Srednja poklicna in tehniška šola Murska Sobota

Naziv poklicne (strokovne) izobrazbe

Elektrotehnik računalništva

Izdelek oz.storitev in zagovor

Dokumentacija programa

**Vodenje športne dvorane**

Mentor: Milan Petrijan Avtor: Uroš Rac

Razred:4R2

Šolsko leto: 2009/2010

Kazalo

[Uvod 4](#_Toc259566067)

[Microsoft Visual Studio .NET 2003 4](#_Toc259566068)

[Začetna stran programa 6](#_Toc259566069)

[Vpis novih podatkov 7](#_Toc259566070)

[Vpis novih podatkov programska koda 8](#_Toc259566071)

[Deklaracije objektov so: 8](#_Toc259566072)

[Začetne vrednosti za prenos vpisanih podatkov so: 8](#_Toc259566073)

[Izračunane vrednosti so: 9](#_Toc259566074)

[Funkcija za izračun vseh mest za sektor in vseh mest: 9](#_Toc259566075)

[Funkcije za vračanje opozoril: 9](#_Toc259566076)

[Funkciji za določitev označenih sektorjev 11](#_Toc259566077)

[Funkciji za izračun števila mest 11](#_Toc259566078)

[Datumska funkcija 11](#_Toc259566079)

[Razporeditev gledalcev glede na sektorje 11](#_Toc259566080)

[Izpis vseh podatkov 12](#_Toc259566081)

[Brisanje podatkov 13](#_Toc259566082)

[Razporeditev gledalcev glede na sektorje 14](#_Toc259566083)

[Začetne definicije programa 14](#_Toc259566084)

[Povezava z bazo podatkov 14](#_Toc259566085)

[Deklaracije objektov in funkcij za prenos 17](#_Toc259566086)

[Vpisovanje podatkov 17](#_Toc259566087)

[Začetni vpis vrednosti ob pojavu okna 17](#_Toc259566088)

[Upravljanje s podatkovno bazo 19](#_Toc259566089)

[Premikanje po bazi in shranjevanje 19](#_Toc259566090)

[Funkcija za izračun števila sektorjev 21](#_Toc259566091)

[Izpis podatkov 22](#_Toc259566092)

[Začetne definicije programa 22](#_Toc259566093)

[Povazava z bazo podatkov 22](#_Toc259566094)

[Deklaracije objektov in funkcij za prenos 23](#_Toc259566095)

[Izpis vrednosti programa 23](#_Toc259566096)

[Začetna določitev 23](#_Toc259566097)

[Izpis podatkov 23](#_Toc259566098)

[Shranjevanje podatkov 24](#_Toc259566099)

[Tabele podatkov 27](#_Toc259566100)

[Definicija programa 27](#_Toc259566101)

[Povezava s podatkovno bazo 27](#_Toc259566102)

[Prikaz podatkov 29](#_Toc259566103)

[Spreminjanje in shranjevanje zapisov 29](#_Toc259566104)

[Iskanje 30](#_Toc259566105)

[Definicija programa 30](#_Toc259566106)

[Povezava z bazo podatkov 30](#_Toc259566107)

[Deklaracija objektov in spremenljivke za iskanje 31](#_Toc259566108)

[Začetna določitev podatkov 31](#_Toc259566109)

[Določitev iskanja podatkov 32](#_Toc259566110)

[Filtriranje podatkov 33](#_Toc259566111)

[Spreminjanje in shranjevanje 36](#_Toc259566112)

[Povezava v inštalacijsko datoteko Inno Setup 5 36](#_Toc259566113)

[Zaključek 41](#_Toc259566114)

[Viri 41](#_Toc259566115)

# Uvod

Vodenje športne dvorane je program, ki omogoča upravljanje s podatki o športni dvorani.

Napisan je v razvojnem okolju Visual C# .NET 2003, ki omogoča vizualno programiranje in pisanje programske kode, ter podpira programiranje za več vrst aplikacij.

Kot so določanje sektorjev v športni dvorani, število mest za posamezen sektor glede na število vrst, cena, vstopnice, datum ter število gledalcev.

Program na podlagi vpisanih podatkov omogoča dostop do drugih funkcij kot so izračun osnovnih podatkov, razporejanje gledalcev po sektorjih, shranjevanje podatkov, prikaz tabel o vpisanih podatkih in iskanje ter filtriranje.

# Microsoft Visual Studio .NET 2003

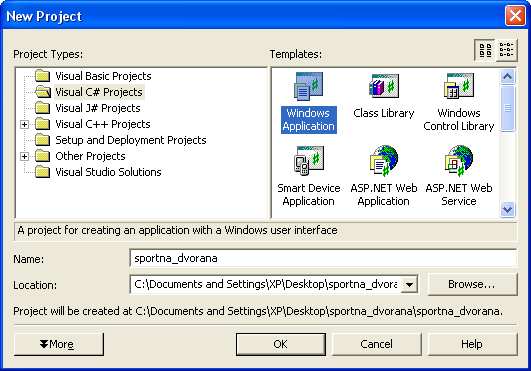
Microsoft Visual C# .NET je komplet orodij za gradnjo spletnih storitev XML in programske opreme, ki uporablja tehnologijo Microsoft .NET za okolje Microsoft Windows in splet.

Uporablja komponentno usmerjen razvojni jezik C# in začetnim, ter izkušenim razvijalcem s področja jezikov C++ in Java omogoča razvijanje naslednje generacije programske opreme. Podpira tudi številne izboljšave in nove funkcije za učinkovito opravljanje poslovnih procesov. Nekateri izmed teh so:

Slika\_1: Microsoft Visual Studio .NET 2003

* Podpora za hitro načrtovanje, razvoj in uvajanje pri ustvarjanju in uporabi spletnih storitev.
* Orodja za oblikovanje obrazcev in vizualne kontrolnike za izdelavo aplikacij za Windows.
* Orodja in storitve za izdelavo zmogljivih strežniških režitev s tehnologijo Microsoft .NET.
* Selitvena orodja za pretvorbo projektov v razvojno okolje Microsoft .NET.

Razvijalci lahko z jezikom Visual C# .NET 2003 izdelajo rešitve za najširšo paleto odjemalcev, vključno z okoljem Windows, spletom in mobilnimi ali vdelanimi napravami.



Slika\_2:Ustvarjanje novega projekta

Visual studio podpira več vrst programskih jezikov kot so Basic, C#, J#, C++ in drugi. Če izberemo eno kategorijo programskega jezika lahko izbiramo med vrsto aplikacije, ki je za program najbolj smiselna..

Za ustvarjanje novega projekta zaženemo Microsoft Visual Studio .NET 2003 in izberemo File→New→Project… ali pritisnemo tipko CTRL+SHIFT+N.

Izberemo programski jezik in vrsto aplikacije. V primeru športne dvorane je to Windows Application, določimo še ime in lokacijo novo ustvarjenega projekta. Potem lahko začnemo pisati programsko kodo in vizualno programirati(odvisno od vrste aplikacije), kar omogoča dostop do elementov, kot so gumbi, tabele, kolendar, grafični zapisi, povezavni elementi z bazo podatkov itd.

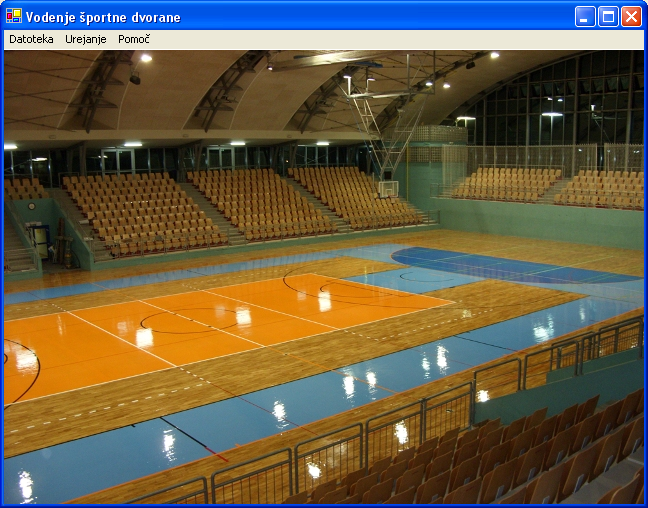
# Začetna stran programa

Na začetni strani je prikaz naziva programa in slike športne dvorane, ta je vključena preko Windows Form Picture Box.

In osnovni izbor Datoteka(bližnjica do Datoteka je ALT D) in Urejanje(bližnjica do Urejanje je ALT D) za dostop do funkcij programa,ki so mogoči preko Windows Form MainMenu in se nahaja v Toolbox-u.

main_menu.bmp

Slika\_3: Komponenta mainMenu1, ki je vključena v začetno stran



Slika\_4: Začetna stran programa

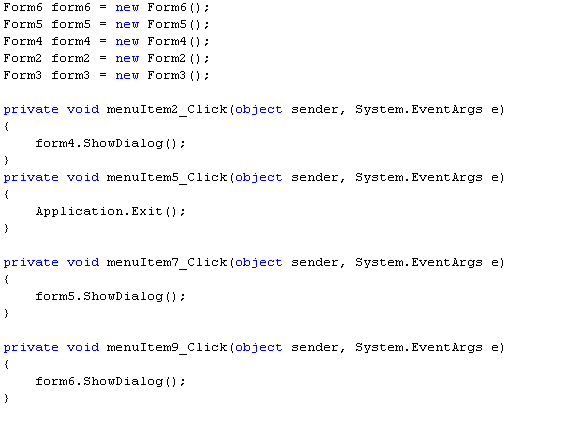
Pod kategorijo Datoteka je omogočen dostop do vpisa podatkov, ki je označen z Vpiši podatke(bližnjica do Vpiši podatke je ALT P in CTRL+N),ki prikaže okno za vpis novih podatkov. In Izhod(bližnjica do Izhod je ALT I),ki zapusti program.

Pod kategorijo Urejanje je omogočen dostop do Tabele podatkov(bližnjica do Tabele podatkov ALT T in CTRL+T),ki prikaže shranjene zapise v tabelah podatkov in Iskanje(bližnjica do Iskanje ALT S in CTRL+I),ki lahko filtrira iz izpiše podatke glede na pogoje.

Pod kategorijo Pomoč(bližnjica ALT O) je dostop do vizitke(bližnjica ALT Z in CTRL+ V),ki prikaže okno z vizitko.

Bližnjice so v programu določene glede na zapis znaka & v imenu izbire pred določenim znakom(ki je bližnjica) do katere je mogoč dostop preko tipke ALT tako, da pritisnemo tipko ALT in izberemo črko na tipkovnici, ki omogoča dostop.

Lahko pa so bližnjice določene tudi z CTRL in določenim znakom, ki ga izberemo, ko pritisnemo tipko CTRL.

Na sliki levo je prikaz dela programske kode začetne strani, kjer so določene povezave z ostalimi okni.

Na začetku je deklaracija obstoječega okna in objekta, ki se nanj sklicuje(v primeru, če ta obstaja). V funkcijah pa so določeni prikazi teh oken v primeru izbire zapisa v vizualnem delu programa.

Slika\_5: Del programske kode začetne strani

Na primer: vse izbire odprejo nova okna samo izbira Izhod zapre aplikacijo(ukaz: Application.Exit()).

Deklaracija Form[število] form[število] = new Form[število]();

določa na katero okno se objekt sklicuje.

Ukaz: form[število].ShowDialog(); odpre novo okno tako da ni mogoče preklapljati med drugimi okni programa in s tem zagotavlja varnost podatkov.

form[število] predstavlja objekt na katerega se program sklicuje in je pri deklaraciji določeno preko katerega okna deluje.

# Vpis novih podatkov

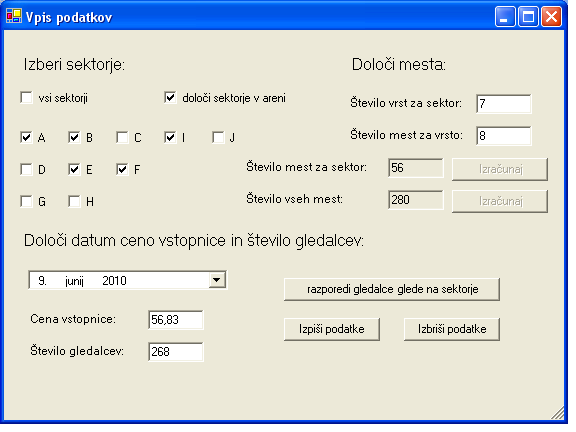
Pri vpisu podatkov lahko označimo kateri od desetih sektorjev bodo omogočeni v športni dvorani med njimi sta tudi dva sektorja v areni. Prav tako tudi število vrst za sektor in število mest za posamezno vrsto.

Če izberemo sektorje v areni najprej označimo določi sektorje v areni. Lahko pa tudi označimo vsi sektorji kar označi vse sektorje in če so določeni v areni se tudi ti označijo.

Če sta parametra določena lahko izračunamo število mest za posamezen sektor v primeru določitve sektorjev pa tudi število vseh mest (polji v kateri se vpiše izračunana vrednost sta samo za branje).

Izberemo lahko tudi datum, katerega vrednost je na začetku enaka današnjemu. Določimo lahko tudi ceno vstopnice in število gledalcev.

Ko so podatki določeni jih lahko razporedimo glede na sektorje, izpišemo ali izbrišemo zapisane vrednosti.



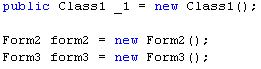
Slika\_6: Prikaz vpisa podatkov

# Vpis novih podatkov programska koda

## Deklaracije objektov so:

\_1 je objekt, ki vzpostavi povezavo z razredom in preko njega dostopamo do spremeljivk razreda.

form2 in form3 sta objekta, ki vzpostavita povezavo z drugimi okni, ki so navedena v deklaraciji in omogoča prenos vrednosti iz enega okna do drugega. Spremeljivka se iz tega dela programa prenesejo preko ukaza do drugega dela(okna), ki sprejme prenesene vrednosti.



Slika\_7: Začetne deklaracije objektov

## Začetne vrednosti za prenos vpisanih podatkov so:

st\_sektorjev,st\_vrst\_sekt,st\_mest\_sekt,st\_z\_mest, ki so celoštevilčne vrednosti za število sektorjev, število vrst in mest za sektor ter število zasedenih mest.

[]niz\_sektorjev, []sek\_arena,ki sta tipa string lahko shranita vrednosti znakov za začetno oznako sektorja.

doloci\_sektorje, izracunaj sta spremeljivki logičnega tipa, ki omogočata določanje sektorjev in izračunanje podatkov pod določenimi pogoji.

datum je spremeljivka tipa string, ki lahko shrani določen datum.

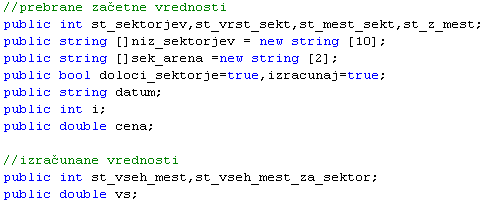
i je spremeljivka, ki šteje število sektorjev in določa zapis v index .

cena je spremeljivka, ki shrani realno vrednost.

## Izračunane vrednosti so:

st\_vseh\_mest, st\_vseh\_mest\_za\_sektor sta spremeljivki celoštevilčnega tipa, ki prevzameta izračunane vrednosti o številu vseh mest in število vseh mest za sektor.

vs je spremeljivka realnega tipa, ki prevzame vrednost vsote, ki jo določa cena vstopnice in število gledalcev.



Slika\_8: Začetne vrednosti razreda Class1 pri vpisu podatkov

## Funkcija za izračun vseh mest za sektor in vseh mest:

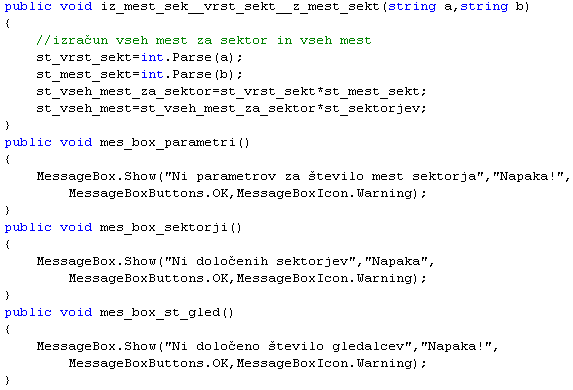
funkcija sprejme vrednost o število vrst in mest v argumentih funkcije za določen sektor in pretvori v celoštevilčni podatkovni tip. Potem izračuna število vseh mest za sektor in število vseh mest glede na znane podatke.

## Funkcije za vračanje opozoril:

mes\_box\_parametri() javi opozorilo v primeru, če niso določeni parametri.

mes\_box\_sektorji() javi opozorilo v primeru, če sektorji nico določeni.

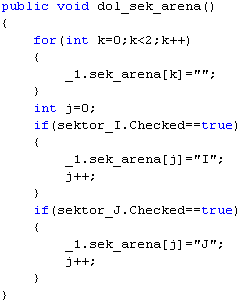
mes\_box\_st\_gled() javi opozorilo v primeru, če niso določeni gledalci.

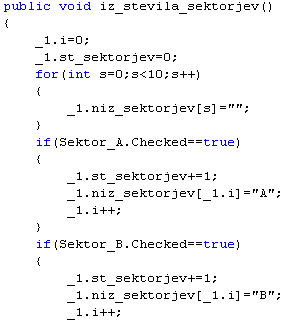


Slika\_9: Izračun podatkov in opozorila za vpis podatkov iz Class1

Na sliki 10 sta predstavljeni dve funkciji leva funkcija za vpis števil sektorjev in oznak sektorjev: Na začetku izvajanja programa funkcija priredi vrednost spremeljivki i in st\_sektorjev vrednost 0. Spremeljivki niz\_sektorjev se vrednost zapisov zbriše.

Funkcija preverja, če so označeni določeni sektorji in vpiše njihove vrednosti v niz\_sektorjev in sešteje število sektorjev.





Slika\_10: Vpis oznak sektorjev in dolocitev števila sektorjev

Desna funkcija deluje isto kot leva le, da se vpisujejo samo sektorji v areni.

## Funkciji za določitev označenih sektorjev

Je funkcija ,ki se izvede z označitvijo vseh sektorjev in označi vse sektorje v primeru, če so omogočeni sektorji v areni tudi te označi.

Druga funkcija je za določitev sektorjev v areni, ki oznaki za sektorja v areni omogoči za označevanje. Če pa od označimo sektorje v areni se oznaki za sektorja v areni zbrišeta in prav tako tudi oznake, če so določene.

## Funkciji za izračun števila mest

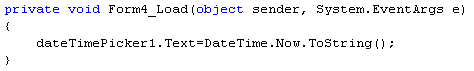
Prva funkcija izračuna število vseh mest za posamezen sektor. S tem, da se število vseh mest za sektor izračuna le, če je določeno število vrst in število mest za posamezno vrsto v nasprotnem primeru javi opozorilo, da niso določeni parametri za izračun. Funkcija se izvede s klikom na gumb izračunaj, ki takrat postane neaktiven za klik.

Če se spremeni zapis v textBox-ih se gumbu spremeni stanje na aktivno za klik, da se podatki lahko ponovno izračunajo.

Druga funkcija izračuna število vseh mest, kjer mora biti poleg parametrov določen vsaj en ali več sektorjev. Deluje tako, določi število označenih sektorjev s klicom funkcije in pod pogojem, če je število sektorjev različno od nič in so parametri za mesta določeni prebere parametre za vrsto in število mest za posamezno vrsto ter izračuna število vseh mest. Če se število mest izračuna se tudi gumb za izračun onemogoči. Omogoči pa se v primeru spremembe parametrov ali sektorjev. Če pogoji niso ustrezno izpolnjeni se najprej preveri, če so sektorji sploh določeni in, če niso vrne opozorilo. V primeru nedoločitve parametrov se sklicuje opozorilo, da parametri niso določeni toda le v primeru, če so določeni sektorji zaradi zaporedja določanja opozoril.

## Datumska funkcija

V programu imamo možnost določitve datuma zapisa podatkov, kjer se na začetku izvajanja programa določi današnji datum s pomočjo funkcije DateTime.Now, ki prenese vrednost današnjega datuma v polje za izbiro datuma tudi v primeru, če izbrišemo vse podatke.



Slika\_11: Na začetku programa se vrednost današnjega datuma prenese v polje za izbiranje datuma

## Razporeditev gledalcev glede na sektorje

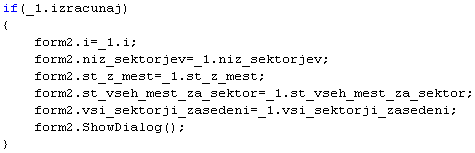
Ko vpišemo podatke o novem zapisu lahko kliknemo na gumb za razvrstitve gledalcev glede na sektorje, ki preveri če so vpisani in določeni vsi podatki za razvrščanje gledalcev glede na sektorje. To so označitev sektorjev, parametri za mesta in število gledalcev. Cena vstopnice ni nujna, ker se pri razvrščanju ne uporablja.

Funkcija deluje tako da najprej preveri, če so določeni sektorji, potem parametri sektorja in nazadnje število gledalcev ali število zasedenih mest. Če je vse določeno program kliče funkciji za določitev števila sektorjev in izračun vseh mest ter število mest za posamezen sektor, vrednost števila gledalcev se prenese v spremeljivko za število zasedenih mest.

Ko se program nadaljuje preveri še, če število vseh mest presega ali je eneko številu zasedenih sektorjev. Če je pogoj izpolnjen program vrne informacijo o zasedenosti vseh sektorjev in spremeljivki za stevilo gledalcev priredi vrednost vseh sektorjev in v polje stevilo\_gledalcev vpiše vrednost števila vseh mest. Informacijo o zasedenosti vseh sektorjev si tudi zapomni z spremembo stanja logične spremeljivke o zasedenosti vseh sektorjev na true.

Skozi celotno izvajanje programa ali funkcije se upošteva tudi stanje logične spremeljivke izracunaj, ki je na začetku izvajanja na logičnem stanju true. In se v primeru neizpolnitve pogojev postavi na logično stanje false, kar onemogoči prenos vrednosti na naslednji form(okno), ki so nujne za razvrstitev gledalcev glede na sektorje. S tem, da se na koncu programa vrednost spremeljivke izracunaj postavi na true v primeru ponovnega izvajanja programa(funkcije).

Deluje tako, da se sklicuje na objekt določenega form-a in funkcije, ki je na določena v tisti programski kodi, ki na drugi strani ali v drugem formu in sprejme podatke iz tega form-a glede na spremeljivke, ki so posredovane. In prikaže okno za razvrščanje v načino ShowDialog, ki onemogoča preklapljanje med odprtimi okni.



Slika\_12: Del funkcije, ki prenese spremeljivke za razvrščanje sektorjev

## Izpis vseh podatkov

Na začetku izvajanja funkcije se vse vrednosti(spremeljivke), ki se uporabljajo za izpis podatkov postavijo na vrednost 0, če tipa int ali double in spremeljivkam tipa string se izbriše njihova vrednost. Isto kot pri funkciji za razvrščanje sektorjev se preveri vpis spremeljivk, ki jih program potrebuje v zaporedju število sektorjev, parametrov za število mest, določenosti gledalcev in ceno.

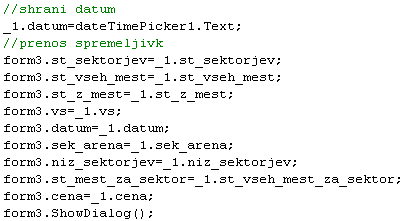
Če so podatki določeni se program nadaljuje z izvajanjem zaradi logične spremeljivke izračunaj, ki ima vlogo izračunavanja in prikaz novega okna v primeru, če se ohrani vrednost true, ki je že določena na začetku in se v primeru vseh dolocenih spremeljivk za izračun ne spremeni na vrednost false. Na koncu funkcije se spremeljivki izracunaj doloci vrednost true v primeru ponovnega izvajanja funkcije.

V primeru vseh vpisanih podatkov se preostale vrednost izračunajo preko funkcij, kot so izračun števila sektorjev, funkcija za določitev oznak sektorjev v areni in izračun vseh mest ter vseh mest za sektor. Funkcija določi tudi število zasedenih mest, ki jih prebere iz polja za število gledalcev in preveri, če jih je več ali enako kot število vseh mest. Če je pogoj izpolnjen se spremeljivki za število zasedenih mest priredi vrednost vseh mest in polju stevilo\_gledalcev se doloci vrednost vseh mest.

Funkcija nadaljuje izvajanje z branjem cene vstopnice in izračuna vsoto glede na pogoj števila mest. Če je število gledalcev večje ali enako od števila vseh mest se vsota izračuna tako, da se cena vstopnice zmnoži z številom vseh mest v primeru, če število gledalcev ni večje od števila vseh mest se vsota izračuna tako da se zmnoži cena vstopnice in število zasedenih mest.

Na koncu funkcije se prebere še izbrani datum in se shrani v spremeljivko tipa string. S tem da se lahko prenese tudi neposredno preko objekta drugega form-a. Vsi podatki za izpis ali spremeljivke kot so število sektorjev, število vseh mest za sektor, število vseh mest, število zasedenih mest, vsota, datum, cena, niz oznak vseh sektorjev in sektorjev v areni se prenesejo preko objekta form-a in določene funkcije, ki sprejme vrednost spremeljivke ali niza v drugem form-u na katerega se ta objekt le sklicuje s prireditvijo spremeljivke določeni funkciji.

Na koncu program(funkcija) prikaže okno(form) za izpis podatkov.



Slika\_13: Del funkcije, ki prenese podatke za izpis vseh podatkov.

## Brisanje podatkov

Omogoča izbris vseh označenih in določenih podatkov, ter spremeljivk pri pri vpisu.

Deluje tako da vse oznake sektorjev postavi na false, izbriše vpisane vrednosti polj za vpis parametrov sektorja, izračunanih podatkov vseh mest za sektor in vseh mest, ceno vstopnice in število gledalcev.

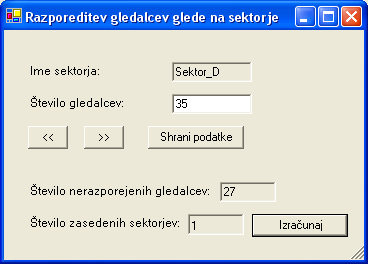
V naslednjem delu programa postavi vrednosti spremeljivk za število sektorjev, število vrst, število mest za vrsto, število zasedenih mest, število vseh mest, število vseh mest za sektor, ceno in vsoto na 0.

Tudi vrednosti nizov za niz vseh sektorjev in sektorjev v areni z for zanko in spremeljivko celoštevilčnega podatkovnega tipa za orientacijo po nizu izbriše zapisane znake.

Na koncu še priredi vrednost polju za datum na današnji z funkcijo DateTime.Now.

# Razporeditev gledalcev glede na sektorje

## Začetne definicije programa



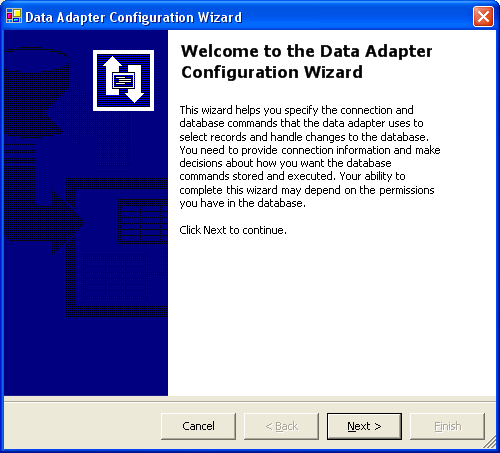
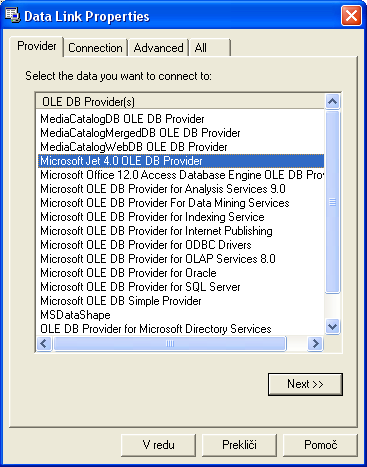
Slika\_14: Primer razporejanja podatkov glede na sektorje

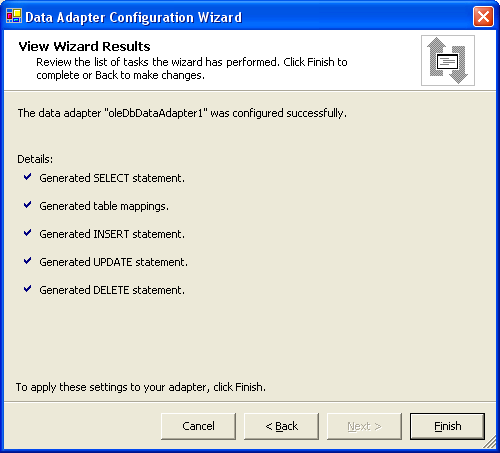
## Povezava z bazo podatkov

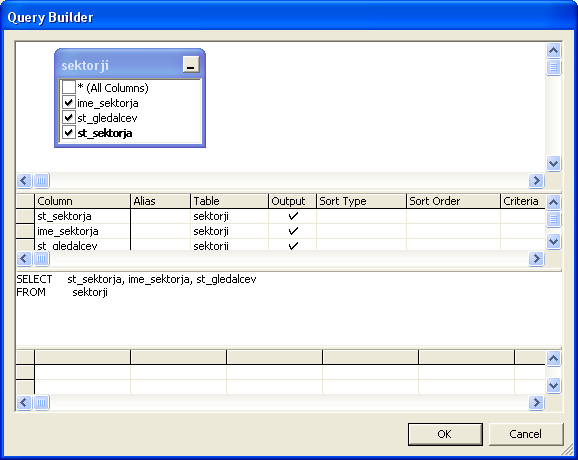
Povezavo z bazo podatkov ustvarimo tako, da vključimo data adapter in po možnosti ostale komponente za upravljanje s podatkovno bazo.

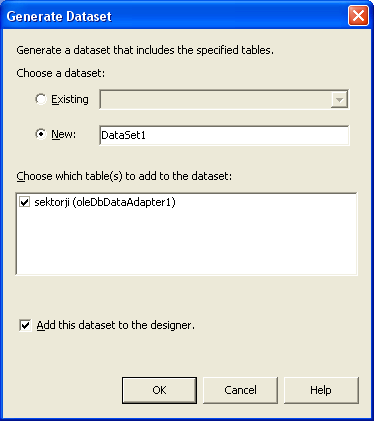
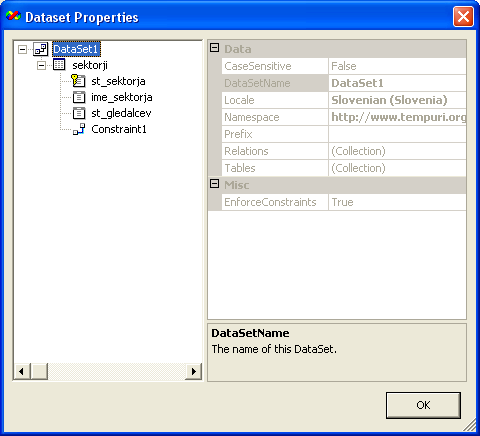
baz_pod_vkl_raz_sek.bmp

Slika\_15: Komponente za upravljanje z bazo, ki so vključene pri razvrščanju gledalcev glede na sektorje

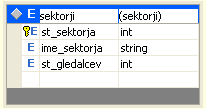




Slika\_16: Configuracija data adapterja pomembnejši koraki



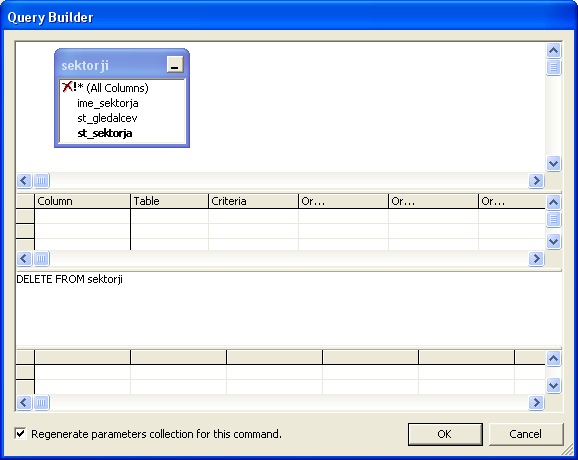
Slika\_17: Primer generacije dataset11 levo in desno prikaz tabele



Slika\_18: Primer prikaza tabele in podatkovnih tipov spremeljivk

con\_1 določa povezavo s podatkovno bazo, ki uporablja v programu in se ustvari po configuraciji data adapterja, ki v tem primeru vsebuje podatke o lokaciji podatkovne baze glede na način povezave.

Delete1 je ukaz, ki se uporablja za brisanje podatkov iz podatkovne baze, deluje tako, da izbriše vse zapise v bazi.

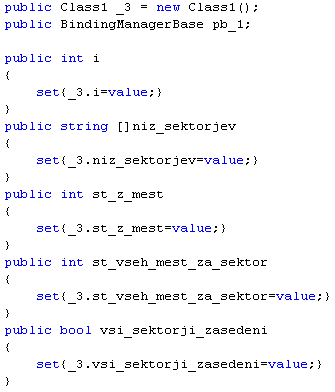


Slika 19: Princip delovanja ukaza Delete1, ki je zapisan v Access-u

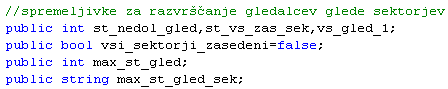
## Deklaracije objektov in funkcij za prenos

Na začetku programa je deklariran objekt, ki se sklicuje na razred Class1 in objekt pb\_1 za povezovanje z bazo podatkov.

Določene so tudi funkcije ,ki sprejmejo vrednosti prejšnjega form-a in jih prenesejo. Deklarirane so tako da jim določimo ime kot spremeljivki in podatkovni tip označimo jo kot javno spremenljivko. V glavnem delu funkcije z ukazom set določimo v katero spremeljivko prenesemo vrednost ali value.

using_ole_db.bmp

Slika\_20: Deklaracija objektov in funkcij za prenos vrednosti in vključitev upravljanja z bazo podatkov desno



Slika\_21: Spremenljivke za razvrščanje gledalcev glede na sektorje

## Vpisovanje podatkov

### Začetni vpis vrednosti ob pojavu okna

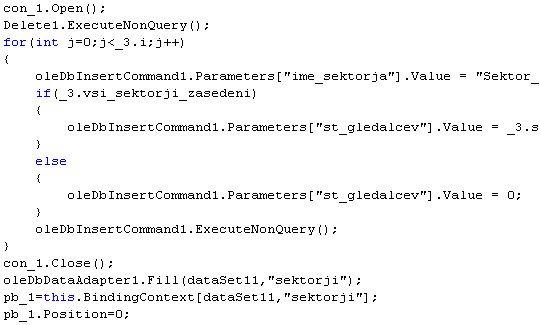
Ko se pojavi okno za razvrščanje gledalcev glede sektorjev se izbrišejo prejšnji zapisi v podatkovni bazi s klicom ukaza Delete1.ExecuteNonQuery().

V nadaljnem delu programa se preko for zanke, ki ima končno vrednost števec sektorjev ali i(prenesena spremenljivka) vpišejo podatki o imenih sektorjev preko oleDbInsertCommand1 s sklicevanjem na parametre in privzeta vrednost za število gledalcev je 0, če je spremenljivka za določanje zasedenosti vseh sektorjev false. V nasprotnem primeru je število gledalcev v bazi podatkov enako številu vseh mest za sektor. Za vpis podatkov na koncu napišemo oleDbInsertCommand1.ExecuteNonQuery().

Ta del programa mora potekati pri opprti povezavi, ki se določi z ukalom con\_1.Open() in se zapre z ukazom con1.Close().

Funkcija vpiše zapise od dataSet11 in tabele sektorji ta preko data adapterja vpiše zapise. Tako da v primeru tabele ali polj, ki so povezana s podatkovno bazo se v te vpišejo vrednosti iz podatkovne baze.

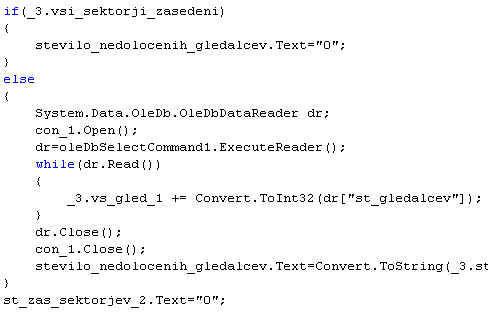
Tudi objekt pb\_1 se poveže z dataSet 11 in tabelo sektorji in je postavljen na začetno pozicijo v podatkovni bazi 0.



Slika\_22: Začetni ukazi za vpisovanje podatkov

Potem program preveri vrednost logične spremenljivke za vse zasedene sektorje, če je vrednost true potem se v polje za število nerazporejenih gledalcev vpiše vrednost 0 v nasprotnem primeru se deklarira števec dr, ki se poveže z oleDbSelectCommand1, ki se tudi določi kot števec. Potem se z while zanko podatki o številu gledalcev berejo iz podatkovne baze in seštevajo ter shranjujejo v spremeljivko vs\_gled\_1 . Potem se števec dr zapre in vse od začetka povezave števca z oleDbSelectCommand1 do konca izvajanja poteka pri odprti povezavi con\_1.

Vrednost nedolocenih gledalcev se prenese v polje število nerazporejenih gledalcev tako, da se od števila zasedenih mest odšteje vs\_gled\_1(vsota gledalcev, ki so razporejeni). Polje število zasedenih sektorjev pa se postavi na vrednost 0.



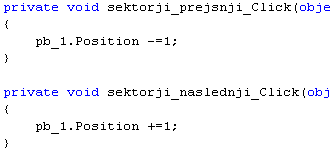
Slika\_23: Ukazi za določanje nerazporejenih gledalcev

Na začetni strani je polje za ime sektorja je povezano z bazo sektorji s stolpcem ime\_sektorja polje število gledalcev je povezano s stolpcem st\_gledalcev.

## Upravljanje s podatkovno bazo

### Premikanje po bazi in shranjevanje

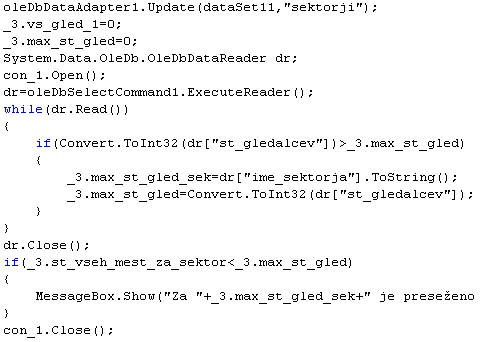
Določena sta dva gumba, ki omogačata premikanje po podatkovni bazi z povečevanjem ali zmanjševanjem števca, ki je deklariran že na začetku programa in se poveže z bazo pri odprtju okna(form).



Slika24: Premikanje po podatkovni bazi z spremembo števca

Če kliknemo gumb za shranjevanje podatkov se vpisani podatki v bazi shranijo vrednost spremenljivk vs\_gled\_1 in max\_st\_gled pa se postavi na vrednost 0. Potem se vzpostvi povezava z podatkovno bazo in deklarira se števec, ki se poveže z oleDbSelectCommand1. V nadaljevanju se preverja pogoj, če je število prebranih gledalcev iz zapisanih podatkov večje od vrednosti spremenljivke max\_st\_gled. Če je se spremenljivki max\_st\_gled priredi trenutna vrednost gledalcev prav tako se spremenljivki za določitev sektorja z največ gledalci priredi vrednost imena sektorja. Vse se vpisuje glede na trenutno lokacijo v podatkovni bazi. Podatki se preverjajo po vrsti kot so zapisani.

Objekt branja podatkov zaključi izvajanje in v primeru, če največje število gledalcev, ki je vpisano v podatkovni bazi presega določeno število mest za sektor javi napako za določen sektor, da presega število gledalcev za sektor. In povezava z bazo se prekine.



Slika\_25: Del funkcije za preverjanje števila vpisanih sektorjev

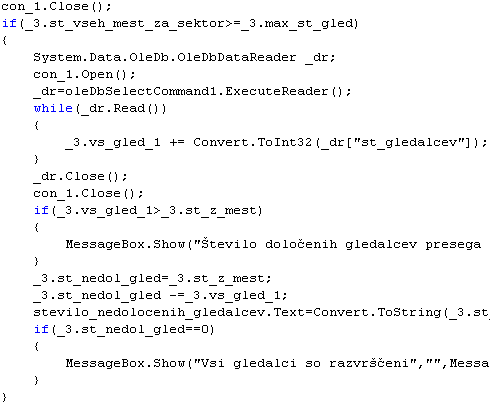
Funkcija nadaljuje izvajanje tako, da v primeru, če je število vseh mest za sektor večje ali enako največjemu številu sektorjev. Ali, če vrednost katerega izmed vpisanih sektorjev ne presega število določenih mest za posamezen sektor.

Tudi ta del funkcije generira števec prebere vse podatke o številu gledalcev, ki jih shrani spremenljivka vs\_gled\_1. Na koncu se zaključi branje in povezava, ki se na začetku vzpostavi se zapre.

V nadaljevanju se preveri, če je vsota sektorjev večja od števila zasedenih mest program(funkcija) javi napako, da število določenih gledalcev presega število nerazporejenih. In od nedoločenega števila gledalcev se odšteje vsota seštetih gledalcev, ki so vpisani v bazi, ter prenese v polje za število nerazporejenih gledalcev.

Tako, da velja, če so v polju pozitivna števila vsi gledalci niso razvrščeni, če je negativna je preveč določenih gledalcev (negativno število nakazuje koliko je teh gledalcev). In če je vrednost 0 so vsi podatki razvščeni.

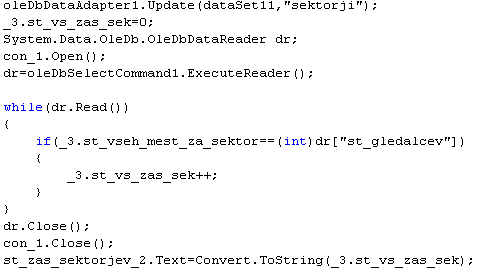
Tako program na koncu še preveri pogoj, če je vrednost nerazvrščenih gledalcev nič in v primeru, če je javi informacijo o zasedenosti vseh sektorjev.



Slika\_26: Del funkcije, ki določa spremembo razvrstitve gledalcev

## Funkcija za izračun števila sektorjev

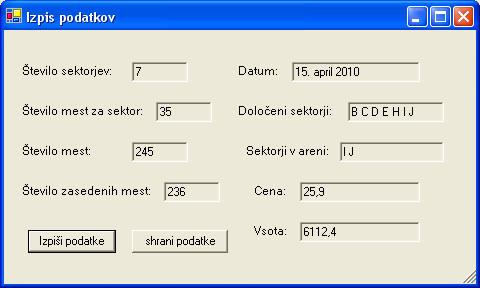
Funkcija shrani obstoječe podatke v podatkovni bazi v primeru spremembe in spremenljivko st\_vs\_zas\_sek postavi na vrednost nič. Genererira se števec, ki se na isti način kot pri drugih funkcijah poveže z bazo podatkov. S tem da preverja, le če število vseh mest za sektor sovpada s številom gledalcev za posamezen sektor glede na poziciji na kateri se nahaja. Če sovpada se vrednost števca, ki šteje število zasedenih sektorjev poveča za ena. Ko prebere in primerja vse podatke v bazi podatkov se vrednost števca prenese v polje za prikaz zasedenih sektorjev.



Slika\_27: Funkcija, ki izračuna število zasedenih sektorjev

# Izpis podatkov

## Začetne definicije programa

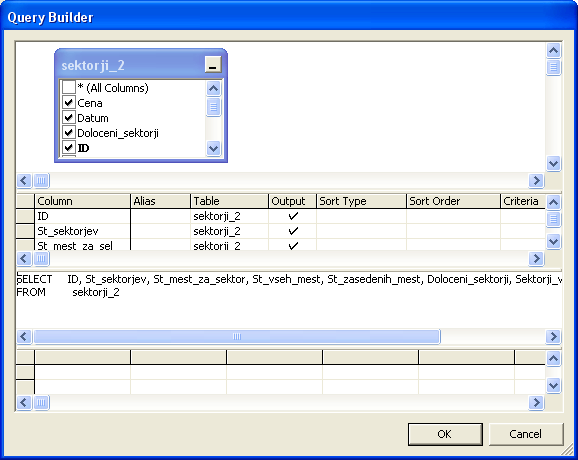


Slika\_28: Primer izpisa vpisanih podatkov in izračunanih vrednosti

## Povazava z bazo podatkov

izpis_pod_dol_baze.bmp

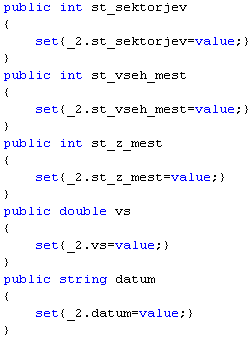
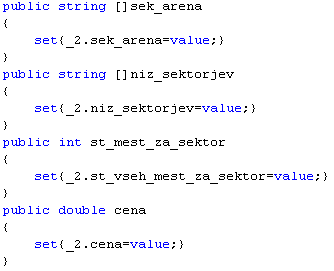
Slika\_27: Komponente za upravljanje z bazo, ki so vključene pri izpisu podatkov



Slika\_29: Razvrstitev stolpcev pri podatkovni bazi za izpis podatkov z ukazi v Access-u

## Deklaracije objektov in funkcij za prenos

razred_dol.bmp

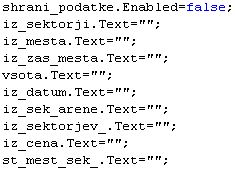


Slika\_30: Deklaracija razreda in funkcij ter nizov za prenos prejšnjih vrednosti

Razred se deklarira za dostop do spremeljivk razreda in funkcije za vpis vrednosti spremeljivk iz prejšnjega forma. Niz sprejme podatke podobno kot spremeljivka le, da je pred imenom niza določen znak [](kar označuje niz).

# Izpis vrednosti programa

## Začetna določitev

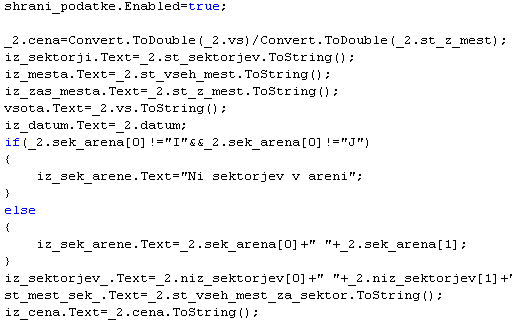


Slika\_31: Začetna določitev vrednosti programa vsem poljem se zbrišejo zapisi in gumb za shranjevanje je onemogočen

## Izpis podatkov

Pri kliku na gumb Izpiši podatke se gumb shrani podatke omogoči in ostale vrednosti se prenesejo v indexe(polja).

Prenesejo se vrednosti za število sektorjev, število vseh mest, število vseh mest za sektor, število zasedenih mest, vsota, datum, cena. Vpišejo se še oznake sektorjev po indexih, če so zapisane. In sektorji v areni pod pogojem, da ni zapisa se izpiše Ni sektorjev v areni, če so sektorji določeni se vpišeta indexa teh. Drug index se vpiše le če je določen.



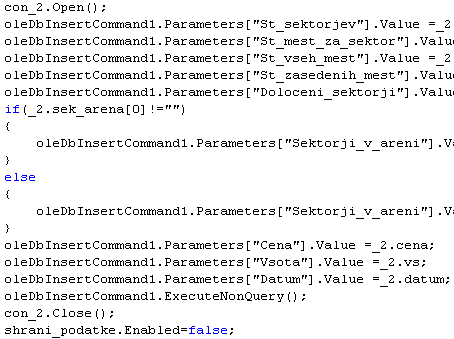
Slika\_32: Funkcija, ki vpiše vrednosti v polja(indexe),ki so določeni v formu(oknu).

## Shranjevanje podatkov

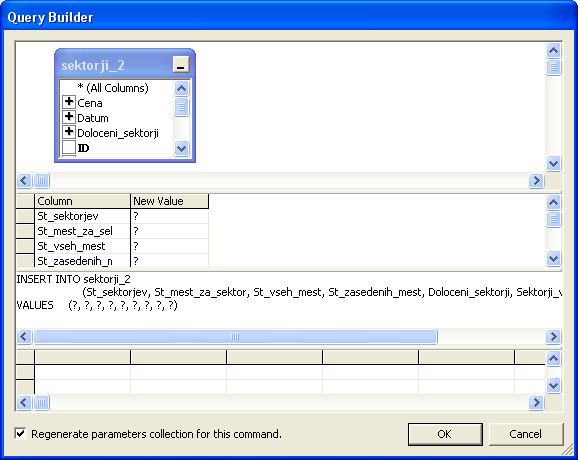
Shranjevanje se izvede s klikom gumba shrani podatke, če je omogočen.

Funkcija,ki se izvaja vzpostavi povezavo z bazo in preko ukaza oleDbInsertCommand1 in s sklicevanjem na vpisane parametre vpiše vrednosti spremenljivk. Za vpis sektorjev v areni se preveri določitev prvega indexa in, če je določen se sektorji vpišejo, če ni določen pomeni, da ni sektorjev v areni in v index se vpiše vrednost Ni sektorjev v areni.

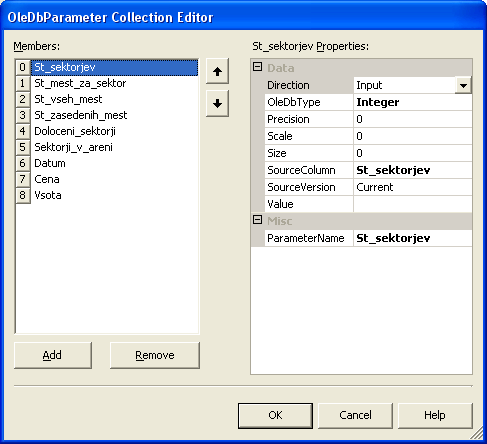
Na koncu ukaz oleDbInsertCommand1.ExecuteNonQuery() omogoči vpis vseh podatkov. In povezava z bazo se zaključi. Gumb za shranjevanje se onemogoči.



Slika\_33: Prikaz funkcije in načina vpisovanja podatkov



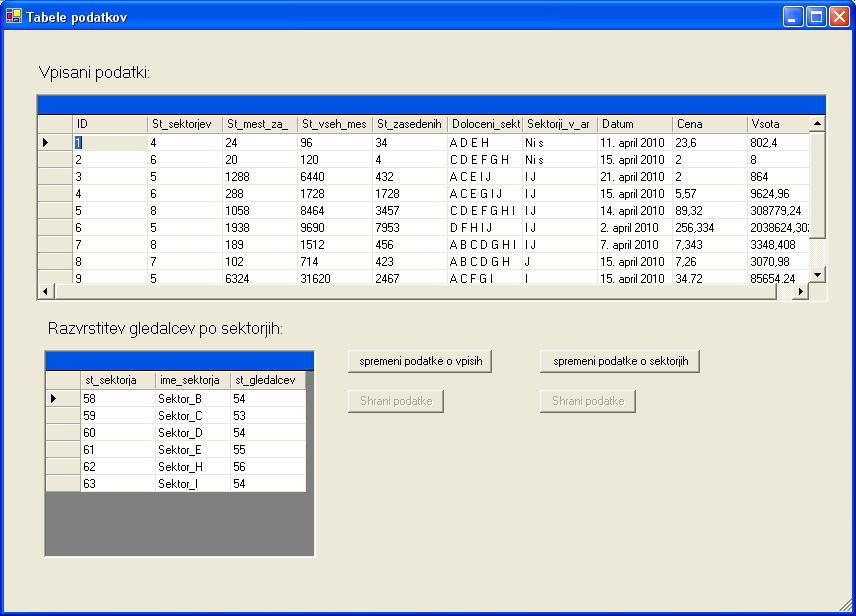
Slika\_34: Določitev ukaza za vpisovanje podatkov v Access-u, ki se ponavadi generira ob zaključku konfiguracije data adapterja



Slika\_35: Določitev parametrov za vpisovanje podatkov. Določeni so tudi podatkovni tipi posameznih parametrov

# Tabele podatkov

## Definicija programa

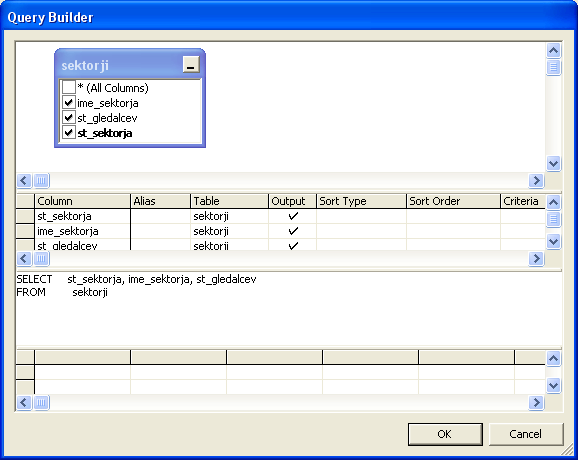


Slika\_36: Prikaz tabel vseh vpisanih podatkov

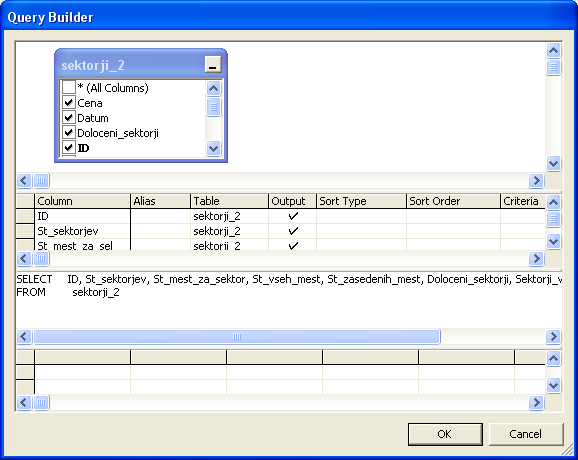
## Povezava s podatkovno bazo

komp_tabele.bmp

Slika\_37: Komponente za prikaz tabel podatkov



Slika\_38: Določitev prikaza stolpecv sektorjev v tabeli sektorji, ki je povezana preko data adapterja 3

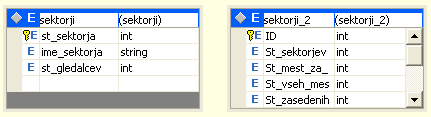


Slika\_39: Določitev prikaza stolpcev vseh zapisov v tabeli sektorji\_2, ki je povezana preko data adapterja 4

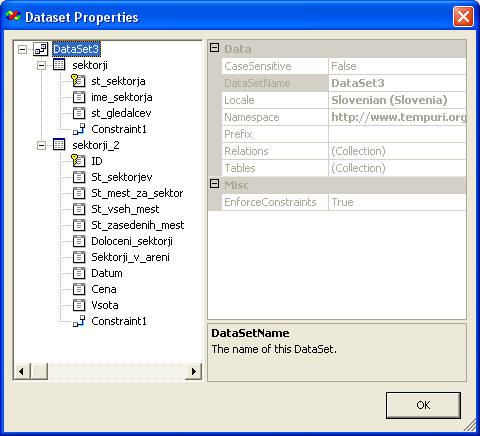
con\_3 določa kje se nahaja podatkovna baza sektorji: «C:\Vodenje športne dvorane\sektorji.mdb«.

con\_4 določa kje se nahaja podatkovna baza sektorji\_2: »C:\Vodenje športne dvorane\sportna\_dvorana.mdb«.

dataSet31 vsebuje podatke o obeh tabelah, ki sta vključeni v dataSet31 pri generaciji.



Slika\_40: Primer prikaza tabel in podatkovnih tipov za obe tabeli



Slika\_41: Primer prikaza polj obeh tabel, ki jih zajema dataSet31

## Prikaz podatkov

Na začetku je dataGrid1 povezan s tabelo preko dataSet31 in dataGrid2 s tabelo sektorji\_2. Ko se pojavi okno se podatki iz tabel vpišejo v dataGrid1 in dataGrid2. Preko ukaza data adapter-ja z izbiro dataSet31 in tabele.



Slika\_42: Vpis vrednosti, ki so zapisane v podatkovni bazi

## Spreminjanje in shranjevanje zapisov

Če kliknemo na gumb spremeni podatke o vpisih ali spremeni podatke o sektorjih se glede na izbiro omogoči vpisovanje ali spreminjanje zapisov določene tabele preko dataGrind, ki je bila prej označena samo za branje se spremeni tako, da lahko vpisujemo ali spreminjamo podatke. Gumb shrani podatke, ki se tudi navezuje na določeno izbiro postane omogočen za klik.

sprem_1.bmp

sprem_2.bmp

Slika\_43: Funkciji za omogočanje spreminjanja podatkov in kasnejšega shranjevanja

V primeru, če je kateri od gumbov za shranjevanje omogočen se pri kliku na ta gumb preko data adapterja shranijo spremenjeni ali nespremenjeni zapisi določene tabele. Določen dataGrind (prikaz tabele) se spremeni na samo berljivega in gumb se onemogoči za klik.

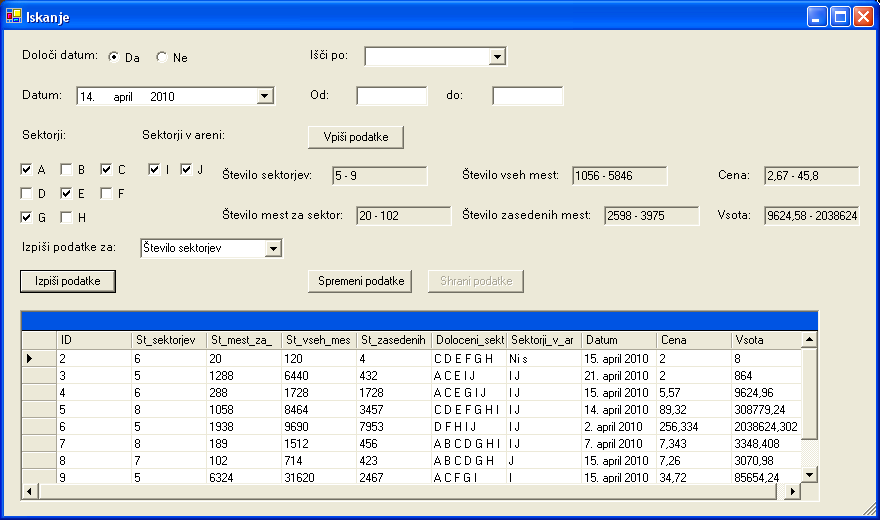
shrani_1.bmp

shrani_2.bmp

Slika\_44: Funkciji za shranjevanje podatkov, in označitve samo za branje

# Iskanje

## Definicija programa

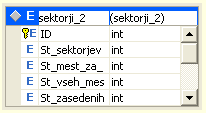


Slika\_45: Primer prikaza ispisa podatkov glede na iskanje

## Povezava z bazo podatkov

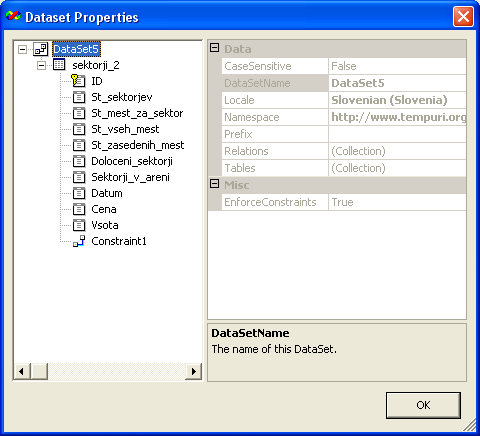
baze_pod_kom.bmp

Slika\_46: Komponente za upravljanje z bazo, ki so vključene pri iskanju



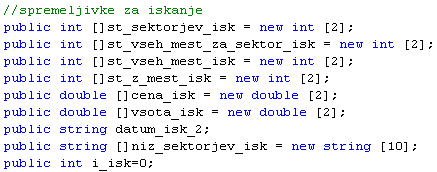
Slika\_47: Primer prikaza tabele in podatkovnih tipov od dataSet51

Med komponente za povezovanje z podatkovno bazo je vključen tudi dataView1, ki je povezan z dataSet51 in lahko filtrira podatke, ki jih program pozneje prikaže. S tem, da lahko pozneje dataView1 povežemo z določeno tabelo ali dataGrid1 ,ki prikaže filtrirane podatke.



Slika\_48: Določitev parametrov za vpisovanje podatkov. Določeni so tudi podatkovni tipi posameznih parametrov

## Deklaracija objektov in spremenljivke za iskanje

System_OleDb.bmpclass_6.bmp

Slika\_49: Določitev spremenljivk in nizov za iskanje podatkov levo, desno vključitev ukaza za upravljanje z bazo podatkov in deklaracija objekta razreda za dostop do spremenljivk in nizov

## Začetna določitev podatkov

Na začetku pojava okna se vrednost datuma postavi na današnji in označi se RadioButton(dol\_datum\_ne\_isk),ki polje za izbiro datuma onemogoči. Polja za določitev iskanja se izbrišejo(\_isk,polje za parametre).

Vse oznake sektorjev se postavijo na false(niso označene).

Zapis comboBox1 za iskanje se izbriše, gumba za spreminjanje in shranjevanje filtriranih podatkov nista omogočena za klik.

Vrednost vseh polj, ki vsebujejo vpisane podatke o zgornji in spodnji meji se izbrišejo in prevzamejo vrednost Ni definirano.

Podatki se filtrirajo glede na tiste, kjer je vrednost ID 0 in zato tabela ne prikaže nobenih podatkov.

Vrednosti vseh nizov, ki imata dva indexa za zgornjo in spodnjo mejo se preko for zanke pastavijo na vrednost nič. To so nizi za število sektorjev,število vseh mest za sektor, število vseh mest, število zasedenih mest, ceno in vsoto.

## Določitev iskanja podatkov

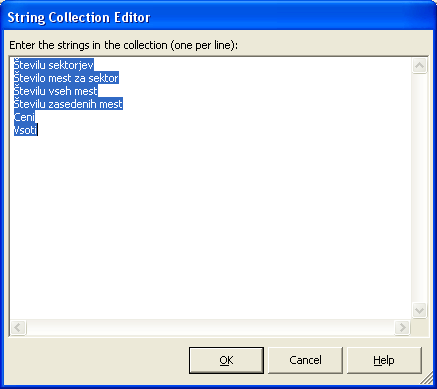
Na začetku programa je določena opcija o izbiri datuma z dvema izbirama (Radio button) s katerima lahko določimo ali je omogočeno iskanje po datumu. Na začetku je izbira datuma onemogočena.

dat_dol_1.bmp

dat_dol_2.bmp

Slika\_50: Omogočanje in onemogočanje izbiranje datuma glede na izbiro

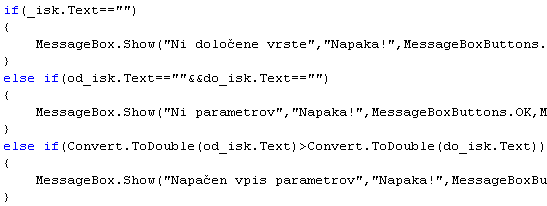
Desno od določitve datuma je comboBox označen z imenom \_isk s katerim lahko določimo iskanje določenih podatkov, katerih zgornje in spodnje meje določimo v dveh poljih pod \_isk, ki sta označeni z od\_isk do\_isk.



Slika\_51: Izbira glede določitve iskanja za določanje zgornjih in spodnjih mej comboBox-a \_isk

Ko so podatki določeni lahko s klikom na gumb Vpiši podatke shranimo vrednost spremenljivk zgornje in spodnje meje, ki se lahko pozneje uporabi za iskanje.

V primeru, če ni določena vrsta iskanja program javi opozorilo, da ni določene vrste iskanja, v primeru, če niso določeni parametri javi opozorilo, da ni parametrov in če vpis parametra prvega polja presega vrednost drugega javi napako o napačnem vpisu parametrov.



Slika\_52: Javljanje napak v primeru napačnih vpisov ali določitve

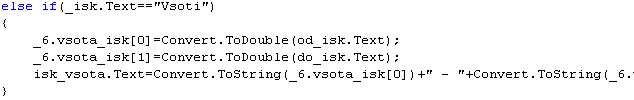
Ko so podatki določeni ali v primeru spremembe parametrov ali vrste iskanja se gumb Vpiši podatke omogoči za klik, ki se pa ob kliku na gumb Vpiši podatke ponovno onemogoči. In vrednost parametrov v poljih se izbriše.

_dol_na_koncu.bmp

Slika\_53: Prikazuje onemogočanje gumba za vpis in izbriše parametre iskanja v poljih

V primeru, če so podatki prav določeni se preveri vrsta izbire in, če sovpada z katero izmed vrst, ki so določene se vrednost zgornje in spodnje meje prenese v niz z dvema poljema s tem da se pretvori v ustrezen podatkovni tip.

Izpiše se tudi vrednost zgornje in spodnje meje v določenem polju, ki je označeno samo za branje glede na vrsto izbire iskanja.

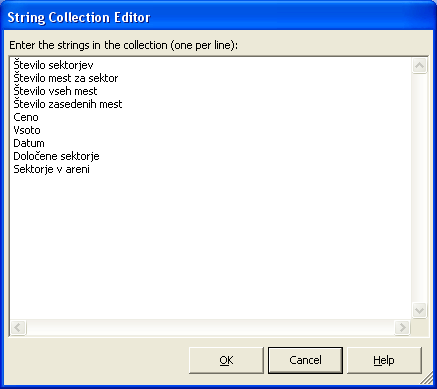


Slika\_54: Primer vpisa vsote

## Filtriranje podatkov

Levo pod določitvijo datuma je izbira za določitev oznak vseh sektorjev za iskanje.Z izbiro oznak lahko iščemo samo določene sektorje. Desno pa so prikazi določitve zgornjih in spodnjih mej glede na prej vpisane podatke. Če podatki niso bili določeni je zapis v polju Ni definirano.

Pod določitvijo izbire sektorjev je comboBox1 za izbiro, ki določa iskanje določenih podatkov, ki se pozneje lahko prikažejo v tabeli.

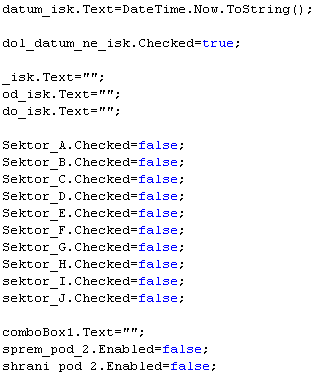
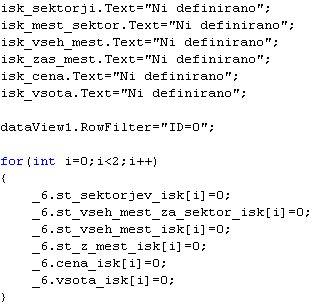


Slika\_55: Izbira glede iskanja podatkov, ki se določi v comboBox-u

Če kliknemo na gumb Izpiši podatke se najprej vrednost števca i\_isk postavi na vrednost nič. In vsi prejšnji sektorji v nizu se zbrišejo z znakom za preslednico preko for zanke.

V nadaljevanje se preveri označenost sektorjev in v primeru označitve se vpiše oznaka v niz in poveča se vrednost števca za 1 v primeru naslednjega vpisa glede na pogoj.

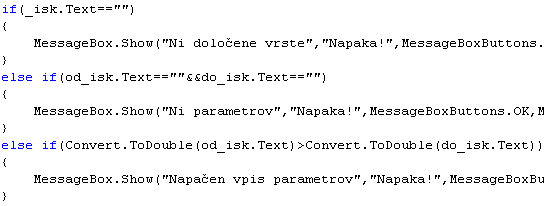
Zbriše se tudi vrednost spremenljivke za datum in vpiše se nov datum, če je določitev datuma omogočena.



Slika\_56: Začetna določitev vrednosti

Potem se vzpostavi povezava z podatkovno bazo preko con\_7 z ukazom con\_7.Open().

Najprej se preveri, če obstaja zapis v polju comboBox, če ne obstaja program javi opozorilo, da iskanje ni določeno.



Slika57: Opozorila v primeru napačne določitve

Potem se preveri zapis v comboBox-u in v polju, ki je določeno samo za branje(vse je zajeto v nizih večih if in else stavkov) in, če se ujema s comboBox-om glede na njegovo vrsto zapisa in je to polje različno od Ni definirano se podatki preko dataView1 filtrirajo glede na pogoje zgornje in spodnje meje, uporabljajo pa se ukazi v Access-u ,ki se navezuje na določen stolpec v tabeli podatkovne baze. To so primeri pogojev za iskanje, kjer so določene spremenljivke celih ali realnih števil.

primer_fil_cena.bmp

Slika\_58: Primer filtriranja cene

V primeru določitve datuma se pri drugem pogoju poleg vrednosti zapisa v comboBox-u preveri označitev RadioButton-a(dol\_datum\_da\_isk),če je vrednost tega true se filtrirajo podatki glede na izbran datum.

primer_filt_datum.bmp

Slika\_59: Primer filtriranja datuma

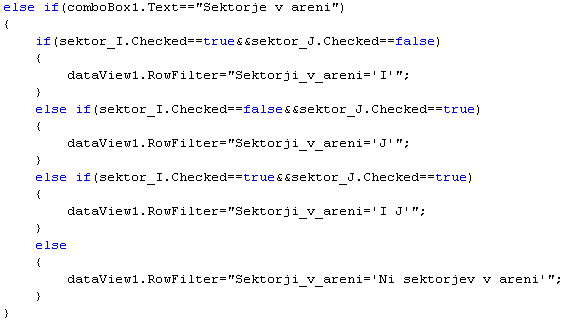
V primeru določenih sektorjev se preveri, če je vrednost prvega index-a niza sektorjev različna od nezapisanih vrednosti in se v tem primeru tudi izvede filtriranje s preverjanjem vseh možnih zapisanih sektorjev. Nezapisani so označeni s preslednico zaradi načina vpisa in uspešnega filtriranja.

primer_filtr_dol_sek.bmp

Slika\_60: Primer filtriranja določenih sektorjev

Če pa so določeni sektorji v areni se preverijo vse možne kombinacije določitve dveh sektorjev v if stavkih. Če pa niso določeni se preverja zapis Ni sektorjev v areni.

Na koncu se preko data adapterja vpišejo podatki v dataSet51 zaradi česar se filtrirani podatki pojavijo tabeli. In na koncu se prekine povezava z bazo podatkov z ukazom con\_7.Close(). S tem, da se na koncu še preveri, če je vrednost comboBox-a različna od nezapisane vrednosti in v tem primeru se gumb za spreminjanje podatkov za klik omogoči.



Slika\_61: Primer filtriranja sektorjev v areni

## Spreminjanje in shranjevanje

Gumb Spremeni podatke je omogočen za klik v primeru izpisa ali neizpisa filtriranih podatkov glede na pogoj. S klikom na gumb se vpisovanje in spreminjanje podatkov preko dataGrid1 omogoči in gumb za shranjevanje podatkov se tudi omogoči za klik.

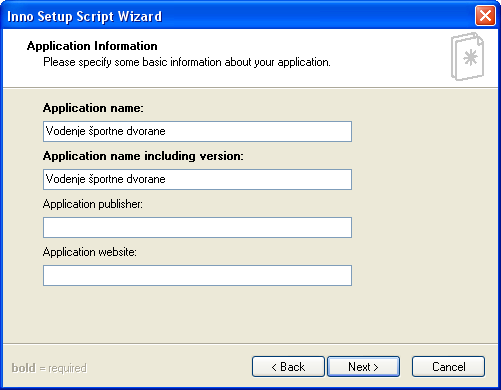
S klikom na gumb Shrani podatke se podatki v bazi podatkov shranijo preko data adapterja in dataGrid1 se onemogoči za vpisovanje in spreminjanje podatkov. Prav tako se gumb Shrani podatke onemogoči za klik.

# Povezava v inštalacijsko datoteko Inno Setup 5

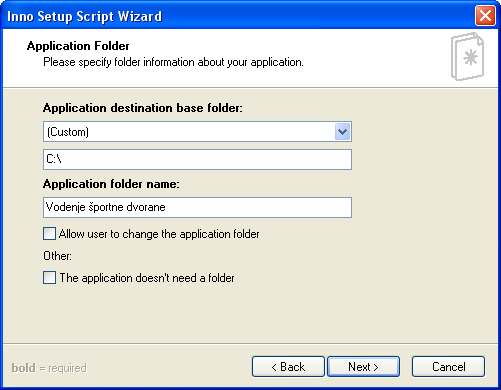
Ko zaženemo Inno setup 5 kliknemo File→New ali CTRL+N. Pojavi se čarovnik za povezavo programa v inštalacijsko datoteko.



Slika\_62: Začetno okno Inno Setup 5

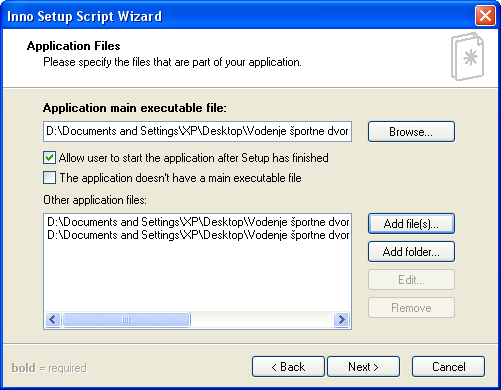


Slika\_63: Naslednje okno za določitev imen in osnovnih podatkov

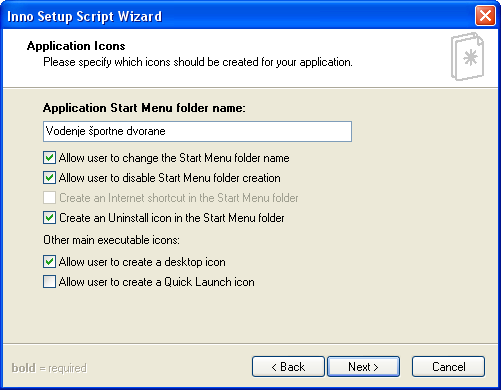


Slika\_64: Okno za določitev inštalacijske lokacije in njeno ime, ter možnost brez lokacije in spreminjanje inštalacijske lokacije.

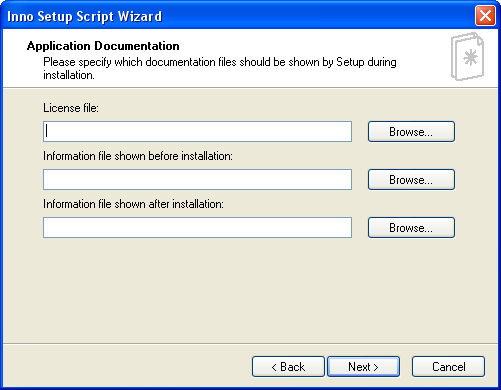
Za vodenje športne dvorane mara biti določena stalna lokacija zaradi načina povezave podatkovne baze in se ne more spreminjati.



Slika\_65: Izbira izvršilne datoteke in ostalih komponent (v tem primeru podatkovnih baz).



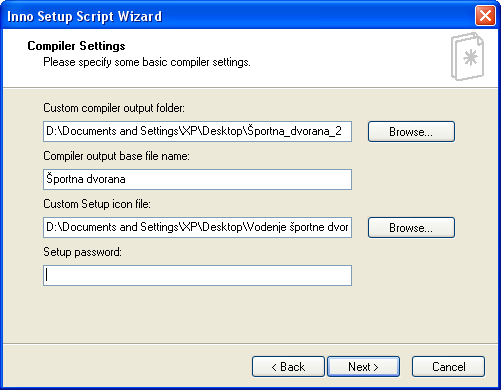
Slika\_66: Določitev aplikacije v Start Menu mapi in ikon na namizju



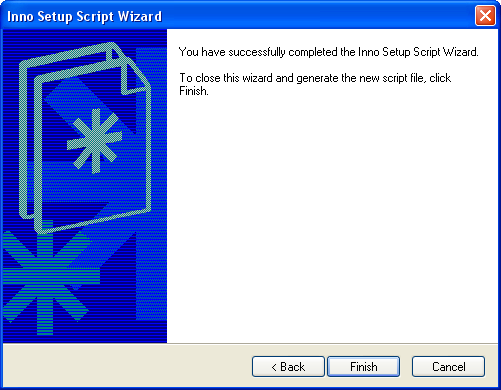
Slika\_67: Lahko vključimo podatke o licenci ter druge podatke pred in po inštalaciji



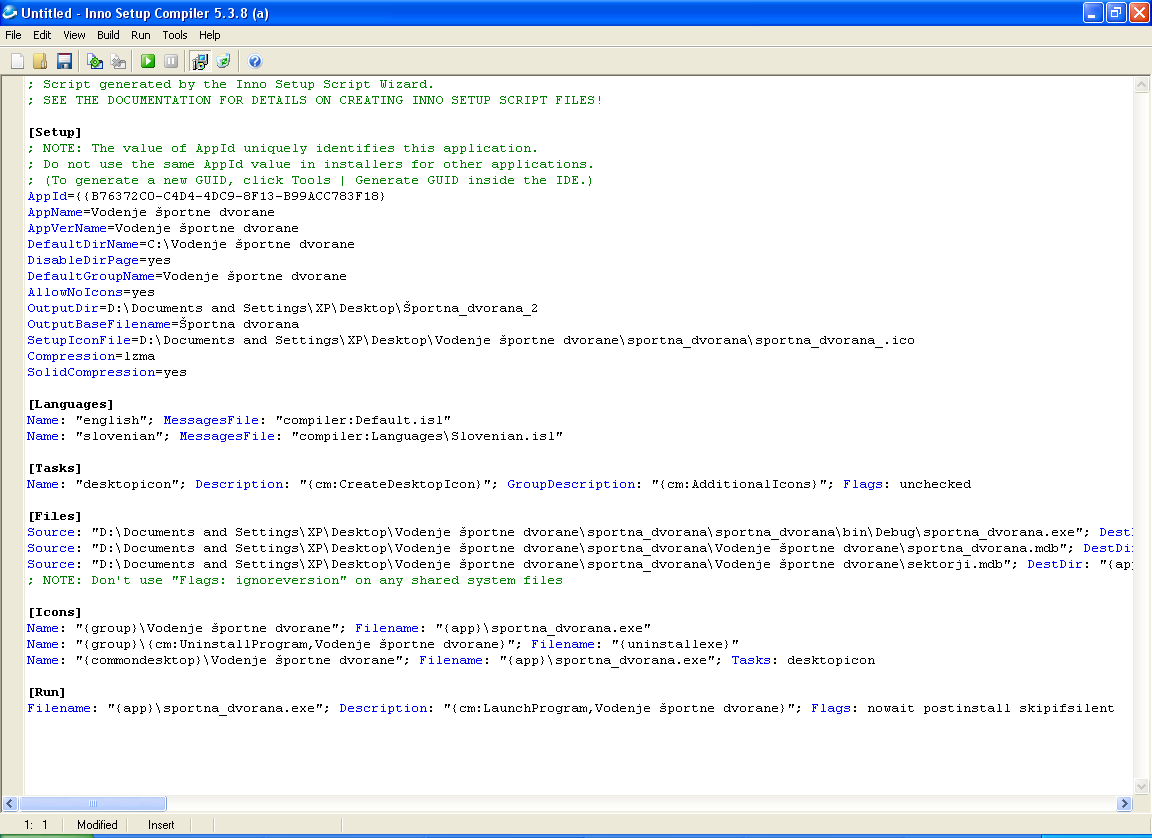
Slika\_68: Določitev izbire jezika



Slika\_69: Določitev lokacije inštalacijske datoteke in ime, ter možnost vključitve ikone in zaščitnega gesla



Slika\_70: Zaključitev povezovanja programa v izvršilno datoteko



Slika\_71: Programska koda prikaže vse podatke za povezovanje, če zaženemo program povežemo vse določene elemente v inštalacijsko datoteko

# Zaključek

Ko program prevedemo ga lahko zaženemo in inštaliramo brez programa C# ali izvršilne datoteke. Program je tako prenosljiv na več računalnikov.

Program vodenje športne dvorane se lahko uporablja za vpisovanje določenih podatkov, ki se uporabljajo za vodenje športne dvorane. Kot so določanje sektorjev, števila gledalcev, cene vstopnice itd. Omogoča tudi razvrščanje gledalcev glede sektorjev, izpisovanje in njihovo shranjevanje.

Zadnji opciji sta, da lahko prikaže tabele vpisanih podatkov, ki se lahko spreminjajo in filtrira podatke glede na določene parametre, ter jih prikaže v tabeli.

# Viri

http://www.microsoft.com/slovenija/msdn/csharp/pregled.mspx