

ECDL és számítógép-kezelői vizsga-előkészítő

# Adatbázis-kezelés

MS Office XP

*Dr. Péterny Kristóf*

Mercator  
Studio

Dr. Pétery Kristóf

# **Adatbázis-kezelés MS Office XP-vel**

**ECDL és számítógép kezelői vizsga előkészítő 5.**

Mercator Stúdió  
2003.

Minden jog fenntartva, beleértve bárminemű sokszorosítás, másolás és közlés jogát is.

Kiadja a Mercator Stúdió  
Felelős kiadó a Mercator Stúdió vezetője  
Lektor: Gál Veronika  
Szerkesztő: Pétery István  
Műszaki szerkesztés, tipográfia: Dr. Pétery Kristóf

ISBN 963 9496 36 7

© Dr. Pétery Kristóf PhD, 2003  
© Mercator Stúdió, 2003

Mercator Stúdió Elektronikus Könyvkiadó  
2000 Szentendre, Harkály u. 17.  
T/F: 06-26-301-549  
06-30-30-59-489

Dr. Pétery Kristóf: Adatbázis-kezelés MS Office XP-vel

# TARTALOM

<b>TARTALOM</b> .....	<b>4</b>
<b>ELŐSZÓ</b> .....	<b>9</b>
<b>A KÖNYV HASZNÁLATA</b> .....	<b>12</b>
Tanulási tanácsok .....	14
<b>BEVEZETÉS</b> .....	<b>15</b>
Az 5. ECDL modulról .....	15
A vizsgáról.....	15
Egyszerű adatbázis tervezése.....	15
Létező adatbázis behívása vagy belépés egy létező adatbázisba .....	16
Értékelés .....	16
A könyv fejezetei.....	16
Összefoglalás .....	17
<b>ELSŐ LÉPÉSEK</b> .....	<b>18</b>
Fogalmak magyarázata .....	18
Adatbázis.....	19
Adatmodell .....	20
A relációs adatmodell alapfogalmai .....	22
Adatbázisok tervezése .....	24
1. lépés: Követelményelemzés .....	25
2. lépés: Entitások, táblák meghatározása .....	26
3. lépés: Attribútumok, mezők megadása.....	27
4. lépés: Az azonosítók meghatározása .....	28
5. lépés: A kapcsolatok meghatározása .....	30
6. lépés: Teszt .....	34
7. lépés: Analizálás Access eszközzel .....	35
8. lépés: Adatbevitel és más objektumok .....	40
Az adatbázis-kezelő program megnyitása .....	40
Az Access alkalmazásablaka .....	46
Feladat .....	50

Létező adatbázis megnyitása alapértelmezés szerinti elrendezésben .....	56
Feladat .....	68
Az adatbázis ablak kezelése .....	69
Létező adatbázis rekordjának módosítása .....	71
Feladat .....	73
Adatbázis mentése merevlemezre vagy floppy lemezre.....	74
Feladat .....	75
Adatbázis bezárása .....	75
A Súgó használata .....	76
A program súgója .....	76
Súgó tartalom és tárgymutató .....	78
Alapvető beállítások elvégzése .....	83
Megjelenítendő elemek beállítása .....	83
Általános beállítások .....	85
Szerkesztés és keresés beállításai .....	86
A billentyűzet alapvető beállításai .....	86
Adatlap beállításai .....	87
Úrlapok és jelentések beállításai .....	87
Lapok beállításai .....	88
Speciális beállítások .....	88
Nemzetközi beállítások.....	88
Helyesírás beállítások .....	89
Automatikus javítás beállításai .....	89
Táblák és lekérdezések beállításai.....	91
Nézetmódok közti váltás .....	91
Eszköztár kinézetének módosítása.....	93
A menük beállítása .....	100
Összefoglalás .....	101
<b>ADATBÁZIS LÉTREHOZÁSA.....</b>	<b>102</b>
Alapvető műveletek.....	102
Adatbázis elvi és gyakorlati tervezése.....	102
A táblákról részletesebben .....	103
Tábla létrehozása .....	104
Adattábla létrehozása adott mezőkkel és attribútumokkal.....	105
Feladat .....	106

Feladat .....	108
Tábla készítése Tábla Varázslóval .....	110
Feladat .....	116
Mozgás az adattáblán belül .....	116
Adatok bevitele az adattáblába .....	118
Feladat .....	119
Kulcsok definiálása .....	119
Elsődleges kulcs definiálása .....	119
Index beállítása .....	120
Feladat .....	121
Kapcsolatok .....	123
Összefoglalás .....	129
<b>ADATTÁBLA MÓDOSÍTÁSA .....</b>	<b>130</b>
Táblaszerkezet módosítása .....	130
Adattábla formai jellemzői .....	130
Új mezők létrehozása és mezők törlése .....	131
Mező attribútumok (tulajdonságok) beállítása .....	131
Mezőtulajdonságok beállítása .....	132
Adatbázis karbantartása .....	156
Adattábla adatainak módosítása .....	156
Adattábla adatainak törlése .....	156
Rekordok bevitele .....	157
Összefoglalás .....	157
<b>ÚRLAPOK HASZNÁLATA .....</b>	<b>158</b>
Űrlap létrehozása .....	158
Űrlapok létrehozása .....	159
AutoŰrlap alkalmazása .....	159
Feladat .....	160
Feladat .....	161
Űrlap létrehozása varázslóval .....	162
Több táblát kezelő űrlap .....	164
Segédűrlap hozzáadása létező űrlaphoz .....	165
Feladat .....	165
Nézetek .....	167
Űrlaptervezés .....	168
Elemek elhelyezése .....	171

Vezérlőelemek .....	172
Kötött vezérlőelem létrehozása .....	173
Kötetlen vezérlőelem létrehozása .....	174
Számított vezérlőelem létrehozása .....	175
Vezérlőelem mezőhöz kapcsolása .....	176
Vezérlőelem formázása .....	179
Vezérlőelem áthelyezése .....	187
Vezérlőelem méretezése .....	188
Vezérlőelem igazítása .....	189
Adattal kapcsolatos tulajdonságok .....	190
Eseményekkel kapcsolatos tulajdonságok .....	192
A vezérlőelem egyéb tulajdonságai .....	192
Címkék és beviteli mezők .....	193
Jelölőnégyzet, választókapcsoló, váltógomb .....	194
Vezérlőelem csoportok .....	194
Feladat .....	195
Lista és kombinált lista .....	197
Feladat .....	198
Parancsgomb .....	200
Segédúrlap .....	202
Diagramok .....	203
Kép és más objektumok .....	207
Bejárési sorrend .....	210
Több karton kialakítása .....	210
Ablakkezelő ikonok .....	212
Léptetőgombok .....	212
Görgetősáv .....	212
Szakaszok .....	213
Szakasz-osztóvonal elrejtése úrlapon .....	215
Egy vezérlőelem vagy szakasz színei, hatásai .....	215
Beépített formátumok .....	215
Előugró úrlapok .....	217
Nem modális előugró úrlap .....	217
Egyéni párbeszédpanel létrehozása .....	217
Összefoglalás .....	218
<b>INFORMÁCIÓ-LEKÉRDEZÉS .....</b>	<b>219</b>

Alapvető műveletek.....	219
Létező adatbázis megnyitása vagy bejelentkezés egy adatbázisba .....	219
Adott feltételeknek megfelelő rekord megkeresése .....	221
Egy érték előfordulásainak megkeresése .....	222
Adatkeresés értéklístával.....	223
Adatkeresés szűrővel .....	225
Szűrő használata táblán, lekérdezésben, úrlapon .....	233
Rekordkeresés a rekordszámmal .....	235
Mezőre ugrás hosszú rekordban .....	236
Rekordok rendezése táblában .....	236
Rekordok rendezése úrlapon .....	237
Egyszerű lekérdezés létrehozása .....	238
Lekérdezéstípusok.....	238
Választó lekérdezések.....	239
Paraméteres lekérdezések.....	241
Keresztáblás lekérdezések.....	242
Módosító lekérdezések.....	244
Azonos rekordok megkeresése .....	249
Azonos rekordok automatikus törlése.....	251
Összefoglalás .....	252
<b>JELENTÉSEK .....</b>	<b>253</b>
A jelentések felépítése .....	253
Jelentés létrehozása .....	256
Feladat .....	258
Jelentés módosítása .....	259
Fejléc és lábléc létrehozása, egyéniesítése .....	260
Címkék készítése.....	260
Jelentés nyomtatása .....	265
Összefoglalás .....	269
<b>IRODALOM .....</b>	<b>270</b>



# ELŐSZÓ

Az ECDL (European Computer Driving Licence) – magyarul Európai Számítógép-használói Jogosítvány – olyan bizonyítvány, amely tanúsítja, hogy birtokosa sikeresen letett egy információ-technológiai alapismereteket mérő elméleti, és hat számítógép-használói jártasságot mérő gyakorlati vizsgát. Az okmány tehát nem elsősorban az informatikai, hanem a felhasználói ismereteket igazolja, melyet minden európai polgár megszerezhet.

Az eredetileg európai programként indult rendszert, illetve ahhoz hasonlót vezetnek be Kanadában, Ausztráliában, Dél-Afrikában és az Egyesült Államokban is. Itt a vizsgarendszer neve: ICDL – International Computer Driving Licence.

A számítógépes ismeretek napjainkban egyre alapvetőbbé válnak az élet minden területén. Az ECDL tanúsítvány igazolja birtokosának számítógép-használói ismereteit. Az ilyen igazolást jól használhatják azok, akiknek munkája megköveteli a számítógépes ismereteket – függetlenül tudományágtól – diákok, munkavállalók és munkáltatók egyaránt. A vizsgákra való felkészítés pedig hasznos lehet azoknak is, akik kedvtelésből akarják megtanulni a számítógépek használatát.

Az ECDL vizsgákat az ismeretek igazolására eddig több mint 26 országban vezették be. Előreláthatólag 2004-re több ötmillió feletti létszám rendelkezik majd az ECDL-bizonyítvánnyal.

Az ECDL szándéka:

- ✚ az általános számítástechnikai tudásszint emelése a jelenlegi és a leendő munkavállalók körében,
- ✚ a számítógéppel dolgozók munkájának eredményesebbé tétele,
- ✚ az információ-technológiai befektetések hatékonyságának növelése,
- ✚ a felhasználók megismertetése a legújabb, és a legmagasabb színvonalú módszerekkel.

Az ECDL-bizonyítvány megszerzéséhez szükséges egy elméleti és hat gyakorlati vizsga: it-alapismeretek (elmélet), operációs rendszerek, szövegszerkesztés, táblázatkezelés, adatbázis-kezelés, pre-

zentáció, azonkívül információ és kommunikáció. Sőt az ECDL Start keretében az ECDL-végbizonyítvány megszerzéséhez a hét modul helyett elegendő négy modulvizsgát letenni a nemzetközi bizonyítványhoz. Az ECDL Start kötelező moduljai az operációs rendszerek, a szövegszerkesztés, az információ és kommunikáció, míg a negyedik modul szabadon választható az it-alapismeretek, a táblázatkezelés, az adatbázis-kezelés és a prezentáció közül.

A vizsga modulonként tehető le az arra feljogosított vizsgaközpontokban. A vizsgarendszer felelőse és jogtulajdonosa Magyarországon a Neumann János Számítógép-tudományi Társaság.

A hét vizsga kissé részletesebben:

✚ **Információ-technológia alapismeretek:** Elméleti vizsga az alapvető számítógépes fogalmakban való jártassággal kapcsolatosan és a számítógépek alkalmazási területeinek ismeretéről.

Gyakorlati vizsgák:

✚ **Szövegszerkesztés:** Számítógép segítségével készülő szövegek létrehozása, szerkesztése, formázása, tárolása és nyomtatása. E terület fontosságát kiemeli, hogy napjainkban az írásos dokumentumok nagy részét szövegszerkesztő programokkal állítják elő.

✚ **Táblázatkezelés:** Táblázatok, jegyzékek és listák számítógépes előállítás, kezelése. A táblázatkezelő programok jelentősége azért is nagy, mert ezeken a funkciókon kívül alkalmasak még a számítások, keresések, szűrések elvégzésére, valamint az adatok szemléletes, grafikus bemutatására is. Ennek köszönhetően az ilyen programokat költségvetések, előrejelzések, üzleti és műszaki számítások, pénzügyi jelentések elkészítésére, kisebb adatállományok kezelésére alkalmazzák.

✚ **Adatbázis-kezelés:** A nagy mennyiségű adatok nyilvántartására, gyors és rugalmas kezelésére, elérésére használható. Ilyen igényekkel nap, mint nap találkozhatunk a valós életben.

✚ **Prezentáció és grafika:** A prezentáció elképzeléseink, terveink és más témák látványos előadásában segít. A számítógép támogatásával végzett bemutatók az eredményes kommunikáció eszközévé váltak az üzleti életben és az oktatásban egyaránt. A grafika illusztrációs eszközei, nemcsak az építészeknek, mérnököknek, illusztrátoroknak és grafikusoknak fontosak, hanem

használatuk beépül a szövegszerkesztő és bemutató készítő programokba.

✚ **Információs hálózati szolgáltatások:** A számítógépes hálózatok teszik elérhetővé sokak számára az erőforrásokat és a kommunikációt. Ezek biztosítására napjainkra már az egész világot behálózó kapcsolatrendszert fejlesztettek ki több millió számítógép összekapcsolásával. A modul az információs szupersztráda eredményes használatához szükséges ismereteket nyújtja a felhasználók számára.

✚ **Operációs rendszerek:** A többi modul elvárt szintű alkalmazásához, valamint a számítógép minél több hasznos szolgáltatásának kiaknázásához elengedhetetlenül fontos ismerni a számítógépes rendszert működtető lényeges funkciókat.

Az első vizsga előtt a jelentkező egy vizsgakártyát kap, amelyre minden sikeres vizsgát rávezetnek. Az összes vizsga letétele után a vizsgaközpont a kártyáját elküldi az NJSZT ECDL irodájába, ahol ennek alapján kiállítják az ECDL bizonyítványt. A hét vizsgát a regisztráció megkezdésétől számított három éven belül kell letenni bármelyik hivatalos európai ECDL vizsgaközpontban.

A vizsgakérdéseket bármely szolgáltatótól származó szoftver alapján összeállíthatják. Néhány vizsgának különböző változatai lehetnek a vizsgaközpont felszereltségétől, a rendelkezésére álló eszközöktől függően.

A Mercator Stúdió sorozatával az eredményes felkészülést kívánja szolgálni. Minthogy a vizsgaközpont felszereltsége eltérő lehet, illetve az alkalmazott eszközök az informatikában megszokott módon, viszonylag rövid átfutási idővel cserélődnek, sorozatunk köteteit mindig a leggyakrabban használt rendszereknek megfelelően dolgozzuk át. Eközben azonban mindig ragaszkodunk a Neumann János Számítógép-tudományi Társaság által kiadott syllabus-hoz.

A sorozat e-book formájában jelenik meg, közvetlen előzményének és ajánlott szakirodalomnak tekinthető a kiadó e témában mára már száz fölé emelkedett köteteinek sora. Szintén kiegészítésként ajánljuk hagyományos „papíros” könyveinket is más kiadók igen fontos példatárait (kiemelve a Kossuth Kiadó ECDL-vizsgapéldatárát). A tárgyalt ismeretek néhány OKJ- (Országos Képzési Jegyzékben szereplő) szakma számítástechnikai feltételeinek is megfelelnek.

# A KÖNYV HASZNÁLATA

A könyv kiadásával az egyéni, számítógép felhasználásával végzett tanulást kívánjuk támogatni. Ennek előnye a teljes időbeli megkööttöttségtől mentesség, szabad időbeosztás mellett az is, hogy az elektronikus könyvet a képernyő egy részén magunk előtt tartva, a képernyő egy másik részén a tárgyalat alkalmazás futtatásával követhetjük a leírtakat.

Ez a könyv az ingyenes Acrobat Reader 5.0 vagy az Acrobat e-Book Reader segítségével olvasható. Akinek nincs ilyen programja, az letöltheti többek közt a [www.adobe.com](http://www.adobe.com) webhelyről is. Az ilyen típusú könyvek igen előnyös tulajdonsága, hogy a képernyőn megjeleníthető a tartalomjegyzék, amelynek + ikonjaival jelölt csomópontjaiban alfejezeteket tartalmazó ágakat nyithatunk ki. A tartalomjegyzék bejegyzései ugyanakkor ugróhivatkozásoként szolgálnak. Ha egy fejezetre akarunk lépni, akkor elegendő a bal oldali ablakrészben megjelenített könyvjelző-lista megfelelő részére kattintani. Sőt az ilyen könyvek teljes szövegében kereshetünk.

A sorozat könyveinek tartalma az NJSZT syllabusához igazodik. A kiadványok összeállításakor a közérthetőség mellett a legfontosabb szempont az volt, hogy sikeresen támogassuk az ECDL vizsgák letételére készülő Olvasót. Minthogy mindegyik kötet sok ismeretet tárgyal, a könnyebb kezelhetőség érdekében néhány olyan tipográfiai megoldást alkalmaztunk, amelyek felhívják a figyelmet a könyv speciális funkciójú részeire.

Ezek egy részét újabban a „papíros” könyvek margóin helyezik el, ami ez elektronikus könyv használatát nemcsak megkönnyítené, hanem néha bizony megnehezítené is. Ezért helyettük a könyvben való tájékozódást segítő csak a bekezdés elején megjelenő szimbólumokat, illetve háttérszínezést alkalmaztunk. A jelek segítségével könnyebben megtalálhatók az új ismereteket leíró részek, a célok és a feladatok. Mindegyik ECDL kötetünkben, mindegyik modul tárgyalásakor azonos jelöléseket használtunk. A parancsok és a párbeszédpanelek nevét **fékövéren**, a párbeszédpanelek listáiban szereplő elemeket, illetve könyvtárakat, mappákat *dólt*en szedtük. A billentyűket és kombinációikat bekeretezve jelöljük.

A könyv részeit jelző rajzok és jelentésük:



**Célkitűzés.** A fejezetek elején bemutatjuk a fejezet végigolvasásával, ismereteinek elsajátításával, gyakorlatainak, feladatainak megoldásával elérendő célt. A fejezet csak a célhoz vezető úthoz szorosan kapcsolódó és a korábbi fejezetekben tárgyalt ismeretekre alapozó ismeretanyagot tartalmaz. Amennyiben ez a célkitűzés elolvasása alapján ismertnek tűnik, ugorjunk a következő fejezethez.



**Időtartam.** A célkitűzést követően minden tanulási egység megkezdésekor bemutatjuk, átlagosan mennyi idő szükséges az adott tananyag elsajátításához. A becsült időtartam az összes feldolgozási időre vonatkozik, amelyet érdemesebb több részre bontva teljesíteni. A rész időtartamokat mindenki tetszése, képességei és előismeretei szerint maga válassza meg.



**Új ismeretek.** Ezzel a jellel hívjuk fel a figyelmet egy korábban nem tárgyalt ismeretre. Persze a teljesen kezdők számára minden újdonság lehet, mégis ezt a jelet csak a legfontosabb esetekben alkalmaztuk. Ahol ez a jel szerepel, rendszerint leíró magyarázat mutatja be az új fogalmakat, ismereteket.



**Közösen oldjuk meg.** A kötetben számos feladat szerepel. A mellékelt szimbólummal jelöljük azokat, amelyekhez részletes megoldási kulcsot is adunk. Ezekben lépésről-lépésre leírjuk a feladat megoldásához vezető utat, alkalmazandó fogásokat. Természetesen hasznosabb, ha a feladat kiírása után előbb magunk próbáljuk a megoldást megkeresni, és csak ha megakadtunk, akkor nézzük meg a bemutatott megoldást.



**Önállóan oldjuk meg.** A mellékelt szimbólummal jelöljük azokat a feladatokat, amelyeket a fejezet korábbi részeinek elolvasása után az Olvasó önmaga is meg tud oldani. Ha mégis nehézségei támadnak, akkor lapozzon a fejezetben előrébb, mert a megoldás ott megtalálható. Az ilyen feladatok alkalmasak a megszerzett tudás ellenőrzésére is.



**Több megoldás is van.** Rendszerint több megoldás is alkalmazható egy-egy feladat esetében. Ezzel a szimbólummal jelezzük, hogy a feladat több megoldását is bemutatjuk. A több megoldás közül általában első helyen mutatjuk be azt, amelyet gyorsasága vagy egyszerűsége miatt gyakrabban

alkalmaznak. Ilyen esetekben legalább az egyik megoldást el kell sajátítani.



Elértük a célt. A fejezet végén összefoglaljuk a megszerzett ismereteket. A szimbólum jelzi, hogy megoldottuk a fejezet elején kitűzött feladatokat. Az összefoglaló segít elhelyezni az új tapasztalatokat és fogalmakat az ismeretek rendszerében.



Megjegyzés. A szimbólum jelzi az adott témához kapcsolódó fontosabb és szélesebb körű információt. Ezeket rendszerint a „papíros” könyvek a margón helyezik el.



Trükk, ötlet. A témához kapcsolódó speciális megoldást mutat be. A tárgyalt eljárás rendszerint az ECDL vizsganyagon túlmutató hasznos fogás, ami az Olvasó részére a teljesség, a jobb megértés, vagy egyenesen a számítógépes szoftverüzemeltető képezés megszerzése céljából ajánlatos.

## Tanulási tanácsok

Bár az ECDL vagy egy szoftverüzemeltető vizsgára készülők rendszerint már többféle ismerettel rendelkeznek, köztük tanulási tapasztalatokkal is, de az elektronikus könyv használata minden bizonnyal számukra is tartalmaz újdonságokat.

A tanulást ezért mindenképpen a könyv kezelésének elsajátításával, szerkezetének megismerésével kezdjük. Olyan ablakméretet és nagyítást állítsunk be *e-book* vagy *Acrobat Reader* programunkban, amely biztosítja a megerősítés nélküli, kényelmes olvasást, szükség esetén az olvasó ablaka mellett a feladatok kipróbálását is. A gyors tájékozódás érdekében használjuk a tartalomjegyzéket, könyvjelzőket, illetve az olvasók beépített keresőszolgáltatásait.

A fejezetek logikus, a programokat kezelő felhasználók ismereteinek, a funkciók használatának sorrendjében követik egymást.

Az ismeretek elsajátításához tűzzünk magunk elé ésszerű, be tartható határidőket. Az egyes fejezetek és feladatok között ne tartunk túlságosan nagy szüneteket, mert gyakorlás nélkül hamar felejtünk. Minden feladatot oldjunk meg, szükség esetén és az ismeretek rögzítése érdekében többször ismételjünk.

# BEVEZETÉS

## Az 5. ECDL modulról

Az *Adatbázis kezelés* modulban a jelöltnek bizonyítania kell, hogy érti az adatbázisok lényegét és képes a személyi számítógépen lévő adatbázis használatára. A modul két részből áll; az első rész a jelöltnek azt a képességét méri, hogy képes-e egy az adatbázis elméleti és gyakorlati megtervezésére egy általánosan használt adatbázis kezelő segítségével; a második rész azt vizsgálja, tudja-e alkalmazni az adatbázis kezelő alkalmazás rendezési és kiválasztási eszközeit és a lekérdezéseket egy meglévő adatbázis adatainak kinyeréséhez.







## A vizsgáról

A modul azt méri fel, hogy a vizsgázó létre tud-e hozni egy kisebb adatbázist standard adatbázis programcsomag segítségével, valamint egyszerű lekérdezéseket és jelentéseket készíteni létező adatbázisból. A vizsga két feladatból áll, mindkettőt meg kell oldani.

Az első feladatban a vizsgázónak egy meghatározott célra saját adatbázist kell felépítenie, meg kell határoznia a rekordok szerkezetét, és adatokat kell bevinnie.

A második feladatban a vizsgázónak le kell töltenie egy adatbázist és lekérdezések segítségével válaszolnia kell az adatokról felteht kérdésekre.

## Egyszerű adatbázis tervezése

-  Rekordok szerkezetének kialakítása.
-  Adatok bevitele az adatbázisba.
-  Adatok megszerkesztése.
-  Új adatsorok beillesztése.
-  Rekordok törlése.
-  Adatbázis-kulcsok meghatározása.

## Létező adatbázis behívása vagy belépés egy létező adatbázisba

- ✚ Adatok bevitele az adatbázisba.
- ✚ Adatok szerkesztése.
- ✚ Új rekordok beillesztése.

A következő részfeladatok fordulhatnak elő a feladatokon belül:

- ✚ Adatok megkeresése, kiválasztása és osztályozása megadott kritériumok szerint.
- ✚ A kiválasztott adatok meghatározott sorrendben történő megjelenítése a képernyőn és jelentésekben.
- ✚ Az adatbázis szerkezetének módosítása.
- ✚ Súgó funkciók használata

## Értékelés

A rendelkezésre álló idő 45 perc. Minden feladatot meg kell oldani, az elégséges szint eléréséhez 80%-os teljesítmény kell.

## A könyv fejezetei

A könyv fejezetei a tárgyalt témakör nehézsége és a korábbi feladatokban szerzett jártasság elvárása alapján egymásra épülnek. Segítségükkel a teljesen gyakorlatlan felhasználó fokozatosan sajátíthatja el az adatbázis-kezeléshez szükséges ismereteket. A fejezetek tehát építenek a korábbi fejezetekben megszerezhető tapasztalatokra. A didaktikai vezérlő elv: fokozatosan, az egyszerűtől a bonyolultig.

Az összeállított feladatok részben a valós életben előforduló eseteket példázzák, részben azt a szándékot tükrözik, amellyel saját és mások oktatási tapasztalatai alapján az ismeretfeldolgozási nehézségek felbukkanásákor szeretnénk támogatást nyújtani. A témák mégoly közérthető tárgyalása is csak akkor követhető, és a megszerzett ismeretek akkor rögzülnek, ha működés közben sajátíthatják el a program alkalmazását, valamint meggyőződhetnek a feladatmegoldás hasznosságáról.



Ennek érdekében a fejezeteket célkitűzéssel kezdjük, amelyben tömören bemutatjuk az adott fejezetben elérhető ismeretszintet, annak rendeltetését és a későbbi hasznosítását, vagyis a megismert funkciók beépülését a többi feladat és általában a táblázatkezelés sorába. A célkitűzést követően megadjuk a fejezet elsajátításához szükséges az átlagos feldolgozási időt is. A ténylegesen szükséges idő természetesen az egyéni képességek, adottságok, illetve gyakorlat, korábbi ismeretek függvénye.

## **Összefoglalás**

A könyv elolvasása, illetve a feladatok elvégzését követően az ECDL vizsgafeladatok minden bizonnyal sikeresen megoldhatók. Ugyanakkor azt is megemlíjtük, hogy ez a szint bár a mindennapokban rendszerint elegendőnek bizonyul, de a programhasználatnak csak egy jó közepes szintjét jelenti. Ennél magasabb szintre csak rengeteg gyakorlással, a sűgő és a szakirodalom böngészésével juthatunk. Így tehetünk szert olyan ismeretekre is, amelyekről részint az ECDL szintet meghaladó volta, részint a könyv kötött terjedelme miatt nem szólhattunk. Említést érdemel a program testre szabása, a kimutatások és kimutatásdiagramok készítése, az egyéni párbeszédpanelek, a csoportos munka, a Visual Basic programozás stb.

Megnyugtató lehet viszont, hogy a tárgyalt ismeretek szinte csak kis változtatással alkalmazhatók más adatbázis-kezelőkben is, nemcsak az itt bemutatott Office XP tag, azaz az Access 2002 esetében (például a StarOffice-ban, a Works-ben stb.). Minél magasabb szintű ismeretekre teszünk szert ugyanis, annál inkább várható, hogy a különféle konkurens programtermékeknél eltérő megoldást alkalmaznak, ha egyáltalán kidolgozták az adott probléma megoldását (például a kimutatások készítése az Access sajátossága).

# ELSŐ LÉPÉSEK



E bevezető célja a kezdő programhasználók megismertetése az Office XP részeként kifejlesztett, de önállóan is forgalmazott és telepíthető Access 2002 program indításával, alapvető beállításával és a programkörnyezet főbb elemeivel. A fejezetben ismertetett információk megalapozzák a program későbbi biztos használatát, ezért ismeretük elengedhetetlenül fontos.

Az ECDL 5. moduljával foglalkozó a tananyagban a Microsoft Access 2002 programmal ismerkedünk meg, amely a Microsoft Office XP Professional változatának tagjaként napjaink egyik legelterjedtebb adatbázis-kezelő programjává vált. Mivel a továbbiakban tárgyalt programfunkciók célja, szerepe a korábban bemutatott programok eljárásaival szemben mindenki számára nem feltétlenül azonnal érthető, az egyes részek előtt külön kiemeljük a tárgyalt eljárás hasznosságát, alkalmazási körét.



A fejezet feldolgozásának becsült átlagos ideje négy óra. A terjedelem miatt érdemes félórás részekre bontani.

## Fogalmak magyarázata



Mindennapi életünk során számtalan információ ér bennünket, amelyeket megjegyzünk, tárolunk, összekapcsolunk más információkkal, adatforrásokkal. A nagyobb tömegű adat kezelése, tárolása, megfelelő sebességű visszakeresése már számítógépes megoldást igényel. Kisebb adatmennyiségnél még elfogadható a hagyományos „kartonos” megoldás, azonban ha biztonságos, gyors, könnyen kezelhető eszközt szeretnénk, akkor már kis adatmennyiségnél is szóba jöhet az informatikai módszer. Az gazdasági és társadalmi élet minden szegletében találkozunk adattárolással.

Az *adatbázis* tágabb értelemben egy olyan halmaz, amelynek elemei – az *adatok* és azok *állományai* – egy meghatározott tulajdonságuk alapján összetartozónak tekinthetők.

Az *adatbázis-kezelők* feladata ezen adatok rendezése, a köztük lévő kapcsolat nyilvántartása, az adatokhoz hozzáférés szabályozása, az adatok védelme, az integritás megőrzése, az adatok módosíthatóságának, lekérdezésének biztosítása, különféle szempontok szerinti kigyűjtése, válogatása és egyéb statisztikai funkciók. Az Access programot e feladatok teljesítésére alkották meg.

Az adatbázis-kezelőkkel megoldandó feladatok általános jellemzői tehát a nagy adatmennyiség kezelése, az adatok közötti kapcsolatot, a szerkezet rögzítése, illetve a tárolt adatokkal végzendő, fentiekben részletezett műveletek.

Az adatok egy egyedre vonatkoznak, az adatbázisban minden egyes egyedet hasonló tulajdonságokkal jellemezhetünk. Egy másik adatbázisban ugyanazt az egyedet más tulajdonságok írhatnak le. Például egy adatbázisban előfordulhatnak személyek, mint egyedek. Egy kórház adatbázisában más tulajdonságok jellemzik a beteget, a kórház dolgozóját (holott személy szerint ugyanaz az ember lehet mind a két adatállományban) stb.



Igen fontos megjegyeznünk, hogy egy gondosan megtervezett és karbantartott „napra kész” adatbázis hatalmas értéket képvisel, már csak az adatrögzítésbe fektetett energia és sok más tényező következtében. Ugyanakkor az adatbázis sérülése, elvesztése végzetes is lehet, sokszor nem pótolható. Ezért mindig gondoskodjunk biztonsági mentésekről.

Az adatbázis-kezelővel végzett műveleteknek köszönhetően az adatkezelési tevékenység kevesebb munkával (adatrögzítéssel, programozással) jár, így kevesebb a hibalehetőség is. A kevesebb programozás jóval olcsóbb üzemeltetést is jelent. Minden program, amely az adatbázishoz hozzáfér, ugyanazokkal a naprakész adatokkal dolgozik. Az egyszeres tárolás kevesebb tárolókapacitás igényvel jár.

## Adatbázis

Az adatbázis egy adatmodellel leírható, adott céllal gyűjtött adatok összessége. Az adatbázis szerkezetét írja le az adatmodell.

Az adott céllal, egy meghatározott témakörrel kapcsolatos információk lehetnek például a vevői megrendelések, számlázási, készlet-, termék-, ügyfél-nyilvántartási adatok stb. Az adatbázist számítá-

tógép nélkül, illetve részlegesen „gépesítve” is nyilvántarthatjuk, ekkor azonban a tárolóhelyek, vagy fájlok adminisztrációja sokkal körülményesebb és számos szubjektív hibaforrást rejt magában.

Az azonos típusú információkat korábban külön-külön fájlokban tároltuk, amelyek magukba foglalták a fájlok közti kapcsolatokat is. Minden fájlban egy-egy egyed típusra vonatkozó információt tárolunk. Egyed típus lehet például a *termék* vagy a *beteg*.

A fájlok legkisebb, önállóan kezelhető, feldolgozható részei a rekordok, amelyek az egyed jellemzők konkrét előfordulásainak adatait tartalmazzák. Az egyes egyed típusok tulajdonságait összefoglaló rekordokban a tulajdonsághoz tartozó adatok a mezőkbe kerülnek. A *termék* egyed típushoz tartozó, külön mezőben tárolt tulajdonságok lehetnek például a termék neve, egységára, tömege, és egyéb jellemzői. A *beteg* egyed típushoz tartozó, külön mezőben tárolt tulajdonságok lehetnek például a beteg azonosító adatai (név, személyi szám, tb. törzsszám, cím), születési éve, neme, súlya stb.

Az adatbázis-rendszerrel szemben támasztott követelmények:

- + biztosítsa nagy mennyiségű adat hatékony kezelését,
- + egyszerre több felhasználó is használhassa,
- + őrizze meg az adatok integritását, feleljen meg a megadott szabályoknak,
- + nyújtson adatvesztés elleni védelmet,
- + tegye lehetővé az egyes felhasználók hozzáférési jogainak szabályozását,
- + továbbfejleszthető legyen.

## Adatmodell

Az adatbázis szerkezetét az adatmodell egyértelműen határozza meg. Az adatmodell írja le az adatok típusát, kapcsolatát, a korlátozó feltételeket és az adatkezelési műveleteket.

A mai gyakorlatban elterjedten használják a következő négyféle logikai adatmodellt:

- + A *hierarchikus adatmodell* az adatokat fa struktúrában ábrázolja. Az adatokat rekordokba rendezi, a rekordok között „szülő-gyerek” kapcsolat van. Így egy szülőhöz több gyerek, de egy gyerekhez egyetlen szülő tartozhat. Ehhez hasonló a

DOS könyvtár-, illetve a Windows mappaszerkezete. Nagyobb, több ezer rekordot tartalmazó adatbázisoknál a keresés és az összetettebb műveletek hosszabb ideig tarthatnak.

- ✚ A *hálós adatmodell* a hierarchikus adatmodell továbbfejlesztéseként jött létre. Az adatokat szintén rekordokban tároljuk, az adatmodellt gráffal szemléltetjük. A rekordok közti kapcsolatok vertikális és horizontális irányúak is lehetnek, így adatmodell rendszerek leírására is alkalmasak. Itt egy gyerekeknek több szülője is lehet. Ilyen adatszerkezettel dolgozik több nagygépes adatbázis-kezelő.
- ✚ Az *objektum-orientált adatmodell* objektumokként képezi le az egyedeket. Az azonos tulajdonságokkal rendelkező objektumokat csoportokba soroljuk, mely csoportok tulajdonságokkal, attribútumokkal rendelkeznek. Az attribútumokat adatként kezeljük. Ez az adatmodell a bonyolult kapcsolatok ábrázolására alkalmas.
- ✚ A *relációs adatmodell* az adatokat táblázatokban rendszerezi, a táblázatok között logikai kapcsolat van. A táblázat oszlopai és sorai teljesítik a következő feltételeket:
  - minden oszlopnak egyértelmű neve van (a név egyedi, nem fordulhat elő a táblában még egyszer),
  - minden sorban ugyanazok az oszlopok vannak (azaz a sorok – rekordok – egyforma hosszúak),
  - az oszlopokban található adatok meghatározott értéket vehetnek fel,
  - az oszlopok soronként csak egy értéket vehetnek fel,
  - a táblázatot a neve egyértelműen azonosítja (ez a név is gyedi, tehát az adatbázisban nem lehet két azonos nevű tábla).

Relációs adatmodellt használnak a személyi számítógépeken igen elterjedt adatbázis-kezelő programok, mint a dBase, Clipper, Oracle, Ingres, Progress, illetve a kötetünkben tárgyalt Microsoft Access is.

## A relációs adatmodell alapfogalmai

Az **adatbázis** az adatok és a köztük lévő összefüggések rendszere, amelyet egymás mellett tárolunk. Az adatbázis hatékony kezelését csak jól átgondolt szerkezet biztosítja.

A relációs adatmodell fő alkotóelemei az egyes egyedeket tulajdonságaikkal leíró táblák, valamint a táblák közötti kapcsolatok.

Az **egyed** az, amit le akarunk írni, amelynek az adatait tároljuk és gyűjtjük az adatbázisban.

Az **attribútum** vagyis tulajdonság az egyed valamely jellemzője. Ez egyed az attribútumok összességével jellemezhető.

A **tábla** foglalja össze a logikailag összetartozó adatokat. A tábla oszlopokból (mezőkből) és sorokból (rekordokból) áll. Annyi sor található egy táblában, ahány konkrét egyed-előfordulást tartunk nyilván (például árucikket vagy beteget)



A **mező** az adatbázis egy oszlopa, amely az adatok, vagyis az egyedek tulajdonságértékeinek tárolására szolgál. Az adat a jelentésétől megfosztott információ, vagyis csak például a tömeget jelzőszám (az adat értelmezésében a mező neve segít).

Az **elemi adat** a táblázat celláiban szereplő érték, vagyis az egyed egy konkrét tulajdonsága.

A **rekord** az adatbázis egy sora, amely az egymással összefüggő adatokat tárolja. A rekordok tehát egymástól több mezőjükben különbözhetnek, több mezőjük tartalma lehet azonos, minthogy különböző egyedek jellemzőit tárolják. Egy fontos dologban mindegyik rekord különbözik a másiktól, mindegyik rekord egyedi azonosítóval rendelkezik.

A **reláció** az adatbázisban található táblák közötti logikai kapcsolat. Az egymással tartalmilag összefüggő táblák közötti kapcsolatot a mindkét táblában előforduló tulajdonságok biztosítják.

A táblák közti **kapcsolatok** az egyedek egymáshoz való viszonyát írják le. Az egyedek közti kapcsolatot háromféleképpen írhatjuk le.

-  **egy az egyhez (1:1) kapcsolat:** az egyik tábla egy eleméhez a másik tábla pontosan egy eleme kapcsolódik
-  **egy a többhöz (1:N) kapcsolat:** az egyik tábla egy eleméhez a másik tábla több eleme is tartozhat

✚ *több a többhöz (N:M) kapcsolat*: bármely tábla elemeihez a másik tábla tetszőleges számú eleme tartozhat

Az **Elsődleges kulcs** a táblázat rekordjainak egyértelmű azonosítója, értéke egyedi, kulcsattribútumnak is nevezik. Előfordulhat, hogy több tulajdonság együttes alkalmazásával összetett kulcsot kell képeznünk.

**Idegen kulcs**: olyan azonosító amelynek segítségével egy másik táblázat elsődleges kulcsára hivatkozhatunk.

Az **anomáliák** egy nem megfelelő modellből eredő problémák, el-  
lentmondások. Egy relációs adatbázisban a következő anomáliák léphetnek fel:

- ✚ **Bővítési anomália**: ha egy rekord felvételekor a már korábban tárolásra került információkat is újra be kell vinni.
- ✚ **Törlési anomália**: amikor az elem megszüntetésekor a nem hozzá tartozó információk is elvesznek.
- ✚ **Módosítási anomális**: amikor az elemi adat módosulásakor az adatbázisban az elemi adat összes előfordulási helyén el kell végezni a módosítást.

Az Access az összes információt egy adatbázis-állományba vonja össze. A fájlban belüli táblák tartalmazzák az adatokat. Minden típusú információ számára egy külön táblát hozunk létre. A táblákban tárolt adatok megtekintésére, bővítésére és frissítésére az űrlapok szolgálnak. Az űrlap szerkezetét, elrendezését az Űrlap Varázslóval vagy akár saját kezűleg is kialakíthatjuk. Az űrlapok felhasználhatók a kívánt adatok megkeresésére is. Űrlapok alkalmazásával ezek a műveletek jelentősen leegyszerűsödnek. Egy űrlap megnyitásakor az Access program a táblák adatait olvassa ki, majd azokat az általunk meghatározott formában jeleníti meg a képernyőn.

A táblák közötti összefüggések és keresőfeltételek megadásával készítjük a lekérdezéseket. A lekérdezés segítségével egyszerre több tábla adatait is vizsgálhatjuk, egyszerre frissíthetünk, vagy törölhetünk több rekordot, illetve beépített és egyénileg megadott számításokat végezhetünk a rekordok adatainak felhasználásával. A lekérdezések alapjául más lekérdezések eredményei is felhasználhatók. Az adatok kigyűjtésének, speciális feldolgozásának eredményét jelentések formájában nyomtatjuk ki. Készíthetünk olyan jelentést, amely adatokat egyesít és összegez, vagy akár olyat,

amely egy speciális formátumban borítékokat címez meg, árucím-kéket nyomtat.

Az Access adatbázis-kezelő objektumai tehát a táblák (*Tables*), űrlapok (*Forms*), lekérdezések (*Queries*), jelentések (*Reports*), valamint a makrók (*Macros*) és az adatbázishoz tartozó – a legújabb változatban Visual Basic nyelvű – programokat tartalmazó modulok (*Modules*), és az Access XP-ben megjelent, Interneten keresztüli adatbázis-kapcsolatot biztosító adatelérési lapok (*Pages*).

A **normalizálás** folyamata során az adatbázisból kiküszöböljük a különféle anomáliákat, a redundanciát (a felesleges és káros adat-többszörözést), csökken a tárolási igény, az adatbázis logikailag áttekinthetőbb lesz. Minimális redundanciára a táblák közötti kapcsolatok biztosítása érdekében mindig szükség van (mivel a kapcsolatba hozott tábláknak tárolni kell ugyanazt az azonosítót).

## Adatbázisok tervezése



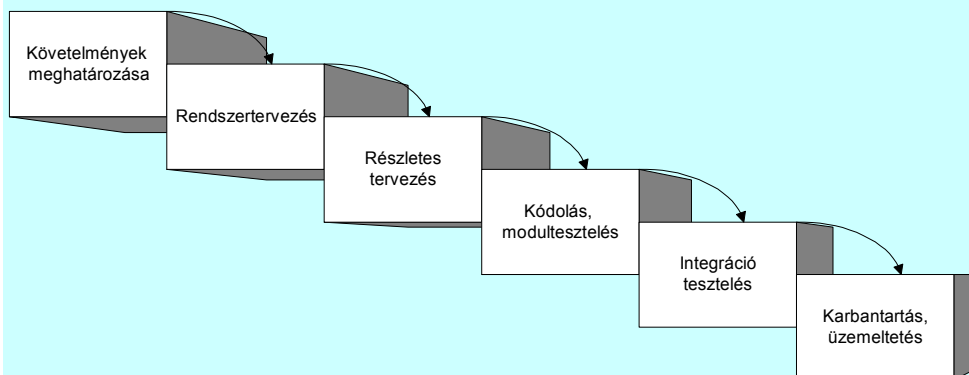
Csak akkor remélhetjük, hogy a programmal létrehozott adatbázisunk megfelelően eredményesen, pontosan és hatékonyan fog működni, ha még az Access betöltése előtt alaposan átgondoljuk a megoldandó feladatot, meghatározzuk a tárolandó egyedtypusokat, az alkalmazás szempontjából fontos tulajdonságaikat stb., megtervezzük az adatbázist. Például egy kórházi betegnyilvántartás szempontjából a személy egyedtypus fontos tulajdonsága a kórelőzmény, a vérnyomás stb. nem lényeges viszont, hogy hány órát dolgozott az adott hónapban. Egy béradatokat számoló, nyilvántartó alkalmazásban pedig éppen ez a tulajdonság a meghatározó.

Információs rendszerek elemzésénél több elfogadott rendszer-életciklus-modell van forgalomban. Röviden ismertetjük legalább az egyiket, az úgynevezett *vizesés életciklus-modellt*, mert a későbbiekben olyan fogalmakról, kifejezésekről ejtünk szót, amelyek megértéséhez legalább erre szükségünk van. Ezt az életciklus modellt támogatja egyébként az Oracle is, a Designer 2000-ről szóló anyagaiban.

Az 1. ábrán látható modell alkalmazása kézenfekvő. Minden fejlesztési szakaszból vissza lehet lépni az előző szakaszba és így



fokozatosan közelítve érjük el a megoldást (ez kissé hasonlít egy másik, a bonyolultabb rendszerekhez kidolgozott spirális életciklus-modellre).



1. ábra. A vízesés életciklus-modell

A következőkben nyolc alapvető lépésben ismertetjük azt a módszert, amelyet az Access adatbázis tervezésénél alkalmazhatunk.

## 1. lépés: Követelményelemzés

Az első lépésben a megoldandó feladatot, az adatbázis célját, az alkalmazás rendeltetését határozzuk meg. Legelőször vizsgáljuk meg az alkalmazási területet, hogyan oldják meg hagyományos eszközökkel a feladatot. Milyen adatokat kapnak a felhasználók, hogyan dolgozzák azokat fel, mi az eredmény, milyen jelentések készülnek, vagyis milyen információkhoz szeretnének jutni az adatbázisból. Vizsgáljuk meg a hasonló felépítésű, szerepű, működő adatbázisokat. Tapasztalatainkat írásban rögzítjük. Készítünk „interjúkat” az adatbázis leendő használóival, gyűjtjük össze az adatok felvételére jelenleg használt űrlapokat. Tudjuk meg, milyen információkhoz szeretnének jutni az adatbázisból. Ezek alapján határozzuk meg, hogy milyen témákról, egyedtípusokról kell adatokat tárolni (táblák), és milyen adatokat kell tárolni az egyes témákról (a táblák mezői). Az adatbázis tervezés elmélete szerint ezzel határozzuk meg az adatbázis belső szerkezetének kialakításához szükséges elemeket. Az első lépésben alkalmazhatók tehát a hagyományos technikák: