიცხად გამოიყენება NOT (არა) ოპერატორი;

□ **თარიღით შესაბამისობა.** გამოიყენება ზუსტი თარიღი ან ოპერატორი DATE(), რომელიც გვიჩვენებს მიმდინარე თარიღს (კომპიუტერის საათით). მაგალითად, 1/12/99 DATE();

□ **შედარების ოპერატორი.** გამოიყენება ჩანაწერის რაიმე პირობასთან შესადარებლად, იხ. ცხრილი 5. 3. მაგალითად, შეიძლება შეიტანოთ < DATE(), იმისათვის რომ დაინახოთ მიმდინარე თარიღამდე არსებული ყველა ჩანაწერი;

□ **ლოგიკური მნიშვნელობა.** თანხმობის ანუ დიახ-ის ნაცვლად იხმარება მნიშვნელობები: Yes, True, On, 1, ხოლო უარყოფისას No, False, Off, 0;

□ **მრავლობითი პირობა.** გამოიყენება ლოგიკური ოპერატორები, ერთი პირობის შიგნით რამდენიმე პირობის მისათითებლად იხ. ცხრილი 5. 4. გამოიყენეთ პირობის ერთი და იგივე სტრიქონი, სხვადასხვა ველის საფუძველზე რამდენიმე პირობის დასაყენებლად, რომელთაგან ყველა უნდა შესრულდეს AND (და). გამოიყენეთ პირობის სხვადასხვა სტრიქონი, რომ მიუთითოთ კრიტერიუმები, რომელთაგან ერთი მაინც უნდა შესრულდეს OR (ან).

შეცვლის ნიშნები

ცხრილი 5. 2

ნიშანი	აღწერა
*	გამოიყენება ნებისმიერი რაოდენობის სიმბოლოების მაგივრად.
?	გამოიყენება ერთი ნებისმიერი სიმბოლოს მაგივრად.
#	ერთი ციფრის მაგივრად (მაგალითად, Like “1# # #“).
[]	შეგიძლიათ მოძებნოთ სიმბოლო, რომელიც ემთხვევა ფრჩხილებში მოთავსებული სიმბოლოებიდან ერთ-ერთს (მაგალითად, N [J Y] შეესაბამება N J ან NY).
!	გამოიყენება იმ სიმბოლოს მოსაძებნად, რომელიც ემთხვევა ფრჩხილში მოთავსებულ სიმბოლოს (მაგ. N [! J Y]).
-	გამოიყენება სიმბოლოების დიაპაზონის განსაზღვრისათვის.
Like	გამოიყენება ნებისმიერი სიმბოლოს მოსაძებნად. მაგ. Like “[A – D]*” შეგიძლიათ იმ გვარის მოსაძებნად გამოიყენოთ, რომელიც იწყება ასოებით A, B, C ან D. Access სიტყვა Like-ს თვითონ უმატებს და მისი აკრეფა არ გჭირდებათ.

შედარების ოპერატორები

ცხრილი 5. 3

ოპერატორი	აღწერა	ოპერატორი	აღწერა
>	მეტია	>=	მეტია ან ტოლია
<	ნაკლებია	< >	არ უდრის
=<	ნაკლებია ან ტოლია	=	უდრის

ლოგიკური ოპერატორები

ცხრილი 5. 4

ოპერატორი	აღწერა
AND	მოითხოვს ყველა პირობის შესრულებას
OR	მოითხოვს ერთ-ერთი ნებისმიერი პირობის შესრულებას
NOT	მოითხოვს რომ პირობა არ შესრულდეს
BETWEEN X AND Y	მოითხოვს რომ მნიშვნელობა იყოს მოცემულ დიაპაზონში

[illegible]

მოთხოვნის შედეგები ნაგულისხმევი პრინციპით დაუხარისხებელი რჩება და გამოვა იმ თანამიმდევრობით, რა თანამიმდევრობითაც იყო ნაპოვნი ჩანაწერში. უფრო ზუსტი შედეგების მისაღებად, შეგიძლიათ დახარისხების თანამიმდევრობის მითითება.

ჯგუფური მოთხოვნის ფორმულარზე დახარისხების სტრიქონი გთავაზობთ შესაძლო არჩევანს - დახარისხების შემდეგ ტიპებს:

- ☐ ზრდის მიხედვით;
- ☐ კლების მიხედვით;
- ☐ დაუხარისხებლად

შედეგების დასახარისხებლად აირჩიეთ ველი და დახარისხების ტიპი.

$\partial \mathfrak{m} \mathfrak{m} \mathfrak{b} \mathfrak{m} \mathfrak{g} \mathfrak{b} \mathfrak{o} \mathfrak{b} \quad \mathfrak{z} \mathfrak{a} \mathfrak{r} \mathfrak{a} \mathfrak{d} \mathfrak{g} \mathfrak{r} \mathfrak{g} \mathfrak{b} \mathfrak{o} \mathfrak{b} \quad \mathfrak{d} \mathfrak{o} \mathfrak{m} \mathfrak{o} \mathfrak{m} \mathfrak{g} \mathfrak{b} \mathfrak{a}$

შეგიძლიათ მოთხოვნის პარამეტრების, ცხრილების ან ველების თვისებების როგორც დათვალიერება, ისე მოდიფიცირება. მათი მოდიფიცირების ან დათვალიერების მიზნით აირჩიეთ მენიუდან View→Properties (ნახვა→თვისებები) ბრძანება. გაიხსნება თვისებების დიალოგური ფანჯრა. დააწკაპუნეთ იმ ობიექტზე, რომლისთვისაც გინდათ მისი დაყენება. ფანჯრის შემკველობა შეიცვლება და აისახება მოცემული ობიექტის თვისებები.

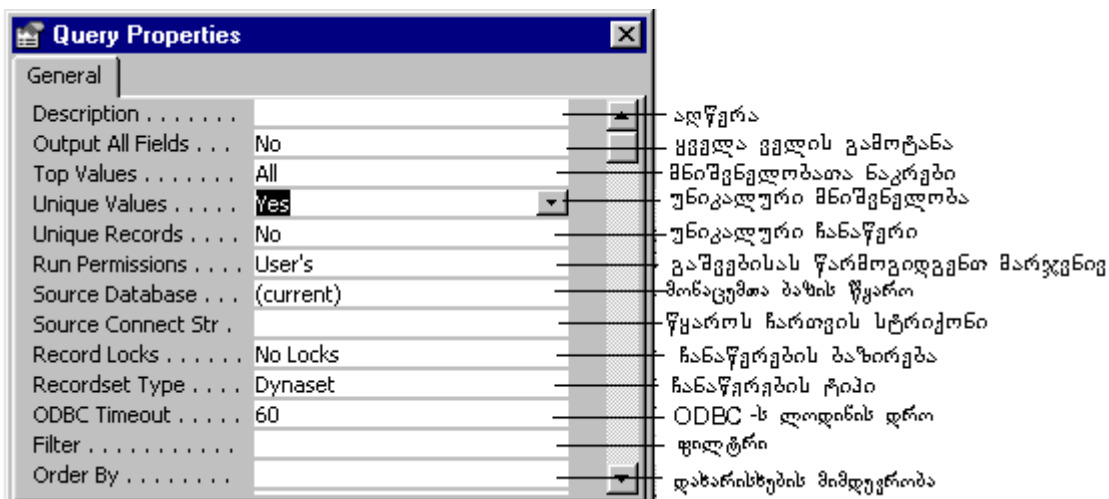
შეიძლება გამოსათვლელი ველის ფორმატის შეცვლა ანუ გამოსატან შედეგში მძიმის შემდეგ ათწილადი ნიშნების რაოდენობის მითითება.

მოთხოვნის მნიშვნელოვანი თვისება მისი უნიკალურობაა. ავტომატურად ამ მნიშვნელობისათვის დაყენებულია უარყოფა - „არა“, მაგრამ თუ გამეორებული ჩანაწერებია, მაშინ ისინი გამოჩნდებიან დინამიურ ცხრილში. გამეორების გამოსარიცხად ამ თვისების მნიშვნელობად დააყენეთ „ლიახ“.

[illegible]

ორი ტექსტური ველის გადაბმა ერთ ველად.

გამოიყენეთ გამოსათვლელი ველი და ოპერაცია კონკატენცია. ცარიელ სვეტში შეიტანეთ ახალი სახელი და ორწერტილი, შემდეგ - ველის სახელი, რომელსაც მოაქცევთ კვადრატულ ფრჩხილებში. ველებს შორის დაწერეთ ამპერსანდის ნიშანი (&). თუ საჭიროა ველებს შორის დაწერეთ ჰარი, გამოიყენეთ ორი ბრჭყალი და სხვა ამპერსანდი. ([ველი1] & “ & [ველი2]), იხ. სურ. 5. 7.



სურ. 5. 7. მოთხოვნის თვისებების ფანჯარა

VI ტ ა ზ ო

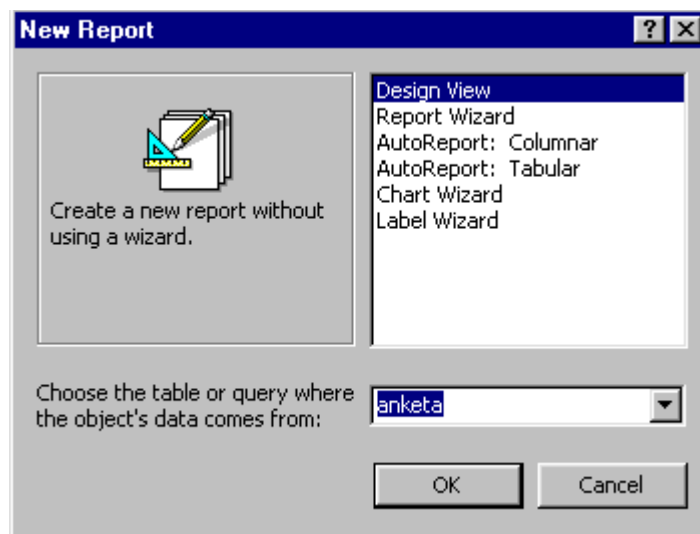
უწყვისების შექმნა

მონაცემთა ბაზის ობიექტი - უწყისი გამოიყენება მონაცემების გარკვეული ფორმით დასაბეჭდად. ბაზაში მონაცემების დაბეჭდვა უშუალოდ ცხრილიდან, ფორმიდან ან მოთხოვნიდან შეუძლებელია. ჯერ რომელიმე მათგანის ბაზაზე უნდა შექმნათ უწყისი და შემდეგ დაბეჭდოთ.

უწყისის შექმნამდე გაიმეტეთ დრო და დაგეგმეთ მისი სტრუქტურა, რისთვისაც:

- ❑ დაათვალიერეთ თქვენს მონაცემთა ბაზაში არსებული ცხრილები, ფორმები და მოთხოვნები;
- ❑ მიუთითეთ მონაცემთა რომელი კომპონენტები შევა უწყისში;
- ❑ დარწმუნდით, რომ ყველა საჭირო მონაცემი შეტანილია;
- ❑ თუ მონაცემების შეტანა გსურთ, გამოიყენეთ ფორმა და არა უწყისი. შეუძლებელია უწყისში მონაცემების შეტანა და მოდიფიცირება;
- ❑ დაფიქრდით მოთხოვნაზე, როგორც უწყისის შექმნის წინასწარ ნაბიჯზე, განსაკუთრებით თუ უწყისში უნდა შედიოდეს სხვადასხვა ცხრილის ველები ან ნაწარმოები ველები;
- ❑ დაათვალიერეთ დაბეჭდილი უწყისები და რომელიმე ნაწილის შესასწორებლად გაესაუბრეთ იმათ, ვინც ამ უწყისებს იყენებს.

უწყისის შექმნის პროცედურა ძალიან ჰგავს ფორმების შექმნას. ახალი უწყისის შესაქმნელად გამოიძახეთ მონაცემთა ბაზის ფანჯარა, გახსენით Reports (უწყისები) და დააწკაპუნეთ New (ახალი უწყისი) ღილაკზე. გამოვა დიალოგური ფანჯარა New Reports (ახალი უწყისი).



სურ. 6. 1. დიალოგური ფანჯარა New Reports (ახალი უწყისი)

დიალოგურ ფანჯარაში ჩამოთვლილია შემდეგი:

- ❑ *Design View (კონსტრუქტორი)*, უწყისის შექმნა კონსტრუქტორის რეჟიმში;
- ❑ *Report Wizard (უწყისის ოსტატი)*, შეკითხვებითა და წინასწარ შექმნილი შაბლონებით გეხმარებათ უწყისის შექმნაში;
- ❑ *Autoreport: Columnar*, გამოაქვს უწყისი სვეტების სახით ანუ თითო ჩანაწერი ვერტიკალურად;
- ❑ *Autoreport: Tabular*, გამოაქვს უწყისი ცხრილის სახით;
- ❑ *Chart Wizard (დიაგრამის უწყისი)* ქმნის უწყისს მონაცემების გრაფიკის ან დიაგრამის მეშვეობით;
- ❑ *Label Wizard (საფოსტო ეტიკეტები)*, სხვადასხვა ფორმატის გამოყენებით ქმნის საფოსტო ეტიკეტებს (იარლიყებს).

თუ ბაზაში გაქვთ ფორმა, რომელიც სტრუქტურით ჰგავს უწყისს, შეგიძლიათ მისი შენახვა უწყისად. ამისათვის თვითონ შეგიძლიათ დააყენოთ შენახვის ღილაკი იარაღების სტრიქონზე, რისთვისაც მენიუდან აირჩიეთ View→Toolbars→Customize (ნახვა→იარაღების ჯგუფი→მომხმარებელი) ბრძანება. გამოვა ფანჯარა, გახსენით Commands (ბრძანებები), კატეგორიებში მონახეთ Form / Report Design (ფორმის/უწყისის კონსტრუქტორი), მონახეთ Save Report (შენახე უწყისი) ღილაკი, გადაათრიეთ იარაღების ჯგუფში და დახურეთ.

6.1. უწყისის შექმნა უწყისის ოსტატის მეშვეობით

უწყისის შესაქმნელად თუ აირჩევთ უწყისის ოსტატის, Access გთავაზობთ დიალოგური ფანჯრების სერიას, რომლებსაც გამოაქვთ კითხვები უწყისის სპეციფიკაციის შესახებ. უწყისებთან მუშაობა ძალიან ჰგავს ფორმებთან მუშაობას და ამიტომ ამ დიალოგური ფანჯრებიდან უმეტესობა თქვენთვის ნაცნობი იქნება.

ოსტატის მეშვეობით უწყისის შესაქმნელად:

1. დააწკაპუნეთ New Object (ახალი ობიექტი) ღილაკზე და აირჩიეთ Report (უწყისი);
2. დაბეჭდეთ ან აირჩიეთ იმ ცხრილის/ფორმის სახელი, რომლის საფუძველზეც ქმნით უწყისს;
3. აირჩიეთ Report Wizard (უწყისის ოსტატი) და მიეცით დასტური;
4. ველების სიიდან აირჩიეთ ისინი, რომლებიც უნდა შედიოდნენ უწყისში და მიეცით დასტური;
5. მიუთითეთ დაჯგუფების სასურველი პარამეტრები და მაჩვენებლით დააწკაპუნეთ Next (მომდევნო) ღილაკზე;
6. მიუთითეთ მონაცემების დახარისხების მიმდევრობა და მაჩვენებლით დააწკაპუნეთ Next (მომდევნო) ღილაკზე;
7. აირჩიეთ უწყისის მაკეტი (ცხრილი, ვერტიკალური ან უწყისების დასაჯგუფებლად, სხვადასხვა ტიპის საფეხურები, ბლოკები და ა.შ.) და შეარჩიეთ ფორმა Landscape (აღბომისებრი) ან Portrait (წიგნისებრი);
8. თუ საჭიროა, მიუთითეთ, რომ ოსტატმა ააწყოს ველების სივანე ერთ გვერდზე მოსათავსებლად;
9. დააწკაპუნეთ Next (მომდევნო) ღილაკზე;
10. აირჩიეთ სტილი (მონიშნული, კომპაქტური, მარტივი, მკაცრი, ფორმალური ან შავ-თეთრი);
11. შეიტანეთ უწყისის სათაური და აირჩიეთ უწყისის დათვალიერების ან უწყისის სტრუქტურის შეცვლის რეჟიმი;
12. აირჩიეთ უწყისთან მუშაობის შესწორება (არ არის აუცილებელი);
13. დააწკაპუნეთ Finish (საბოლოო) ღილაკზე.

6.2. უწყისში დაჯგუფების არჩევა

უწყისის ოსტატით მუშაობისას, თუ აირჩევთ დაჯგუფების პარამეტრს, უწყისი დაიყოფა სხვადასხვა ნაწილად. დაჯგუფების პარამეტრის მნიშვნელობის ყოველი შეცვლა იწვევს ჯგუფის სათაურისა და შენიშვნების (ნაწილობრივი ჯამების) გამოჩენას.

დიალოგურ ფანჯარაში, რომელიც შეკითხვებს იძლევა დასაჯგუფებელი ველების შესახებ, არის Grouping Options (დაჯგუფების პარამეტრები) ღილაკი, რომელიც ხსნის Grouping interval (დაჯგუფების ინტერვალები) დიალოგურ ფანჯარას. ჯგუფის თითოეული ტექსტური ველისათვის შეგიძლიათ აირჩიოთ მთელი ტექსტის ან მხოლოდ პირველი რამდენიმე სიმბოლოს მიხედვით დაჯგუფება. მაგალითად, თუ გსურთ ჯერ მხოლოდ ა - ანით დაწყებული ტექსტი დაჯგუფდეს, შემდეგ ბ - ბანით დაწყებული და ა. შ.

თუ მონაცემებს აჯგუფებთ რიცხვითი ველის მიხედვით, შეგიძლიათ მონაცემების დაჯგუფება ათად, ორმოცდაათად, ასად (1-10 ჯგუფდება ერთად, 11-20 ასევე, ან 1-50, 51-100).

თუ დასაჯგუფებელი ველი შედგება თარიღებისაგან, შეგიძლიათ დააჯგუფოთ წლის, კვარტლის, თვის და დროის სხვა შუალედის მიხედვით.

6. 3. უწყისის მოდიფიცირება, წინასწარ დათვალიერება და ბეჭდვა

უწყისების მოდიფიცირება ისევე ხდება, როგორც ფორმის კონსტრუქტორის რეჟიმში. მოვიყვანოთ მოკლე ჩამონათვალი, რაც უნდა იცოდეთ უწყისებთან მუშაობისას.

1. ერთი ან რამდენიმე ველის დამატება;
2. რამდენიმე ელემენტის ერთდროულად მონიშვნა <Shift> კლავიშის მეშვეობით;
3. <Delete> კლავიშის მეშვეობით მართვის ელემენტების წაშლა;
4. მართვის ელემენტების გადაადგილება, მათი გადართვა თაგვით, როდესაც მაჩვენებელი ღებულობს ხელის მტევნის (ველი წარწერით) ან თითის ფორმას;
5. მართვის ელემენტების ზომების გატოლება ან შეცვლა მენიუდან Format (ფორმატის) ბრძანების გამოყენებით;
6. თუ საჭიროა ხაზების დამატება ხაზის შექმნის იარაღის გამოყენებით;
7. Properties (თვისებები) ფანჯარაში შედწევა, მონიშნულ ობიექტზე თაგვის მარჯვენა კლავიშზე დაწკაპუნებით;
8. ელემენტების გარე სახისა და რიცხვების ფორმატის შეცვლა, თვისებით Format (ფორმატი) და Number (რიცხვების) ათობითი ნიშნებით;
9. გვერდების დაყოფის იარაღის გამოყენება და მუშაობის ცოდნა თვისებით “ახალი გვერდის დაწყება” და “ერთად დაჭერა”;
10. ტექსტის ზომების შეცვლა თვისებით “შეგიძლიათ გაზარდოთ”. თუ მისცემთ თანხმობას Yes (დიახ), მაშინ ტექსტი მოთავსდება რამდენიმე სტრიქონში.

უწყისის წინასწარი დათვალიერება

თუ უწყისის ოსტატის დიალოგურ ფანჯარაში აირჩიეთ აღნიშნული რეჟიმი, მაშინ Access გამოიძახებს უწყისის წინასწარი დათვალიერების რეჟიმის ფანჯარას, რომელიც ეკრანზე გიჩვენებთ უწყისს იმ სახით, როგორც უნდა დაიბეჭდოს. იხ. სურ. 6. 3. თუ გაკმაყოფილებთ, მისცემთ ბეჭდვის ბრძანებას, თუ არა, მაშინ დაბრუნდებით უწყისის მოდიფიცირების რეჟიმში და შეცვლით ფორმატს.

Characteristics by first. Family	Child's name	Characteristics	Nationalities	d. ragdaria. Telephone
ბერიძე	7500.00	მამაკაცი	ქართველი	5/15/83 33-45-40
ცხიშვილი	6000.00	ქალი	ქართველი	13/25/81 22-34-65
ცხიშვილი	600.00	ბავშვი	ქართველი	3/18/82 33-65-32
Summary for 'Characteristics' = average (3 detail records)	4700.00			
ბერიძე	9000.00	მამაკაცი	სომხური	4/30/83 65-50-55
ვენიშვილი	700.00	ბავშვი	ქართველი	6/15/82 34-77-55
Summary for 'Characteristics' = average (2 detail records)	4850.00			
ბერიძე	500.00	ბავშვი	ქართველი	5/20/85 23-45-55
ცხიშვილი	500.00	ბავშვი	ქართველი	10/15/83 22-75-85
Summary for 'Characteristics' = average (2 detail records)	500.00			

თებერვალი, 09, 2002 Page 1 of 1

სურ. 6. 3. უწყისის წინასწარი დათვალიერების რეჟიმი

ამ რეჟიმში კურსორის ფორმა იცვლება და ღებულობს ლუბის ფორმას. თუ უწყისზე დააწკაპუნებთ, გაიზრდება უწყისის ფორმის მასშტაბი და იგი შეგიძლიათ დაათვალიეროთ ლიფტის მეშვეობით. მაჩვენებლის ხელახლა დაწკაპუნებით უწყისი აღიდგენს საწყის ზომას, უწყისს შეგიძლიათ დაარქვათ სახელი და შეინახოთ ან დაბეჭდოთ.

VII თაზო

ზოგადად Visual Basic-ის შესახებ

Visual Basic-ის საშუალებით შეგიძლიათ ადვილად და სწრაფად შექმნათ პროექტი. იგი შეიცავს გრაფიკულ საშუალებას, რომელშიც შეიძლება თქვენი პროექტისათვის საჭირო ეკრანული ფორმებისა და მართვის ელემენტის კონსტრუირება. გარდა ამისა, იგი შეიცავს ისეთ საშუალებებს, რომლებიც აადვილებენ პროგრამისტის შრომას, ესენია: პროექტები, ფორმები, ობიექტების შაბლონები, პროგრამირების ენა, არასტანდარტული მართვის ელემენტები და სხვა.

Visual Basic შექმნილია Internet-ის გათვალისწინებით და მასში გამოჩნდა დამუშავების ახალი მეთოდი - Windows-ის გამცილებელი, რომლის საშუალებითაც ხდება წინასწარ შექმნილი ნიმუშების გამოყენება. ნაცნობი მდგომარეობა კი მომხმარებელს ეხმარება უშუალოდ დაიწყოს პროექტის დამუშავება Visual Basic-ში.

მოთხოვნები

VB-ში მუშაობისათვის უნდა იცნობდეთ Windows-ში მუშაობის პრინციპებს, შეგუდოთ კლავიატურისა და თაგვის ხმარება, ერკვეოდეთ მის ინტერფეისში იმდენად, რომ შეძლოთ პროექტის გაშვება. გარდა ამისა იგულისხმება, რომ კომპიუტერში გაქვთ VB-ის კომპაქტ დისკიდან მაგალითები, რომლებსაც შემდეგში განვიხილავთ.

გრაფიკული ინტერფეისი

VB-ში მუშაობის მთავარი საშუალებაა გრაფიკული ინტერფეისი. იგი არის ეკრანული ფორმებისა და გამოსახულებების ერთობლიობა, რომელიც მასზე მოთავსებული ობიექტების საშუალებით უზრუნველყოფს მომხმარებლისა და პროგრამის ურთიერთკავშირს. გრაფიკული ინტერფეისის მთავარი ელემენტებია: ფორმა, მართვის ელემენტები და ობიექტები.

ფორმა გრაფიკული ინტერფეისის მთავარი ობიექტია. იგი არის ფანჯარა მასზე მოთავსებულია სხვადასხვა ელემენტი.

მართვის ელემენტი გრაფიკული ინტერფეისის მთავარი იარაღია. მას მიეკუთვნება: ბრძანებითი ღილაკები, აღმები, გადამრთველები, ტექსტური და გრაფიკული ველები, მენიუები და სხვ. ისინი მოთავსებულია VB-ის სამუშაო გარემოს მთავარ ფანჯარაში იარაღების (ხელსაწყოების) პანელზე.

ობიექტები ფორმაზე მოთავსებული მართვის ელემენტებია, რომელთა მახასიათებლებია: თვისებები, მეთოდები და მოვლენები.

ობიექტთა კლასები ერთნაირი თვისებების, მეთოდებისა და მოვლენების მქონე ობიექტებია. ფორმაზე მოთავსებისას ობიექტები ინომრება. მაგალითად, Text1, Text2, Text3 და ა. შ. ობიექტთა კლასებია: TextBox (ტექსტური ველი), CommandButton (ბრძანებითი ღილაკები), ListBox (სიის ველი), OptionButton (გადამრთველი ღილაკები) და სხვ.

პროგრამული კოდი. გრაფიკული ინტერფეისის ობიექტისათვის იწერება პროგრამის ტექსტი - პროგრამული კოდი. იგი შედგება ერთი ან რამდენიმე ოპერატორის შემცველი სტრიქონისაგან და მისი აკრება ხდება სპეციალურ ფანჯარაში.

ობიექტთა თვისებები გრაფიკული ინტერფეისის ყოველი ობიექტი ხასიათდება თვისებებით, ესაა ობიექტის სახელი, ზომა, ფერი, წარწერა მასზე, ეკრანზე გამოჩენა, დამალვა და სხვ. თავდაპირველად მათ აქვთ საწყისი მნიშვნელობები, რომელთა შეცვლა შესაძლებელია სამუშაო გარემოს Properties (თვისებების) ფანჯარაში. თვისებებზე მიმართვის ზოგადი ფორმაა - ობიექტი, თვისება.

მაგალითად, Command1. Caption აღნიშნავს ბრძანებით ღილაკზე წარწერის გაკეთებას.

ობიექტთა მეთოდები გარკვეულ ობიექტზე მიბმული და პროგრამულ კოდებში ჩაწერილი ოპერატორებია, რომელთა საშუალებითაც ხდება ობიექტის თვისებების შეცვლა და მასზე სხვადასხვა ოპერაციის შესრულება. მეთოდის ჩაწერის ზოგადი სახეა: ობიექტი, მეთოდი (პარამეტრები).

მაგალითად, Form1. Print “მოგესალმებით”. ამ ბრძანებით ხდება Form1 ობიექტზე Print (ბეჭდვის) მეთოდით ტექსტის “მოგესალმებით” ეკრანზე გამოტანა.

მოვლენა არის ქმედება, რომელზეც ობიექტი რეაგირებს. მოვლენა გამოწვეულია მომხმარებლის ზემოქმედებით თავზე ან რომელიმე კლავიშზე დაჭერით, ობიექტის გადაადგილებით, ფაილის გახსნით, დახურვით და სხვ.

პროცედურები. პროგრამული კოდები მოთავსებულია პროცედურებში, რომლის სტრუქტურის ზოგადი სახეა: Sub Command1-Click() /პროცედურის სათაური/

პროგრამული კოდი /პროცედურის ტანი/

End Sub /პროცედურის დასასრული/

პროექტი VB-ში შექმნილი პროდუქტია, რომელშიც აღწერილია ფორმის ობიექტთა მოვლენითი პროცედურები და მათ შორის არსებული კავშირები. პროექტი შეიძლება შეიცავდეს ერთზე მეტ ფორმას და შესაბამისად რამდენიმე მოდულს.

ფორმა და მასთან დაკავშირებული პროგრამული მოდული ინახება .frm გაფართოების ფაილში, პროექტი კი .vbp გაფართოების ფაილში.

7.1. Visual Basic-ის სამუშაო გარემო

VB-ის პროექტის მომზადება ხდება მის სამუშაო გარემოში, რომელსაც დაპროექტების გარემო ეწოდება და რომლის საშუალებითაც ხდება გრაფიკული ინტერფეისის შექმნა, პროგრამული კოდის შედგენა, გამართვა და შესრულება.

VB-ის გაშვება ხდება ბრძანებით Start → Programs → Visual Basic ან მის ნიშნაკზე თავის ორჯერ დაწკაპუნებით.

გაშვების შემდეგ ეკრანზე ჩნდება დიალოგური ფანჯარა New Project (ახალი პროექტი). იხილეთ სურ. 7.1.

ჩანაცვლება New-ში შეგიძლიათ აირჩიოთ პროექტის შაბლონების სიიდან თქვენი პროექტისათვის შესაფერი შაბლონი. დიალოგურ ფანჯარას აქვს ფურცლის სამი ჩანაცვლება.



სურ. 7.1. New Project Wizard (ახალი პროექტის ოსტატის) დიალოგური ფანჯარა.

1. New (ახალი პროექტი);
2. Existing (არსებული პროექტი);
3. Recent (ახლახანს გამოყენებული პროექტები).

New ჩანაცვლებაში შედის შემდეგი პროექტების შაბლონები:

- ❑ Standard EXE (სტანდარტული EXE) ფაილი;
- ❑ ActiveX EXE (ActiveX -ის EXE) ფაილი;
- ❑ ActiveX DLL (ActiveX -ის DLL) ბიბლიოთეკა;
- ❑ ActiveX Control (მმართველი ელემენტი ActiveX);
- ❑ Visual Basic Application Wizard (პროექტის ოსტატი);
- ❑ Addin (დამატება);
- ❑ ActiveX Document DLL (ActiveX-ის საბუთის DLL ბიბლიოთეკა);
- ❑ ActiveX Document EXE (ActiveX საბუთის EXE) ფაილი.

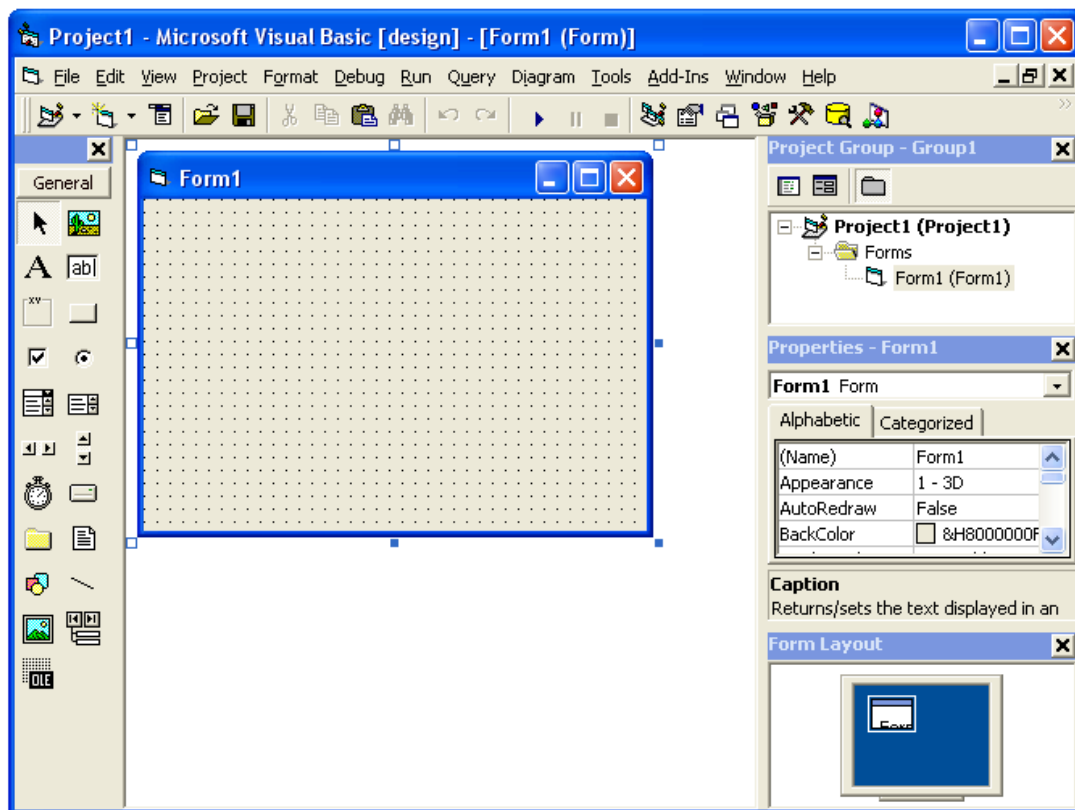
შაბლონის არჩევით, Visual Basic-ს ავალზეთ ძირითადი პროექტის შექმნას, რითაც დაზოგავთ საკმარის დროს, რაც განსაკუთრებით მნიშვნელოვანია თუ თქვენ ახალბედა ხართ. უფრო ხშირად გამოიყენება შაბლონი Standard EXE.

ჩანაცვლებაში Existing (არსებული) შეგიძლიათ აირჩიოთ არსებული პროექტი. ეს შეიძლება იყოს Visual Basic-ის კომპლექტში არსებული მაგალითი, ან სხვა პროექტი, რომელზედაც ადრე გიმუშავიათ.

ჩანაცვლება Recent (უახლესი) საშუალებას გაძლევთ იმუშაოთ ახლახანს გამოყენებულ პროექტთან. გარეგნულად იგი ჰგავს ჩანაცვლებას Existing, მაგრამ აქ ჩამოთვლილი იქნება მხოლოდ ბოლო დროს შესრულებული პროექტები.

7. 2. დამუშავების ინტეგრირებული საშუალება - IDE

პროექტის ოსტატთან მუშაობის დამთავრების შემდეგ, ეკრანზე დაინახავთ Visual Basic-ის დამუშავების ინტეგრირებულ საშუალებას ანუ IDE-ს. ეს არის Visual Basic-ის მნიშვნელოვანი ნაწილი, აქ მოიყვანთ მთლიანობაში თქვენს დანართს, იხ. სურ. 7. 2.



სურ. 7. 2. Visual Basic-ის დამუშავების ინტეგრირებული საშუალება

ტერმინი IDE პროგრამირებაში ჩვეულებრივ აღწერს ინტერფეისს და იმ საშუალებებს რომელშიც ხდება პროექტის დამუშავება. საშუალებებს ჰქვია ინტეგრირებული იმდენად, რამდენადაც ეკრანიდან შეიძლება გამოიძახოთ ნებისმიერი საჭირო იარაღი. დამუშავების

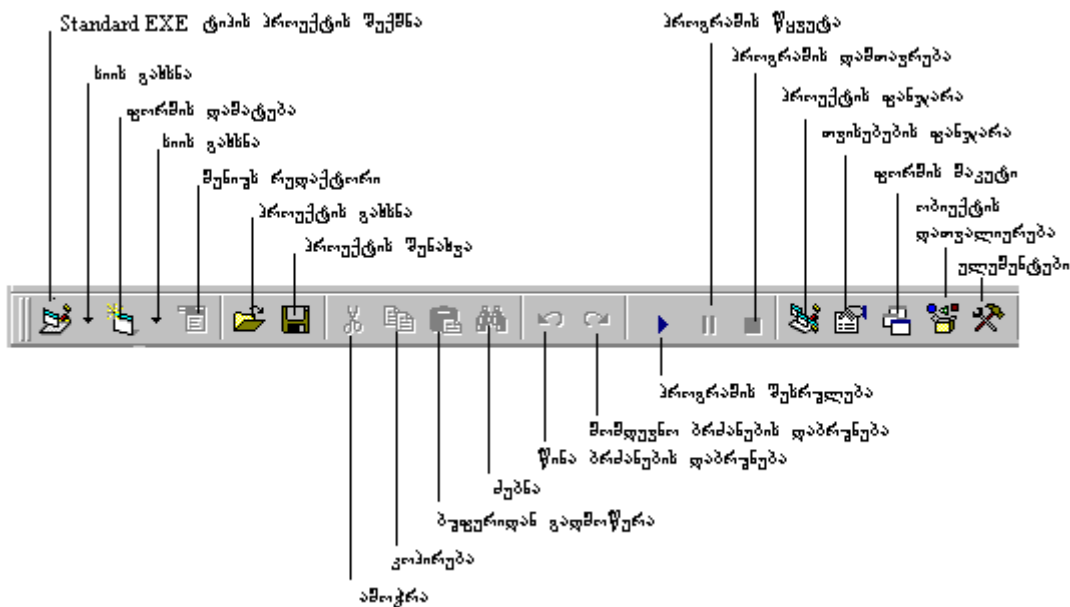
ინტეგრირებულ საშუალებებს ხშირად უწოდებენ კონსტრუირების საშუალებებს, კონსტრუირების პროგრამებს ან უბრალოდ IDE –ს. შემდეგში ვისარგებლებთ ამ ტერმინით.

IDE შედგება რამდენიმე კომპონენტისაგან:

- ❑ სტანდარტული იარაღების ჯგუფი:
- ❑ პროექტის ფანჯარა:
- ❑ თვისებების ფანჯარა:
- ❑ ელემენტების ჯგუფი:
- ❑ ფორმების დიზაინერი.

სტანდარტული იარაღების ჯგუფი

მენიუს სტრიქონის ქვეშ მოთავსებულია Visual Basic-ის სტანდარტული იარაღების ჯგუფი, იხ. სურ. 7.3 . თუ იგი არ ჩანს, შეასრულეთ ბრძანება View→Toolbars→Standard. საერთოდ, ყველაფრის მართვა შეგიძლიათ მენიუდან, მაგრამ სტანდარტული იარაღების ჯგუფით უფრო ადვილია საჭირო ბრძანების გამოყენება.



სურ. 7. 3. Visual Basic-ის სტანდარტული იარაღების ჯგუფი

ბრძანება View→Toolbars საშუალებას გაძლევთ გამოაჩინოთ იარაღების ჯგუფი ეკრანზე ან დამალოთ.

პროექტის ფანჯარა

ეკრანის მარჯვენა მხარეს, იარაღების ჯგუფის ქვეშ მოთავსებულია პროექტის ფანჯარა, იხ. სურ. 7.4. მისი დანიშნულებაა სწრაფად მოგაწოდოთ ინფორმაცია პროექტის ობიექტების - ფორმების, კლასებისა და მოდულების შესახებ.

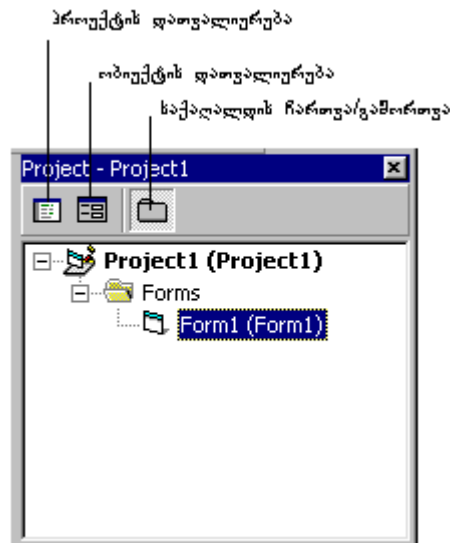
პროექტის ფანჯარა ჰგავს Windows-ის Explorer-ის (გამცილებელი) ფანჯარას და ისევე შეგიძლიათ მასში ჩალაგებული საქაღალდეების გახსნა და დახურვა, როგორც Windows-ის გამცილებელის ფანჯარაში.

პროექტში გაერთიანებულია მისი შემადგენელი ყველა ობიექტი. პროექტის ტესტირებისა და გამართვის მიზნით, მისი შემდგომში გამოყენებისათვის Visual Basic-ი პროექტის შენახვის დროს ფაილს ანიჭებს .VBP გაფართოებას (რაც ნიშნავს Visual Basic Project). მარტივი პროექტი, ჩვეულებრივ, ერთ ფორმას შეიცავს.

იმისათვის, რომ დაათვალიეროთ ფორმა, აირჩიეთ იგი პროექტის ფანჯარაში და დააწკაპუნეთ ღილაკზე View Object, ან ორჯერ დააწკაპუნეთ არჩეული ფორმის სახელზე.

პროექტის ფანჯარაში ფორმის გვერდით არის მოდულებისა და კლასების ჩამონათვალი.

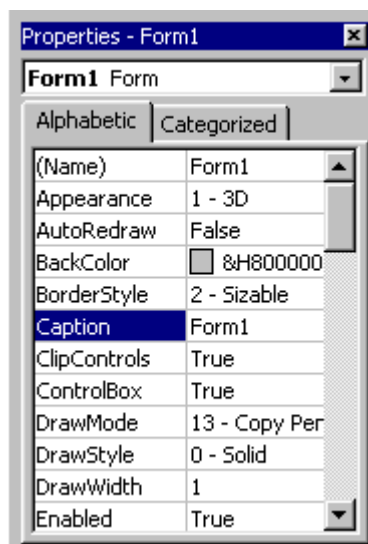
თუ გსურთ, პროექტიდან ობიექტის წაშლა ან დამატება, გამოიტანეთ კონტექსტური მენიუ და შესაბამისად აირჩიეთ ბრძანება Remove, Delete, Add, Print.



სურ. 7. 4. პროექტის ფანჯარა

თვისებების ფანჯარა

პროექტის ფანჯარის ქვეშ მოთავსებულია თვისებების ფანჯარა. მასში ჩამოთვლილია მონიშნული ობიექტის სხვადასხვა ატრიბუტი. პროექტის ყოველი ფორმა არის ობიექტი. ამ ფორმაზე ყოველი მმართველი ელემენტი (მაგალითად, ღილაკი) აგრეთვე არის ობიექტი. ყველა ობიექტს აქვს თავისი ატრიბუტები - ფერი და ზომა. ზოგიერთი ატრიბუტი ცვლის ობიექტის არა მარტო გარეგნულ სახეს, არამედ მის ქცევასაც. ამ შემთხვევაში მათ უწოდებენ თვისებებს. ამრიგად, ყოველ ფორმას და მასზე მოთავსებულ ყოველ ელემენტს აქვს ერთდროულად რამდენიმე თვისება. ყველა თვისება აისახება თვისებების ფანჯარაში, იხ. სურ. 7. 5.

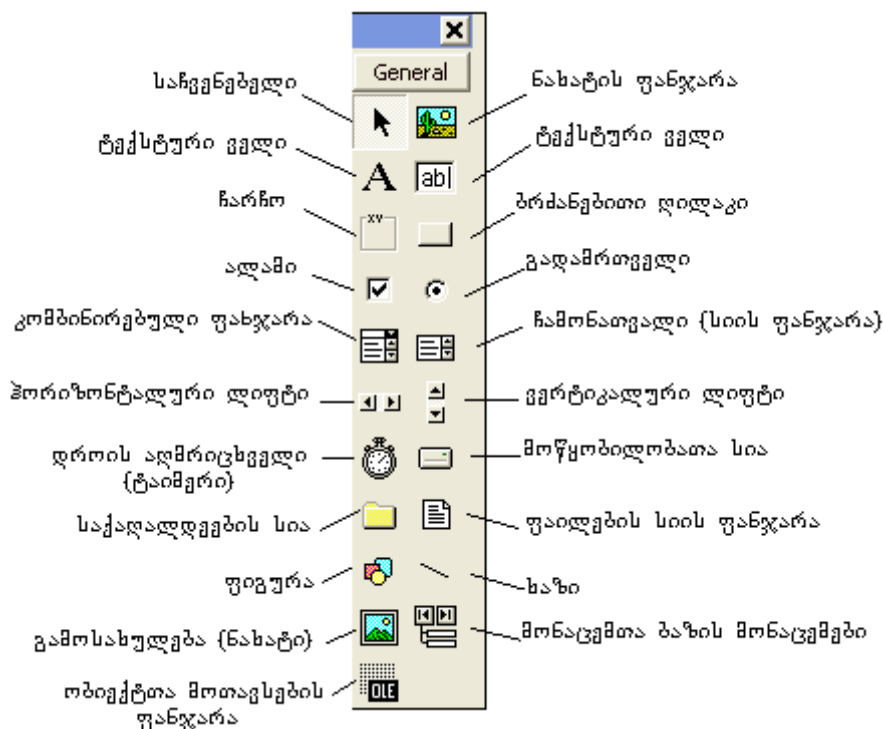


სურ. 7. 5. თვისებების ფანჯარა

ამ ფანჯარაში არის ობიექტის თვისებების ჩამონათვალი. რადგან ეს თვისებები საკმაოდ ბევრია, ამიტომ ფანჯარას აქვს ღილაკი, რომლის საშუალებითაც შეგიძლიათ ნახოთ ყველა თვისება. თვისების სახელი მეტყველებს თვისების დანიშნულებაზე (მაგალითად, Caption (წარწერა), Height (სიმაღლე), Width (სიგანე) და ა.შ.). თუ არ გახსოვთ რომელიმე თვისების დანიშნულება, შეგიძლიათ მისი მოკლე აღწერა ნახოთ თვისებების ფანჯარის ქვედა ნაწილში.

მართვის ელემენტების ჯგუფი

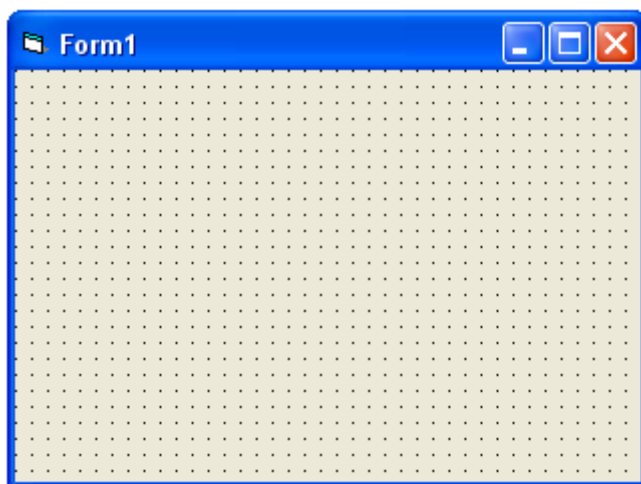
მართვის ელემენტების პანელი არის ფანჯარა, რომელზედაც მოთავსებულია პროექტის შესაქმნელად გამოსაყენებელი სხვადასხვა ელემენტის ნიშნები, იხ. სურ. 7. 6.



სურ. 7. 6. მართვის ელემენტების პანელი

ფორმის დიზაინერი

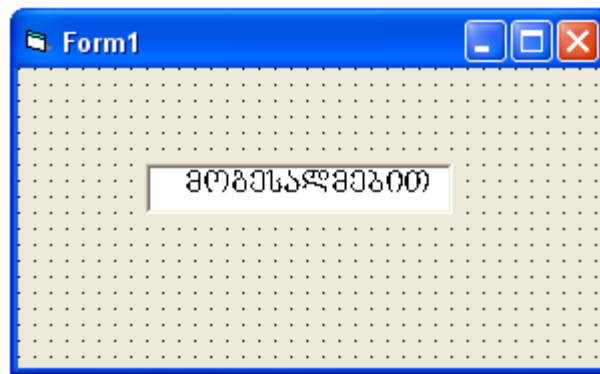
ფორმაზე მართვის ელემენტის მოსათავსებლად შეასრულეთ შემდეგი:



სურ. 7. 7. ფორმის დიზაინერის ფანჯარა

1. მართვის ელემენტების პანელზე აირჩიეთ სასურველი ელემენტი და მაჩვენებლით დააწკაპუნეთ მასზე ერთჯერ;
2. თავის მაჩვენებელი მოათავსეთ ფორმის იმ ადგილას სადაც გინდათ ობიექტის განთავსება (მაჩვენებელი მიიღებს ჯვრის ფორმას);

3. დაჭირეთ თავის მაჩვენებლის მარცხენა კლავიშს და თითის აულებლად ააგეთ სასურველი ზომის ობიექტი.



სურ. 7. 8. ფორმის დიზაინერი ტექსტური ობიექტით

✎ მართვის ელემენტების პანელში არჩეულ ელემენტზე თავის მაჩვენებლის მარცხენა კლავიშზე ორჯერ დაწკაპუნებით, შესაბამისი ობიექტი ავტომატურად მოთავსდება ფორმის ცენტრში და შეგიძლიათ გადაადგილების საშუალებით იგი სასურველ ადგილას დააყენოთ.

თუ ფორმაზე მოათავსეთ მმართველი ელემენტი (მაგალითად, ღილაკი), თვისებების ფანჯარაში აისახება მონიშნული ელემენტის თვისებები. შეგიძლიათ დაათვალიეროთ სხვადასხვა ობიექტისა და თვით ფორმის თვისებები, რისთვისაც წინასწარ უნდა მონიშნოთ ობიექტი მასზე მაჩვენებლის ერთჯერ დაწკაპუნებით. სწორად ობიექტების თვისებების დაყენება ხდება თვისებების ფანჯარაში პროექტის შექმნის დროს, თვისების შეცვლა კი შეგიძლიათ ნებისმიერ დროს.

ობიექტების ზოგიერთ თვისებათა ჩამონათვალი

ცხრილი № 7.1.

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
1	2	3
Project (პროექტი)	Name (სახელი)	prjFieldTypes
Form (ფორმა)	Name (სახელი)	frmFieldTypes
	Caption (წარწერა)	ფორმის სათაური
Label (ჭდე)	Name (სახელი)	lbl Label1
	Caption (წარწერა)	
TextBox (ტექსტური ველი)	Name (სახელი)	txtDisplay
	Text (ჩასაწერი ტექსტის შინაარსი)	მოგესალმებით
	BorderStyle	1 – Fixed Single
Frame (ჩარჩო)	Name (სახელი)	fraCharcho
	BorderStyle	1 – Fixed Single
CommandButton(ბრძანებითი ღილაკი)	Name (სახელი)	cmdBtn
	Caption (წარწერა)	&Add
	Height (სიმაღლე)	300
	Index (ინდექსი)	0
	Width (სიგანე)	1200
CheckBox (აღამი)	Name (სახელი)	chkBold
	Caption (წარწერა)	მუქი
	Visable (ხილვადი)	True
OptionButton (გადამრთველი)	Name (სახელი)	optDefault
	Value (cvladi)	True
	Visable (ხილვადი)	True

1	2	3
ComboBox (კომბინირებული ველი)	Name (სახელი)	cboProducts
	ItemData	(List)
	Font	სასურველი ფონტი
	Text	შინაარსი(მაგალითად, პური)
	List	(List)
ListBox (სიის ველი)	Name (სახელი)	lstGroceries
	Font	სასურველი ფონტი
Image (გამოსახულება)	Name (სახელი)	imgPicture1
	BorderStyle	(None)
	Picture	(Bitmap)
HScrollBar (ჰორიზონტალური ლიფტი)	Name (სახელი)	hscVolume
Label	Name (სახელი)	lblVolume
	Caption (წარწერა)	ფურცლების რაოდ.
Timer (საათი)	Name (სახელი)	tmrTime
	BorderStyle	1 – Fixed Single
Data (მონაცემთა ბაზის მონაცემი)	Name (სახელი)	მონაც. ბაზის სახელი
	DatabaseName (ბაზის სახელი)	ბაზის მისამართი
	RecordSource	არჩეული ცხრილი

გაითვალისწინეთ, რომ სახელის მნიშვნელობის ველში, ობიექტის სახელის ჩაწერისას, სახელს წინ ეწერება ობიექტის სახელის შესაბამისი თავსართი. მაგალითად, ტექსტური ობიექტისათვის txtსახელი, ჭდისათვის lblსახელი და ა.შ. ეს სახელი გამოიყენება პროგრამულ კოდში და მისი სწორად აკრება აუცილებელია.

7.3. მართვის ელემენტების გამოყენება

ფორმაზე ობიექტის მოთავსება შეგიძლიათ როგორც ზემოთ აღწერილი გზით, ისე მართვის პანელზე მოთავსებულ ელემენტზე თავისი მაჩვენებლის ორჯერ დაწკაპუნებით, რის შედეგადაც ობიექტი განთავსდება ფორმის ცენტრში, შემდეგ კი შეგიძლიათ გადაათრიოთ თქვენთვის სასურველ ადგილას. ფორმაზე ობიექტის მოთავსების შემდეგ, თავისი მაჩვენებლით ერთჯერ დააწკაპუნეთ მასზე, რათა იგი გააქტიურდეს და დაიწყოთ თვისებების ფანჯარაში მისი თვისებების დაყენება (ჩასწორება).

მართვის ელემენტებიდან პროექტებში ყველაზე ხშირად იყენებენ ღილაკებს. მათი საშუალებით ხდება როგორც ინფორმაციის შეტანა, ისე ფორმასთან დაკავშირებული სპეციალური ფუნქციის გამოძახება.

ტექსტურ ველებთან მუშაობა

VB-ის პროექტში სულ მცირე ერთი ტექსტური ველი მაინც იქნება. ეს ელემენტი გამოიყენება მონაცემების შესატანად ან მომხმარებლისაგან ინფორმაციის მისაღებად, ამიტომაც არჩეულია მისი თვისებები.

ყველა ობიექტის თვისებები იწევა Name-ით (სახელით). ტექსტური ველის ეს თვისება იწევა თავსართით. ტექსტურ ველს არა აქვს თვისება Caption (წარწერა), მისი შიგთავსი განისაზღვრება თვისებით Text (ტექსტი).

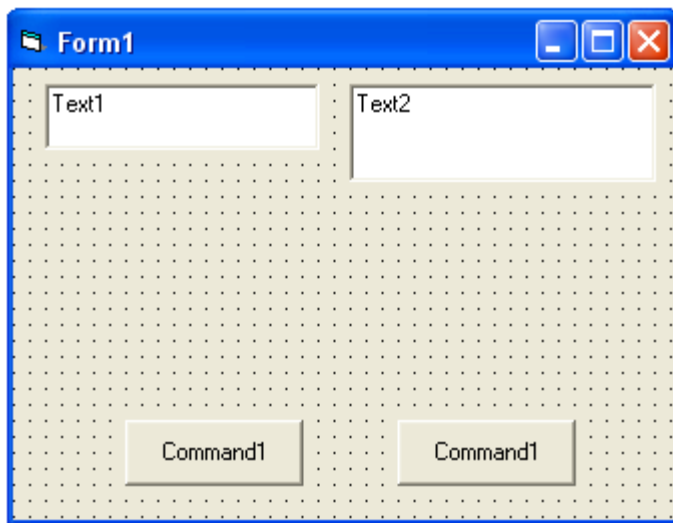
თვისება SetFocus ხშირად გამოიყენება პროექტში ტექსტურ ველთან მუშაობისას. თუ ღილაკის პროცედურა მოვლენა Click მთავრდება SetFocus – მეთოდით, მაშინ ფოკუსი გადაეცემა ტექსტურ ველს მონაცემების შესატანად, რის გამოც მომხმარებელს აღარ სჭირდება თავისი მაჩვენებლით ზედმეტი დაწკაპუნება ან Tab კლავიშის გამოყენება. ოპერატორს აქვს შემდეგი სახე:

txtMyTextBox. SetFocus

განვიხილოთ მაგალითი, რისთვისაც შეასრულეთ მიმდევრობა:

1. ჩატვირთეთ VB, აირჩიეთ ახალი პროექტი Standart Exe-ის საშუალებით;
2. ფორმის დიზაინერის ზედა ნაწილში მოათავსეთ ორი ტექსტური ველი;

3. ფორმის ქვედა ნაწილში კი მოათავსეთ ორი ბრძანებითი ღილაკი. ფორმას ექნება შემდეგი სახე, იხ. სურ. 7.9.



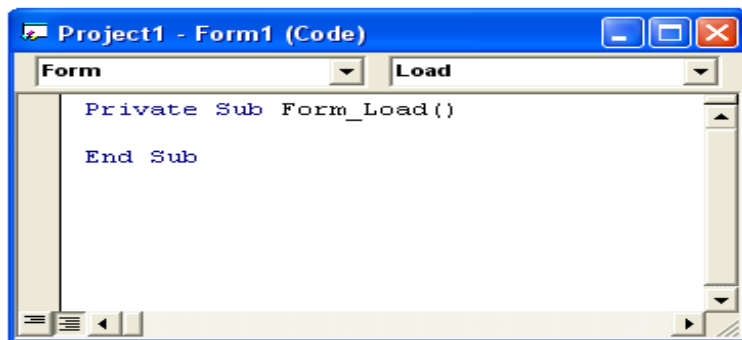
სურ. 7. 9. ფორმა ტექსტური ველით და ბრძანებითი ღილაკებით

ფორმის თითოეული ობიექტი ცალ-ცალკე მონიშნეთ აირჩიეთ თვისებები №72 ცხრილის შესაბამისად.

ცხრილი № 7.2

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
TextBox	Name	txtDisplay
	Font	AcadNusx
	Text	სასურველი ტექსტი
TextBox	Name	txtMulti
	Font	AcadNusx
	Multiline	True
	Tex ()	სასურველი ტექსტი
CommandButton	Name	cmdReset
	Caption	აღდგენა
	Font	AcadNusx
CommandButton	Name	cmdClose
	Caption	დახურვა
	Font	AcadNusx

თაგვის მაჩვენებლით ორჯერ დააწკაპუნეთ ფორმაზე, ეკრანზე გამოვა კოდების ფანჯარა, რომელიც შედგება ორი ნაწილისაგან მარცხენა - Form და მარჯვენა - Load. მარცხენა ნაწილში ჩაწერილია ფორმაზე თქვენს მიერ დაყენებული ობიექტების სახელები, მარჯვენაში - ამ ობიექტებზე შესასრულებელი მოქმედებები.



სურ. 7. 10. კოდების ფანჯარა

დაუბრუნდით ფორმის დიზაინერს, რისთვისაც მაჩვენებლით ორჯერ დააწკაპუნეთ პირველ ტექსტურ ველზე. კოდების ფანჯრის მარცხენა ნაწილის სათაურში სრულად დაიწერება პირველი ტექსტური ველის სახელი (txtDisplay), მარჯვენა ნაწილის სათაურში კი - (change) ცარიელი პროცედურის ზოგადი სახე:

```
Private Sub txtDisplay_Change()
```

```
End Sub
```

ანალოგიურად გამოიტანეთ ცარიელი პროცედურა მეორე ტექსტური ველისათვის, რომლის ზოგადი სახეა:

```
Private Sub txtMulti_Change()
```

```
End Sub
```

დაუბრუნდით ფორმას, მაჩვენებლით ორჯერ დააწკაპუნეთ პირველ ბრძანებით ღილაკზე და შესაბამის ცარიელ პროცედურაში ჩაწერეთ შემდეგი კოდი:

```
Private Sub cmdReset_Click()
```

```
    'ორიგინალი ტექსტის აღდგენა
```

```
    txtDisplay.Text = "ამ მაგალითში MultLine (მრავალხაზოვანი) თვისება მცდარია"
```

```
    optDefault.Value = True
```

```
End Sub
```

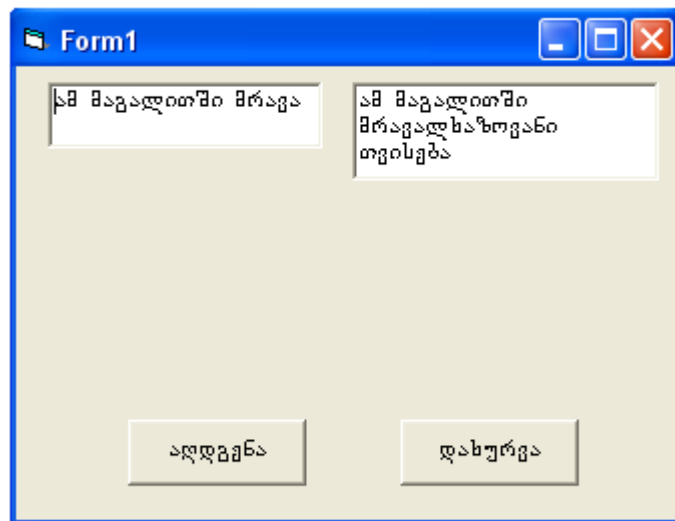
იგივე გააკეთეთ მეორე ბრძანებითი კლავისათვის და შესაბამის ცარიელ პროცედურაში ჩაწერეთ შემდეგი კოდი:

```
Private Sub cmdClose_Click()
```

```
    Unload Me ' ფორმის დახურვა.
```

```
End Sub
```

სტანდარტული იარაღების სტრიქონში მონახეთ კლავიში - Start და თქვენი პროექტი გაუშვით შესრულებაზე, შედეგი ეკრანზე ასე გამოისახება:



სურ. 7. 11. ერთსტრიქონიანი ტექსტის მრავალ სტრიქონად ჩაწერის პროექტის შედეგი

სტანდარტული იარაღების სტრიქონში მონახეთ კლავიში – End, დაამთავრეთ პროექტის შესრულება, რის შედეგადაც დაბრუნდებით ფორმის დიზაინერზე და შეინახეთ პროექტი.

ჩარჩო და გადამრთველები

მართვის ელემენტების პანელზე მონახეთ Frame-ჩარჩო და დაამატეთ თქვენს ფორმას. ჩარჩოში მოათავსეთ გადამრთველი. გადამრთველის საშუალებით ჯგუფიდან შეგიძლიათ აირჩიოთ ერთ-ერთი ვარიანტი. გადამრთველებს აჯგუფებენ პირდაპირ ფორმაზე ჩარჩოში. ამიტომ ჩარჩოზე მოათავსეთ კიდევ ოთხი გადამრთველი, თუ ადგილი არ გყოფნით გაზარდეთ ფორმის დიზაინერის ფანჯარა და ჩარჩო. მართვის ელემენტების პანელიდან აირჩიეთ Label - ჭდე და მოათავსეთ ჩარჩოს მარჯვნივ თავისუფალ ადგილას. გადართ

თვისებების ფანჯარაში და თითოეულ ობიექტისათვის აირჩიეთ შესაბამისი თვისება №7. 3 ცხრილიდან.

ცხრილი № 7. 3.

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
OptionButton	Name	OptDefault
	Font	AcadNusx
	Caption	გაჩუმებით
	Value	True (ჩართავს გადამრთველს)
OptionButton	Name	optEnd
	Font	AcadNusx
	Caption	ბოლოში გადატანა
	Value	False
OptionButton	Name	optInsert
	Font	AcadNusx
	Caption	5 სიმბოლოს შემდეგ
	Value	False
OptionButton	Name	optSelect
	Font	AcadNusx
	Caption	მთელი ტექსტის მონიშვნა
	Value	False
OptionButton	Name	optText
	Font	AcadNusx
	Caption	ტექსტის ჩასმა
	Value	False
Frame	Name	fraCharcho
	Font	AcadNusx
	Caption	კურსორის დაყენება
Label	Name	lblHelp
	Font	AcadNusx
	Caption	დააყენეთ გადამრთველი და დააკვირდით პირველ ველში ცვლილებას

მაჩვენებელი ორჯერ დააწკაპუნეთ პირველ გადამრთველზე, გაიხსნება კოდების ფანჯარა, შემდეგი პროცედურით:

```
Private Sub optDefault_Click()
```

```
End Sub
```

ჩაწერეთ მასში კოდების ფრაგმენტი

```
Private Sub optDefault_Click()
```

```
' კურსორს აყენებს ტექსტის დასაწყისში
```

```
txtDisplay.SelStart = 0
```

```
txtDisplay.SetFocus
```

```
End Sub
```

დატოვებულ როგორც არის და დაბრუნდით ფორმის დიზაინერზე. მაჩვენებლით ორჯერ დააწკაპუნეთ მეორე გადამრთველზე, კოდების ფანჯარაში გაჩნდება ცარიელი პროცედურა:

```
Private Sub optEnd_Click()
```

```
End Sub
```

ჩაწერეთ შემდეგი ოპერატორები:

```
' მოძებნეთ ტექსტის სიგრძე და ჩასვით კურსორი მის ბოლოში
```

```
txtDisplay.SelStart = Len(txtDisplay.Text)
```

```
' ტექსტის განახლება
```

```
txtDisplay.SetFocus
```

პროცედურა მიიღებს საბოლოო სახეს:

```
Private Sub optDefault_Click()
```

```
' მოძებნეთ ტექსტის სიგრძე და ჩასვით კურსორი მის ბოლოში
```

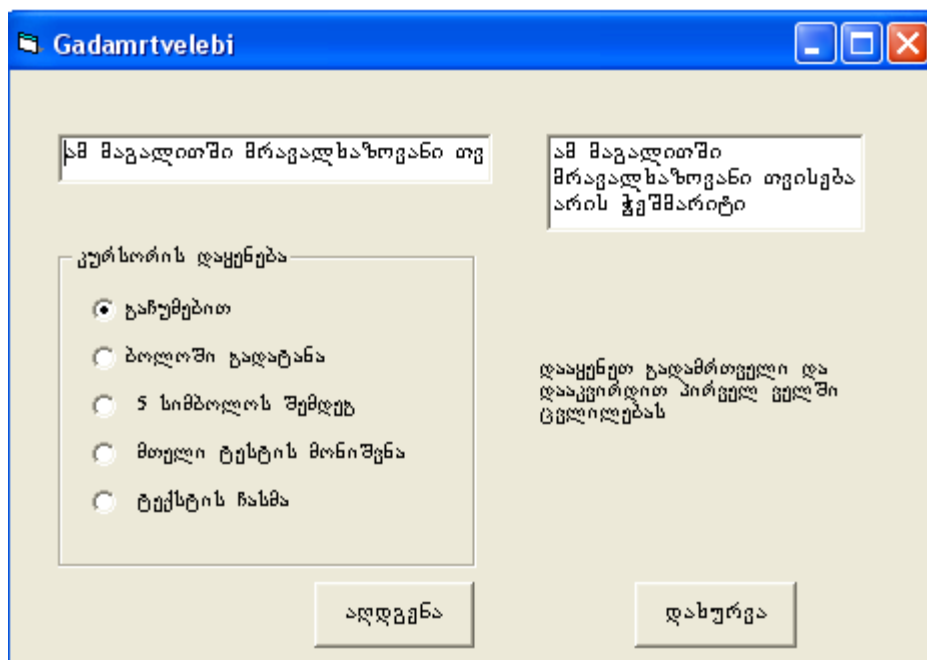
```
txtDisplay.SelStart = Len(txtDisplay.Text)
'ტექსტის განახლება
txtDisplay.SetFocus
End Sub
```

ანალოგიურად გაიმეორეთ დანარჩენი გადამრთველებისათვის და თითოეულ მათგანში ჩაწერეთ შესაბამისი ოპერატორები, მიიღებთ შემდეგი პროცედურების მიმდევრობას:

```
Private Sub optInsert_Click()
'ჩასვით კურსორი 5 სიმბოლოს შემდეგ
txtDisplay.SelStart = 5
'ტექსტის განახლება
txtDisplay.SetFocus
End Sub
```

```
Private Sub optSelect_Click()
'ჩასვით კურსორი ტექსტის თავში
txtDisplay.SelStart = 0
'ტექსტის სიგრძის განსაზღვრა
txtDisplay.SelLength = Len(txtDisplay.Text)
'ტექსტის განახლება
txtDisplay.SetFocus
End Sub
```

```
Private Sub optText_Click()
'ახალი ტექსტის ჩასმა და ძველის წაშლა
txtDisplay.SelText = "axali teqsti"
'ტექსტის განახლება
txtDisplay.SetFocus
End Sub
```



სურ. 7. 12. გადამრთველებზე პროექტის შედეგი

მონახეთ ფორმის დიზაინერის ობიექტი – აღდგენა, დააწკაპუნეთ მასზე ორჯერ და კოდების ფანჯარაში კი ჩაწერეთ შემდეგი ფრაგმენტი:

```
Private Sub cmdReset_Click()
```

```

' ტექსტის ორიგინალის აღდგენა
txtDisplay.Text = "The MultiLine property is set to False in this example"
' გადამრთველის ფუნქციის აღდგენა
optDefault.Value = True
End Sub

```

გაუშვით შესრულებაზე პროექტი, შედეგი ეკრანზე აისახება ისე, როგორც ეს ნახაზზეა, იხ. სურ. 7.12.

დაამთავრეთ პროგრამული კოდის შესრულება და შეინახეთ.

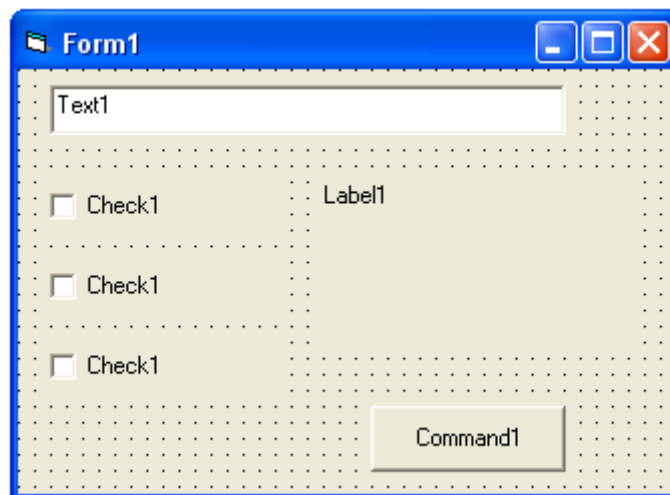
ალამი

ელემენტი ალამი ჰგავს ელემენტი გადამრთველს. ხშირად რამოდენიმე ალამი ერთიანდება ჯგუფში ჩარჩოს საშუალებით, ხოლო თვისება Value-ს დანიშნულების მიხედვით შეიძლება განსაზღვროთ ალამის მიმდინარე მდგომარეობა. ალამსა და გადამრთველს შორის ორი მნიშვნელოვანი განსხვავებაა. ალამი შეიძლება გამოიყენოთ როგორც ცალკე, დამოუკიდებელი ელემენტი, გადამრთველი ცალკე, ჯგუფში გაერთიანების გარეშე, აზრს მოკლებულია. გადამრთველი შეიძლება იყოს ჩართული ან გამორთული; ამიტომ Value თვისება მხოლოდ ორ მნიშვნელობას იღებს: False და True. ალამი კი შეიძლება იყოს დასმული, მოხსნილი ან განუსაზღვრელ მდგომარეობაში, ამიტომ Value თვისება ღებულობს სამ მნიშვნელობას. გაურკვეველ მდგომარეობაში მყოფი ალამი გვამცნობს, რომ სხვა ფორმაზე ჩართულია რომელიმე ალამი. განვიხილოთ მაგალითი.

1. შექმენით ახალი პროექტი File→New Project ბრძანებით;
2. ფორმის ზედა ნაწილში მოათავსეთ ტექსტური ველი;
3. ტექსტური ველის ქვემოთ, მარცხენა მხარეს მოათავსეთ სამი ალამის ობიექტი;
4. ალმების გასწვრივ მარჯვენა მხარეს მოათავსეთ ჭდე;
5. ჭდის ქვემოთ მოათავსეთ ბრძანებითი ღილაკები.

ფორმას დაარქვით Check.frm, პროექტს კი - Almebi.vbp. და შეინახეთ;

ფორმა მიიღებს შემდეგ სახეს, იხ. სურ. 7. 13.



სურ. 7. 13. ალმების ელემენტების მოთავსება ფორმაზე

გადადიოთ თვისებების ფანჯარაში, შეცვალეთ ყოველი ობიექტის თვისება №7. 4 ცხრილის გამოყენებით.

ცხრილი № 7. 4.

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
Form	Name	frmCheck
	Caption	Almebis magaliti
TextBox	Name	txtDisplay
	Font	AcadNusx
	Text	საქართველოს ტექნიკური უნივერსიტეტი
CheckBox	Name	chkBold

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
	Font	AcadNusx
	Caption	მუქი
	Value	False
CheckBox	Name	chkUnderLine
	Font	AcadNusx
	Caption	ხაზგასმული
	Value	False
CheckBox	Name	chkItalic
	Font	AcadNusx
	Caption	დახრილი
	Value	
Label	Name	lblEnter
	Font	AcadNusx
	Caption	დააყენეთ ალამი რიგრიგობით და დააკვირდით ტექსტის ცვლილებას
CommandButto	Name	
	Font	
	Caption	

თვისებების შეცვლის შემდეგ გადით ფორმის დიზაინერში, მაჩვენებლით ორჯერ დააწკაპუნეთ ფორმის ნებისმიერ თავისუფალ ადგილზე, რის შედეგადაც ეკრანზე გამოვა კოდების ფანჯარა. გაიტანეთ კურსორი ცარიელი პროცედურის გარეთ და ჩაწერეთ მასში პროცედურების შემდეგი მიმდევრობა:

```
Private Sub chkBold_Click()
```

```
    'ეს კოდი ამუშავდება მაშინ, როდესაც ალამი დაისმება ან მოიხსნება
```

```
    If chkBold.Value = 1 Then 'tu alami dasmulia
```

```
        txtDisplay.FontBold = True
```

```
    Else 'თუ ალამი დასმული არ არის
```

```
        txtDisplay.FontBold = False
```

```
    End If
```

```
End Sub
```

```
Private Sub chkUnderLine_Click()
```

```
    'ეს კოდი ამუშავდება მაშინ როდესაც ალამი ჩაირთვება ან გამოირთვება
```

```
    If chkUnderLine.Value = 1 Then
```

```
        txtDisplay.FontUnderline = True 'თუ ალამი დასმულია
```

```
    Else 'თუ ალამი დასმული არ არის
```

```
        txtDisplay.FontUnderline = False
```

```
    End If
```

```
End Sub
```

```
Private Sub chkItalic_Click()
```

```
'ეს კოდი ამუშავდება მაშინ, როდესაც ალამი დაისმება ან მოიხსნება
```

```
    If chkItalic.Value = 1 Then 'თუ ალამი დასმულია
```

```
        txtDisplay.FontItalic = True
```

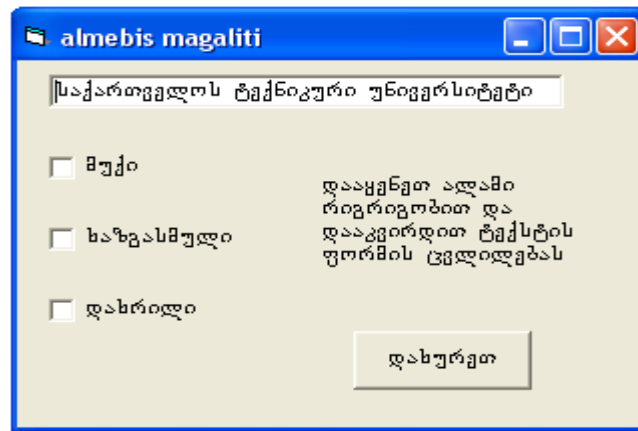
```
    Else 'თუ ალამი დასმული არ არის
```

```
        txtDisplay.FontItalic = False
```

```
    End If
```

```
End Sub
```

Private Sub cmdClose_Click()
 Unload Me ' ფორმის დახურვა
 End Sub



სურ. 7. 14. ალმების გამოყენების პროექტის შედეგი

პროექტის შესრულების შედეგად მივიღებთ 7.14 სურათზე გამოსახულ დიალოგურ ფანჯარას.

სია

სია იდეალური საშუალებაა რამდენიმე ჩამონათვალთან სამუშაოდ. მომხმარებელს შეუძლია დაათვალიეროს სია და დასამუშავებლად აირჩიოს ერთი ან რამდენიმე სტრიქონი. თვისება Columns (სვეტები) შექმნის რამდენიმე სვეტისაგან შემდგარ სიას, მაგრამ VB-ში მოუხერხებელია ასეთ სიასთან მუშაობა. თვისება List (სია) იძლევა ან დააბრუნებს სიაში შემავალი სტრიქონის შინაარსს, რისთვისაც საჭიროა სტრიქონის ნომრის მითითება.

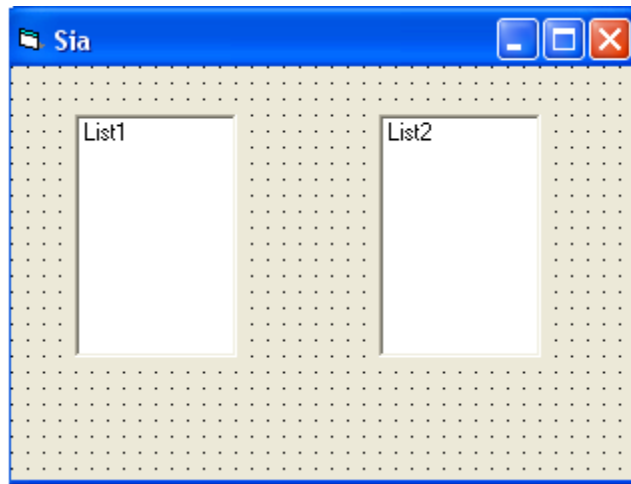
სიის მეთოდებიდან აღსანიშნავია სამი: AddItem (სტრიქონის დამატება), Clear (სიის გაწმენდა) და RemoveItem (სტრიქონის ამოღება). განვიხილოთ სიასთან მუშაობის ნიმუში.

1. შექმენით ახალი პროექტი ბრძანებით File→New Project და პროექტის ოსტატის ფანჯარაში აირჩიეთ ნიშნაკი Standard EXE;
2. მართვის ელემენტების პანელიდან აირჩიეთ ელემენტი List (სია) და ფორმის დიზაინერზე შექმენით სიის ორი ველი;
3. თვისებების ფანჯარაში შეცვალეთ თითოეული სიის ველის თვისება ცხრილი №7.5-ის შესაბამისად;

ცხრილი №7. 5

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
Form	Name	frmSia
	Caption	Sia
List	Name	lstList1
	Font	AcadNusx
List	Name	lstList2
	Font	AcadNusx

დაპროექტებული ფორმის სახე იხ. სურ. 7. 15.



სურ. 7. 15. სიის ელემენტების მოთავსება ფორმაზე

4. თავისი მაჩვენებლით ორჯერ დააწკაპუნეთ ფორმაზე, გაიხსნება პროგრამული კოდების ფანჯარა და ჩაწერეთ შემდეგი ფრაგმენტი:

```
Private Sub Form_Load()
    List1.AddItem "ქანები"
    List1.AddItem "ჭანჭიკები"
    List1.AddItem "ღურსმნები"
    List1.AddItem "კავეები"
    List1.AddItem "ჩაქუნები"
    List1.AddItem "ხერხები"
    List1.AddItem "ქლიბები"
    List1.AddItem "ხუმფარები"
    List1.AddItem "შალაშინები"
End Sub
```

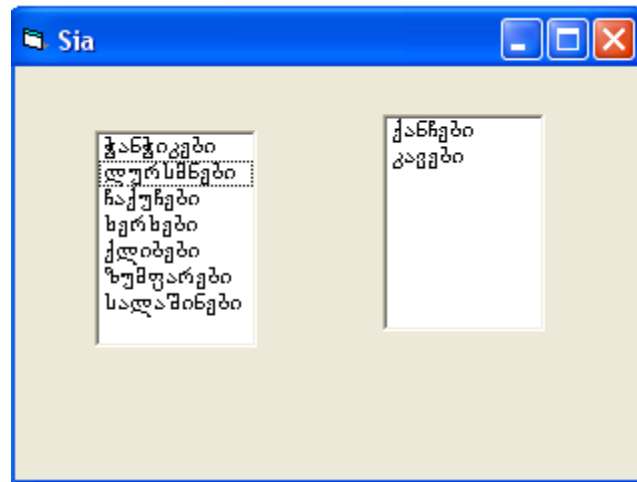
5. გადადით ფორმის დიზაინერზე და პირველი სიის ველზე ორჯერ დააწკაპუნეთ მაჩვენებლით, გაიხსნება კოდების ფანჯარა და კურსორი იქნება DoubleClick პროცედურის შიგნით, მასში ჩაამატეთ შემდეგი ფრაგმენტი:

```
Private Sub List1_DblClick()
    ' სტრიქონის ჩამატება სხვა სიაში
    List2.AddItem List1.Text
    ' მოცემულ სიაში სტრიქონის წაშლა
    List1.RemoveItem List1.ListIndex
End Sub
```

ანალოგიურად, მეორე სიაში ჩაამატეთ შემდეგი ფრაგმენტი:

```
Private Sub List2_DblClick()
    ' სტრიქონის ჩამატება სხვა სიაში
    List1.AddItem List2.Text
    ' მოცემულ სიაში სტრიქონის წაშლა
    List2.RemoveItem List2.ListIndex
End Sub
```

6. თვისებების ფანჯარაში თითოეულ სიის ველს ჩაურთეთ ქართული ფონტი და გაუშვით პროგრამა შესრულებაზე ბრძანებით Start, ეკრანზე მივიღებთ 7. 16 სურათზე გამოსახულ დიალოგურ ფანჯარას.



სურ. 7. 16. სიის პროექტის შედეგი

პირველი ველის ნებისმიერ სტრიქონზე მანვენებლის ორჯერ დაწკაპუნებით, ეს სტრიქონი გადადის სიის მეორე ველში. თუ იგივეს გაიმეორებთ სიის მეორე ველზე სტრიქონი გადავა სიის პირველ ველში (შეამოწმე თვითონ). მეთოდი AddItem წინ უნდა უსწრებდეს მეთოდს RemoveItem.

კომბინირებული ველი

სიის ველში მომხმარებელს უწევს სტრიქონებით მუშაობა, რის გამოც პირდაპირი რედაქტირება შეუძლებელია. ასეთ შემთხვევაში სარგებლობენ კომბინირებული ველით, რომელიც აერთიანებს ტექსტური ველისა და სიის შესაძლებლობებს. კომბინირებული ველი საშუალებას იძლევა სიიდან აირჩიოთ წინასწარ განსაზღვრული სტრიქონი, ან შეიტანოთ ის მნიშვნელობა, რომელიც არ არის სიაში. კომბინირებული ველის გახსნა შეიძლება, რის გამოც ფორმაზე იგი იკავებს უფრო ნაკლებ ადგილს, ვიდრე ჩვეულებრივი სია. არსებობს სამგვარი კომბინირებული ველი: განშლადი კომბინირებული, მარტივი კომბინირებული და განშლადი სია.

□ განშლადი კომბინირებული ველი სიის ველის მსგავსია, იმ განსხვავებით რომ ველს მარჯვნივ აქვს ისრიანი ღილაკი. ღილაკზე დაჭერით ტექსტური ველის ქვეშ ჩამოიშლება სია და მომხმარებელს შეუძლია აირჩიოს სიის სტრიქონი ან შეიტანოს ველში საკუთარი ტექსტი;

□ მარტივი კომბინირებული ველი არის ზემოთ აღწერილი ტექსტური ველის სახესხვაობა. განსხვავება ისაა, რომ სია მუდამ გაშლილი რჩება, ამ ვარიანტს ირჩევთ მაშინ, როდესაც ფორმაზე საკმარისი ადგილია;

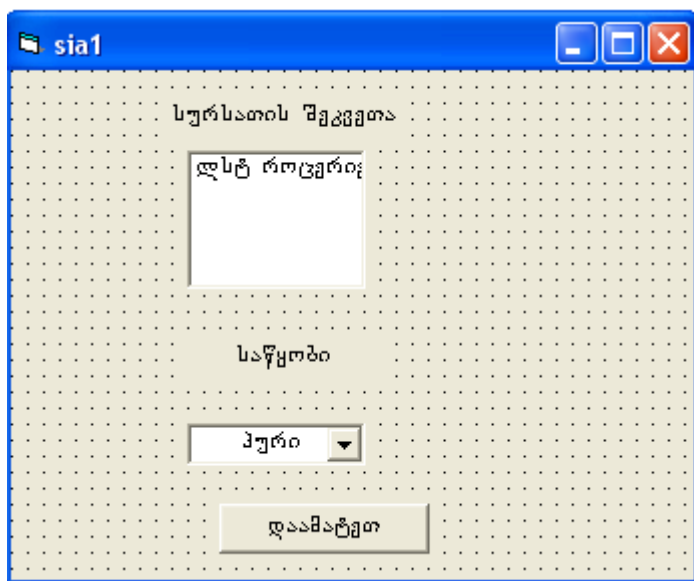
□ განშლადი სია, გარეგნულად განშლადი კომბინირებული ველის მსგავსია, მომხმარებელს შეუძლია მხოლოდ სიაში შემავალი სტრიქონებით სარგებლობა. ამ სტილის უპირატესობა ისაა, რომ იგი ნაკლებ ადგილს იკავებს.

კომბინირებული ველი განისაზღვრება Style თვისებით. თვისებებით, მოვლენის პროცედურებით და მეთოდებით კომბინირებული ველი ჰგავს სიის ველს. თვისება - Text კომბინირებულ ველში სხვაგვარად მუშაობს. ამ თვისების მნიშვნელობა შეიძლება მივცეთ შესრულებისას - ტექსტი შეიტანება იმ შემთხვევაშიც, როდესაც სიაში სტრიქონი არ არის. მონაცემთა ბაზაში მუშაობის დროს კომბინირებულ ველს აქვს დამატებითი თვისებები.

განვიხილოთ მაგალითი, სასურსათო მაღაზიისათვის შევადგინოთ შეკვეთა. მაგალითის მიზანია აჩვენოს კომბინირებული ველიდან სტრიქონის აღება და სიის ველში ჩასმა.

1. შექმენით ახალი პროექტი, ფორმაზე მოათავსეთ სიის ველი;
2. შექმენით კომბინირებული ველი;
3. მოათავსეთ ფორმაზე წარწერა სიის თავში;
4. მოათავსეთ ფორმაზე წარწერა კომბინირებული ველის თავში;

5. ფორმის ქვემოთ მოათავსეთ ბრძანების ღილაკი;
6. თვისებების ფანჯარაში შეცვალეთ თვისებები №7.6 ცხრილის შესაბამისად.



სურ. 7. 17. სიის და კომბინირებული ველის მოთავსება ფორმაზე
ცხრილი № 7. 6.

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
Form	Name	frmCombo
	Caption	Sia1
List	Name	lstGroceries
	Font	AcadNusx
ComboBox	Name	cboProducts
	Font	AcadNusx
	Text	puri
Label	Name	lblLabel1
	Caption	sursaTis SekveTa
	Font	AcadNusx
Label	Name	lblLabel2
	Caption	sawyobi
	Font	AcadNusx
CommandBox	Name	cmdAdd
	Caption	&daamateT
	Font	AcadNusx

თავის მაჩვენებლით დააწკაპუნეთ ფორმაზე, ეკრანზე გამოვა კოდების ფანჯარა, პროცედურაში მოათავსეთ შემდეგი ფრაგმენტი:

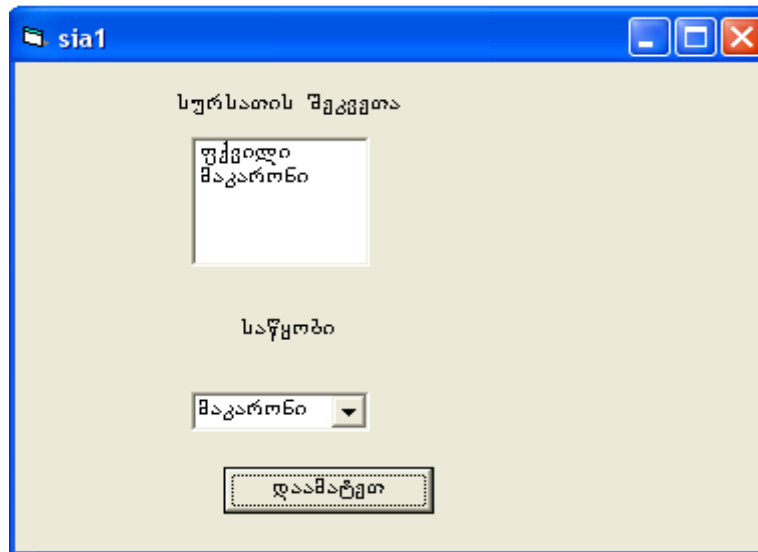
```
Private Sub Form_Load()
    ' სიის გაწმენდა
    lstGroceries.Clear
    დაამატეთ სტრიქონი კომბინირებულ ველში
    cboProducts.AddItem "პური"
    cboProducts.AddItem "ფქვილი"
    cboProducts.AddItem "ხორცი"
    cboProducts.AddItem "მკარონი"
    cboProducts.AddItem "შოკოლადი"
    cboProducts.AddItem "ლიმონათი"

End Sub
```

კოდების ფანჯრის მარცხენა ნაწილიდან აირჩიეთ cmdAdd და პროცედურის ტანში დაამატეთ ფრაგმენტი:

```
Private Sub cmdAdd_Click()
    lstGroceries.AddItem cboProducts.Text
    'cmdAdd.Value = True
End Sub
```

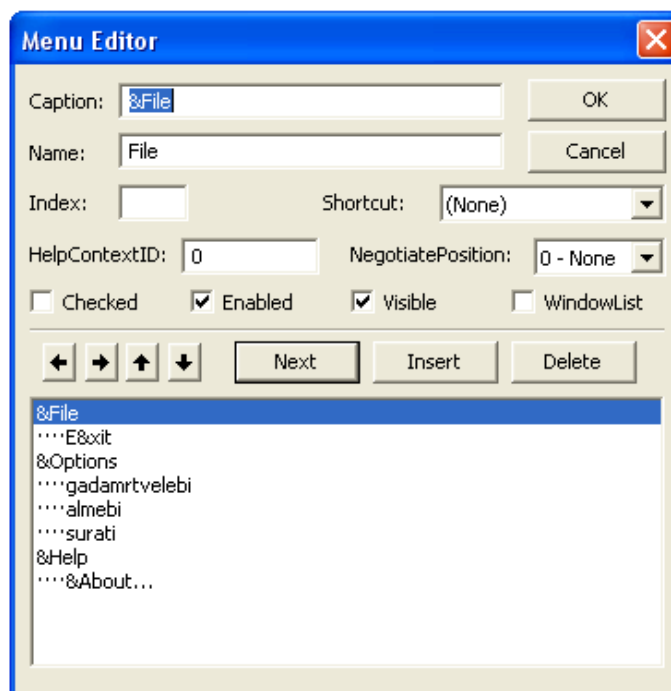
პროგრამული კოდები Start ბრძანებით გაუშვით შესრულებაზე, ეკრანზე მივიღებთ 7. 18 სურათზე გამოსახულ დიალოგურ ფანჯარას.



სურ. 7.18. სიის და კომბინირებული ველის პროექტის შედეგი

მენიუს შექმნა

შექმენით ახალი პროექტი, ძირითადი მენიუდან Tools→Menu Editor ბრძანებით გამოიძახეთ მენიუს რედაქტირების ფანჯარა. ფანჯრის Name (სახელი) და Caption (წარწერა) ველში ჩაწერეთ სახელისა და წარწერის მნიშვნელობა, ხოლო ქვემენიუს ბრძანებების ჩასაწერად გამოიყენეთ ისრები. ყოველი მომდევნო ბრძანების ჩასაწერად გამოიყენეთ Next (მომდევნო) კლავიში. იხ. სურ. 7. 19.



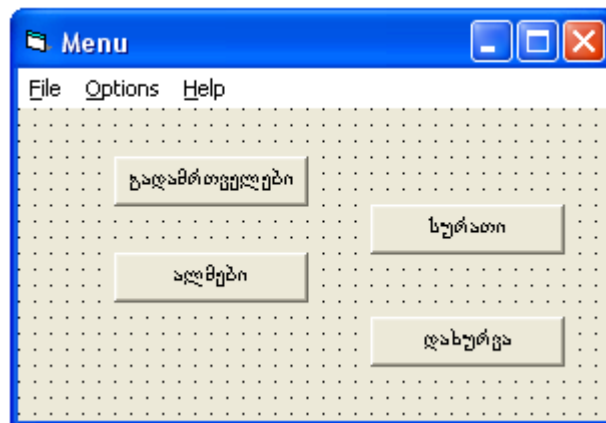
სურ. 7. 19. Menu Editor-ის ფანჯარა

ფორმაზე მოათავსეთ ოთხი ბრძანებითი კლავიში, თვისებების ფანჯარაში შეიტანეთ ცვლილებები №7. 7 ცხრილის მიხედვით.

ცხრილი № 7. 7.

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
Form	Name	frmMain
	Caption	Menu
	Font	AcadNusx
CommandButton	Name	cmdgadamrtvelebi
	Caption	გადამრთველები
CommandButton	Name	cmdalmebi
	Caption	აღმები
CommandButton	Name	cmdSurati
	Caption	სურათი
CommandButton	Name	cmdExit
	Caption	დასურვა

თვისებების შეცვლის შემდეგ ფორმა მიიღებს 7. 20 სურათზე გამოსახული დიალოგური ფანჯრის ფორმას.



სურ. 7. 20. მენიუს ელემენტების მოთავსება ფორმაზე

ნახატები

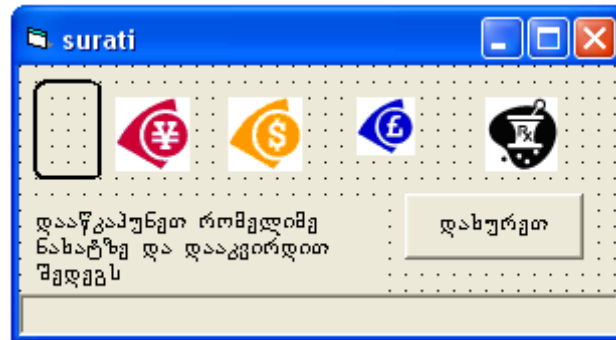
აიღეთ იმავე პროექტში ახალი ფორმა რისთვისაც პროექტის ფანჯარაში მონიშნეთ ფორმის ნიშნაკი, მარჯვენა კლავიშით გამოიტანეთ მენიუ და აირჩიეთ Add→Form ბრძანება, Add Form დიალოგურ ფანჯარაში მონიშნეთ Form ნიშნაკი და მაჩვენებლით დააწკაპუნეთ Open ღილაკზე. ფორმა დაემატება პროექტს. ფორმის ღიზაინერზე მოათავსეთ Shape (ფიგურა) ობიექტი, სასურველი რამდენიმე ნახატი, CommandButton (ბრძანების ღილაკი), Label (ტეკსტი) და PictureBox (გრაფიკული ველი).

ცხრილი № 7.8

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
Form	Name	picStatus
	Caption	surati
	Font	AcadNusx
Shape	Name	shpCard
	Shape	4-Rounded Rectangle
Image	Name	imgImage1
Image	Name	imgImage2
Image	Name	imgImage3
Image	Name	imgImage4

ობიექტის სახელი	თვისებები	თვისების მნიშვნელობა
Label	Name	lblEnter
	Caption	დააწკაპუნეთ რომელიმე ნახატზე და დააკვირდით შედეგს
CommandButton	Name	cmdClose
	Caption	დახურეთ
PictureBox	Name	picStatus
	Align	2-Align Bottom

თვისებების დაყენების შემდეგ ფორმა მიიღებს 7. 21 სურათზე გამოსახულ დიალოგური ფანჯრის ფორმას.



სურ. 7. 21. ნახატის ელემენტების მოთავსება ფორმაზე

ფორმაზე დაწკაპუნებით გადით კოდების ფანჯარაში და ჩაწერეთ შემდეგი კოდების ფრაგმენტი:

```
Private Sub Form_Load()
    shpCard.Left = -500
End Sub

Private Sub imgImage1_Click()
    shpCard.Left = imgImage1.Left
    picStatus.Cls
    picStatus.Print "moniSnulia: naxati1 ";
End Sub

Private Sub imgImage2_Click()
    shpCard.Left = imgImage2.Left
    picStatus.Cls
    picStatus.Print "moniSnulia: naxati2";
End Sub

Private Sub imgImage3_Click()
    shpCard.Left = imgImage3.Left
    picStatus.Cls
    picStatus.Print "moniSnulia: naxati3 "
End Sub

Private Sub imgImage4_Click()
    shpCard.Left = imgImage4.Left
    picStatus.Cls
    picStatus.Print "moniSnulia: naxati4"
```

```
End Sub
```

```
Private Sub cmdClose_Click()  
    Unload Me  
End Sub
```

პროექტის ფანჯარაში დაამატეთ უკვე შექმნილი ფორმები: frmGadamrtvelebi და frmCheck, რისთვისაც პროექტის ფანჯარაში მონიშნეთ form ნიშნაკი, მარჯვენა კლავიშით გამოიტანეთ მენიუ, აირჩიეთ Add→Form ბრძანება, დიალოგურ ფანჯარაში გადადით Existing ჩანაცვლებაზე, აირჩიეთ ზემოთ აღნიშნული ფორმები და დააწკაპუნეთ Open ლილაკზე. გაააქტიურეთ frmMain ფორმა, გადადით კოდების ფანჯარაში და ჩაუწერეთ შემდეგი კოდი:

```
Private Sub About_Click()  
    frmAbout.Show  
End Sub
```

```
Private Sub almebi_Click()  
    frmChek.Show  
End Sub
```

```
Private Sub cmdalmebi_Click()  
    almebi_Click  
End Sub
```

```
Private Sub cmdExit_Click()  
    End  
End Sub
```

```
Private Sub cmdgadamrtvelebi_Click()  
    gadamrtvelebi_Click  
End Sub
```

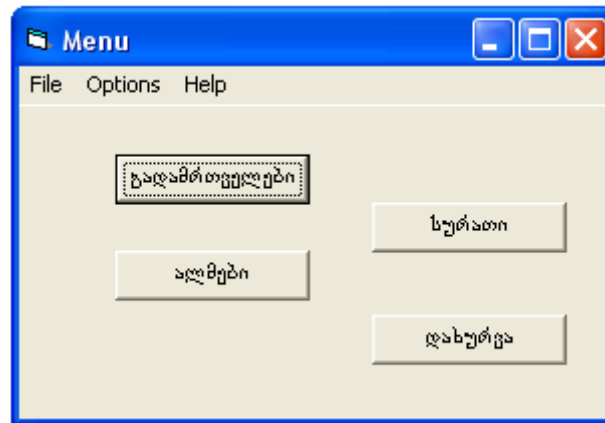
```
Private Sub cmdSurati_Click()  
    surati_Click  
End Sub
```

```
Private Sub Exit_Click()  
    cmdExit_Click  
End Sub
```

```
Private Sub gadamrtvelebi_Click()  
    frmGadamrtvelebi.Show  
End Sub
```

```
Private Sub surati_Click()  
    picStatus.Show  
End Sub
```

შესრულების შედეგად მიიღება 7.22 სურათზე გამოსახული დიალოგური ფანჯარა.



სურ. 7. 22. მენიუს საშუალებით შესრულებული მაგალითი დააწკაპუნეთ ღილაკებზე და გამოიძახეთ სხვადასხვა ფორმა.

VIII თავი

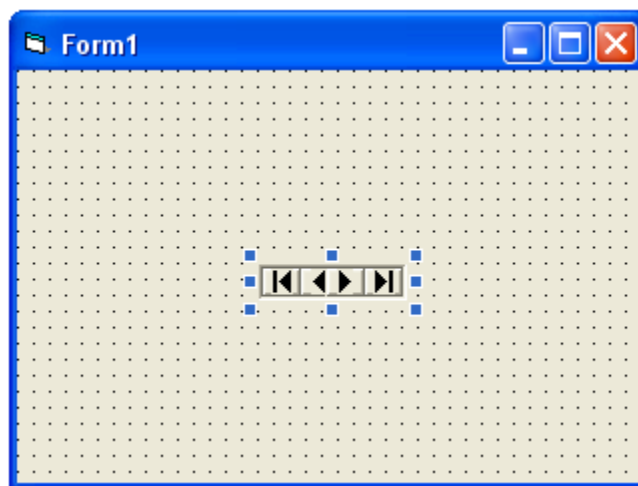
მონაცემთა ბაზასთან მუშაობა

8. 1. ახალი პროექტის გაშვება Visual Basic-ში

იმისათვის რომ შექმნათ ახალი პროექტი, თუ Visual Basic-ი უკვე ჩატვირთულია თქვენს კომპიუტერზე, მენიუდან აირჩიეთ File→New Project (ფაილი→ახალი პროექტი) ბრძანება. თუ არ არის ჩატვირთული, მაშინ ჩატვირთეთ, აირჩიეთ Standard EXE და დიალოგურ ფანჯარაში დააწკაპუნეთ Open (გახსენი) ბრძანებაზე. ახლა უკვე ყველაფერი მზადაა შესატანი ეკრანული ფორმების შესაქმნელად.

მონაცემთა ბაზის მართვის ელემენტის დამატება

მართვის ელემენტების პანელზე მონახეთ Data (მონაცემი) ელემენტი და ორჯერ დააწკაპუნეთ მასზე. ფორმის ცენტრში გაჩნდება მონაცემთა მართვის ელემენტი, რის შემდეგაც ფორმა მზადაა მონაცემთა ცხრილის გასახსნელად. ახლა თქვენგან მოითხოვება მართვის ელემენტისათვის რამდენიმე თვისების დაყენება. იხ. სურ. 2. 1.



სურ. 8. 1. მონაცემთა მართვის ელემენტი ფორმაზე პირველი მოთავსების შემდეგ

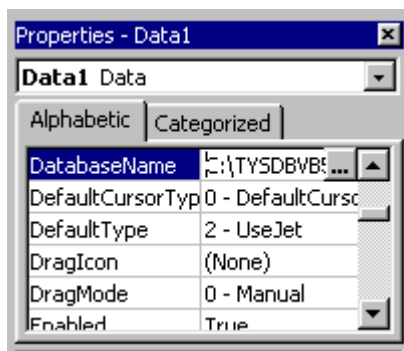
DatabaseName (მონაცემთა ბაზის სახელის) და RecordSource (ცხრილის ჩანაწერის) თვისებების დაყენება

მონაცემთა ბაზასთან მონაცემთა მართვის ელემენტის დასაკავშირებლად უნდა მიანიჭოთ შემდეგი თვისებები;

DataBaseName (არჩეული მონაცემთა ბაზის სახელი);

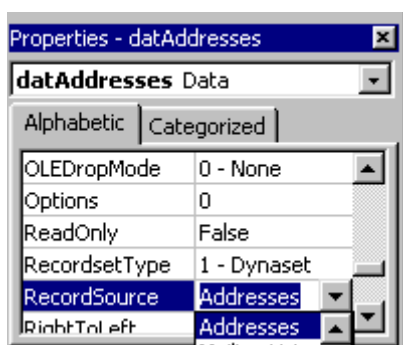
RecordSource (არჩეული მონაცემთა ცხრილი).

ისარგებლეთ მონაცემთა ბაზით BOOKS5.MDB, რომელის მისამართია C:\TYSDVB5\Source\Data. იმისათვის რომ მიანიჭოთ მართვის ელემენტის თვისება DataBaseName (მონაცემთა ბაზის სახელი), ჯერ მონიშნეთ ეს ელემენტი ფორმაზე, გადადით Properties (თვისებების) ფანჯარაში, სადაც გამოჩნდება მონაცემთა მართვის ელემენტის თვისება DataBaseName (მონაცემთა ბაზის სახელი), დააწკაპუნეთ მასზე, მარჯვნივ გამოჩნდება სამი წერტილი (. . .), დააწკაპუნეთ მაჩვენებელი ამ სამ წერტილზე, ეკრანზე გამოჩნდება ფაილის გახსნის სტანდარტული დიალოგური ფანჯარა, აირჩიეთ TYSDVB5\Source\Data\Books5.mdb, რის შემდეგაც ეკრანზე აისახება შემდეგი, იხ. სურ. 8. 2.



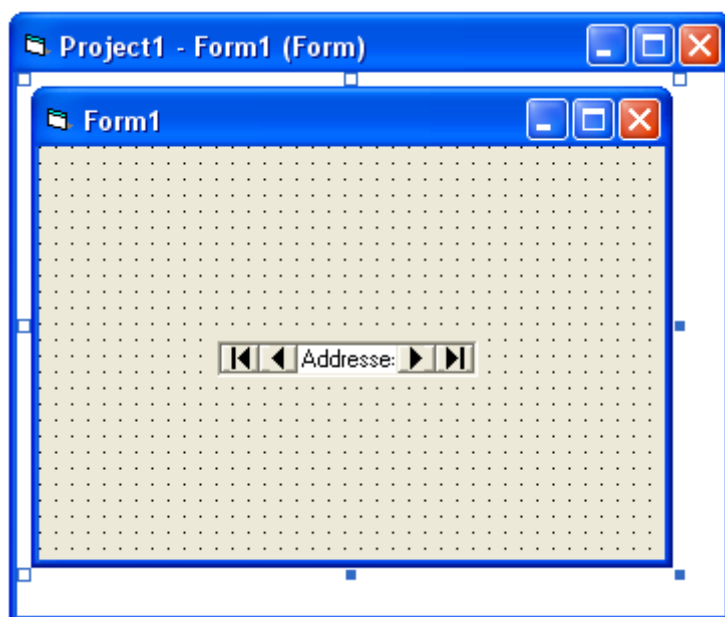
სურ. 8. 2. თვისებების ფანჯარაში მონაცემთა ბაზის მისამართის დაყენება

მონაცემთა ბაზის არჩევის შემდეგ უნდა აირჩიოთ მის შიგნით თქვენთვის საჭირო ცხრილი, რისთვისაც RecordSource თვისებაში მონახეთ Addresses ცხრილი და დააწკაპუნეთ მასზე, რის შედეგადაც ცხრილის სახელი ავტომატურად ჩაჯდება თვისებების ფანჯარაში, იხ. სურ. 8. 3.



სურ. 8. 3. RecordSource - თვისების დაყენება Addresses ცხრილში

Caption (წარწერის) და Name (სახელის) თვისებების დაყენება



სურ. 8. 4. მონაცემთა მართვის ელემენტზე Caption თვისების მნიშვნელობის გამოჩენა

მონაცემთა მართვის ელემენტს დაუყენეთ კიდევ ორი თვისება. მართალია, ეს თვისებები აუცილებელი არ არის, მაგრამ მათი დაყენება ითვლება პროგრამირების კარგ სტილად, რადგანაც პროგრამის წაკითხვა იოლდება. ეს თვისებებია:

Caption - მონაცემთა მართვის ელემენტზე გამოტანილი ტექსტი;

Name - პროგრამაში მონაცემთა მართვის ელემენტის სახელი.

თვისებაში Caption იწერება ტექსტი, რომელიც უნდა აგმოჩნდეს მონაცემის მართვის ელემენტზე ისრებს შორის.

თვისებაში Name (სახელი) ჩაიწერება ტექსტი, რომელიც უნდა გამოიყენოს პროგრამისტმა პროგრამის დაწერისას. მართალია, ეს თვისება არ გამოჩნდება, მაგრამ მას უნდა ჰქონდეს საერთო Caption თვისებასთან. ახლა მონაცემის მართვის ელემენტის მონიშნული კვადრატებიდან ერთ-ერთზე დააყენეთ მახვენებელი და ჩარჩო გაჭიმეთ სიგანეზე, რომ გამოჩნდეს თქვენს მიერ დაწერილი ტექსტი. თქვენი ფორმა უნდა გამოიყურებოდეს ისე, როგორც ეს ნაჩვენებია სურათზე 8. 4.

პროექტის შენახვა

ახლა უნდა შეინახოთ თქვენი ნამუშევარი დისკზე. ამისათვის აირჩიეთ მენიუდან File→Save Project (ფაილი →პროექტის შენახვა) ბრძანება. გამოვა ფაილის შენახვის დიალოგური ფანჯარა, ფაილის სახელის ველში შეიტანეთ Datcontrol.frm ფორმისათვის, პროექტისათვის კი Datcontrol.vbp. გირჩევთ თქვენი ნამუშევრის ხშირად შენახვას.

8. 2. დაკავშირებული შეტანის მართვის ელემენტების დამატება

მას შემდეგ, რაც მონაცემთა მართვის ელემენტის მეშვეობით ფორმა დააკავშირეთ მონაცემთა ბაზასთან და აირჩიეთ ცხრილი, ფორმაზე უნდა მოათავსოთ შეტანის მართვის ელემენტი. Visual Basic გთავაზობთ მართვის ისეთ ელემენტებს, რომლებიც შეგიძლიათ უშუალოდ დააკავშიროთ მონაცემთა ცხრილებთან, რომლებშიც გასურთ შედგენა. ამისათვის საკმარისია ფორმაზე მოათავსოთ რამდენიმე ასეთი ობიექტი და ჩაამატოთ ისინი უკვე არსებულ მონაცემთა მართვის ელემენტთან.

დაამატეთ დაკავშირებული მართვის პირველი ელემენტი ცხრილ Addresses-ის შესატან ფორმას. რისთვისაც იარაღების ჯგუფში მონახეთ TextBox control (ტექსტური ველი) ღილაკი და მასზე ორჯერ დააწკაპუნეთ. შედეგად ფორმის შუა ადგილას გაჩნდება ტექსტური ველი. გადაათრიეთ იგი თავის საშუალებით სასურველ ადგილას, შეცვალეთ ზომები და დაუყენეთ თვისებები პროექტის თვისებების ფანჯარაში, კერძოდ მართვის ამ ელემენტის Name (სახელი) თვისებაში აირჩიეთ txtAddresses. ამ ტექსტური ველის აღწერისათვის დაამატეთ Label (ჭდე), რისთვისაც იარაღების ჯგუფში მონახეთ ობიექტი Label control და ორჯერ დააწკაპუნეთ მასზე. დაუყენეთ Name თვისებაში, მნიშვნელობა lblAddresses, ხოლო Caption თვისებაში - Addresses.



მართვის ელემენტს, რომელიც დაკავშირებულია მონაცემთა რომელიმე წყაროსთან, ეწოდება დაკავშირებული შეტანის მართვის ელემენტი.

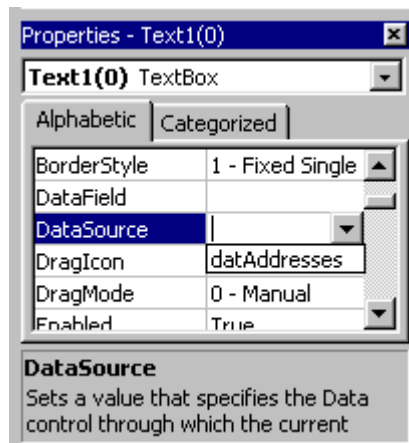
თვისებების Data Source და Data Field-ის დაყენება

ტექსტური ველი რომ ურთიერთქმედებაში იყოს მონაცემთა მართვის ელემენტთან უნდა დაუყენოთ ორი თვისება:

1. DataSource (მონაცემთა ბაზის სახელი);
2. DataField (ცხრილის ველის სახელი).

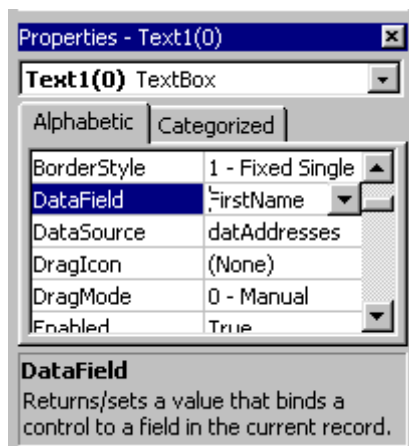
ფორმაზე მონიშნეთ ტექსტური ველი, შემდეგ თვისებებში DataSource მონახეთ მონაცემთა მართვის ელემენტის სახელი და დააწკაპუნეთ მასზე, გამოვა მართვის ელემენტების ჩამონათვალი, დაუყენეთ მნიშვნელობა datAddresses, იხ. სურ. 8. 5.

ამ ორი თვისების დაყენებით ცხრილის ველებსა და მართვის ელემენტს შორის მყარდება კავშირი. მართვის ელემენტში ნებისმიერი მონაცემის შეტანა და მათი ასახვა უშუალოდ დაკავშირებულია თქვენს მიერ არჩეულ ცხრილთან / ველთან.



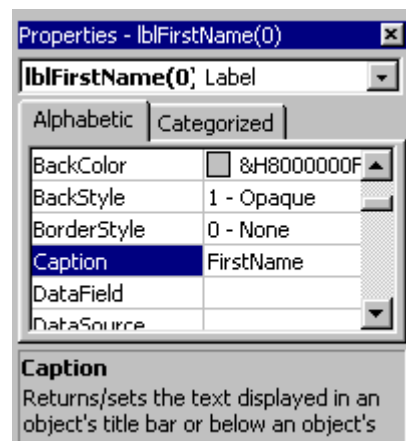
სურ. 8. 5. დაკავშირებულ ტექსტურ ველზე თვისება DataSource-ს დაყენება

Data Field-ის დაყენება, მართვის ელემენტს აკავშირებს ცხრილის კონკრეტულ ველთან. თვისებებში გახსენით DataField (ცხრილის ველი), გახსენით სია, გამოჩნდება ველები და აირჩიეთ FirstName, იხ. სურ. 8.6.



სურ. 8. 6. დაკავშირებული ტექსტური ველის თვისებისათვის DataField მნიშვნელობის არჩევა

ახლა მონიშნეთ ჭდე, გადადით თვისებებში, მონახეთ Name და დაუყენეთ მნიშვნელობა lblFirstName, ხოლო თვისებაში Caption დაუყენეთ მნიშვნელობა FirstName.



სურ. 8. 7.

ა) ჭდის სახელის დაყენება

ბ) ჭდეზე ტექსტის ასახვა

იგივე გაიმეორეთ დანარჩენ ტექსტურ ველებზე და დაუკავშირეთ ველები LastName და City. შემდეგ კი იარაღების სტანდარტულ სტრიქონში მონახეთ ღილაკი Run და დააწკაპუნეთ ერთჯერ. შეგიძლიათ დაამთავროთ ღილაკზე დაწკაპუნებით, რის შედეგადაც დაბრუნდებით თქვენს ფორმაზე. შეგიძლიათ ცხრილში მოახდინოთ ნებისმიერი ცვლილება და შესაბამისი შესწორება შეიტანოთ როგორც ფორმაში, ისე მის თვისებებში

სურ. 8. 8. ცხრილში Addresses მონაცემების შესატანად მომზადებული ფორმა

*Add (დამატება) და Delete (წაშლა)
ბრძანების ღილაკების დამატება*

ჯერ მოათავსეთ ფორმაზე ბრძანების ღილაკები, შემდეგ შეიტანეთ თვისებებში სათანადო მნიშვნელობები, გახსენით ქვეპროგრამა ამ ღილაკზე ორჯერ დაწკაპუნებით და ჩაწერეთ შესაბამისი ბრძანება.

*ფორმაზე ბრძანების ღილაკების თვისებები
ცხრილი 8.1.*

Name	Caption
cmdAdd	&Add
cmdDelete	&Delete

DatAddressess.Recordset.AddNew - დამატეთ ახალი ჩანაწერი ცხრილში, რისთვისაც ორჯერ

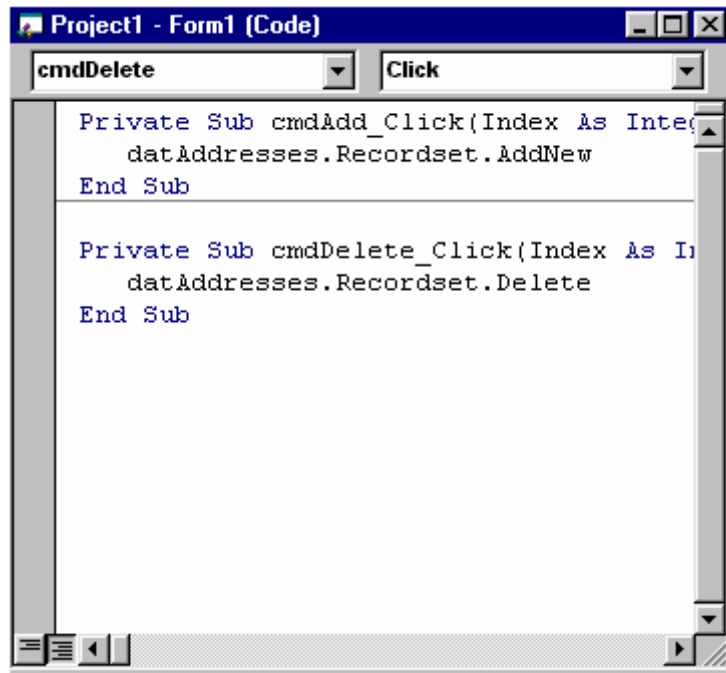
სურ. 8. 9. ფორმის სახე ბრძანების ღილაკების დამატების შემდეგ

დააწკაპუნეთ ბრძანებით ღილაკზე Add (დამატება). გამოვა კოდების ფანჯარა, ცარიელ პროცედურაში შეიტანეთ ოპერატორი `datAddresses.Recordset.AddNew` და პროცედურას ექნება შემდეგი სახე:

```
Private Sub cmdAdd_Click()  
    datAddresses.Recordset.AddNew  
End Sub.
```

1. Visual Basic-ი ავტომატურად ახდენს სტრიქონის `Sub _ End Sub` გენერირებას ყოველი ახლად შექმნილი პროცედურისათვის. პროგრამული კოდი უნდა ჩაწეროთ ამ ორ სტრიქონს შორის.

2. ბრძანების ღილაკის თვისებაში `Captions` ასო, რომელიც დგას & ამპერსანდის შემდეგ იქნება ხაზგასმული, რაც იმას ნიშნავს, რომ ეს არის სწრაფი, ანუ ცხელი კლავიში.



სურ. 8. 10. კოდების ფანჯარა ბრძანების ღილაკების გასააქტიურებელი პროცედურებით

შეინახეთ თქვენი პროექტი და შესრულების შედეგად ეკრანზე მიიღება ფორმა 8. 11 სურათზე გამოსახული დიალოგური ფანჯარა.

სურ. 8. 11. შექმნილი ფორმის საბოლოო სახე

IX თავი

მონაცემთა ბაზის შემწეა

9.1. რელაციურ მონაცემთა ბაზა

რელაციურ მონაცემთა ბაზა არის მონაცემების ერთობლიობა, მის ელემენტებს შორის კავშირის მითითებით.

დაკავშირებული ელემენტების ერთობლიობის ასაგებად საჭიროა სამი ძირითადი ბლოკი:

- ველები (მონაცემთა ცხრილის სვეტები);
- ჩანაწერები (მონაცემთა ცხრილის სტრიქონები);
- ცხრილები (მონაცემთა ცხრილები).

ველი. რელაციური ბაზის საწყისი ასაგები ბლოკია ველი. იგი შეიცავს მონაცემთა ბაზის უმცირეს ელემენტს. თითოეული ველი შეიცავს მხოლოდ მონაცემების ერთ ელემენტს. მაგალითად, ველი რომლის სახელია მომხმარებლის სახელი, ან მიმდინარე ანგარიშის ბალანსი, მომხმარებლის მისამართი და ა.შ. ხშირად მონაცემთა ბაზაში ინახება ცალკე მომხმარებლის სახელი და ცალკე მომხმარებლის გვარი. როდესაც იწყებთ მონაცემთა ბაზის შექმნას, კარგად იფიქრეთ მონაცემთა არჩევის სხვადასვა ხერხებზე. თქვენი მონაცემთა ბაზის ხარისხი და სარგებელიანობა დაფუძნებულია მის მთლიანობაზე და ველების სიმცირეზე.

ჩანაწერი(*სტრიქონი*) არის ერთმანეთთან დაკავშირებული ველების ერთობლიობა. თქვენს მაგალითში, ჩანაწერი შეიძლება შეიცავდეს მომხმარებლის სახელს, გვარს და მიმდინარე ანგარიშის ბალანსს.

მონაცემთა ცხრილის სტრიქონები და სვეტები

ველებისა და ჩანაწერების გაერთიანებით მიიღება რელაციური ბაზის ზოგადი ელემენტი – ცხრილი. ეს ელემენტი შეიცავს ჩანაწერების სიმრავლეს, თითოეული ჩანაწერი კი – ველების სიმრავლეს. მონაცემთა ცხრილის სახელი (მაგ. მომხმარებლის ცხრილი, ან ანგარიში), ისევე როგორც ველები (გვარი, სახელი, ანგარიში) ინფორმაციულია. ასეთი სახელები გეხმარებათ დაიმახსოვროთ ცხრილის ელემენტები და ველები.

თანაფარდობა

ვიცით რომ ჩანაწერი შეიცავს ერთმანეთთან დაკავშირებულ რამდენიმე ველს მონაცემთა ბაზა კი ერთმანეთთან დაკავშირებულ ცხრილებს. ურთიერთდამოკიდებულება რთული მონაცემების შენახვის ხერხია. მაგ. ცხრილი, რომელიც შეიცავს მყიდველების სახელებს (გვარებს), შეიძლება დააკავშიროთ სხვა ცხრილთან, რომელიც შეიცავს დამკვეთების მიერ ნაყიდი საქონლის სახელებს, ეს კი თავის მხრივ შეიძლება დააკავშიროთ თქვენს მიერ ყველა გაყიდული საქონლის სახელებთან. ცხრილებს შორის აზრობრივი დამოკიდებულების დამყარებით შეიძლება შექმნათ და შეინარჩუნოთ მონაცემთა მოქნილი სტრუქტურა, ცხრილების მონაცემებს შორის დამოკიდებულება მყარდება ველების მახვენებლით, ან კვალიფიკატორით.

ველი – კვალიფიკატორი გამოიყენება დამატებითი ინფორმაციის შემცველ ცხრილში ჩანაწერის გადსაგზავნად. იგი აღწერს ერთ-ერთთან დამოკიდებულებას. მაგალითად, დამოკიდებულება მყიდველების ცხრილის ერთი ჩანაწერისა და მომწოდებლის მისამართების ცხრილის ერთ ჩანაწერს შორის.

ველი – მიმთითებელი მიუთითებს იმ ცხრილის ერთ ან რამდენიმე ჩანაწერზე, რომლებიც შეიცავენ მოცემული ჩანაწერის შესახებ ინფორმაციას. ველი – მიმთითებელი აღწერს ე. წ. დამოკიდებულებას ერთი – მრავალთან. მაგალითად, მყიდველზე ერთ ჩანაწერსა და ამ მყიდველის მიერ შეუსრულებელი დამკვეთების ერთ ან რამდენიმე ჩანაწერზე. ურთიერთდამოკიდებულება ერთი-მრავალთან გამოიყენება მთავარი ცხრილის ჩანაწერების დასაკავშირებლად სხვა ცხრილის რამდენიმე ჩანაწერთან. ცხრილებს შორის დამოკიდებულების დასამყარებლად არსებობს სხვადასხვა მიდგომა. ეს თანაფარდობა მყარდება საგასაღებო ველების საშუალებით.

გასაღებიანი ველები

როგორც წესი, მონაცემთა ყოველ ცხრილში ერთი ველი მაინც არის გასაღებიანი ველი. რელაციურ მონაცემთა ბაზაში გასაღებიანი ველი გამოიყენება მონაცემთა განსაზღვრისა და მთლიანობის შესანარჩუნებლად, ასევე ცხრილებს შორის დამოკიდებულების დასამყარებლად. მონაცემთა ცხრილებში გასაღები იქმნება ერთი ან რამდენიმე ველის მითითებით პირველად ან გარე გასაღებად. მონაცემთა ცხრილს შეიძლება ჰქონდეს მხოლოდ ერთი პირველადი გასაღები და რამდენიმე გარე გასაღები. პირველადი გასაღები გამოიყენება მონაცემთა ასახვის მიმდევრობის მართვისათვის. გარე გასაღები კი გამოიყენება მოცემული ცხრილის ველების სხვა ცხრილის ველებთან დასაკავშირებლად.

 გასაღებიანი ველი და ინდექსური ველი ვიზუალურ ბეისიკში ითვლება სინონიმებად (იღენტურია).

პირველადი გასაღებიანი ველი განსაზღვრავს ველში უნიკალურ ჩანაწერს. მაგალითად, გვარი არის პირველადი გასაღებიანი ველი ცხრილში Customer (მყიდველი). ეს ნიშნავს, რომ ამ ცხრილში არ შეიძლება რომელიმე ორ ჩანაწერს ველში LastName (გვარი) ჰქონდეს ერთნაირი მნიშვნელობა. ყოველი ცდა, დაამატოთ ერთზე მეტი ჩანაწერი იმავე პირველადი გასაღებით, მთავრდება შეცდომით.

ცხრილს შეიძლება ჰქონდეს რამდენიმე გარე გასაღებიანი ველი, ან საერთოდ არ ჰქონდეს იგი. ცხრილში Orders (ჩანაწერები), გარე გასაღებიან ველად შეიძლება გამოიყენოთ LastName (გვარი) ველი. ეს იმას ნიშნავს, რომ იგი მიუთითებს რომელიმე გარე გასაღებიან ველზე. ყოველი ცდა, Orders (ჩანაწერები) ცხრილში დაუმატოთ ისეთი ჩანაწერი, რომელიც არ შედის Customer (მყიდველი) ცხრილში, მთავრდება შეცდომით.

9. 2. მონაცემთა ბაზის Visual Basic-ის ველის ტიპები

Visual Basic მონაცემებს ინახავს ცხრილის ველებში. განასხვავებს მონაცემთა ველების 14 ტიპს. 9. 1 ცხრილში ჩამოთვლილია ეს ტიპები.

□ ველის ტიპი LONGBINARY გამოიყენება გამოსახულებისა და OLE ობიექტების შესანახად;

□ ველის ტიპი BOOLEAN ხშირად გამოიყენება აღმებით დაკავშირებული არჩეული შედეგების შესანახად. მასში ინახება მხოლოდ – 1 (ჭეშმარიტი) და 0 (მცდარი). თუ შეტანის ველში შეიტანეთ მაგალითად 13, მაშინ დაიმახსოვრებს მხოლოდ – 1-ს და არ გამოიტანს შეცდომის შეტყობინებას. ამიტომ ამ ტიპის გამოყენებისას ფრთხილად უნდა იყოთ.

□ ველის ტიპი BYTE აიღებს რიცხვებს 0–255 დიაპაზონში, ყველა სხვა მნიშვნელობას აღიქმის როგორც შეცდომა (შეცდომა 524).

□ ველის ტიპი COUNTER–განსაკუთრებული შემთხვევა. ნებისმიერი ჩანაწერი მხოლოდ წაიკითხება და რაც არ უნდა ეცადოთ ამ ველში რაიმის ჩაწერას, გამოიტანება შეცდომის შეტყობინება. იგი გამოიყენება უნიკალური პირველადი გასაღებიანი ველებისათვის.

□ ველის ტიპები MEMO და TEXT დასაშვებია ნებისმიერი სიმბოლური მონაცემის შესატანად.

ველების ტიპები Visual Basic-ში

ცხრილი 9. 1.

მონაცემთა ველის ტიპი	ზომა	Visual Basic -ში მონაცემის ტიპი	შენიშვნა
BINARY	V	არავითარი	შემოსაზღვრულია 255 ბაიტამდე
BOOLEAN	1	Boolean	ინახავს მხოლოდ 0-ს ან 1-ს
BYTE	1	Integer	ინახავს მხოლოდ 0 - 255
COUNTER	8	Long	
CURRENCY	8	Currency	ათობითი მძიმის მარცხნივ 15 სიმბოლოს და მძიმის მარჯვნივ კი 4 - ს
DATETIME	8	Date/Time	ათობითი მძიმის მარცხნივ თარიღი, მარჯვნივ კი საათი
DOUBLE	8	Double	

GUID	16	არავითარი	გლობალური უნიკალური იდენტიფიკატორების შესანახად
INTEGER	2	Integer	მთელი ტიპის
LONG	8	Long	
LONGBINARY	V	არავითარი	OLE - ობიექტების შესანახად
MEMO	V	String	სიგრძე 1.2 გიგაბაიტი
SINGLE	4	Single	
TEXT	V	String	სიგრძე შეზღუდულია 255 ბაიტამდე

9. 3. პროექტი ველის ტიპების ტესტირებისათვის

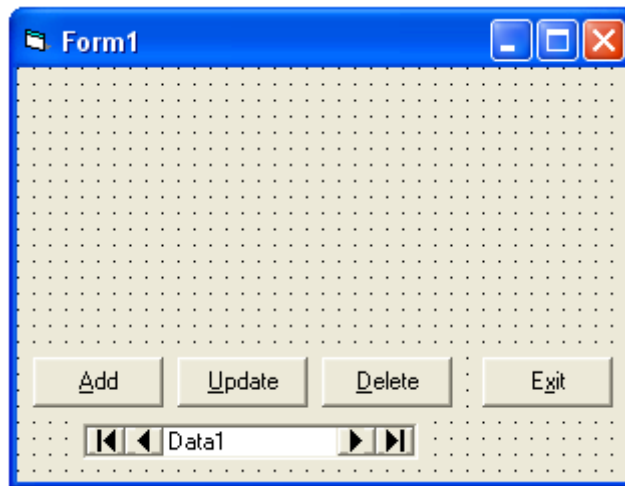
1. შექმენით ახალი პროექტი, რისთვისაც მენიუდან აირჩიეთ File→ New Project ბრძანება, იხედვით 9. 2 ცხრილით და შეავსეთ ფორმა.

2. ახლა დაამატეთ კოდი ბრძანების ღილაკებისათვის, რისთვისაც ორჯერ დააწკაპუნეთ ნებისმიერ ბრძანებით ღილაკზე. გამოჩნდება საწყისი კოდის ფანჯარა, შეიტანეთ დასამუშავებელი ღილაკის შესაბამისი მნიშვნელობები 9. 2 ცხრილიდან.

მონაცემთა ველის ტიპების ტესტირებისათვის მართვის ელემენტების პროექტი

ცხრილი № 9. 2.

მართვის ელემენტი	თვისება	მნიშვნელობა
პროექტი	Name	prjFieldTypes
ფორმა	Name	frmFieldTypes
	Caption	Visual Basic-ის მონაცემთა ველის ტიპი
ბრძანების ღილაკი	Name	cmdBtn
	Caption	&Add
	Height	300
	Index	0
	Width	1200
ბრძანების ღილაკი	Name	cmdBtn
	Caption	&Update
	Height	300
	Index	1
	Width	1200
ბრძანების ღილაკი	Name	cmdBtn
	Caption	&Delete
	Height	300
	Index	2
	Width	1200
ბრძანების ღილაკი	Name	cmdBtn
	Caption	E&xit
	Height	300
	Index	3
	Width	1200
მართვის ელემენტი Data	Name	datFieldTypes
	DatabaseName	FIELDTYPES.MDB (სწორი გზის ჩათვლით)
	RecordSource	FieldTypes



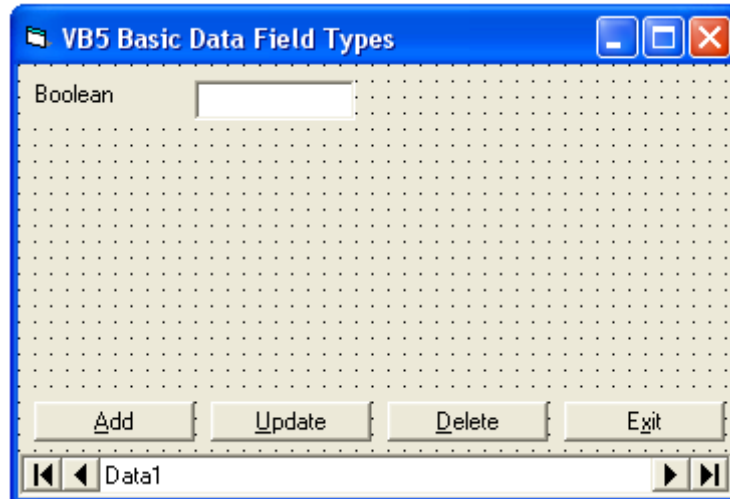
სურ. 9. 1. მონაცემთა ველის ტიპების ტესტირებისათვის ფორმის პროექტი.

ბრძანების კლავიშებზე დაწკაპუნების დამმუშავებლის კოდი კოდების ფრაგმენტში ზოგი სიახლე შეგხვდებათ. მაგალითად, იმის მიხედვით თუ მომხმარებელი რომელ ლილაკს დააჭერს ხელს სრულდება კოდის სხვადასხვა სტრიქონი. ლილაკზე დაჭერა მიუთითებს პარამეტრს -Index, რომელიც გადაეცემა დაწკაპუნების (Click) დამმუშავებელს. ექსპერიმენტის გასაადვილებლად ფორმაზე დამატებულია შეცდომის დამუშავების კოდი.

```
Private Sub cmdBtn_Click(Index As Integer)
'
' ბრძანების ლილაკებზე დაწკაპუნების დამუშავება
'
On Error GoTo LocalError
'
Select Case Index
Case 0 ' Add
    datFieldTypes.UpdateRecord
Case 1 ' Update
    datFieldTypes.UpdateRecord
    datFieldTypes.UpdateRecordset.Bookmark = datFieldTypes.UpdateRecordset.LastModified
Case 2 ' Delete
    datFieldTypes.UpdateRecord.Delete
    datFieldTypes.UpdateRecord.MovePrevious
Case 3 ' Exit
    Unload Me
End Select
Exit Sub
'
LocalError:
    MsgBox Err.Description, vbCritical, Err.Number
'
End Sub
```

ახლა პროექტი უნდა შეინახოთ FieldTypes.vbp ფაილში, ფორმა კი - FieldTypes.frm ფაილში.

BOOLEAN ტიპის მონაცემის ტექსტირება



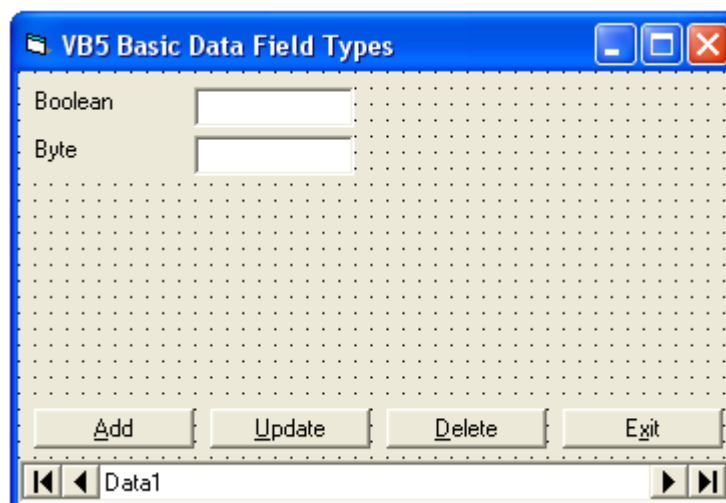
სურ. 9. 2. Boolean ტიპის მონაცემის ტექსტური ველის დამატება

ახლა ფორმაზე მოათავსეთ ტექსტური ველი და ჭდე. გაუკეთეთ ჭდეს სათაური Boolean. ტექსტური ველისათვის DataSource თვისებაში შეიტანეთ datFieldTypes მნიშვნელობა, ხოლო DataField თვისებაში - BooleanField მნიშვნელობა. Text თვისება დატოვეთ ცარიელი და გაუშეით პროგრამა შესრულებაზე. დაინახავთ, რომ ველი ცარიელია. იმისათვის, რომ გააკეთოთ ახალი ჩანაწერი, დააწკაპუნეთ Add ღილაკს, ხოლო იმისათვის რომ შეინახოთ იგი, დააწკაპუნეთ Update ღილაკზე. დაინახავთ, რომ შეტანის ველში პირველი ჩანაწერი არის 0.

ახლა შეიტანეთ რიცხვი 13 და დააწკაპუნეთ Update ღილაკზე. მონაცემის მართვის ელემენტი შეინახავს შეტანილ მნიშვნელობას ცხრილში და განაახლებს ეკრანზე მის ასახვას. დაინახავთ, რომ 13-ის ნაცვლად ჩაწერილი იქნება - 1, რადგან ნულისაგან განსხვავებული ყოველი რიცხვი Boolean ტიპის ველში გარდაიქმნება - 1-ად.

Byte-ის ტიპის მონაცემის ტექსტირება

ახლა შექმენით Byte-ის ტიპის ტექსტური ველი და ჭდე. შრომის გასაადვილებლად უკვე შექმნილი ჭდე და ტექსტური ველი მონიშნეთ ჩარჩოთი (ორივე ერთად), გამოიყენეთ მენიუდან Edit→Copy, Edit→Paste ბრძანებები და გააკეთეთ ფორმაზე ასლი. შეცვალეთ შესაბამისად თვისებები ახალი ველისათვის, რის შემდეგაც თქვენი ფორმა უნდა გამოიყურებოდეს ისე როგორც 9. 3 სურათზეა გამოსახული.



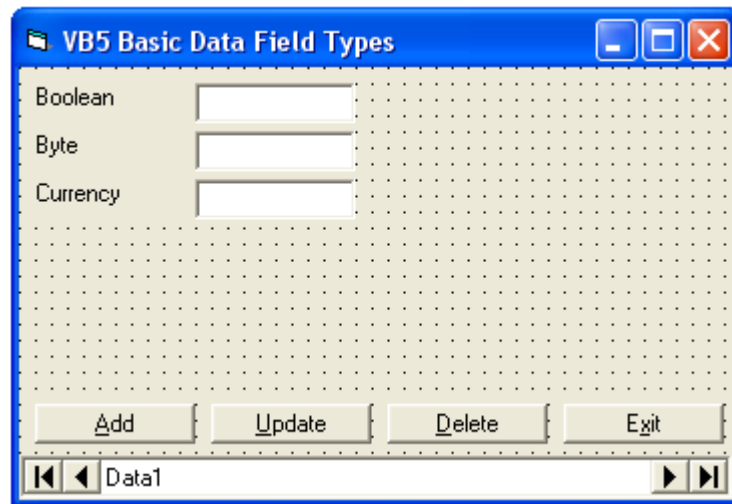
სურ. 9. 3. ფორმაზე Byte-ის ტიპის ველის დამატება

შეინახეთ პროექტი და გაუშვით პროგრამა შესრულებაზე. დააწკაპუნეთ Add ღილაკზე, რითაც დაემატება ახალი ჩანაწერი, ტექსტურ ველში შეიტანეთ რიცხვი 256 და დააწკაპუნეთ Update ღილაკზე. ამ მნიშვნელობის შენახვისას გამოვა შეტყობინება შეცდომის შესახებ, რადგანაც Byte-ის ტიპის ველი ინახავს მხოლოდ 0-დან 255-მდე დიაპაზონის რიცხვებს.

Currency-ის ტიპის მონაცემის ტესტირება

წინა შემთხვევის ანალოგიურად გააღებლეთ ტექსტური ველი და ჭდე. მიანიჭეთ შესაბამისი თვისებების მნიშვნელობები, შეინახეთ პროექტი და გაუშვით შესრულებაზე.

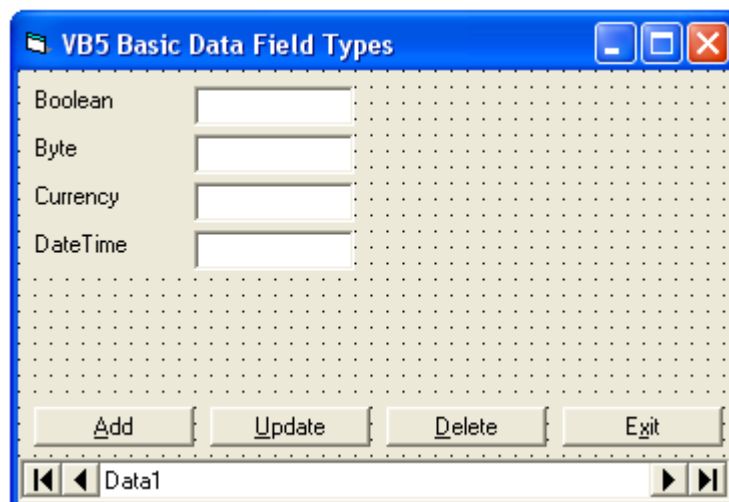
შემდეგ დაამატეთ ჩანაწერის მნიშვნელობა 1.00001 და სცადეთ შენახვა. რადგან ველი რომლის ტიპია Currency ინახავს მხოლოდ ისეთ რიცხვებს რომლის ათწილად ნაწილში არის ოთხი ციფრი, ამიტომ თქვენს მონაცემს იგი ვერ შეინახავს და მოგცემთ შეტყობინებას შეცდომის შესახებ. გაასწორეთ და შეინახეთ.



სურ. 9. 4. ფორმაზე Currency ტიპის ველის დამატება

DATETIME-ის ტიპის მონაცემის ტესტირება

Visual Basic ასრულებს ფართომასშტაბიან შემოწმებას DataTyme-ის ტიპის ველში მნიშვნელობის ჩასაწერად. ამ ტიპის ველის მეშვეობით გარანტირებული ხარტ, რომ თარიღის მნიშვნელობა სწორად იქნება შეტანილი. თარიღის ტიპის ველის შექმნის შემდეგ თქვენი ფორმა უნდა გამოიყურებოდეს ისე, როგორც ეს ნაჩვენებია 3. 5 სურათზე.



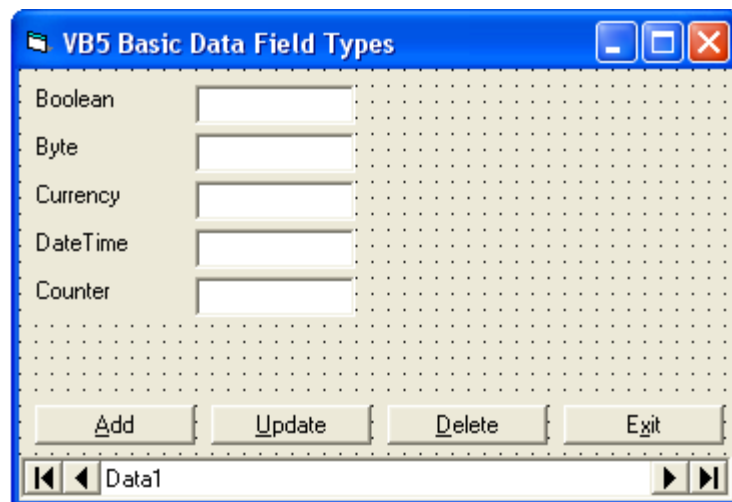
სურ. 9. 5. ფორმაზე DateTime-ის ტიპის მონაცემის დამატება

როგორ ირჩევს Visual Basic-ი თარიღის ფორმატს? ამისათვის იგი იღებს Windows-ის მართვის ჯგუფიდან (Regional/Settings)-ს. ჩაატარეთ ექსპერიმენტი, რისთვისაც აირჩიეთ (Start/Settings/Control Panel/ Regional Settings). შეცვალეთ თარიღის ფორმატი და დაუბრუნდით თქვენს პროექტს, შედეგი არ დააყოვნებს.

✎ ევროპულ ფორმატში, სადაც ჩაიწერება თარიღი ფორმატით დღე-თვე-წელიწადი, თქვენი პროგრამა იმუშავებს უპრობლემოდ. ამერიკული ფორმატია თვე-დღე-წელიწადი. თუ თქვენ თარიღს ინახავთ სტრიქონული ფორმატით თვე-დღე-წელიწადი, ან რიცხვითი ფორმატით წელიწადი-დღე-თვე, მაშინ თქვენი პროგრამა არასწორად ასახავს თარიღს რეგიონის მიხედვით.

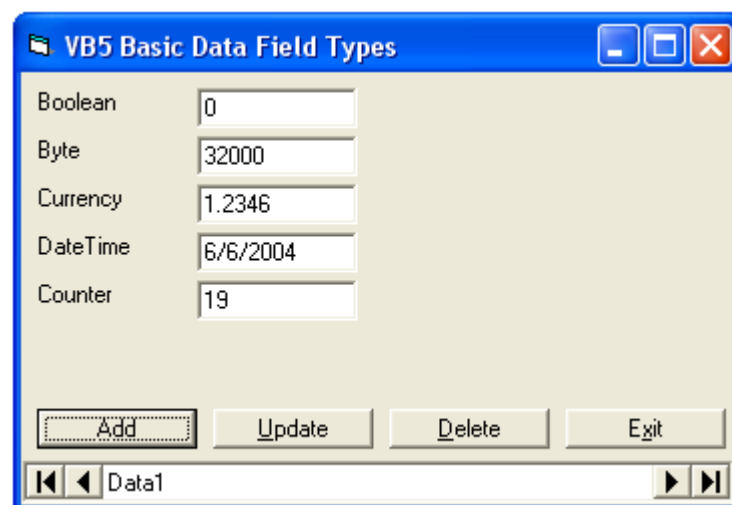
COUNTER-ის ტიპის მონაცემის ტესტირება

Visual Basic ყოველთვის, როდესაც თქვენ ამატებთ ახალ ჩანაწერს, ახდენს ამ ტიპის მონაცემის ავტომატურ ფორმირებას. ამ ტიპის ველი გამოიყენება პირველადი გასაღებისათვის, რადგანაც Visual Basic-ს შეუძლია შეინახოს Counter ტიპის ველში მილიარდზე მეტი უნიკალური მნიშვნელობა. დააყენეთ თქვენს ფორმაზე ამ ტიპის ველი.



სურ. 9. 6. ფორმაზე Counter-ის ტიპის მონაცემის დამატება

მიანიჭეთ ჭლის სათაურს მნიშვნელობა Counter, ხოლო ტექსტური ველის თვისებაში DateField დააყენეთ მნიშვნელობა AutoIncrField. შეინახეთ პროექტი და პროგრამა გაუშვით შესრულებაზე. თქვენი ფორმა უნდა გამოიყურებოდეს ისე, როგორც 9. 7 სურათზეა ნაჩვენები.



სურ. 9. 7. შესრულების შედეგი

☞ თუ მონაცემთა ცხრილი შექმნილია Access-ში, მაშინ მონაცემთა ბაზის მართვის ელემენტის გამოსაყენებლად, Access-ისა და Visual Basic-ის ვერსიები უნდა შეესაბამებოდეს ერთმანეთს. თუ ეს ვერსიები არ შეესაბამება ერთმანეთს, მაშინ გახსენით Access-ში შექმნილი ცხრილი, აირჩიეთ ბრძანება მენიუდან Tools→Database Utilities→Convert Database→To Access 97 File Format..., გამოვა შეკითხვა ცხრილის დახურვის შესახებ, დაეთანხმეთ, რის შემდეგაც მოგთხოვთ მონაცემთა ბაზის შენახვას. შეინახეთ ბაზა და Visual Basic-ში უკვე შეძლებთ მის გამოყენებას.

Visual Basic-ის მონაცემთა დამატებითი ტიპები

ცხრილების დაროქტებისას გაითვალისწინეთ შემდეგი:

- გამოვა შეტყობინება შეცდომის შესახებ, თუ ველში რომლის ტიპია TEXT შეიტანეთ 50-ზე მეტი სიმბოლო;
 - LONGBINARY-ის ტიპის ველი გამოიყენება გრაფიკული გამოსახულების შესანახად და ნებისმიერი ალფაბეტური სიმბოლოს მონაცემთა შეტანისა და შენახვის საშუალებას გაძლევთ;
 - ველის ტიპებისათვის DOUBLE, INTEGER და SINGLE შეამოწმეთ Visual Basic-ის ოპერატიულ ცნობარში მნიშვნელობების დიაპაზონი;
 - BOOLEAN-ის ტიპის ველში დასაშვებია 0-ისა და 1-ისაგან განსხვავებული რიცხვები;
 - CURRENCY ტიპის ველი ინახავს მძიმის შემდეგ მხოლოდ ოთხ ათობით ნიშანს. თუ შეგაქვთ მძიმის შემდეგ ოთხ ათობით ციფრზე მეტი, მაშინ Visual Basic-ი დაამრგვალებს ოთხ ციფრამდე;
 - DATETIME ტიპის ველში Visual Basic-ი არ მოგცემთ შესაძლებლობას, რომ დაიმახსოვროს თარიღი დასაშვები ფორმატის გარეშე;
 - GUID -ის ტიპის ველი გამოიყენება სპეციალური 128 თანრიგიანი რიცხვების ე. წ. “გლობალურ უნიკალური იდენტიფიკატორების” შესანახად;
 - BINAR-ის ტიპის ველი საშუალებას იძლევა დამახსოვროს 0-დან 255 ბაიტამდე მონაცემი და აქვს შეზღუდვითი გამოყენება. თუ იყენებთ Visual Data Manager-ს, მაშინ ცხადია რომ ველის ტიპი BINARY იგივეა რაც LONGBINARY;
 - MEMO და LONGBINARY-ის ტიპის ველებს უწოდებენ “დიდი მნიშვნელობების” ველებს, რამდენადაც თითოეული მათგანი შეიძლება შეიცავდეს 1. 2 გიგაბაიტის მონაცემს.
- შექმნილი პროექტი შეგიძლიათ შეინახოთ შემდგომში ჩასათვირთად ან დასაბეჭდად. გარდა ამისა, შეგიძლიათ გამოიყენოთ სხვა პროექტის შედგენის დროს როგორც ნიმუში. Visual Basic-ის მონაცემთა ბაზებთან მუშაობისა და პროგრამირების შესაძლებლობების გაერთიანებით შეგიძლიათ შექმნათ სრულყოფილი პროექტი. Visual Basic-ის პროგრამირების ელემენტებს გაეცნობით წიგნის მომდევნო ნაწილში.

ლიტერატურა:

1. С. Браун “Visual Basic 5“, с самого начала, ПИТЕР Санкт-Петербург, Москва •Харьков •Минск, 1998 г.
2. Майкл Амундсен, Кертис Смитт “Программирование баз данных на VISUAL BASIC 5“ Полное руководство для самостоятельного обучения, Перевод с английского под редакцией В. Карташова, Москва • ЗАО « Издательство **БИНOM** » • 1998 г.

ს ა რ ჩ ე ვ ი

შესავალი	2
თავი I. მონაცემთა ბაზის შექმნა	3
1.1. Access-ის გაცნობა	4
1.2. ობიექტები მონაცემთა ბაზაში	4
1.3. მონაცემთა ბაზის შექმნა	5
1.4. მონაცემთა ბაზის სტრუქტურის დაგეგმვა და შექმნა	7
1.5. დამატებითი მოწყობილობები	8
თავი II. ცხრილების შექმნა	10
2.1. ახალი ცხრილის შექმნა	10
2.2. ცხრილების შექმნა ცხრილის რეჟიმში	11
2.3. ცხრილის შექმნა ცხრილის ოსტატით	12
2.4. ცხრილის შექმნა კავშირის დაყენების საშუალებით	14
2.5. ცარიელი ცხრილის შექმნა	15
2.6. ცხრილში ველებთან მუშაობა	17
2.7. ინდექსების თვისებების დაყენება	21
2.8. ცხრილების მოდიფიკაცია	23
2.9. ცხრილებს შორის კავშირის დამყარება	24
თავი III. მონაცემთა დათვალიერება და რედაქტირება	26
3.1. ცხრილის დათვალიერება ცხრილურ ფორმატში	26
3.2. მონაცემების ძებნა	30
3.3. მონაცემების დახარისხება და გაფილტვრა	31
თავი IV. ფორმების შექმნა	34
4.1. ფორმების დაგეგმვა	34
4.2. ფორმების შექმნა	32
4.3. ფორმების მოდიფიკაცია და უწყისის სტრუქტურის დამუშავება	37
4.4. მართვის ელემენტებთან მუშაობა	38
თავი V. მონაცემთა ბაზის მოთხოვნების შექმნა	45
5.1. მოთხოვნა	45
5.2. ახალი მოთხოვნის შექმნა	46
5.3. მოთხოვნის ფორმულების აღწერა	49
5.4. გამოსათვლელი ველების დამატება	50
5.5. ჯგუფური გამოთვლების დამატება	50
5.6. არჩევის კრიტერიუმების გამოთვლა	51
თავი VI. უწყისების შექმნა	54
6.1. უწყისის შექმნა უწყისის ოსტატის დახმარებით	55
6.2. უწყისში დაჯგუფების არჩევა	55
6.3. უწყისის მოდიფიკაცია, წინასწარი დათვალიერება და ბეჭდვა	56
თავი VII. Visual Basic-ის გაცნობა	57
7.1. Visual Basic-ის სამუშაო გარემო	58
7.2. დამუშავების ინტეგრირებული საშუალება	59
7.3. მართვის ელემენტების გამოყენება	64
თავი VIII. მონაცემთა ბაზასთან მუშაობა	79
8.1. ახალი პროექტის გაშვება Visual Basic-Si	79
8.2. დაკავშირებული შეტანის მართვის ელემენტების დამატება	81
თავი IX. მონაცემთა ბაზის შექმნა	85
9.1. რელაციური მონაცემთა ბაზა	85
9.2. მონაცემთა ბაზის Visual Basic-ის ველის ტიპები	86
9.3. პროექტი ველის ტიპების ტესტირებისათვის	87
ლიტერატურა	93

დასკვნა მეცადინეობის შედეგად გაიგეთ შემდეგი:

1. როგორ გამოიყენოთ მართვის ელემენტი Data მონაცემთა ბაზასთან ფორმებისა და ცხრილების დასაკავშირებლად თვისებების DatabasName და DataSource-ს საშუალებით;
2. როგორ მიაბათ ფორმაზე მოთავსებული დაკავშირებული შეტანის მართვის ელემენტი ცხრილის და ველის მონაცემთან თვისებების DataSource და DataField-ის გამოყენებით.
3. როგორ გავაერთიანოთ სტანდარტული ბრძანების ლილაკები მეთოდებით Add New და Delete, რომ უზრუნველყოფილი იქნეს შესატანი ფორმებისათვის ჩანაწერების დამატება და წაშლა.

კითხვები

1. რომელი ორი თვისება უნდა დაუყენოთ აუცილებელი მიმდევრობით, მართვის ელემენტს Data-ს, როდესაც არსებულ მონაცემთა ბაზასა და ცხრილს აკავშირებთ ფორმასთან?
2. რომელი თვისება უნდა დაუყენოთ, მართვის ელემენტს Data-ს, რომ შეძლოს მონაცემთა ცხრილის სახელის ასახვა?
3. ტექსტური ველის თვისებებში რომელი ორი თვისებაა დასაყენებელი მის დასაკავშირებლად ფორმაზე მოთავსებულ Data - მართვის ელემენტთან ?
4. შეტანის ფორმაზე მართვის ელემენტის Data-ს გამოყენების დროს Visual Basic-ის პროგრამული კოდის რამდენ სტრიქონს მოითხოვს ჩანაწერების დამატებისა და წაშლის ფუნქციები?
5. რომელ კლავიშებს ეძახიან სწრაფ ანუ ცხელ კლავიშებს?

სავარჯიშოები:

1. დაამატეთ სათაური “The Addresses Program “ახლახან შექმნილ შესატან ფორმაში;
 2. მოათავსეთ შესატან ფორმაზე დამატებითი ბრძანების ლილაკი Exit. დააპროგრამეთ ეს ლილაკი ისე, რომ მასზე დაწკაპუნებით პროგრამამ დაამთავროს მუშაობა;
- მოახდინეთ Add ლილაკის მოდიფიკაცია ისე, რომ მასზე დაწკაპუნებით კურსორი გადაადგილდეს შესატანი ფორმის პირველ ტექსტურ ველზე (txtISBN). (კარნახი: Visual Basic-ის ოპერატიულ ცნობებში მონახეთ მეთოდი Set Focus).