

Andmebaasi mõiste ja kasutusvaldkonnad

Kõige lihtsam viis on andmeid hoida mingite säilitusühikute kaupa: raamatukogus raamatute ja riulite kaupa, arvutis failide ja kaustade kaupa.

Klassikalise definitsiooni järgi on **andmebaas** korrastatud ja loogiliselt seostatud andmete hulk, mida säilitatakse andmekandjal minimaalse liiasusega ning mida võivad üksteisest sõltumatult kasutada mitu rakendusprogrammi. Termin "andmebaas" (*data base, Datenbank*) ilmus käibesse 20. sajandi 60.aastate keskel, kui arvutis hakati programsete vahenditega arendama andmete integreeritud kasutust.

Andmebaasid on näiteks:

- raamatukogu kartoteek,
- telefoniraamat,
- klassipäevik,
- teie isiklik telefonimärkmik,
- rahvastikuregister,
- ARK register,
- piirivalve andmebaas,
- jne.

Lühidalt:

Andmebaas on kogum andmetest, mida mingil põhjusel on vaja säilitada.

Andmebaas:

- ✓ *realiseerib andmemudeli*
- ✓ *võimaldab ühiskasutust*
- ✓ *minimeerib andmete liiasuse*
- ✓ *eristab andmete füüsilise säilitamise andmekandjal nende kasutamisest,*
- ✓ *võimaldab andmete muutmist ilma süsteemi teisi komponente muutmata.*

Tabelitöötluste ja andmebaaside võrdlus

Elektrooniline andmebaas võib olla...

- **lameandmebaas:**

Lameda andmebaasi mõistet kasutatakse mõnikord ühe-tabeli-süsteemi puhul (näiteks Exceli tööleht või selle osa). Sel juhul polegi andmemudelit vaja, sest tegemist on vaid ühe olemiga (seoseid ei saa kajastada). Kõige parem on lamedat andmebaasi ette kujutada andmetabelina (millele füüsiliselt võib vastata fail). Tabeli ridu nimetatakse kirjeteks ja tabeli veerge väljadeks, mis kirjeldatakse tavaliselt (tabeli päises) järgmiste komponentide abil:

- ◆ nimi;
- ◆ tüüp (näit. *integer, real, character, array, date, picture*);
- ◆ kitsendused (näit. minimaal- ja maksimaalväärtus, väärtusvaru, maksimaalne sümbolite arv jne.).

Selliseks andmebaasiks võib olla näiteks elektrooniline telefoniraamat.

- **relatsioonandmebaas:**

Relatsiooniline andmebaas koosneb tabelitest ehk relatsioonidest. Erinevalt lamedast andmebaasist sisaldab ta mitut seostatud tabelit. Kasutaja tasandil hakkame me MS Accessis tegutsema just selle andmemudeliga.

Relatsioonilise andmebaasi loomine ja kasutamine MS Access-i vahenditega

Üldine ülesehitus:

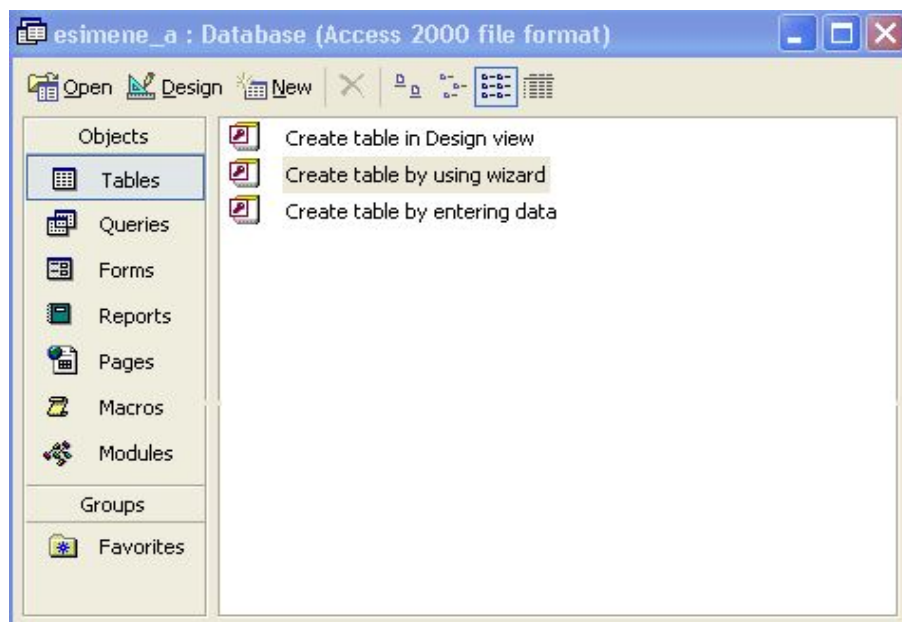
Microsoft Access võimaldab andmeid kirjeldada nende sisulist kokkukuuluvust arvestades. Andmetabeleid on võimalik omavahel siduda ja kasutada andmeid paljudest tabelitest korruga. Tabelite sidumine võimaldab andmete paremat säilitamist ja paindlikumat toimetamist, realiseerides seega andmebaasidele omaseid funktsioone.

Andmebaasi avamine ja salvestamine:

Uue andmebaasi loomiseks valige peale MS Access'i avamist algdialoogis (paremal pool aknas) Blank Database või menüüst File>New Database. Sisestage andmebaasile nimi ja salvestage soovitas kaustas.

Olemasoleva andmebaasi avamiseks valige algdialoogis Open an Existing Database või menüüst File> Open Database...

Andmebaasi aken:



Accessi andmebaas on konteinerfail (laiendiga *.mdb), mis võib sisaldada järgmist tüüpi objekte:

- tabelid (*tables*) on andmebaasi põhikomponendiks; ta peab sisaldama primaarvõtit - see on väli tabelis, mille sisu on iga kirje jaoks erinev; tabelite vahel on võimalik defineerida seoseid,
- päringud (*queries*) on põhivahend vajaliku info kättesaamiseks ühest või mitmest seotud tabelist,
- (ekraani)vormid (*forms*) on graafilised elemendid andmete sisestamiseks ja muutmiseks; ekraanvorme on otstarbekas kasutada kui andmeid tuleb sisestada, muuta või sageli vaadata.
- (trüki)aruanded (*reports*) esitavad andmebaasi andmeid ülevaatlikul kujul paberil; põhieesmärgiks on teha vahe- ja lõppkokkuvõtteid, lisada tekste, jaotada tekst lehekülgedeks, nummerdada lehekülgi.

Neid saab valida andmebaasiakna lipikute abil. Hiireklõps lipikul avab vastavat tüüpi objektide loetelu, millest tuleb topeltklõpsuga valida vajalik tabel, vorm jne.

Korruga saab töötada ainult ühe andmebaasiga, konteinerfaili füüsilisele struktuurile kasutaja ligi ei pääse (näiteks ei saa kasutada korraldust *Save as...*). Andmebaasi suurus piirab praktiliselt ainult kõvaketta maht.

Andmebaasi loomine:

Esimene asjana, kui üldse andmebaasi tegema hakata, tuleb meil selgeks teha, millist infot peaks loodav andmebaas endas sisaldama. Andmebaasist pole kasu, kui sealt vajalikku infot kätte ei saa. Andmebaas peab rahuldama nii hetke- kui tulevikuvajadusi. Seetõttu tuleb taotleda seda, et vajalike objektide kohta sisestataks piisavalt põhjalik info, kuid välditaks üleliigse info sisestamist.

Kõige esmalt tuleb paberile kirja panna, milliste objektide kohta infot vajatakse ja milliseid andmeid on iga objekti kohta vaja. Kui tabeli koostamiseks ei kavatseta kasutada tabelitarka, siis tuleb eelnevalt määrata ka iga välja tüüp ja pikkus. Iga välja kohta tuleb otsustada, kas see sisaldab ainult numbraid (*Number*) või mitmesuguseid märke, viimisel juhul on tegemist tekstiväljaga (*Text*).

Andmetüübid, nende omadused:

Igal tabeli andmeväljal (veerul) on kindlad omadused, mida iseloomustavad:

- **väljanimi** - võivad koosneda mitmest sõnast, sisaldada eesti täppidega tähti; pikkus kuni 64 sümbolit,
- **välja tüüp:**

<i>Tüüp</i>	<i>Sisu</i>	<i>Pikkus</i>
Tekst	Sümbolid	kuni 255 baiti
Memo	Sümbolid	Kuni 32000 baiti
Number	Täis- või murdarvud	1,2,4 või 8 baiti
Kuupäev/aeg	Kuupäevad ja kellaajad	8 baiti
Raha	Rahasummad	8 baiti
Loendaja	Kirjete järjekorranumbrid	4 baiti
Jah/Ei	Loogilised väärtused	1 bitt
OLE objekt	Graafikud, pildid, helid või muud binaarfailid	Kuni 1GB

- **välja kirjeldus** - kirjeldus kuni 64 sümbolit (väljastatakse teadeteraale – näiteks juhis välja täitmiseks),
- **välja omadus:**

<i>Omadus</i>	<i>Mida määrab</i>
<i>Field size</i>	Teksti- või numbrivälja pikkuse
<i>Format</i>	Kuidas andmeid kuvatakse
<i>Decimal Places</i>	Kümnenkohtade arvu pärast koma
<i>Input Mask</i>	Kuidas andmeid tuleb sisestada
<i>Caption</i>	Välja nimetus, mida kasutatakse vormidel ja aruannetel

<i>Omadus</i>	<i>Mida määrab</i>
<i>Default value</i>	Vaikeväärtus
<i>Validation Rule</i>	Kontrollimiseeskiri andmete sisestamisel
<i>Validation Text</i>	Veateade sisestusvea korral
<i>Required</i>	Kas väli peab olema täidetud
<i>Allow Zero Length</i>	Kas nullpikkusega stringid on lubatud
<i>Indexed</i>	Kas välja sisu kasutatakse indeksina

Numbrite vormingud:

- *standard* (kaks kümnendkohta)
- *currency* (lisatakse rahaühik)
- *fixed* (ümardatakse lähima täisarvuni)
- *percent* (korrutatakse sajaga)
- *scientific* (esitatakse poollogaritmilisel kujul)

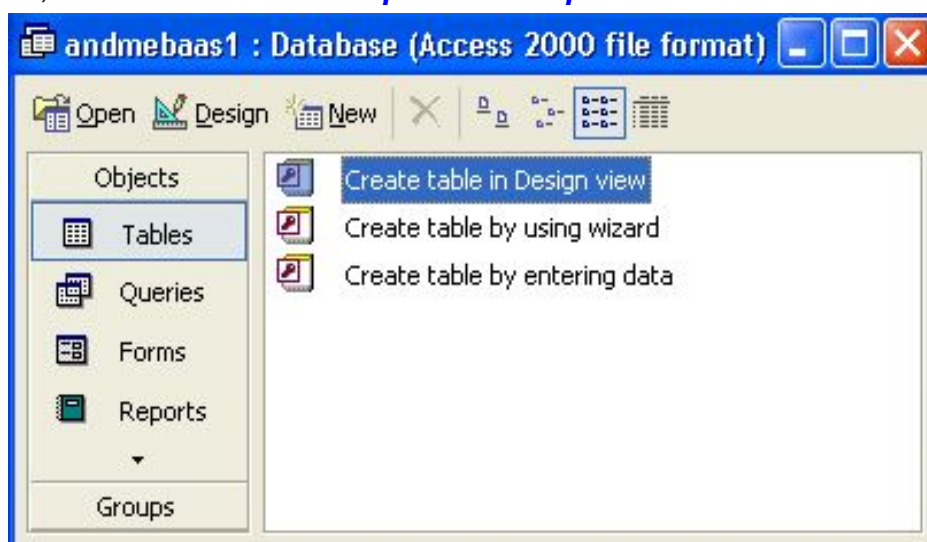
Numbrite pikkused:

- *byte* (arvud 0-255)
- *single* ($\pm 3,4 \cdot 10^38$)
- *double* ($\pm 1797 \cdot 10^{308}$)
- *integer* (± 32768)
- *long integer* (± 2147483648)

Kümnendkohti võib olla single puhul 7, double puhul 15.

Tabeli loomine

Uut andmebaasi on kõige lihtsam luua andmebaasitarga abil. Kui olete avanud MS Access'i ja valinud uue andmebaasi (*Blank Database*) ja selle salvestanud, siis avaneb teile uus aken, mis kannab nime: **teie poolt antud pealkiri: Database**.



Antud aknas saab valida objekti tüüpi, millega tahate antud andmebaasi teha. Vaatleme esmalt tabeleid. Tabeli veerge nimetatakse väljadeks (*Field*), ridu aga kirjeteks (*Record*). Igal väljal asuvad kindlat tüüpi andmed. Andmetüüpidest on juba eespool juttu olnud. **Kirje** on ühe objekti andmete kogum.

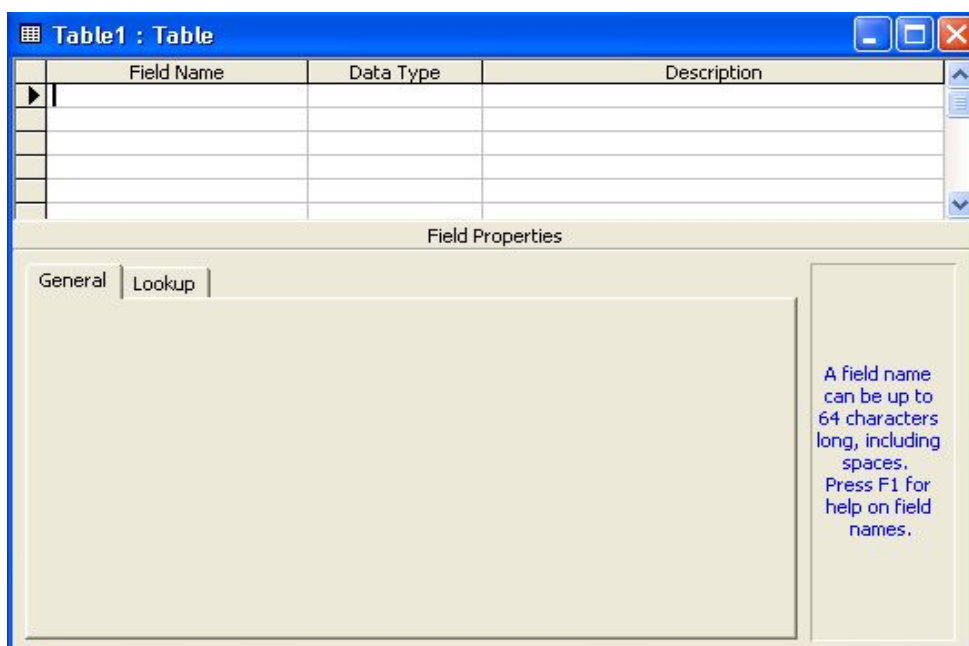
Esiteks aga tuleb paika panna, mille kohta andmebaas üldse luua. Proovime teha näiteks tabelid klassitunni ja asukoha kohta ning hiljem vaatame, kuidas on võimalik neid kahte tabelit omavahel siduda.

Tabelite loomiseks on mitu võimalust.

Esimesel juhul:

Soovitav on kasutada tabeli loomist kujundusvaates (*Design View*). Selleks valige andmebaasi aknas *Tables* ja topelt hiireklõps valikul *Create table in Design View*.

Avanevas aknas



tuleb määrata:

- väljade nimed (*Field Name*) – ei või sisaldada punkti,
- andmetüübid (*Data Type*) – vali lahtrisse sisenemisel avanevast rippmenüüst,
- võib lisada ka välja kirjelduse (*Description*), mis ilmub tabelivaates ekraani allservas olevale infoleale.

Täidame klassitunni tabeli selliste andmetega:

<i>Field Name</i>	<i>Data Type</i>	<i>Description</i>
Nimi	Text	Tunni nimetus
Nime_ID	Text	Tunni lühend
Esitaja	Text	Tunni läbiviija
Kestvus	Text	Tunni läbiviimisaeg
Ruum	Text	Tunni asukoht

Peale andmete sisestamist, tuleb määrata **primaarvõti** ehk tunnusmärk, mis iga kirje jaoks on erinev. Access ei luba sisestada tabelisse kirjet, mille primaarvõti langeb kokku juba tabelis oleva kirjega. Antud tabelis võiks selleks olla **Nime_ID**. Seega, et teha tunni lühendit primaarvõtmeks, tuleb minna antud tabelis Nime_ID peale ja valida parema hiireklahviga *Primary Key*. Siiani näeb tabel välja selline (ilma primaarvõtmega):

Field Name	Data Type	Description
Nimi	Text	Tunni nimetus
Nime_ID	Text	Tunni lühend
Esitaja	Text	Tunni läbiviija
Kestvus	Text	Tunni läbiviimisaeg
Ruum	Text	Tunni asukoht

Field Properties

General | Lookup

Läheme antud akna alumise poole juurde. Selleks valite ülemises pooles vajaliku välja nime ja vajutate klahvi F6. Cursor siirdub alumisse aknapoolde, kus on kuvatud valitud välja omadused. Selles pooles saab iga sisestatud välja sisu struktuuri muuta.

Väljade omadustest on juttu olnud juba eelmistel lehekülgedel, kuid antud juhul võiks teha sellise muudatuse – valida sobilikult *kestvust*. Selleks lähete ülemises aknapooles väljale Kestvus ja liigute eespool kirjeldatult alumisse aknapoolde. Et kuju määrata, tuleb valida *Input Mask* rida. Lähete antud rea lõpus olevale kolmele punktile ja klikkide sellel. Selle tulemusena soovitakse teada, kas tahate tabelit salvestada ja te valite *Yes*.

Salvestage nimega **TUND**.

Peale seda avaneb teile selline aken:

Input Mask Wizard

Which input mask matches how you want data to look?

To see how a selected mask works, use the Try It box.

To change the Input Mask list, click the Edit List button.

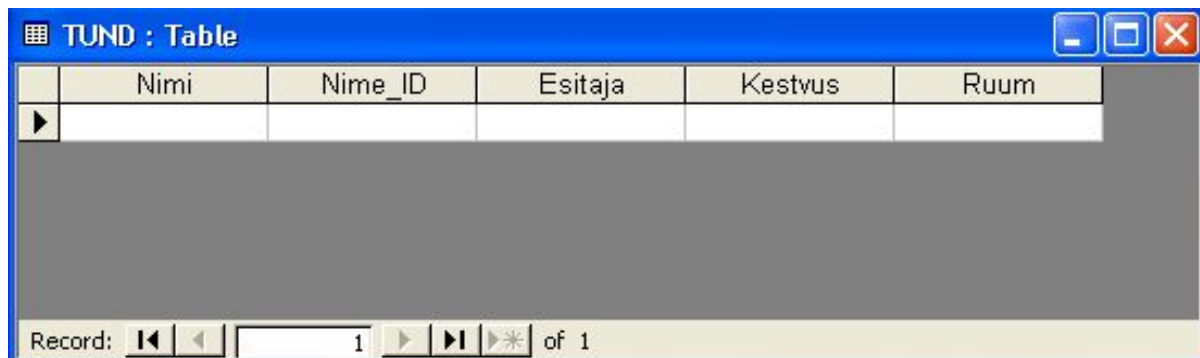
Input Mask:	Data Look:
Phone Number	(206) 555-1212
Social Security Number	531-86-7180
Zip Code	98052-6399
Extension	63215
Password	*****
Long Time	1:12:00 PM

Try It:

Edit List Cancel < Back Next > Finish

Antud loetelust valite *Short Time* ja valite *Next*. Järgmises osas võite jätta numbri kuju samaks ja *Finish*.

Nüüd võiks asuda tabelit täitma andmetega. Selleks valite aknamenüüribalt *View => Datasheet View*. Teie käest küsitakse uuesti, kas tahate salvestada. Peale *Yes* vastamist avaneb teile aken:



Antud aknasse tuleb kanda puuduvad andmed. Täitke tühjad lahtrid selliste andmetega:

<i>Nimi</i>	<i>Nime_ID</i>	<i>Esitaja</i>	<i>Kestvus</i>	<i>Ruum</i>
Eesti keel	EK	A. Kuusk	00:45	24
Inglise keel	IK	K. Rähk	00:45	75
Saksa keel	SK	T. Muru	01:45	73
Matemaatika	Mat	E. Puu	02:00	23
Arvutiõpetus	A	M. Klement	01:45	78
Füüsika	F	M. Jõgi	01:45	35
Keemia	K	K. Pärtel	00:45	41
Bioloogia	Bio	S. Visnapuu	00:45	62

Ja ongi esimene tabel koos andmetega valmis. Salvestage.

Teisel juhul:

Enne kui läheme tabelite sidumise kallale, vaatame teist võimalust teha tabel andmebaasitarga abil, seega valite *Create table by using wizard*. Sellel klikkides, avaneb selline aken:

Which of the sample tables listed below do you want to use to create your table?

After selecting a table category, choose the sample table and sample fields you want to include in your new table. Your table can include fields from more than one sample table. If you're not sure about a field, go ahead and include it. It's easy to delete a field later.

Business
 Personal

Sample Tables:
Mailing List
Contacts
Customers
Employees
Products
Orders

Sample Fields:
MailingListID
Prefix
FirstName
MiddleName
LastName
Suffix
Nickname
Title
OrganizationName
Address

Fields in my new table:

Rename Field...

Cancel < Back Next > Finish

Nüüd teeme teise tabeli, mis võiks kanda pealkirja Koht. Valime *Table Wizard*'i aknas *Personal* tüüpi tabeli. *Sample Fields* lahtrist paigutame märgi „>“ abil *Fields in my new table* lahtrisse need väljad, mis peaksid kajastuma meie uues tabelis. Need võiksid siis olla: *AddressID*, *Address* ja *Notes*. Kuna väljade nimed on kirjutatud hetkel inglise keeles, siis nupu *Rename Field...* on võimalik muuta meie tabeli jaoks valitud väljad eestikeelseteks.

Which of the sample tables listed below do you want to use to create your table?

After selecting a table category, choose the sample table and sample fields you want to include in your new table. Your table can include fields from more than one sample table. If you're not sure about a field, go ahead and include it. It's easy to delete a field later.

Business
 Personal

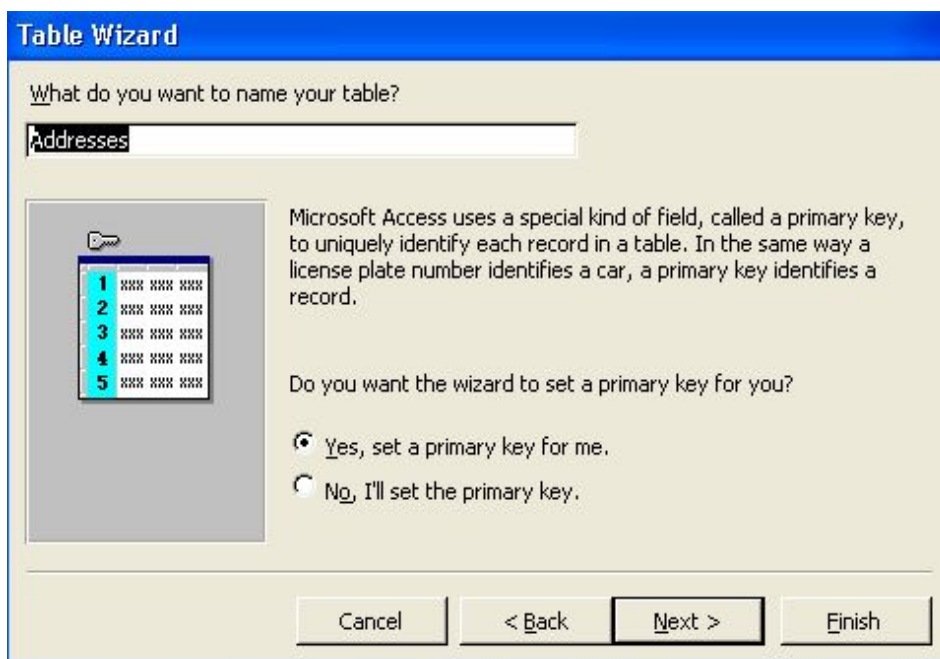
Sample Tables:
Addresses
Guests
Categories
Household Inventory
Recipes
Plants

Sample Fields:
FirstName
LastName
SpouseName
ChildrenNames
Address
City
StateOrProvince
PostalCode
Country/Region
EmailAddress

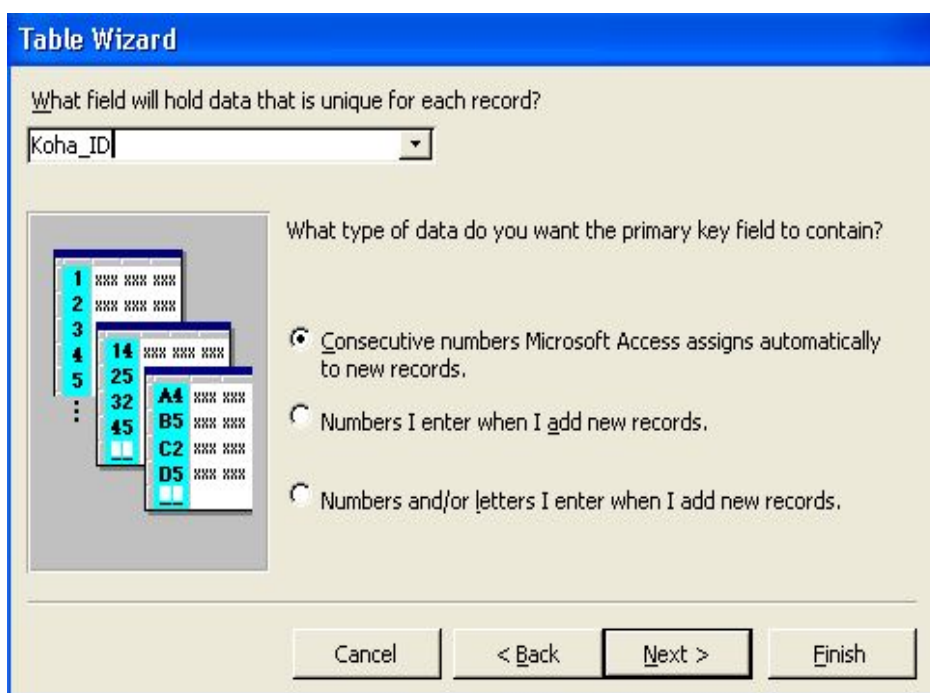
Fields in my new table:
Koha_ID
Kohtade arv
Aadress

Rename Field...

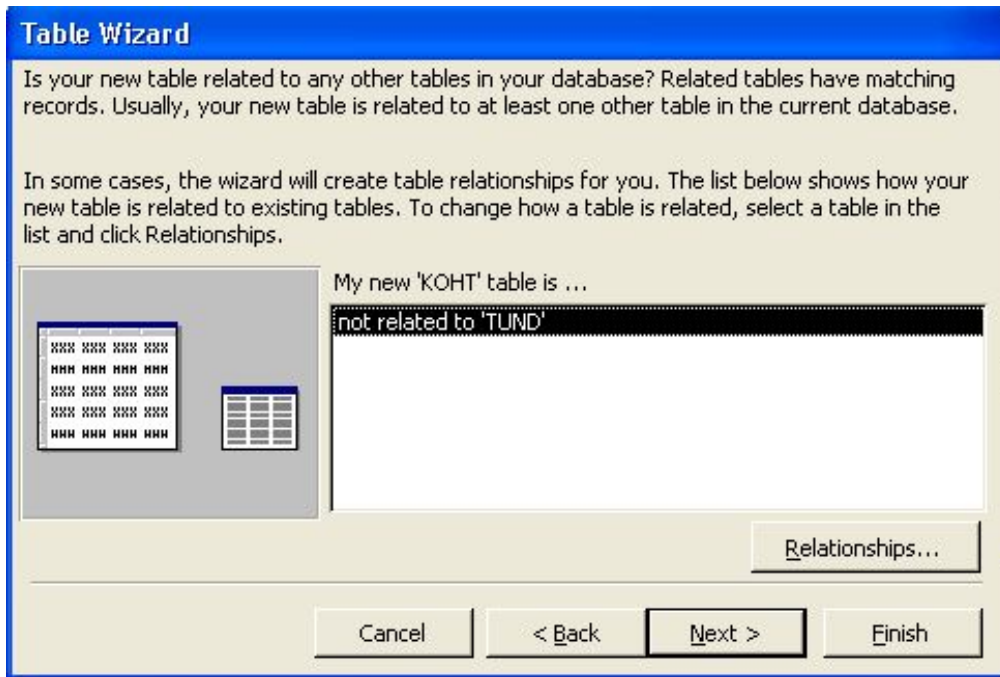
Cancel < Back Next > Finish



Edasi - järgmises osas tuleb panna pealkiri uuele tabelile. Olgu selleks siis **KOHT**. Veel soovitakse teada, kas soovite, et tabelitark määraks teie eest primaarvõtme või soovite seda ise teha. Valime jälle isetegemise – *No, I'll set the primary key*.



Edasi – soovitakse teada, millise välja andmed on unikaalsed. Olgu selleks Koha_ID. Siis soovitakse teada, mis tüüpi andmeid me tahame, et primarvõtme väli sisaldaks. Jääme kolmanda valiku juurde.



Edasi – järgnevalt antakse teada, kas tehtud tabel KOHT on seotud varem tehtud tabeliga TUND. Jätame praegu nad mitte seotuks. Seega valime *Finish*. Nüüd saame hakata täitma tabelit KOHT selliste andmetega. Peale täitmist salvestage.

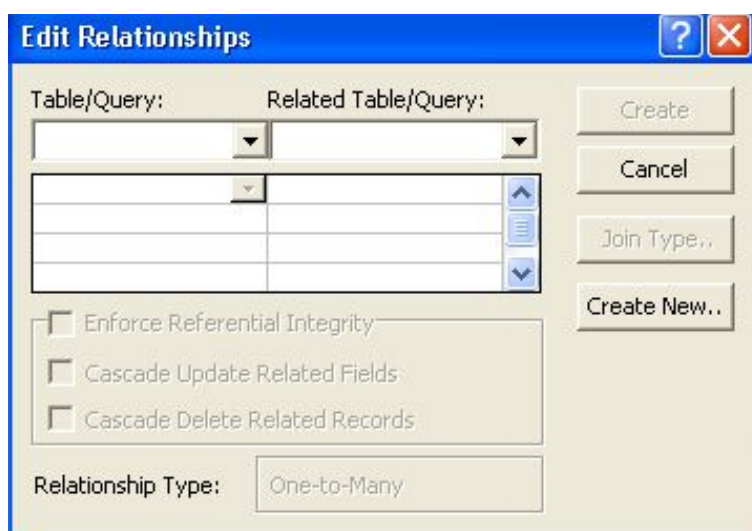
<i>Koha_ID</i>	<i>Kohtade arv</i>	<i>Aadress</i>
LK_2	100	Kruusa 6
LK_3	45	Kastani 34
TM_6	50	Marja 12
MM_8	80	Tina 4

Tabelite seostamine

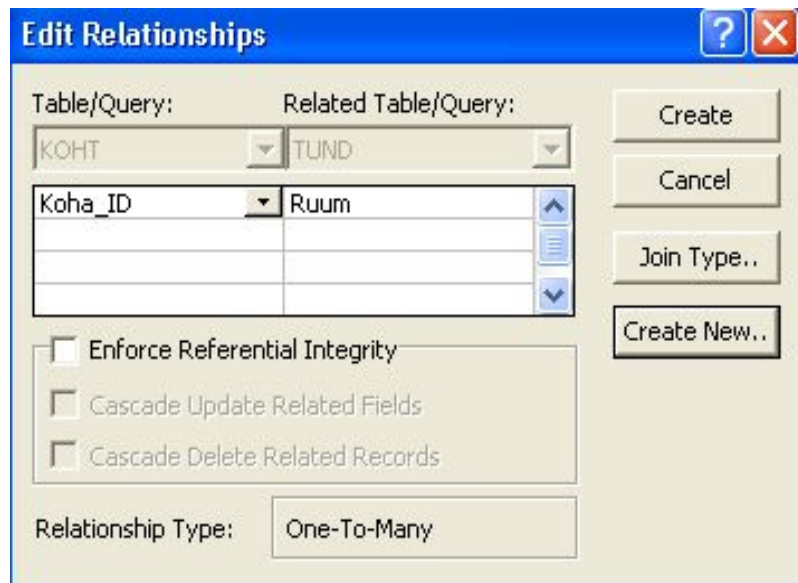
Relatsioonilise andmebaasi "selgrooks" on tabelite seostamine, sest käsitletav valdkond on tavaliselt kirjeldatud paljude tabelitega. Accessis saab seoseid interaktiivselt luua ja graafiliselt näha. Seose tüüpi saab määrata otse ja kaudselt. Proovime nüüd tehtud kahte tabelit omavahel siduda. Selleks valime MS Accessi aknas **Tools => Relationships**. Selle tulemusena kuvatakse kaks aknet, kus **Relationships** aknasse tulevad vormingud ja **Show Table** aknas võetakse loetelust need tabelid, mida tahetakse siduda. Hetke seisuga on seal kaks tabeli – KOHT ja TUND – lisame need (Add). Suleme Show Table akna. Seega jääb aken Relationships, mis on selline:



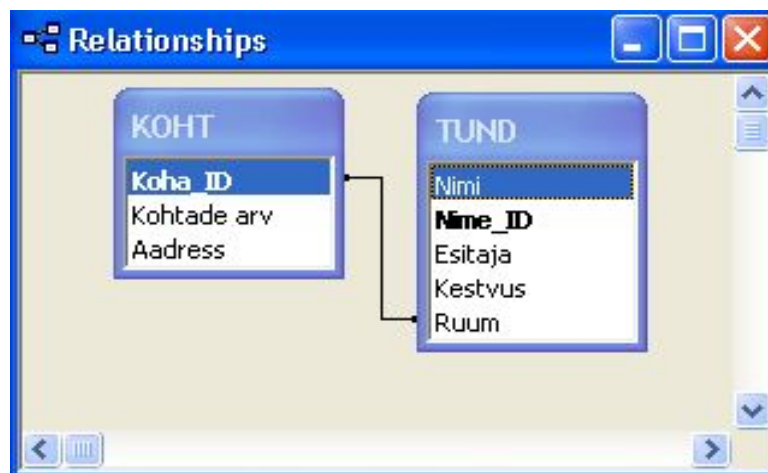
Kuna alguses KOHT tabeli loomisel me ei sidunud teda tabeli TUNNIGA, siis tuleb võtta MS Access'i menüüst **Relationships => Edit Relationships**, mille tulemusena avaneb taoline aken:



Klõikime *Create New...*. Uues aknas paigutame vasakpoolsesse lahtrisse KOHT ja parempoolsesse TUND. Alumised lahtrid täidame vastavalt selle, kuidas need kaks tabelit seotud on. Antud juhul võiksid nad seotud olla Koha_ID ja Ruumiga. Seega peale nende valikute määramist OK. Aken Edit Relationships muutub selliseks:



Peale Create nupu vajutamist saame Relationships tabeli kuju selliselt (kaks tabelit on seotud):



Salvestage.