

Peter Belko

Microsoft®



Access 2013

Podrobná
uživatelská
příručka

Od základů až po administraci databáze
Dokonalé využití novinek verze 2013
Spolupráce s Microsoft SharePoint
Otázky a odpovědi za každou kapitolou

computer
press

Peter Belko

Microsoft Access 2013

Podrobná uživatelská příručka

**Computer Press
Brno
2014**

Microsoft Access 2013

Podrobná uživatelská příručka

Peter Belko

Překlad: Martin Herodek

Obálka: Martin Sodomka

Odpovědný redaktor: Libor Pácl

Technický redaktor: Jiří Matoušek

Objednávky knih:

<http://knihy.cpress.cz>

www.albatrosmedia.cz

eshop@albatrosmedia.cz

bezplatná linka 800 555 513

ISBN 978-80-251-4125-0

Vydalo nakladatelství Computer Press v Brně roku 2014 ve společnosti Albatros Media a. s. se sídlem Na Pankráci 30, Praha 4. Číslo publikace 18253.

© Albatros Media a. s. Všechna práva vyhrazena. Žádná část této publikace nesmí být kopírována a rozmnožována za účelem rozšiřování v jakékoli formě či jakýmkoli způsobem bez písemného souhlasu vydavatele.

1. vydání

 **ALBATROS** MEDIA a.s.

Obsah

Úvod	9
Poděkování	12

1

Základy práce s databází 13

Microsoft Access – úvodní teoretické informace	14
Co je Microsoft Access	14
Kdy je vhodné použít Access	14
Jednoduché vysvětlení, co je databáze	15
Možnosti databáze	16
Dva typy aplikací v Accessu 2013	16
Základní části databáze Microsoft Access	17
Tabulky	17
Dotazy	18
Formuláře	18
Sestavy	19
Makra	19
Moduly	19
Možnosti spuštění Microsoft Access	20
Spuštění z nabídky Start nebo Úvodní obrazovky Windows 8	20
Spuštění pomocí odkazů	20
Prostředí Accessu 2013	20
Vytvoření prázdné databáze	20
Vytvoření databáze ze šablony	21
Získání pomoci	29
Základní návrh jednoduché databáze	30
Určení typu databáze a jejího použití	31
Návrh tabulek	34
Datové typy	37
Vztahy mezi tabulkami	38
Návrh uživatelského rozhraní	39
Návrh tiskových výstupů	40
Otázky a odpovědi	41
Stručný souhrn	41

2

Tabulky 43

Vytvoření tabulky	44
Nová tabulka	44
Návrh tabulky	46
Seznamy služby SharePoint	48

Práce s tabulkou	53
Označení, otevření a zavření tabulky	53
Možnosti zobrazení tabulky	54
Přejmenování, kopírování a mazání tabulek	54
Zobrazení datového listu	55
Práce s poli tabulky	55
Vložení a název pole (sloupce)	56
Nastavení datového typu a vlastností	57
Odstranění pole	59
Práce se záznamy (řádky)	59
Přidání, změna a odstranění záznamu	59
Filtrování a řazení záznamů	60
Další možnosti filtrování záznamů	62
Vyhledávání a nahrazování údajů v záznamech	64
Formátování textu	65
Vlastnosti tabulky	66
Návrhové zobrazení tabulky	66
Vytvoření polí, datové typy a popis	67
Nejdůležitější vlastnosti polí	68
Velikost pole	68
Formát	69
Přesnost	69
Počet desetinných míst	69
Vstupní maska	69
Titulek	72
Výchozí hodnota	72
Ověřovací pravidlo	72
Ověřovací text	73
Povolit nulovou délku	73
Je nutno zadat	73
Indexovat	73
Vyhledávání	73
Přehled nejdůležitějších datových typů	76
Krátký text	76
Dlouhý text	76
Číslo	76
Datum a čas	76
Měna	77
Automatické číslo	77
Ano/Ne	77
Počítané pole	77
Další nastavení tabulky	77
Primární klíč	77
Indexy	78
Údajová makra	80
Relace mezi tabulkami	82
Vytvoření relace	82
Referenční integrita	85
Otázky a odpovědi	85
Stručný souhrn	87

3

Dotazy

89

Typy dotazů v Accessu	90
Výběrový dotaz	90
Vytvářecí dotaz	90
Přidávací dotaz	90
Aktualizační dotaz	91
Křížový dotaz	91
Odstraňovací dotaz	91
SQL dotazy	91
Vytvoření výběrového dotazu pomocí průvodce	91
Jednoduchý podrobný dotaz	91
Jednoduchý souhrnný dotaz	95
Vyhledání duplicitních údajů pomocí dotazu	97
Úprava a použití dotazu v návrhovém zobrazení	98
Spuštění dotazu	99
Zobrazení a odstranění tabulek a dotazů	99
Vztahy mezi tabulkami a dotazy	100
Přidání polí	102
Kritéria v dotazu	103
Řazení	108
Využití tvůrce dotazů při úpravě pole v tabulce	109
Výpočty a souhrny v dotazu	111
Vytvoření výpočtu	112
Tvůrce výrazů	113
Nejčastěji používané výpočty	114
Vytvoření souhrnu	122
Parametrické dotazy	124
Vytvoření parametrického dotazu	124
Nastavení parametrů dotazu	125
Křížový dotaz	126
Akční dotazy	129
Vytvoření a spuštění akčního dotazu	129
Vytvoření tabulky pomocí dotazu	130
Přidání údajů pomocí dotazu	132
Aktualizace údajů pomocí dotazu	135
Odstranění údajů pomocí dotazu	139
Sjednocení údajů pomocí příkazů jazyka SQL	140
Otázky a odpovědi	143
Stručný souhrn	144

4

Formuláře

147

Vytvoření formuláře	149
Jednoduchý formulář	150
Součástí formuláře	151
Rozdělený formulář	152

Více položek	153
Datový list	154
Prázdný formulář	154
Vytvoření formuláře pomocí průvodce	155
Práce s údaji na formuláři	159
Zobrazení rozložení a úpravy formuláře	160
Seznam vlastností	161
Návrh	162
Uspořádání	165
Formát	167
Návrhové zobrazení formuláře	169
Vytvoření formuláře v návrhovém zobrazení	169
Práce v návrhovém zobrazení	171
Ovládací prvky formuláře	172
Formátování a úprava formuláře	174
Nastavení, použití a vlastnosti ovládacích prvků	178
Textové pole	178
Změna textového pole na pole se seznamem	181
Průvodce rozevírácím seznamem	183
Obdélník, čára a popisek	186
Uzamknutí a zpřístupnění ovládacího prvku	189
Příkazové tlačítko	190
Zaškrtačací políčko	191
Podformulář	192
Formulář jako dialogové okno	194
Navigační formuláře	198
Otázky a odpovědi	200
Stručný souhrn	201

5

Sestavy

203

Použití a tisk sestavy	204
Zobrazení sestavy	205
Náhled před tiskem	206
Tisk sestavy	207
Vytvoření a úprava sestavy	208
Základní sestava	208
Průvodce sestavou	209
Zobrazení rozložení	222
Návrhové zobrazení sestavy	230
Návrh sestavy, prázdná sestava a podsestava	238
Vytvoření a úprava faktury k tisku pro zákazníky	243
Otázky a odpovědi	247
Stručný souhrn	249

6

Makra

251

Vytvoření a úprava jednoduchého makra	253
Okno se zprávou	254
Podmínka If ... Else If ... Else ... End If	255
Submakro a přesouvání akcí	256
Spustit Makro	257
Komentář	258
Spustit Příkaz Nabídky	259
Ukončit Access	260
Zobrazit Ukazatel Přesýpací Hodiny	260
Nastavit Varování	261
Použití vytvořeného makra	261
Spuštění v návrhovém zobrazení a testování	261
Spuštění z navigačního podokna	263
Spuštění při události ve formuláři/sestavě	263
Vytvoření vloženého makra Při kliknutí	274
Konvertování makra do jazyka VBA	275
Komplexní úlohy na procvičení	279
Vlastní ovládací formulář s makry	280
Přesouvání záznamů pomocí akčních dotazů a maker	289
Otázky a odpovědi	295
Stručný souhrn	297

7

Import a export

299

Možnosti importu a exportu	300
Import a propojení údajů	302
Import údajů z Excelu	302
Připojení kopií záznamů k tabulce	305
Odkaz a propojení na sešit Excel	305
Import a propojení údajů z jiné accessové databáze	307
Propojení tabulky z SQL serveru	309
Import údajů z textového souboru	311
Export údajů	313
Export do excelového sešitu	313
Export do textového souboru	315
Export do PDF	317
Export e-mailem	317
Export do Accessu	317
Exportování do hromadné korespondence Wordu	318
Další možnosti exportu	318
Uložení nastavení importu a exportu	318
Spolupráce se serverem SharePoint	320
Otevření a propojení údajů ze SharePointu v Accessu	321
Export tabulky na SharePoint on-line	325
Základní informace o vytvoření webové aplikace	326

Vytvoření vlastní jednoduché webové aplikace	331
Zveřejnění aplikace na SharePoint	345
Otázky a odpovědi	350
Stručný souhrn	352

8

Správa databáze 355

Navigační podokno	356
Možnosti zobrazení podokna	356
Vyhledávání	357
Závislosti objektů	357
Nastavení databázové aplikace	358
Název a ikona aplikace	360
Zobrazit formulář	360
Rozdělení databáze	360
Důvody rozdělení databáze	360
Rozdělení databáze	361
Správa propojených tabulek	362
Zabezpečení databáze	363
Zašifrování heslem	363
Možnosti distribuce databáze	364
Access RunTime na distribuci databáze	367
Důvěryhodné databáze a umístění	368
Předcházení problémům s databází	368
Komprimování a oprava databáze	369
Zálohování databáze	370
Uložení databáze	371
Dokumentace databáze	372
Analýza výkonu databáze a tabulek	373
Porovnání databází	375
Důležité novinky v Accessu 2013	377
Přechod z Accessu 2010	377
Přechod z Accessu 2007	378
Přechod z Accessu 2003	379
Otázky a odpovědi	379
Stručný souhrn	381
Použitá literatura	382
Rejstřík	383

Úvod

Publikace, kterou držíte v rukou, vám poslouží k získání vstupních znalostí do světa databází. Databází, které si dokáže vytvořit i zkušený uživatel bez znalostí programování na svém vlastním počítači. Access často začínají využívat uživatelé, kterým už nestačí na zpracování údajů Excel. Vzhledem k tomu, že Access patří do rodiny kancelářského balíku Microsoft Office, je takovýto přechod poměrně jednoduchý, pokud uživatelé znají základní postupy při vytváření databází.

Tato publikace je napsána na příkladu jednoduché aplikace na zpracování faktur. Postupy dokážete využít nejen u faktur, ale i u objednávek, rezervací, evidence zaměstnanců, sportovců, uchazečů o práci, záznamů o vypůjčených knihách, CD/DVD, soupisu majetku a dalších. Princip je vždy velmi podobný. Stačí, pokud si faktury zaměníte za jiný subjekt a přizpůsobíte návrh specifickým požadavkům v oblasti, pro kterou databázi vytváříte.

Pokud jste dosud s databázemi v Accessu nepracovali, doporučujeme, abyste knihu četli postupně, jelikož jednotlivé kapitoly na sebe navazují. Pokud vynecháte například první nebo druhou kapitolu, nebudete moct začít s formuláři nebo sestavami. Na jejich vytvoření potřebujete tabulky. Prvních šest kapitol postupně popisuje krok za krokem vytvoření databáze na zpracování faktur. Poslední dvě jsou zaměřené na import/export údajů a správu databáze. Pokud jste s accessovými databázemi už pracovali, některé popisované postupy, které jsou vám známé, můžete vynechat, respektive si je projít jen stručně, abyste věděli, jak se vytvářená databáze postupně mění a přibývají v ní nové databázové objekty.

V úvodní kapitole jsou popsány základní informace o aplikaci Access, kdy ho použít, z jakých objektů se skládá a jak ho spustit. Následuje popis prostředí aplikace, které se v porovnání s předchozími verzemi příliš nezměnilo. Zásadní změna nastala při přechodu z verze 2003 na novější, jelikož bylo uživatelské prostředí kompletně přepracováno. Ve stručnosti popisujeme základní návrh databáze, který je vhodné u komplexnějších projektů načrtnout na papír a až potom začít vytvářet databázi v Accessu. Co se týká návrhu databáze, kniha nemá takový rozsah, abychom mohli popsat celou teorii databází. Proto pokud chcete poznat všechny teoretické databázové detaily, je vhodné přečíst si vedle této publikace i speciální knihy a učebnice, které popisují detailně návrh databázové aplikace.

Druhá kapitola se věnuje nejdůležitějšímu databázovému objektu – tabulkám, bez kterých není možné dále pracovat a vytvářet jiné databázové objekty. Popisujeme několik způsobů, jak se dá tabulka vytvořit. Vycházíme z návrhu databáze Faktury, který je navržen v první kapitole. Následuje popis práce s tabulkou a jejími záznamy, pokud byste ji chtěli využívat na analyzování údajů bez vytvoření aplikace. V návrhovém zobrazení tabulky si přečtete, jak upravit datový typ a vlastnosti pole. Tyto jsou důležité na správné uchování údajů. Následuje přehled nejdůležitějších datových typů, jejichž vlastnosti uvádíme podle dostupné literatury a technické specifikace Accessu. Vzhledem k tomu, že Access funguje trochu jinak než Excel, jsou údaje rozdělené ve více normalizovaných tabulkách. Tyto je potřeba vzájemně spojit pomocí primárního a cizího klíče. Tyto informace jsou popsány v závěru kapitoly.

Třetí kapitola popisuje použití dotazů. Jsou to objekty, které se vytvářejí z údajů v tabulkách. Můžete tak spojit informace z více tabulek do jednoho dotazu, který je bude zobrazovat jako

jednu virtuální tabulku. Vymazáním objektu dotazu se nevymažou údaje v tabulkách, jelikož dotaz je jen zobrazuje. Kromě spojení více informací do jednoho celku provádíte v dotazech i výpočty. Můžete spočítat například celkovou cenu z množství a jednotkové ceny. Dotazy jsou výběrové a akční. Výběrové mohou být podrobné a souhrnné. Akční zase manipulují s údaji v tabulkách. Každý dotaz může obsahovat kritéria a parametry, pomocí kterých zpřesňujete zobrazení výsledků. Dotazy vytváříte v návrhovém zobrazení dotazu. Využít můžete i přímý zápis v jazyce SQL, jelikož každý dotaz ho využívá, i když ho u návrhového zobrazení nevidíte. Doporučujeme vám podívat se na vytvořené dotazy i v zobrazení SQL, abyste tyto informace mohli využít při přechodu na SQL server, pokud vám už Access nebude stačit.

Čtvrtá kapitola se věnuje formulářům, které slouží k vytvoření uživatelského prostředí pro koncové uživatele. Koncový uživatel se záznamy pracuje přes formuláře, nikoliv tabulky/dotazy. Formuláře vytváříte z tabulek/dotazů a můžete je upravit tak, aby jejich použití bylo efektivní a co nejjednodušší. Vytváříte je pomocí průvodce, v zobrazení rozložení a v návrhovém zobrazení. Mohou obsahovat i podformuláře a takto zobrazujete údaje z více tabulek na jednom místě. Koncový uživatel bude vidět fakturu jako celek a nemusí se zabývat tím, že údaje jsou rozděleny ve více tabulkách a výpočty jsou realizovány v dotazech. Formuláře můžete kromě práce s údaji využít jako dialogová okna, kdy nejsou navázaná na žádnou tabulku/dotaz. Stejně je můžete použít na vytvoření hlavního navigačního formuláře, ze kterého uživatelé otevírají ostatní objekty.

V páté kapitole najdete popis sestav, které jsou potřeba na vytvoření tiskových výstupů. Fakturu pro odběratele vytisknete jako sestavu, až když ji vytvoříte ve formuláři. Kromě tisku je můžete použít na zobrazení údajů, které jsou různě seskupené a seřazené. Jejich vytvoření je podobné jako u formulářů – pomocí průvodce, v návrhovém zobrazení a z prázdné sestavy. Hlavní rozdíl je v tom, že v sestavě nemůžete měnit údaje, jelikož jsou jen ke čtení. U sestav je důležité zohlednit limitující faktor – velikost papíru, na který se sestava bude tisknout. Proto je potřeba návrhu věnovat dostatečnou pozornost a ovládací prvky rozmístit tak, aby zobrazovaly všechny potřebné údaje na šířku papíru.

Šestá kapitola popisuje použití maker, která přidávají do aplikace logiku ovládání. Na formuláře můžete přidat příkazová tlačítka, která budou otevírat jiné formuláře nebo sestavy. Při vhodném nastavení můžete zabezpečit, že se z formuláře faktury tlačítkem otevře sestava, která zobrazuje jen tu konkrétní fakturu a ne všechny. Projdete si, jak vypadá návrhové zobrazení makra, jak přidat akce a parametry pro akci. Vytvořená makra se spouštějí z událostí na formulářích/sestavách a podíváte se na stručný popis těch nejdůležitějších událostí. Ostatní si můžete vyzkoušet postupně, když budete vytvářet další databáze. Často se stává, že některé události používáte velmi často a některé události nepoužijete nikdy. Od maker je to už jen kousek k programování v jazyce Visual Basic for Applications (VBA). Abyste pochopili jeho základy, je možné makra konvertovat do VBA a tak si prohlédnout strukturu kódu. Detailní znalosti je vhodné nastudovat z knihy, která se věnuje programování ve VBA. V závěrečné části kapitoly je popis vytvoření jednoduchého přepínacího formuláře s makry a popis archivace starých záznamů pomocí akčních dotazů, které jsou spouštěny makrem.

Sedmá kapitola už není přímo navázaná na databázi Faktury, kterou jste postupně vytvářeli pomocí postupů v předchozích kapitolách. Věnuje se importu a exportu údajů z jiných zdrojů. Pomocí nástrojů na import můžete do vlastní databáze přidat údaje z jiných aplikací. Přímo

jsou dostupné nástroje na import z Excelu, webu, textového souboru, Accessu, ODBC – SQL serveru, SharePointu a dalších. Podobně můžete z vaší databáze exportovat údaje do jiného formátu. Postupy využijete i tehdy, pokud nevytváříte databázovou aplikaci, jen potřebujete Access využít ke zpracování a analýze údajů z jiných aplikací. Některé typy propojených údajů jsou jen k zobrazení, některé můžete měnit na obou místech a synchronizují se obousměrně.

Access je výrazně navázaný i na SharePoint, který máte ve své infrastruktuře nebo jej využíváte prostřednictvím služby Office 365. Můžete zobrazovat a používat údaje ze SharePoint seznamů bez toho, abyste museli vždy otevírat internetový prohlížeč a zobrazovat web SharePoint. Kromě propojení seznamů je v Accessu velmi zajímavá novinka, která umožňuje vytvářet webové aplikace. Ty mohou uživatelé využívat i bez toho, aby měli Access. Stačí jim jen webový prohlížeč a přístup na SharePoint. V závěrečné části kapitoly je popis, jak vytvořit jednoduchou webovou aplikaci na SharePointu ve službě Office 365.

Poslední kapitola se věnuje správě databáze, kde si přečtete, jak rozdělit databázi na front-end a back-end. Toto poslouží k bezproblémové úpravě uživatelského prostředí bez ovlivnění uživatelů a údajů, které už jsou v databázi. Přečtete si i možnosti, jak opravit a komprimovat databázi, pokud by se dostala do nestabilního stavu a její velikost by byla neúměrná zadaným údajům. Důležité je databázi průběžně zálohovat, abyste ji v případě havárie mohli obnovit. Pro kritické situace je dobré vytvořit i dokumentaci databáze, pomocí které ji můžete zrekonstruovat. Po vytvoření tabulek a zadání několika testovacích údajů je vhodné využít i nástroje na analýzu databáze a tabulek. Takto můžete hned na začátku vytváření databáze vyřešit problémy s tabulkami, které nejsou navrženy v souladu s normalizací. V úplném závěru je stručný seznam novinek podle toho, z jaké verze Accessu přecházíte na tuto verzi.

Ke knize jsou k dispozici vzorové soubory databáze Faktury ve verzích postupně, jak vznikala. Uložené jsou ve složkách podle jednotlivých kapitol. Název každého souboru je doplněn kvůli přehlednosti pořadovým číslem (Faktury2, Faktury3, ...). Proto si ve svém počítači můžete soubor přejmenovat odstraněním pořadového čísla, abyste měli název databázového souboru v souladu s textem v popisovaných postupech, kde ho uvádíme bez pořadového čísla.

Ideální stav je, pokud budete pracovat s jedním souborem od začátku a postupně mu budete přidávat podle popsaných postupů nové objekty a vytvoříte i vlastní objekty podle potřeby, abyste si procvičili získané znalosti. Příložené soubory v takovémto případě využijete jen jako pomůcku ke kontrole, zda postupujete správně. Kromě databázových souborů najdete v některých složkách i testovací soubory na import/propojení údajů s Accessem nebo na porovnání databází. Vzorové soubory jsou dostupné na adrese <http://knihy.cpress.cz/K2103>. Po stažení je uložte do složky C:\Access 2013, jelikož vytvořená propojení z postupů v kapitolách 7 a 8 jsou nastavena na tuto složku. Pokud je uložíte do jiné, nebudou fungovat soubory z předposlední a poslední kapitoly, dokud nepoužijete správce propojených tabulek. Soubory jsou vždy ve stavu podle všech postupů jako na konci kapitoly.



Poznámka: Databáze Faktury, kterou budete postupně vytvářet ze vzorového příkladu, obsahuje testovací náhodně vygenerované údaje o fiktivních zákaznících, jako i starší údaje z různých ceníků IT produktů, jejichž ceny nejsou v žádném případě aktuální a slouží jen k cvičným a testovacím účelům při práci s databází. Podobnost údajů z databáze Faktury se skutečnými je náhodná.

Zpracování velkého množství údajů a informací se postupně stane důležitou úlohou informačních pracovníků a ne vždy se to podaří provést v Excelu. Proto věříme, že vám tato publikace pomůže při řešení zpracování databázových údajů, které buď postupně do databáze zadáte tak, jak budou vznikat, nebo je importujete z jiných zdrojů. Zkušený uživatel může vytvořit jednoduchou aplikaci s uživatelským prostředím pro koncové uživatele, kteří následně budou s údaji pracovat. Po zvládnutí postupů z této publikace můžete přejít ke studiu programování ve VBA nebo začít postupně poznávat práci s údaji na SQL Serveru. Důležité je všechny postupy z knihy prakticky zkoušet a vytvářet různé nové databáze, jelikož potřebnou zručnost získáte praktickým používáním. Tato publikace vám pomůže zorientovat se v používání aplikace Microsoft Access 2013.

Poděkování

Na tomto místě bych chtěl poděkovat rodině a kamarádům, kteří mi drželi palce a podporovali mě během psaní knihy. Děkuji i kolegům z firmy exe, spol. s r. o., a čtenářům mého blogu za občasně náměty a připomínky z jejich praxe. Podobně přispěli náměty i účastníci mých školení a konzultací ve firmách. Za projevenou důvěru, nabídku napsat tuto knihu a cenné rady během psaní děkuji panu Liborovi Páclovi z vydavatelství Computer Press, jako i překladateli Martinovi Herodkovi.

Speciálně bych chtěl poděkovat Filipovi Mazánovi za cenné připomínky, tipy a rady v databázové teorii.

Peter Belko

1

Základy práce s databází

V této kapitole:

Microsoft Access – úvodní teoretické informace

Základní části databáze Microsoft Access

Možnosti spuštění Microsoft Access

Prostředí Accessu 2013

Základní návrh jednoduché databáze

Otázky a odpovědi

Stručný souhrn

Microsoft Access – úvodní teoretické informace

Co je Microsoft Access

Access 2013 je desktopová relační databázová aplikace, která je součástí sady Office Professional. Využívá se k vytváření jednoduchých databázových aplikací, které se používají buď na jednom počítači samostatně, nebo v síti na více počítačích, kde může s daty pracovat současně více uživatelů. Kromě vytváření databázových aplikací poslouží i k uchovávání množství údajů a jejich analýze pomocí množství vestavěných funkcí. Uživatelé technologie SharePoint mohou Access použít k vytváření webových aplikací stejně jako i analyzování dat uložených na SharePointu. Kromě SharePointu je možné napojení na údaje v dalších zdrojích, jako je SQL server, Excel, jiná databáze Access, textový, HTML nebo XML soubor a několik dalších.

Access mohou využít i uživatelé pouze se základními znalostmi databází, jelikož obsahuje hodně zajímavých předpřipravených šablon databází. Tyto mohou uživatelé upravit podle potřeby. Na úpravu databázových objektů je k dispozici množství přehledných průvodců, pomocí nichž je tvorba a úprava databáze poměrně jednoduchá a není nutné znát programovací jazyky. Koncoví uživatelé vytvořené databáze nepotřebují Access ovládat, stačí, pokud umí používat počítač, a v prostředí vlastní databáze se zorientují po krátkém zaškolení. Koncoví uživatelé pracují s údaji, nikoliv s konfigurací nebo úpravou databázových objektů.

Kdy je vhodné použít Access

Access využijete vždy, když potřebujete uchovávat množství údajů, mezi kterými jsou různé souvislosti, a Excel už na to nestačí kvůli přehlednosti a ne zrovna vhodné týmové práci po síti. Hodí se i v případech, kdy je údajů více, než má Excel dostupných řádků. Můžete samozřejmě využít nástroj PowerPivot, který v Excelu poskytuje více řádků, než je 1 048 576, Access ale umožňuje jednodušší tvorbu uživatelského prostředí databáze a tiskových sestav. Disponuje množstvím průvodců a vizuálních návrhářů.

Access je tedy vhodné použít vždy, když potřebujete uchovávat různé údaje, mezi kterými existují závislosti. Například potřebujete uchovávat seznam zákazníků, jejich objednávek a objednaného zboží, jako i samotný seznam zboží, které nabízíte. Následně potřebujete, aby s těmito údaji efektivně pracovali uživatelé, kteří nejsou databázovými specialisty. Kromě uchovávání údajů je mohou uživatelé různě filtrovat a vytvářet pohledy na ně podle různých obchodních kritérií.

Využijí ho i personální manažeři a společnosti, které evidují údaje o uchazečích o práci. Tyto údaje potom mohou analyzovat a vyhledávat tak vhodné pracovníky podle přesných kritérií znalostí na pracovní pozici.

Access je také vhodný na práci s propojenými údaji z SQL serveru nebo SharePointu. Tyto údaje můžete nejčastěji filtrovat a analyzovat podle různých obchodních a podnikových požadavků. V kombinaci se SharePointem je dostupná i poměrně jednoduchá tvorba webových aplikací.

Access poslouží i vývojářům, kteří připravují databázové aplikace pro své zákazníky přesně podle jejich požadavků nebo připravují databáze, které následně předávají jako samostatné aplikace.

Domácí uživatelé ho využijí k vytvoření adresáře kontaktů, správy domácích účtů nebo seznamu knih/CD/DVD a jejich výpůjček kamarádům či známým spolu s termíny zapůjčení. Následně je možné údaje analyzovat a zjistit, který titul je mezi kamarády nejžádanější.

Využití databázové aplikace je poměrně rozsáhlé a stačí jen zvolit vhodný nápad a návrh, který potom transformujete do konkrétní aplikace, která pomůže dosažení obchodních a podnikových cílů.

Jednoduché vysvětlení, co je databáze

Při práci s počítačem se setkáváte také nepřímě s databázemi, jsou to například kontakty uložené v tabulce nebo dokumentu, který můžete považovat za jednoduchou databázi kontaktů. Podobně můžete zapisovat do tabulky údaje o telefonátech jednotlivým kontaktům. Ty budou zase jednoduchou databází historie komunikace. Má to však jednu nevýhodu. Po čase se údaje rozšíří natolik, že spravovat závislosti mezi historií a kontakty bude značně problematické a neefektivní. Jednoduché úlohy se komplikují a čas na jejich dokončení se prodlužuje. Proto je vhodné sáhnout po relační databázi, která údaje v tabulkách spojuje, a práce s údaji je tak jednodušší a rychlejší.

Aby databáze správně fungovala, je potřeba věnovat jejímu návrhu dostatečnou pozornost. Důležité je připravit tabulky tak, aby splňovaly požadavky normalizace, redukovaly se nadbytečné (redundantní) údaje a předcházelo se anomáliím při vkládání, aktualizaci a mazání záznamů. To znamená, že například údaje o zákaznících a jejich objednávkách zboží by v jediné tabulce způsobovaly redundantní údaje o zákazníkovi. Je zbytečné, aby u každého objednaného zboží byly uvedeny všechny kontaktní údaje zákazníka. Vyvarovat se toho můžete tak, že rozdělíte údaje o zákaznících a objednávkách do více tabulek a tyto budou propojené pomocí identifikačního čísla – primárního klíče v jedné tabulce a pomocí cizího klíče v druhé tabulce.

Postup normalizace tabulek popisuje pět normálních forem (NF). Každá klade na tabulku určité požadavky, které musí být splněny. Pokud chcete mít tabulku v 2NF, musí být tato v 1NF. Podobně to platí i pro 3NF, kdy tabulka musí být v 2NF a samozřejmě i v 1NF. Tabulky v databázi obsahují pole (sloupce) a záznamy (řádky). V polích jsou uvedeny konkrétní údaje, jako je například jméno, příjmení, adresa, telefon apod. Celý řádek tak tvoří záznam o zákazníkovi. V objednávkách potom stačí přidat jen identifikační číslo zákazníka, na základě něhož se identifikuje záznam z tabulky zákazníků. Celé propojení se realizuje primárním klíčem v tabulce zákazníků a cizím klíčem v tabulce objednávek.

Teorii relačních databází se věnuje samostatná kniha (například *Vytváříme relační databázové aplikace* od R. M. Riordan), my v této kapitole podle uvedené dostupné literatury uvádíme jen základní vysvětlení a nejdůležitější pojmy z terminologie relačních databází:

- **Relace** – informace o jednotlivém subjektu, například objednávky, zákazníci, školy a studenti. Relace je obvykle v systému relační databáze uložena jako tabulka. Podrobněji

je tento pojem rozepsán na straně 9 v knize *Vytváříme relační databázové aplikace*, jelikož v Accessu má i význam vztahu mezi tabulkami.

- **Atribut** – konkrétní informace o subjektu, například kontaktní údaje zákazníka. Atribut je v tabulce uložen jako sloupec (pole).
- **Vztah** – způsob, jak jsou informace z jedné relace napojené na informace z jiné relace. Tyto mohou nabývat několika typů, například jeden zákazník má více objednávek a tehdy je vztah 1:N. V případě, že by zákazník mohl objednat jen jedinkrát, jednalo by se o vztah 1:1. Pokud by však zákazník objednával od více dodavatelů a dodavatelé by přijímali objednávky od více zákazníků, jednalo by se o vztah M:N. Tento vztah se v relačních databázích musí upravit pomocí prostředníka tak, aby mezi nimi vznikly vztahy 1:N.
- **Spojení** – způsob, jak jsou údaje konkrétně propojené mezi dvěma tabulkami. Například v objednávkách je identifikační číslo zákazníka, kterým se rozlišuje, komu patří daná objednávka. Spojení je možné zrealizovat jen se stejným datovým typem, tj. číslo s číslem, nikoliv však číslo s textem.

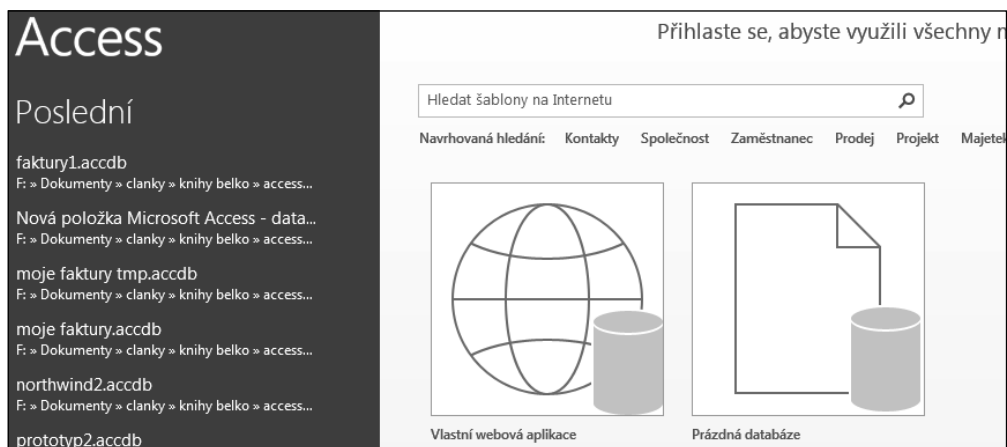
Možnosti databáze

Kniha J. L. Viescase *Mistrovství v Microsoft Access* uvádí, že relační databázový systém poskytuje plnou kontrolu nad definováním údajů, prací s údaji a jejich řízením.

- **Definování údajů** – v databázích definujete, jaké údaje budete ukládat, jejich datové typy a jaké vztahy budou mezi údaji. Můžete také definovat, jak budou údaje formátovány a případně ověřovány, abyste se vyvarovali zadávání nesprávných údajů.
- **Manipulování s údaji** – s uloženými údaji můžete dále pracovat tak, že je filtrujete, řadíte nebo vybíráte jen některá pole k zobrazení. Kromě toho můžete údaje vzájemně spojovat a zobrazovat tak údaje z více tabulek. Databázový systém umožňuje údaje aktualizovat, odstraňovat, přidávat a přesouvat, dokonce i vytvořit novou tabulku se zkopírovanými údaji.
- **Řízení dat** – v databázových systémech určujete, kdo může vidět konkrétní údaje, kdo je může upravovat nebo přidávat. Také určujete, jak se bude pracovat s údaji při používání více uživatelů.

Dva typy aplikací v Accessu 2013

Access 2013 umožňuje vytvářet dva typy databází: *souborové aplikace* a *webové aplikace*. Souborové se vytvářejí a používají stejně jako v předchozích verzích Accessu, tj. vytvoříte tabulky, dotazy, formuláře a další databázové objekty. Databázi spouštíte z lokálního souboru. U webových aplikací potřebujete server SharePoint, na kterém se databáze publikuje. Používá se přes webový prohlížeč a koncoví uživatelé nepotřebují Access ve svých počítačích. Takováto databáze se vytváří v grafickém editoru, a proto na jednoduché aplikace není nutné znát programovací techniky.



Obrázek 1.1: Výběr šablon z úvodní obrazovky – vlastní webová aplikace a prázdná databáze

Základní části databáze Microsoft Access

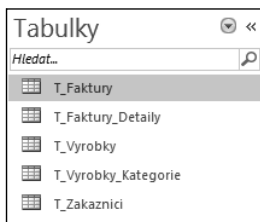
Databázový systém Access se skládá z několika typů objektů. Tyto se ukládají pod konkrétními názvy a jsou dostupné v navigačním podokně na levé straně aplikace.

Tabulky

Nejdůležitější částí databáze jsou tabulky. Tyto uchovávají údaje, s kterými se dále pracuje v jiných objektech. Při návrhu databáze je důležité správně navrhnout tabulky a následně vztahy mezi nimi. Pokud se v této fázi něco zanedbá, další úprava a změny mohou být poměrně náročné. Tabulky je vhodné vytvořit tak, aby splňovaly normální formy.

V každé tabulce je potřeba zvolit i vhodné datové typy. To znamená, že tam, kde budete uchovávat čísla, která budou součástí výpočtů, zvolíte číselný datový typ. Kde budou finanční údaje, zvolíte typ měna, u textů zase textový typ a délku textu a podobně postupujete i u jiných typů údajů.

V databázových tabulkách se standardně nedělají výpočty jako v Excelu, i když Access 2013 dokáže základní výpočet provést při použití nového datového typu **Počítané pole**. Výpočty se provádí v dalších objektech – dotazech, formulářích a sestavách. Také je možné implementovat ověřování zadávaných údajů, aby tyto byly uživateli zadávány správně. Například při zadání prodejní ceny je možné nastavit ověřování jen na kladná čísla.

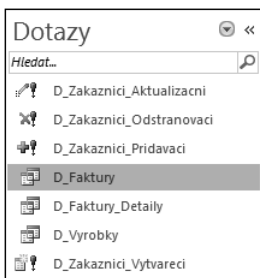


Obrázek 1.2: Seznam tabulek v databázi zobrazený v navigačním podokně

Dotazy

Dotazy se používají k zobrazení údajů z jedné nebo více tabulek, případně jiných dotazů. Umožňují vybrat k zobrazení jen některá pole, uspořádat je v jiném pořadí, než jsou v tabulce, a přidat různá kritéria na zobrazení údajů. Například můžete přidat kritérium na zobrazení jen žen nebo jen mužů v tabulce pracovníků.

Kromě kritérií se v dotazech realizují i výpočty. Pokud potřebujete v tabulce objednávky vypočítat celkovou cenu za počet kusů zboží a jednotkové ceny, provedete to v dotazu. Také se dají realizovat i souhrnné dotazy, které spočítají například sumu celé objednávky. Kromě výběrových dotazů jsou v Accessu i akční dotazy, které přidávají údaje do tabulky, aktualizují je a odstraňují. Akčním dotazem je možné vytvořit i tabulku. Existuje i křížový, sjednocovací, předávající a definiční dotaz. Většinu dotazů můžete vytvořit pomocí průvodců a upravit v návrhovém zobrazení. Poslední tři jmenované se vytvářejí přímo psaním SQL příkazů.



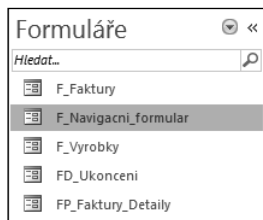
Obrázek 1.3: Seznam výběrových a akčních dotazů v databázi zobrazený v navigačním podokně

Formuláře

Formuláře jsou určeny na práci s údaji a tvoří uživatelské rozhraní databázové aplikace. S formuláři přicházejí do styku pracovníci s informacemi. Jsou založené na údajích z tabulek a dotazů. Mohou též obsahovat výpočty. Existují i formuláře bez údajů, ty se používají hlavně k vytvoření navigačního systému v databázi, aby koncový uživatel mohl s aplikací efektivně pracovat.

Na výběr je několik způsobů vytvoření formuláře a také několik typů různých formulářů, jako je například jednoduchý formulář, rozdělený, datový list apod. U navigačních formulářů

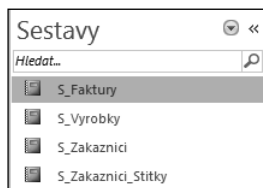
jsou dostupné například vodorovné karty, svislé karty, víceúrovňové apod. Na kartách jsou tlačítka, která slouží k otevírání formulářů s údaji.



Obrázek 1.4: Seznam formulářů v databázi zobrazený v navigačním podokně

Sestavy

Sestavy se nejčastěji používají na tiskové výstupy údajů z tabulek nebo dotazů. Využijete je samozřejmě i k prohlížení údajů na obrazovce. Do sestav není možné přidávat údaje jako do formulářů. Vytvoříte je průvodcem nebo v návrhovém zobrazení. Stejně jako formuláře, i sestavy mohou obsahovat vložené sestavy, a zobrazovat tak údaje z propojených tabulek. Při formátování je vhodné dodržet rozmístění a velikost jednotlivých polí tak, aby se vešla na běžnou velikost kancelářského papíru.



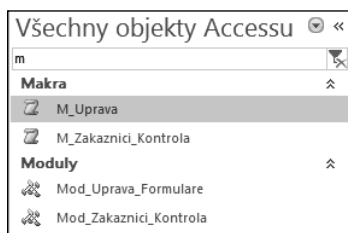
Obrázek 1.5: Seznam sestav v databázi zobrazený v navigačním podokně

Makra

K automatizování některých úloh, ale i nastavování parametrů použijete makra. Ta umožňují například na základě podmínky nastavit hodnotu některého ovládacího prvku, případně ho skrýt/zobrazit nebo pracovat s objekty. Pomocí makra můžete také otevírat a spouštět jiné databázové objekty. Podobně se dají automatizovat i úlohy se záznamy. V podokně úloh **Katalog akcí** jsou dostupné akce maker přehledně rozdělené do skupin.

Moduly

Moduly se používají na vytváření programového kódu jazyka Visual Basic pro aplikace (VBA), kterým taktéž automatizujete úlohy. V tomto případě je potřeba už ovládat programovací techniky. Využít můžete i možnost **Konvertovat makra do jazyka Visual Basic**. Tématu programování pomocí VBA v Accessu se věnují samostatné knihy (například *Access VBA Výukový průvodce* od R. Shepherda).



Obrázek 1.6: Seznam maker a modulů v databázi zobrazený po filtrování v navigačním podokně

Možnosti spuštění Microsoft Access

Spuštění z nabídky Start nebo Úvodní obrazovky Windows 8

Ve Windows 7 spustíte Access klepnutím na tlačítko **Start**, po kterém můžete začít psát název programu a on se zobrazí v horní části nabídky. Následně ho spustíte stisknutím klávesy **Enter** nebo klepnutím myši. Využít můžete i zobrazení programové skupiny Microsoft Office 2013, ve které následně klepnutím spustíte Access. Podobně ho spustíte i ve Windows 8, kde na Úvodní obrazovce začnete psát Access a po jeho vyhledání ho spustíte klávesou **Enter** či klepnutím myši.

Spuštění pomocí odkazů

Pokud používáte Access často, můžete si vytvořit odkaz, který může být na pracovní ploše nebo na hlavním panelu Windows. Ve Windows 7 klepněte v nabídce **Start** pravým tlačítkem myši na Access 2013 a z místní nabídky vyberte **Připnout na hlavní panel**. Odkaz na ploše vytvoříte též pravým tlačítkem myši, z místní nabídky vyberte **Odeslat → Plocha (vytvořit zástupce)**. Podobným způsobem postupujete i ve Windows 8 na Úvodní obrazovce.

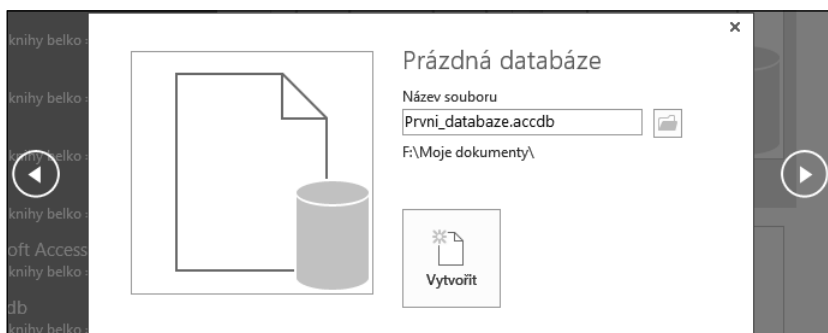
Prostředí Accessu 2013

Po spuštění Accessu se zobrazí úvodní obrazovka, která obsahuje několik šablon databází. Vlevo je seznam naposledy použitých databází, ze kterého můžete přímo některou používanou databázi otevřít.

Vytvoření prázdné databáze

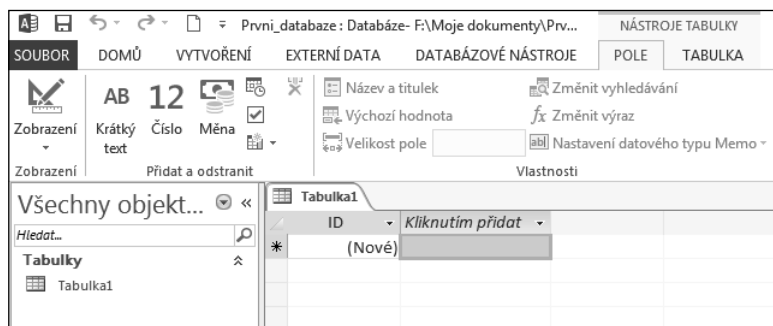
Úplně novou vlastní databázi vytvoříte výběrem šablony **Prázdná databáze z úvodní obrazovky** po spuštění Accessu.

1. Spusťte Access a z Úvodní obrazovky vyberte klepnutím šablonu **Prázdná databáze**.
2. V zobrazeném okně zadejte název souboru databáze `Prvni_databaze` a vyberte umístění, například *Moje dokumenty*.



Obrázek 1.7: Dialogové okno vytvoření nové prázdné databáze

3. Klepněte na tlačítko **Vytvořit**, které zobrazí prostředí Accessu s otevřenou novou prázdnou tabulkou s názvem *Tabulka1*. Tuto byste následně upravili a vytvořili i další objekty podle návrhu databáze. Postup bude popsán v další části.
4. Ukončete Access zavřením jeho okna.

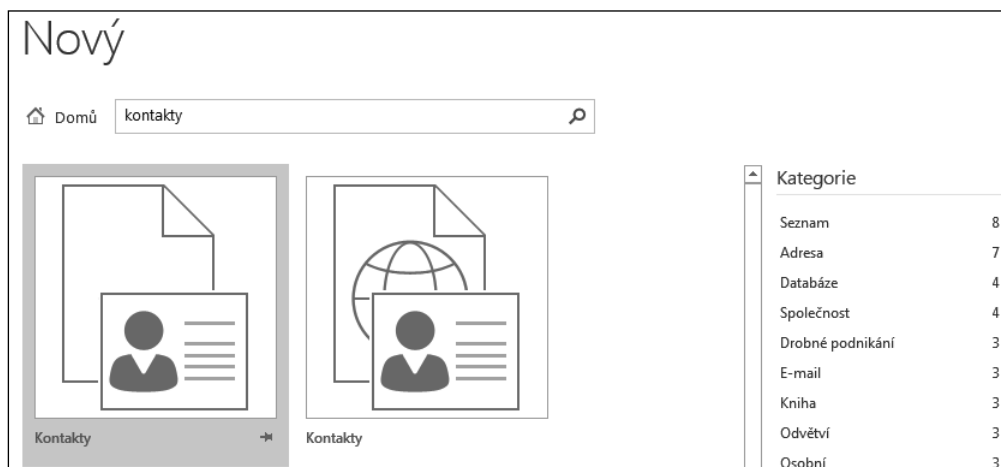


Obrázek 1.8: Zobrazení prostředí Accessu po vytvoření nové prázdné databáze

Vytvoření databáze ze šablony

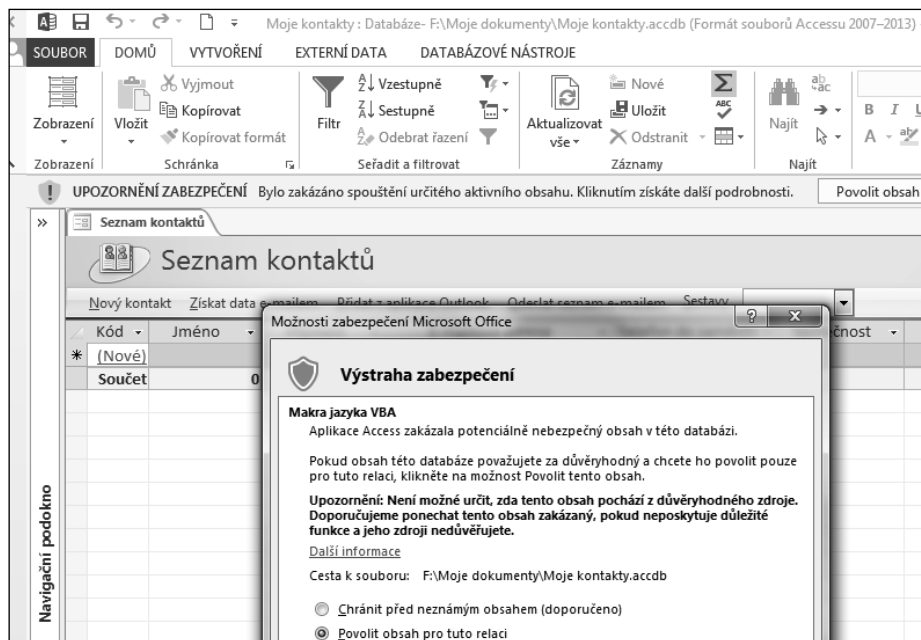
V horní části úvodní obrazovky je možné vyhledávat šablony i z webu, pokud zadáte vhodné klíčové slovo. Můžete využít i navrhovaná klíčová slova, která jsou dostupná pod řádkem vyhledávání. Výsledky nejsou jen pro Access, ale i pro jiné aplikace Office. Ty jsou uvedeny v dolní části zobrazení Backstage. Z výsledků vyhledávání se vrátíte na úvodní obrazovku klepnutím na šipku vlevo nahoře.

1. Spusťte Access a na Úvodní obrazovce vyberte šablonu *Kontakty*. Pozor, dostupné jsou dvě šablony kontaktů, jedna jako webová aplikace a druhá jako lokální. Vyberte lokální, jejíž ikona má v pozadí dokument bez schématu zeměkoule.
2. V zobrazeném okně zadejte název *Moje kontakty* a vyberte umístění, kam se databáze uloží (například *Moje dokumenty*). Klepněte na **Vytvořit**.



Obrázek 1.9: Šablony lokální a webové aplikace na úvodní obrazovce po vyhledávání na základě klíčového slova

- Po zobrazení krátkého průběhu stahování se zobrazí formulář kontaktů a v horní části žlutý panel hlášení **UPOZORNĚNÍ ZABEZPEČENÍ Bylo zakázáno spouštění určitého aktivního obsahu**. Kliknutím získáte další podrobnosti. Klepněte na něj. V zobrazení Backstage se zobrazí informace **Upozornění zabezpečení**, že makra VBA byla zakázána.
- Klepněte na tlačítko **Povolit obsah** → **Upřesnit možnosti**.



Obrázek 1.10: Po prvním spuštění databáze ze šablony se zobrazuje Upozornění zabezpečení a přes kartu Soubor můžete zobrazit i Výstrahu zabezpečení

5. V zobrazeném okně zvolte **Povolit obsah pro tuto relaci** a klepněte na **OK**. Tímto povolíte aktivní obsah jen pro toto jedno spuštění databáze. Po zavření a opětovném spuštění se hlášení zobrazí znovu.

6. Ponechteje databázi otevřenou.



Tip: Pokud klepnete na panelu hlášení na tlačítko **Povolit obsah**, toto způsobí, že databáze se stane důvěryhodnou a při opětovném spuštění se už nebude hlášení zobrazovat. Toto byste měli provést jen u databází, kterým důvěřujete, že pocházejí z bezpečného zdroje. V případě, že chcete zrušit důvěryhodnost databáze, je potřeba v nastavení centra důvěryhodnosti změnit všechny důvěryhodné dokumenty na nedůvěryhodné. Pozor, toto změni všechny databáze, jelikož není možné ze seznamu vybrat jen některé.

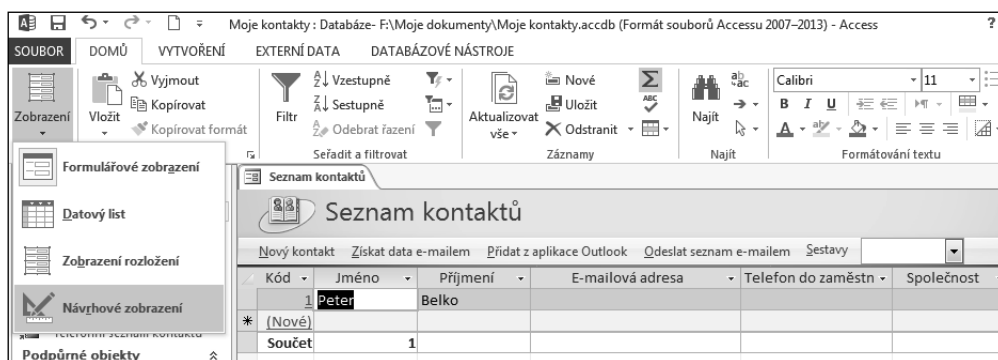
Popis prostředí aplikace a orientace v něm

Access je součástí balíku Microsoft Office, a proto je i jeho prostředí podobné jiným aplikacím z této sady. Uživatel se v něm rychle zorientuje a může vytvářet databáze.

Pás karet a panel nástrojů Rychlý přístup

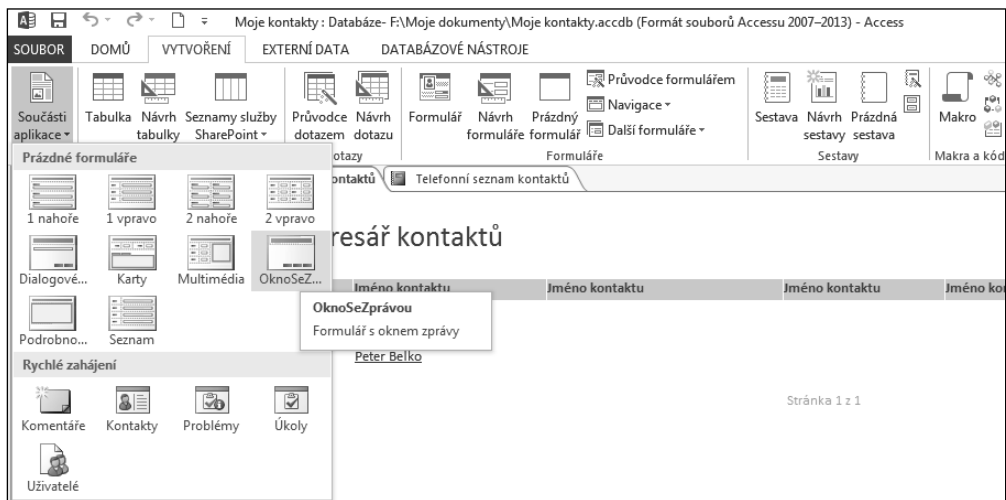
Jako všechny aplikace Office od verze 2007 má i Access pás karet s nástroji, který obsahuje všechny důležité příkazy. Některé položky jsou dostupné až při práci s konkrétním objektem na tzv. kontextových kartách. Hlavní karty jsou **Domů**, **Vytvoření**, **Externí data**, **Data-bázové nástroje**.

Na kartě **Domů** je důležitý příkaz na změnu zobrazení objektu, který přepíná z normálního zobrazení na návrhové a další podle typu objektu, s kterým právě pracujete. Dále jsou k dispozici nástroje schránky, filtrování, práce s údaji, vyhledávání a formátování.



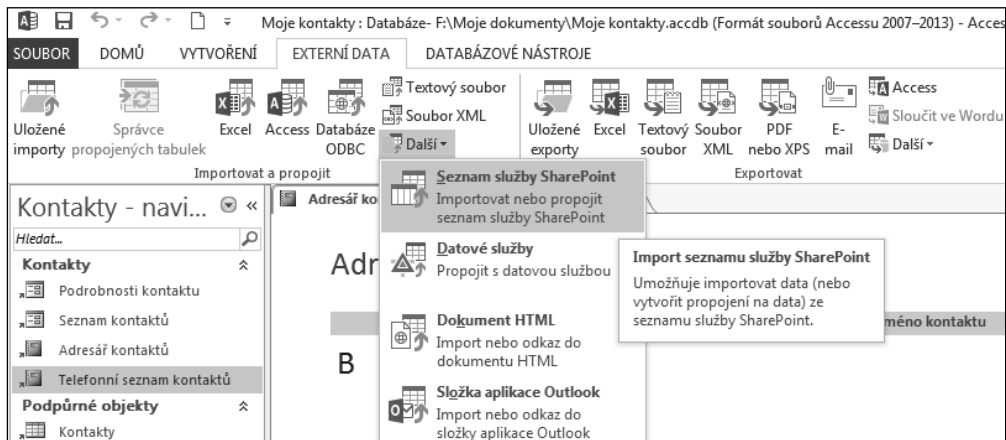
Obrázek 1.11: Karta **Domů** s možnostmi změny zobrazení formuláře v otevřené databázi *Moje kontakty*

Karta **Vytvoření** je důležitá pro každého, kdo vytváří a upravuje databázi. K dispozici jsou příkazy na vytvoření tabulek, dotazů, formulářů, sestav, maker a modulů.



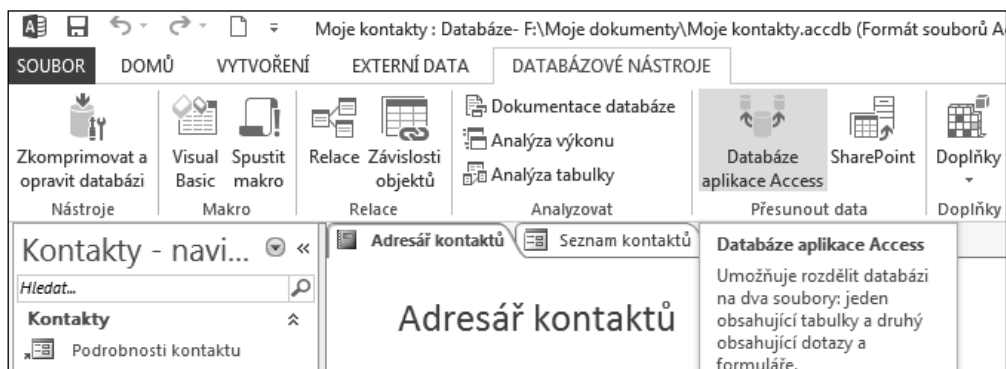
Obrázek 1.12: Karta Vytvoření s příkazy na vytvoření databázových objektů a součástí aplikace

Externí data se používají hlavně k importu a exportu údajů v různých podporovaných formátech. Využijí to hlavně pracovníci s informacemi, kteří potřebují údaje dále zpracovávat.



Obrázek 1.13: Karta Externí data obsahuje příkazy na import, propojení a export údajů v různých formátech

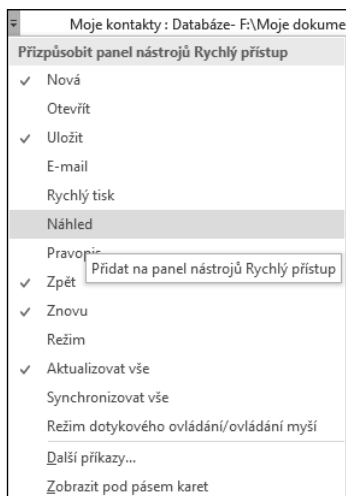
Databázové nástroje obsahují příkazy na kompresi a opravu databáze, rozdělení databáze, analýzu a práci s relacemi.



Obrázek 1.14: Karta Databázové nástroje nabízí příkazy na vytvoření relací, rozdělení databáze a další

K rychlému přístupu k některým příkazům můžete využít panelu nástrojů **Rychlý přístup**. Na tento panel přidáváte příkazy přes místní nabídku, která se zobrazí klepnutím pravého tlačítka myši, volbou **Přidat na panel nástrojů Rychlý přístup**. Další možnosti, jak na panel přidat nástroje, jsou možnosti aplikace.

1. V otevřené databázi *Moje kontakty* se zobrazeným formulářem *Seznam kontaktů* klepněte na pásu karet na kartu **Domů** a vyhledejte příkaz **Aktualizovat vše**.
2. Na příkaz klepněte pravým tlačítkem myši a z místní nabídky zvolte **Přidat na panel nástrojů Rychlý přístup**. Tlačítko se přidá na panel **Rychlý přístup** do horní části vlevo.
3. Klepněte na rozevírací šipku na panelu **Rychlý přístup** a ze seznamu přidejte příkaz **Náhled**.
4. Klepnutím pravým tlačítkem myši na přidávané tlačítko **Aktualizovat vše** a volbou **Odebrat** z panelu **Rychlý přístup** tlačítko odeberte.
5. Ponechteje databázi otevřenou.

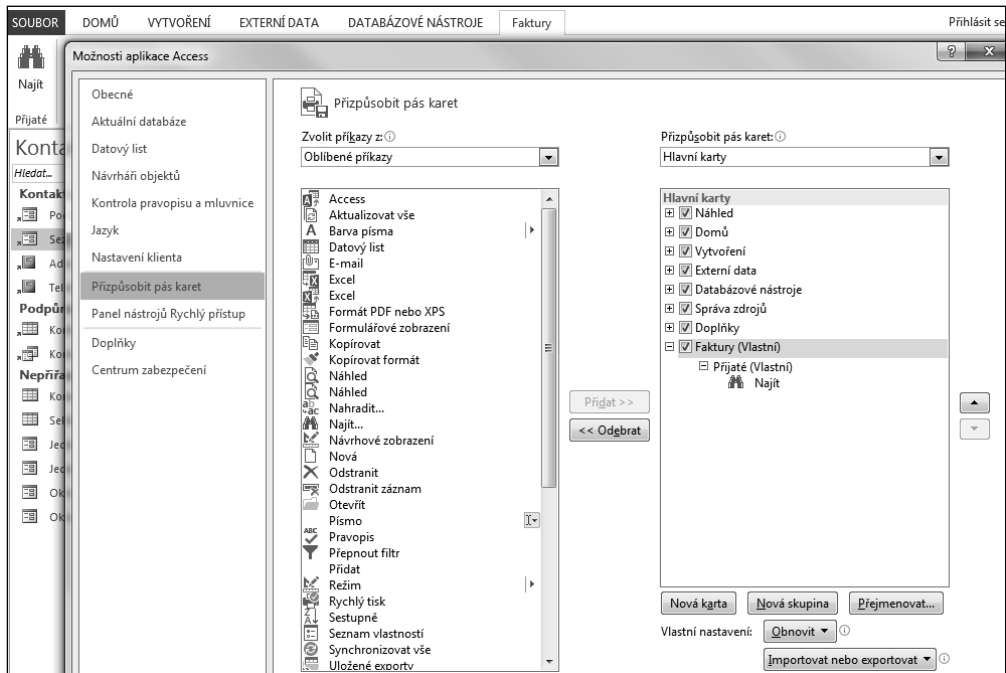


Obrázek 1.15: Přidání příkazů na panel nástrojů Rychlý přístup ze seznamu

Úprava pásu karet

Podobně jako panel nástrojů **Rychlý přístup** můžete upravovat i pás karet a vytvářet karty vlastní.

1. V otevřené databázi **Moje kontakty** klepněte na **Soubor** → **Možnosti** → **Prizpůsobit pás karet**.
2. V levém seznamu jsou dostupné příkazy, které vybíráte podle oblasti v horní části v rozvíracím seznamu. V pravém seznamu jsou příkazy na jednotlivých kartách. Karty můžete skrýt a měnit jejich pořadí.
3. Klepněte na poslední kartu v seznamu a potom v dolní části na tlačítko **Nová karta**, které přidá novou kartu a skupinu.
4. Klepněte na novou kartu a poté na tlačítko **Přejmenovat**. V zobrazeném okně zadejte název **Faktury**.
5. Klepněte na novou skupinu a poté na tlačítko **Přejmenovat**. V zobrazeném okně zadejte název **Přijaté** a klepněte na **OK**.
6. Ze seznamu vlevo z oblíbených příkazů přidejte klepnutím **Najít** a změny potvrďte tlačítkem **OK**. Zobrazí se okno, že změny se zobrazí až po opětovném spuštění databáze, kterou zavřete a znovu spustíte. Pokud jste nepovolili aktivní obsah v žlutém panelu hlášení, povolte jej. Zobrazí se nový pás karet. Stejně přidáte na vlastní pás karet i další příkazy.



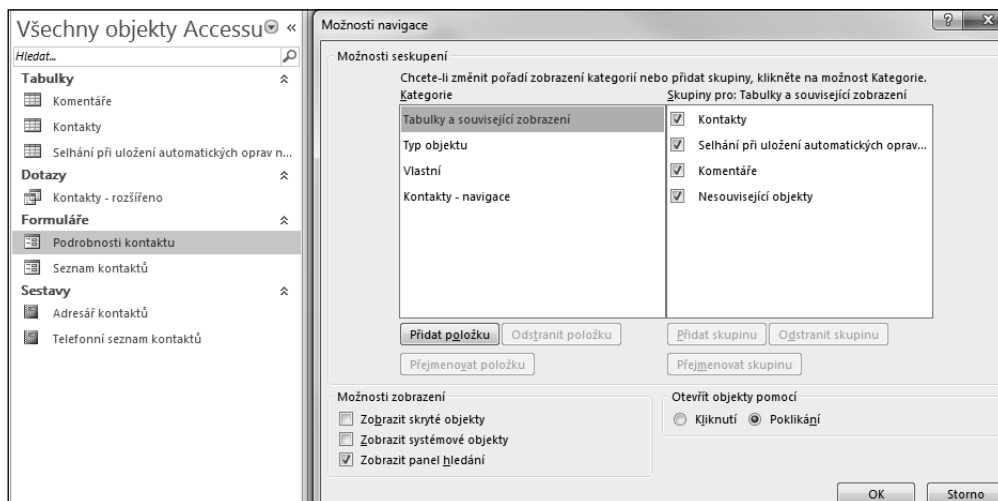
Obrázek 1.16: Úprava pásu karet v okně možností s vytvořenou novou kartou a přidaným příkazem

7. Vlastní karty odstraníte přes **Soubor** → **Možnosti** → **Přizpůsobit pás karet**. Klepněte v seznamu na vlastní kartu **Faktury** pravým tlačítkem myši a zvolte **Odebrat**. Karta se odebere, okno možností zavřete klepnutím na tlačítko **OK**.
8. Ponechte databázi otevřenou.

Navigační podokno

Navigační podokno vlevo je velmi důležitá část uživatelského prostředí Accessu. Obsahuje jednotlivé objekty, s kterými pracujete při vytváření databáze. Každý vytvořený databázový objekt se v podokně zobrazí, můžete ho vymazat, přejmenovat nebo kopírovat. Navigační podokno se může zobrazovat různými způsoby a poskytuje i vyhledávání objektů. Pokud ho přímo v otevřené databázi *Moje kontakty* nevidíte, klepněte na dvojitou šipku vlevo nahoře, která ho zobrazí.

1. Prohlédněte si, jak jsou objekty v databázi seskupeny.
2. Klepněte v horní části na rozevírací šipku a vyberte **Typ objektu**. Objekty se seskupí podle typu – tabulky se seskupí, formuláře se seskupí atd.
3. Vyzkoušejte si i další zobrazení. Často používané zobrazení je podle objektu, ale používejte to, které vám nejlépe vyhovuje.
4. Pokud máte hodně objektů, můžete použít i vyhledávání postupným zadáváním názvu.
5. Další nastavení navigačního podokna změníte přes místní nabídku pravého tlačítka myši přímo v podokně. Můžete měnit zobrazení seznamu, podrobností nebo ikon stejně jako způsob řazení.
6. Klepněte v místní nabídce na volbu **Možnosti navigace** a v zobrazeném okně si prohlédněte možnosti seskupení. To můžete změnit nebo vytvořit další. Kromě toho jsou k dispozici položky na zobrazení skrytých a systémových objektů. Ty potřebujete v určitých specifických případech, například při vytváření vlastního pásu karet pro aplikaci.
7. Tlačítkem **Zrušit** zavřete možnosti navigace a ponechte je databázi otevřenou.



Obrázek 1.17: Podokno navigace a nastavení jeho možností zobrazení

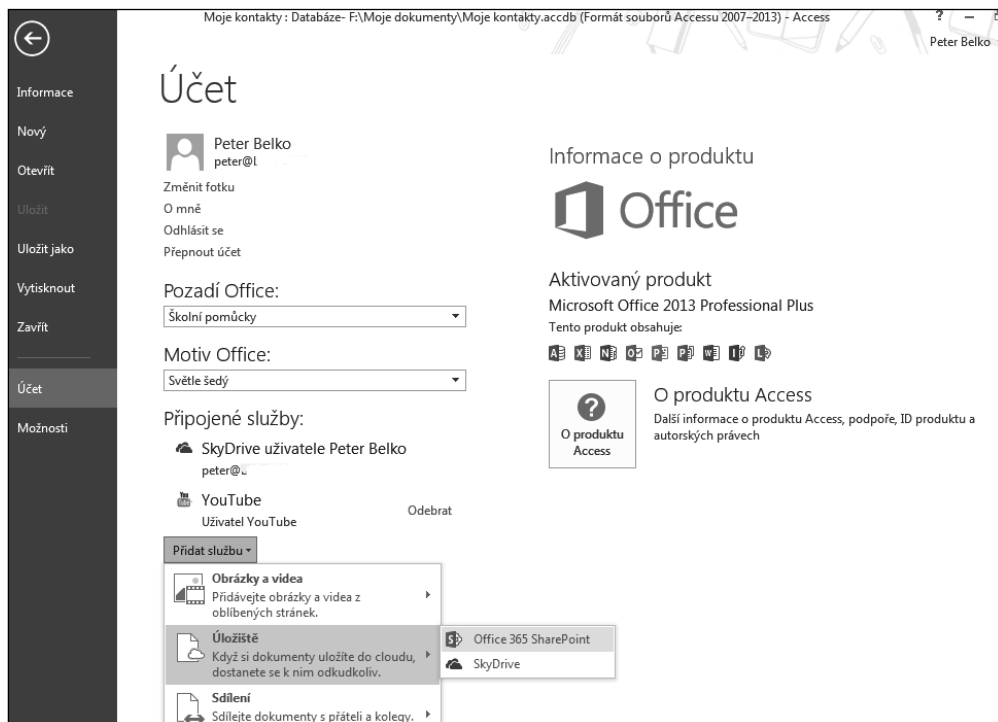


Tip: V případě, že se navigační podokno nezobrazí vůbec, respektive nechcete, aby se zobrazovalo, můžete to změnit přes **Soubor** → **Možnosti** → **Aktuální databáze** → **Navigace** → **Zobrazit navigační podokno**.

Soubor a zobrazení Backstage

Ve verzi Office 2010 byla představena karta **Soubor**, která zobrazuje **Backstage**. Toto zobrazení obsahuje důležité příkazy na práci s aplikací jako celkem. Můžete vidět informace o databázi, otevírat existující databáze příkazem **Otevřít**, vytvářet nové, ukládat databáze v různých formátech, ukládat jednotlivé označené objekty, tisknout, zavřít databázi nebo přistupovat k možnostem aplikace. Novinkou je Účet Microsoft, pomocí něhož máte přímo z aplikací Office jednoduchý přístup ke svému on-line úložnému prostoru na SkyDrive nebo na podnikovém SharePointu, případně na SharePointu online přes Office 365.

1. V databázi **Moje kontakty** klepněte na **Soubor** → **Otevřít** a prozkoumejte dostupné možnosti. Jsou k dispozici naposledy používané databáze, možnosti otevřít ze SkyDrive nebo z lokálního počítače. Také můžete přidat nové místo.
2. Část **Vytisknout** obsahuje **Rychlý tisk**, **Tisk** a **Náhled**.
3. Prozkoumejte i ostatní části zobrazení **Backstage**.
4. Klepnutím na šipku vlevo se vrátíte do prostředí databáze. Ponechejte databázi otevřenou.

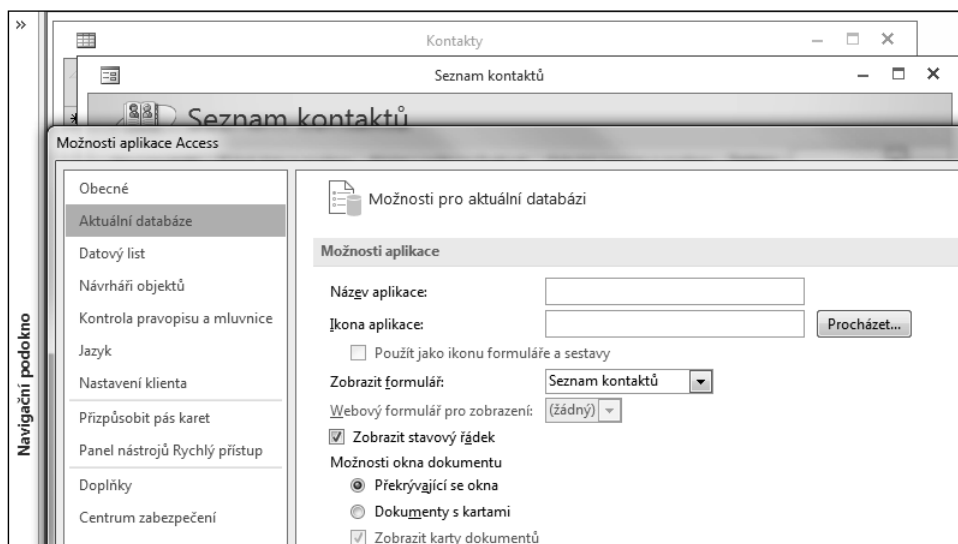


Obrázek 1.18: Účet Microsoft k přímému přístupu k online úložnému prostoru SkyDrive nebo SharePoint, vpravo nahoře zobrazuje i jednoduché grafické pozadí

Karty nebo okna objektů

Uživatelé starších verzí Accessu znají zobrazování jednotlivých databázových objektů v samostatných oknech. Verze 2013/2010 zobrazuje objekty na kartách v horní části obrazovky. V případě, že budete potřebovat ve vlastní databázi zobrazovat objekty v oknech, postupujte v databázi *Moje kontakty* následovně:

1. Klepněte na **Soubor** → **Možnosti** → **Aktuální databáze** a vyhledejte část **Možnosti okna dokumentu**.
2. Klepněte na položku **Překrývající se okna** a potvrďte tlačítkem **OK**.
3. Zobrazí se informace, že se zadaná možnost zobrazí až po novém spuštění aplikace, kterou zavřete a otevřete.
4. Jednotlivé objekty se budou zobrazovat v samostatných oknech, ale jen v této konkrétní databázi.
5. Původní nastavení vrátíte stejným postupem, jen vyberete položku **Dokumenty s kartami**. Ponechte Access spuštěný.



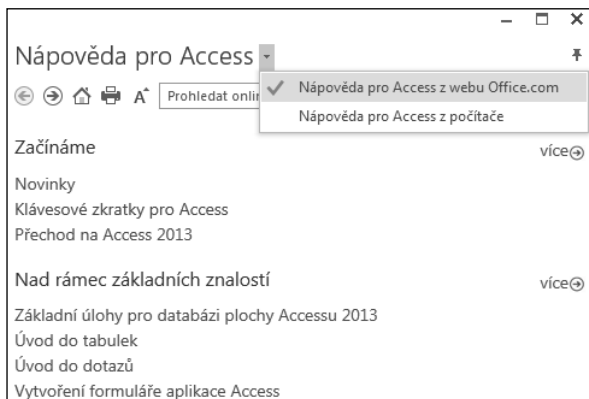
Obrázek 1.19: Zobrazení objektů v samostatných oknech místo karet nastavíte v okně *Možnosti aplikace Access*

Získání pomoci

Access je robustní klientský databázový systém, který obsahuje množství funkcí a nastavení. Tyto jsou zdokumentovány v pomocníku aplikace a na webových stránkách podpory Office. Pokud potřebujete získat pomoc s určitými funkcemi a nastaveními, které často nepoužíváte, vyvoláte pomocníka klávesou **F1** nebo klepnutím na tlačítko s otázníkem v pravé horní části aplikace.

Zobrazí se okno pomocníka, které obsahuje témata a možnosti vyhledávání. Standardně je zapnutý pomocník online. V horní části pomocníka u jeho názvu jej můžete změnit na offline,

který použije jen lokální informace dodané s instalací Office. V okně pomocníka se pohybuje podobně jako na webové stránce, šipkami vlevo a vpravo a tlačítkem domů.



Obrázek 1.20: Okno Nápověda pro Access umožňuje vyhledávat informace lokálně nebo z online zdrojů

Základní návrh jednoduché databáze

Při práci s údaji jste se pravděpodobně už setkali s určitou formou databáze. Pravděpodobně to bylo uspořádání údajů do řádků a sloupců v Excelu nebo jiné tabulkové aplikaci. K údajům jste přidali pravděpodobně i nějaké výpočty. Takovéto tabulky jsou však použitelné jen na základní zpracování velkého množství údajů. V případě, že potřebujete údaje začít spojovat, porovnávat, filtrovat a jinak zpracovávat, zjistíte, že tabulková aplikace na to nebude nejvhodnější.

Komplexnější zpracování údajů je potřeba naplánovat a dobře připravit. Proto nevytvářejte databázovou aplikaci bez důkladné přípravy, abyste ji nemuseli potom několikrát předělávat a upravovat. Dobrá vstupní analýza požadavků a propracovaný návrh databáze jsou základem úspěchu a funkční databázové aplikace. Následné úpravy a přizpůsobení jsou jednodušší než v tabulkách Excelu, které často vznikají podle aktuálních potřeb uživatelů přidáváním různých vzorců bez důkladné celkové analýzy a dokumentace.

Metodologii na dobrý návrh aplikací navrhli experti už v 60. letech 20. století. Uznávání konzultanti J. Martin, E. Yourdon a L. Constantine doporučili, aby se 60 % a více času věnovalo návrhu aplikace ještě předtím, než byl vytvořen jakýkoliv kód. Bylo to kvůli tomu, že úprava kódu aplikace byla cenově a časově náročná. V současnosti je při použití Accessu vytvoření jednoduché databázové aplikace pro zkušeného uživatele otázkou několika hodin. Následné úpravy a doplňování požadavků uživatelů se dají realizovat během provozu s minimálními odstávkami.

Složité aplikace postavené např. na SQL Serveru je potřeba po důkladné analýze požadavků dobře navrhnout a následně postupně vytvořit. V takovýchto robustních řešeních to už není jako u Accessu úkolem pro jednoho uživatele, ale pro celé týmy analytiků, konzultantů, programátorů a testerů, kteří mají na starosti jednotlivé úlohy a pracují na částech aplikace.

Podle různých zdrojů se v knize J. L. Viescase *Mistrovství v Microsoft Access* rozděluje návrh databáze do několika kroků: identifikace úloh, náčrt toku úloh, identifikace datových prvků, uspořádání údajů, navržení prototypu a uživatelského prostředí, vytvoření aplikace, testování, úprava a zdokonalování.

Vzhledem k rozsahu této knihy se nemůžeme věnovat detailnímu popisu teorií relačních databází a návrhu aplikací, jak je tomu v knihách o teorii databází. Proto se v následujícím textu dozvíte hlavně o základních krocích přípravy návrhu jednoduché databáze na zpracování faktur. Tyto kroky využijete i na jiné typy databází Accessu, například zpracování objednávek, skladových zásob, rezervací apod. Většina databázových aplikací využívá podobnou strukturu, jen se mění tabulky například dodavatelů za odběratele, faktury za objednávky, výrobky za služby apod. U komplexních databází evidujících objednávky, faktury a provize můžete mít samozřejmě všechny výše zmíněné tabulky. Všechno závisí na typu databáze a jejím budoucím použití.

Určení typu databáze a jejího použití

Potřeba vytvoření databázové aplikace vychází většinou z určité nespokojenosti uživatelů s dosavadním řešením zpracování dat. To může být způsobeno například tím, že údaje v excelové tabulce se komplikovaněji zpracovávají po síti ve sdílené složce, tabulka už nevyhovuje potřebám uživatelů a zpracování dat je neefektivní. Tehdy nastává fáze rozhodnutí na změnu dosavadního systému zpracování dat. Představte si, že vás malá firma oslovila s požadavkem na vytvoření jednoduché databáze na zpracování faktur.

V prvním kroku je potřeba zjistit, jak používají dosavadní systém a jaké mají procesy zpracování faktur. Kromě zjišťování informací od uživatelů jsou dobrou pomůckou i jejich stávající formuláře, tabulky a další dokumenty včetně papírových. Ty vám poskytnou pohled, jak přibližně bude potřeba realizovat výstupy z databáze. Z těchto informací vznikne představa, co by měla databáze evidovat a zpracovávat. Tuto představu je potřeba konzultovat s uživateli ve firmě, zda jste všechno o jejich zpracování faktur pochopili správně.

Následně je potřeba všechno zaznamenat a začít připravovat struktury databáze a tabulek. Myslete však na to, že zadavatel databáze dopředu neumí odhadnout možnosti růstu svého podnikání. Proto pokud vám řekne, že zákazník nikdy nebude mít více než 5 faktur za rok, připravte návrh databáze pro N faktur za rok a ne jen pět, aby byla připravena na expanzi fakturace.

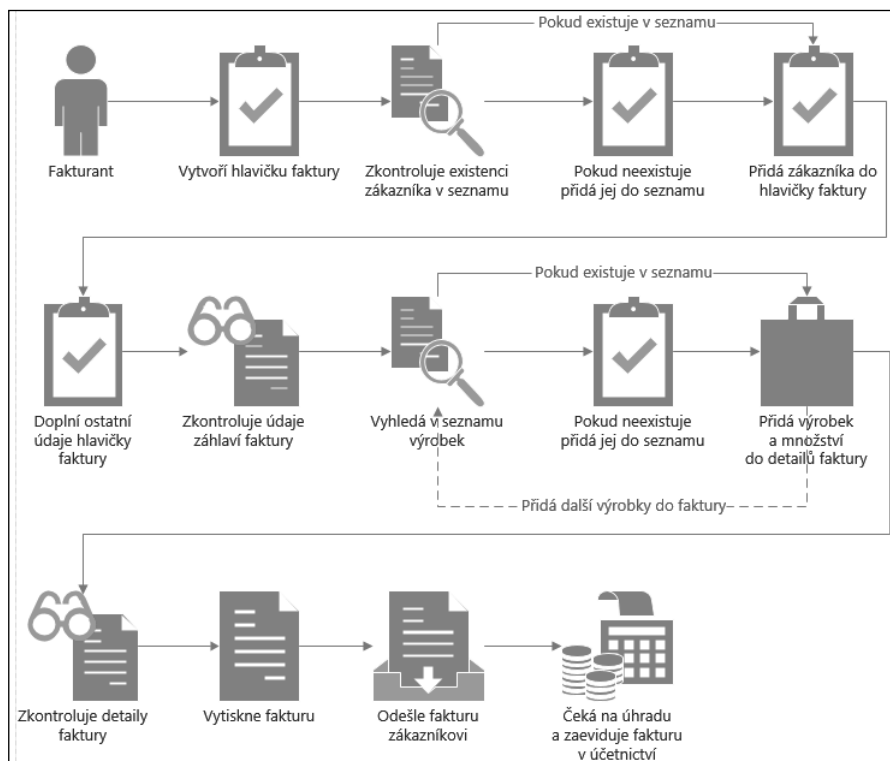
Úvodní sepsané informace by se měly týkat analýzy úloh, které uživatelé realizují při zpracování faktur. Mohly by tam být následující činnosti:

- Zadání údajů o zákazníkovi.
- Zadání údajů o kategoriích výrobků.
- Zadání údajů o výrobcích.
- Spojení výrobků s kategoriemi.
- Zadání údajů do hlavičky faktury.
- Spojení údajů hlavičky faktury s údaji zákazníka.
- Zadání údajů do těla faktury.
- Spojení údajů těla faktury s údaji o výrobcích.

Následně je potřeba sepsat, jaké úlohy se realizují, pokud se fakturuje novému zákazníkovi, jaké úlohy probíhají při opakované fakturaci existujícímu zákazníkovi, co se dělá, pokud v databázi není výrobek, který se má fakturovat. Mohly by vypadat následovně:

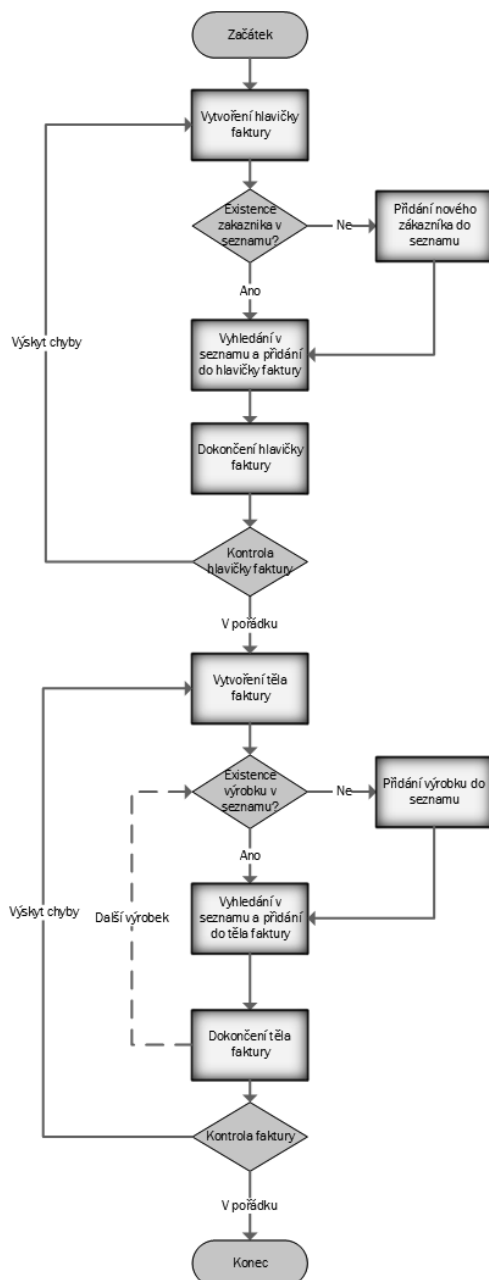
- Vytvoření hlavičky faktury.
- Kontrola, zda se zákazník nachází v databázi.
- Pokud se nenachází, vytvoření nového záznamu zákazníka.
- Pokud se nachází, přidání zákazníka do hlavičky faktury.
- Kontrola hlavičky faktury.
- Vyhledání výrobku v databázi.
- Pokud se nenachází, vytvoření nového záznamu výrobku a přidání do kategorie.
- Pokud se nachází, přidání výrobku do detailů faktury.
- Doplnění množství fakturovaného výrobku.
- Kontrola detailů faktury.
- Tisk faktury.
- Odeslání vytištěné faktury a čekání na úhradu.

Tyto procesy můžete realizovat i graficky v aplikaci Visio 2013 pro lepší přehlednost a následnou dokumentaci, pokud by bylo zapotřebí v návrhu databáze někdy provádět změny.



Obrázek 1.21: Jednoduchý náčrt procesu zpracování faktury v malé firmě

Vytvořit můžete i vývojový diagram, který bude zohledňovat jednotlivé úlohy od začátku až do konce i s rozhodováním v situacích, kdy se ověřuje, zda se zákazník a výrobek nacházejí v databázi. U větších řešení mohou být vývojové diagramy rozděleny na několik samostatných částí.



Obrázek 1.22: Základní vývojový diagram zpracování faktury v malé firmě

V souvislosti s analýzou doporučujeme u robustních databázových řešení nastudovat si příslušnou literaturu, kde jsou detailně popsány jednotlivé kroky a způsoby analýzy a její zpracování a dokumentace. V dalším textu popíšeme, jak z takového popisu úloh navrhnout tabulky.

Návrh tabulek

Návrh tabulek vychází ze všech zjištění od zadavatele databáze a jejích uživatelů. Vaším hlavním úkolem bude navrhnout, jaké tabulky jsou zapotřebí, co mají evidovat a jak mají být vzájemně propojeny. U této úlohy je potřeba držet se teorie návrhu relačních databází a tabulky vytvořit normalizované. Znamená to, že je potřeba navrhnout je tak, aby neobsahovaly duplicitní údaje.

Tabulky jsou tvořeny poli (sloupci) a v řádcích se evidují záznamy. Záznam představují údaje například o konkrétní firmě – název, pracovník, adresa, kontakty apod., jeden záznam tabulky výrobků obsahuje název výrobku, jeho jednotkovou cenu, kód, poznámku apod. Nevhodná tabulka by obsahovala dohromady údaje o zákaznících, hlavičky faktury i o výrobcích, které se fakturují. Vznikly by tak zbytečné údaje o firmě u každého fakturovaného výrobku. Pokud by se fakturovalo například 10 výrobků, všechny údaje o firmě by se v databázi objevily 10krát. Další problémy by vznikly při aktualizaci údajů, kdy je potřeba v nevhodně navržené tabulce aktualizovat tu samou informaci o firmě několikrát.

	A	B	C	D	E	F	G	H	I	J
1	Název	Kontaktní osoba	Adresa	Tel_kont akt	Mobil	email	název_vyrobku	Jednotka va_cena	Mnozstvi	cena
2	Belko Ltd.	Peter Belko	Hlavní 17, Trnava	12345678	905123	peter@ltd.cz	Chicony	5,00 €	2	10,00 €
3	Belko Ltd.	Peter Belko	Hlavní 17, Trnava	12345678	905123	peter@ltd.cz	Genius	7,00 €	1	7,00 €
4	Belko Ltd.	Peter Belko	Hlavní 17, Trnava	12345678	905123	peter@ltd.cz	Philips	100,00 €	2	200,00 €
5	MMP s.r.o.	Mazánová	Boční 9, Brno	258741	321456	mmp@ma.cz	Chicony	5,00 €	2	10,00 €
6	MMP s.r.o.	Mazánová	Boční 9, Brno	258741	321456	mmp@ma.cz	Genius	7,00 €	2	14,00 €
7	MMP s.r.o.	Mazánová	Boční 9, Brno	258741	321456	mmp@ma.cz	Philips	100,00 €	3	300,00 €

Obrázek 1.23: Nevhodně navržená tabulka, která obsahuje nadbytečné údaje o firmách a nejednoznačné pole adresy

Toto je nežádoucí stav, a proto je úkolem analytiků a vývojářů rozdělit tabulku na více tabulek tak, aby dodržovaly podmínky normalizace. Z knihy od J. L. Viescase *Mistrovství v Microsoft Access* vybíráme jen ty nejdůležitější informace o normalizaci.

První pravidlo normalizace: Jedinečnost polí

Každé pole v tabulce by mělo představovat jedinečný typ informace. Zjednodušeně to znamená, že je potřeba z tabulky odstranit údaje, které by generovaly sloučené údaje v jednom poli (ulice a město) a opakuující se údaje. Proto tabulka, která obsahuje firemní údaje, údaje z hlavičky faktury a údaje o výrobcích, musí být rozdělena na samostatnou tabulku zákazníků, samostatnou tabulku údajů hlavičky faktury a samostatnou tabulku výrobků. Jejich vzájemný vztah je vytvořen na základě identifikačních čísel, která reprezentují údaje o zákaznících, údaje o hlavičce faktury a údaje o výrobcích. Tabulky budou propojené těmito čísly a ne všemi údaji. Tak se ušetří místo v úložném prostoru databáze. Z hlediska používání jsou takovéto samostatné tabulky jednodušší. Při hledání některého údaje o výrobku stačí prohledávat jen tabulku výrobků a není nutné zatěžovat se údaji o klientech.

	A	B	C	D	E	F	G
1	Tabulka zakazníci						
2	Nazev	Kontaktní osoba	Adresa	Mesto	Tel_kont akt	Mobil	email
3	Belko Ltd.	Peter Belko	Hlavní 17	Trnava	12345678	905123	peter@ltd.cz
4	MMP s.r.o.	Mazánová	Boční 9	Brno	258741	321456	mmp@ma.cz
5							
6	Tabulka výrobky						
7	nazev_vyrobu	Jednotková cena					
8	Chicony	5,00 €					
9	Genius	7,00 €					
10	Philips	100,00 €					

Obrázek 1.24: Původní tabulka je rozdělena na dvě samostatné a každé pole obsahuje samostatné údaje – adresa, město

V návrhu databáze *Faktury* budou podle zjištěných informací od zadavatele a uživatelů databáze tabulky, které evidují zákazníky, výrobky, kategorie výrobků, faktury a detaily faktury. Každá obsahuje pole jen k určité skupině informací. Tabulka zákazníků bude mít pole kontaktní osoba, adresa, město, PSČ, telefon a e-mail. Výrobky budou obsahovat údaje o výrobcích – kód, název, cena, poznámka. Podobně budou realizovány i další tabulky.

Druhé pravidlo normalizace: Primární klíč

Každá tabulka musí mít jednoznačný identifikátor – primární klíč, který je vytvořen z jednoho nebo více polí tabulky. Znamená to, že každý záznam v tabulce musí být jednoznačně identifikovatelný. Je to něco podobného, jako mají lidé rodné číslo, které je jedinečné, a každý může být podle rodného čísla identifikovatelný. Podobně je to i s výrobky, které by měly mít určitou formu jednoznačné identifikace, například číslo výrobku.

Na základě čísla výrobku je možné vyhledat další informace o výrobku. Při propojení tabulek se může zdát, že se v rozpisu faktury duplikují údaje o čísle výrobku, je to ale vždy efektivnější, než kdyby se v tabulce nacházely i ostatní údaje o výrobcích – název, cena, poznámka apod. Tyto informace zobrazíte v rozpisu faktury pomocí výběrových dotazů, které budou popsány později.

	A	B	C	D	E	F	G	H
1	Tabulka zakazníci							
2	ID	Nazev	Kontaktní osoba	Adresa	Mesto	Tel_kont akt	Mobil	email
3	1	Belko Ltd.	Peter Belko	Hlavní 17	Trnava	12345678	905123	peter@ltd.cz
4	2	MMP s.r.o.	Mazánová	Boční 9	Brno	258741	321456	mmp@ma.cz
5								
6	Tabulka výrobky							
7	ID	nazev_vyrobu	Jednotková cena					
8	1	Chicony	5,00 €					
9	2	Genius	7,00 €					
10	3	Philips	100,00 €					

Obrázek 1.25: Rozdělené tabulky zákazníků a výrobků doplněné o jednoznačná identifikační čísla – primární klíče

Při vytváření tabulky upozorní Access na vytvoření primárního klíče, pokud jste ho nedefinovali. V upozornění se zobrazí, že není povinný, je však doporučený. V případě, že chcete tabulku propojit s jinou tabulkou, primární klíč potřebujete. Pokud tabulka neobsahuje pole

s datovým typem automatické číslo, jako primární klíč takovéto pole při ukládání vytvoří, když potvrdíte, že chcete primární klíč nastavit. Pokud ho nastavíte manuálně, můžete vybrat i jiný datový typ, než je automatické číslo.

V databázi Faktury je potřeba do tabulek nastavit primární klíče. Tabulka zákazníků bude mít jednoznačnou identifikaci bez duplicity `ZakaznikID`. Tabulka výrobků zase `VyrobekID`, kategorie `KategorieID`, faktury budou mít `FakturaID` a rozpis detailů faktury bude mít `Faktura-DetailID`. Tabulky, které budete propojovat, musí mít příslušná pole, která se nazývají cizí klíče. Pole primárního klíče i pole cizího klíče musí mít stejný datový typ, aby se daly propojit.

Třetí pravidlo normalizace: Funkcionální závislost

Pro každou jedinečnou hodnotu primárního klíče se musí hodnoty v datových sloupcích týkat předmětu tabulky a musí tento předmět úplně popisovat. Po definování primárního klíče je potřeba zkontrolovat, zda se tento vztahuje k údajům, které jsou předmětem tabulky. To znamená, že v tabulce zákazníků musí primární klíč identifikovat jednoznačně údaje o zákazníkovi. Pokud by však tabulka obsahovala i údaje o fakturách, tak by primární klíč už nemohl jednoznačně identifikovat předmět tabulky – údaje o zákazníkovi. Proto je nezbytná samostatná tabulka pro faktury, kde její primární klíč bude jednoznačně identifikovat předmět tabulky – údaje o fakturách. Taktéž je potřeba, aby údaje v tabulce popisovaly její předmět úplně.

Jako teoretický příklad by posloužil třeba systém objednávek, kde by v tabulce objednávek kromě jiných polí mohla být fakturační adresa a adresa doručení. Identifikační číslo objednávky jednoznačně popisuje předmět. Objednávka s nějakým konkrétním číslem je fakturovaná na jednu adresu a doručena je na jinou adresu.

Čtvrté pravidlo: Nezávislost polí

Musíte být schopní realizovat změnu údajů v libovolném poli (kromě pole primárního klíče) bez toho, aby byly ovlivněny údaje v kterémkoliv jiném poli. Toto pravidlo ověřuje, zda mohou vzniknout problémy při realizaci změn údajů v tabulkách.

Teoretický příklad by mohl být v tabulce detailů faktury, která by obsahovala pole poznámky k výrobku. Každý záznam fakturovaného výrobku by obsahoval samostatnou poznámku k výrobku. Mohly by tak vzniknout různé poznámky pro jeden konkrétní výrobek. Pokud byste takovouto poznámku chtěli upravit, museli byste měnit hodně záznamů. Proto je vhodnější umístit pole poznámek do tabulky výrobků, kde ji změníte jen jednou, a změni se ve všech propojených tabulkách. Kontrolovat nezávislost polí byste mohli i sledováním, zda se v záznamech neopakují stejné údaje. Toto sledování je ale méně efektivní. Jednodušší je při výskytu chyby v poznámce opravit ji jednou než několikrát v samostatných záznamech. Efektivita přichází hlavně v případě komplikovanějších návrhů, kde se nejedná jen o jedno pole, ale o více polí.

Abyste splnili druhé pravidlo normalizace, musí tabulka splňovat požadavky prvního pravidla. U třetího pravidla musí splňovat požadavky druhého i prvního pravidla atd. V určitých případech se může stát, že budete muset některé pravidlo normalizace porušit. Týká se to většinou obchodních požadavků. Například pokud máte definováno, že cena výrobku se může průběžně měnit v čase, bude potřeba do detailů faktury přidat pole cena. Následně byste makrem nebo procedurou vkládali aktuální údaj o ceně z tabulky výrobků do tabulky detailů faktury. Tímto

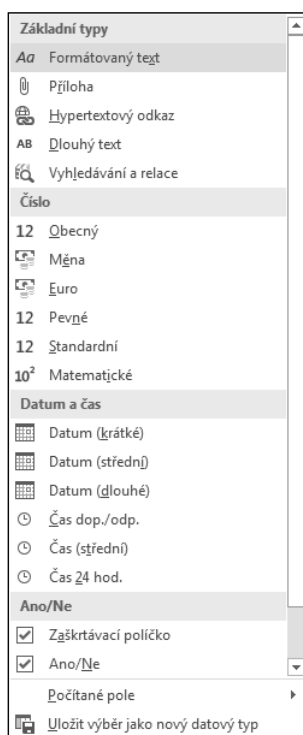
zabezpečíte, že se po změně ceny výrobku ceny nezmění ve všech existujících fakturách, jelikož ceny jsou uloženy v poli detailů faktury. Změněná cena bude platit jen pro nové faktury.

Při návrhu databáze a tabulek jsou důležité i datové typy, které pro jednotlivá pole použijete. Zvolením nesprávného datového typu můžete omezit fungování databáze.

Datové typy

Při vytváření tabulek je potřeba nastavit i datové typy. Ty je samozřejmě vhodné zpracovat už do samotného návrhu databáze, aby při vytváření tabulek nevznikaly zbytečné problémy, jaký typ použít. Týká se to hlavně polí primárního a cizího klíče a taktéž polí, která budou používána ve výpočtech, abyste se vyvarovali problémů s nesprávnými výsledky výpočtů. Ostatní pole mohou mít nastavený krátký text, který umožňuje zadat jakýkoliv znak a číselné hodnoty.

U číselných údajů, které se nepoužívají ve výpočtech, existuje riziko, že u nevhodného číselného datového typu nebudete moci zadat správně požadované číslo kvůli omezením datového typu. Například doplňkové označení výrobku bude číselná hodnota 32800. Pokud nastavíte datový typ **Celé číslo**, bude možné do pole zadat jen hodnotu do 32767.



Obrázek 1.26: Seznam datových typů dostupný v zobrazení datového listu

S takovýmito omezeními je potřeba počítat už při návrhu databáze a zjistit od zadavatele databáze, jakých hodnot mohou údaje nabývat. Není nic horšího, než když vám zadavatel po krátkém čase používání reklamuje odevzdanou databázi kvůli tomu, že nemůže zadat oby-

čejné číslo výrobku. Proto u vstupní analýzy požadavků nenechte nic náhodě. Ptejte se budoucích uživatelů databáze a zapisujte si všechny informace, které vám poskytnou v souvislosti s požadavky na databázi.

Vztahy mezi tabulkami

Kromě normalizovaných tabulek a správných datových typů jsou velmi důležité i vztahy mezi tabulkami. Správným návrhem tabulek jich vznikne několik. Vhodným vytvořením vztahů dáte údaje znovu dohromady. Kdybyste něco podobné chtěli provést v tabulkové aplikaci, tak by se situace poměrně zkomplikovala. Museli byste buď ponechat hodně nadbytečných údajů v jedné tabulce, nebo vymyslet mechanismus, jak by se údaje z více tabulek spojily. Toto všechno za vás udělá relační systém pomocí vztahů. Následně dokáže jednoduše zobrazovat související údaje, například hlavičku faktury, a detaily faktury, jako by byly údaje uloženy v jedné tabulce.

Aby vztahy fungovaly, potřebujete kromě primárního klíče i cizí klíč v jiné tabulce. V návrhu tabulky jste vytvářeli i pole, která slouží ke spojování polí mezi tabulkami. V tabulce `T_Faktury` je to pole `Zakaznici`, které se spojí s primárním klíčem `ZakaznikID` z tabulky `T_Zakaznici`. Podobně jsou navrženy i v dalších tabulkách. Tímto dáváte Accessu informaci, jak chcete spojit a vidět údaje z více tabulek.



Poznámka: Můžete se setkat i s označením primárního klíče jen `ID`. Cizí klíč může mít označení `název_pole_id`. Občas se vyskytují pole primárního a cizího klíče se stejným názvem v obou tabulkách (například i v ukázkové databázi `Northwind Traders`).

V databázích jsou dostupné vztahy `M:N`, `1:1` a `1:N`. Vztah `M:N` – více záznamů z jedné tabulky a více k nim příslušejících záznamů z druhé tabulky – není vhodný, jelikož neobsahuje ve spojených polích nastavení údajů bez duplicity. Proto je potřeba provést úpravu takovýchto tabulek nebo vytvořit ještě jednu tabulku, přes kterou se údaje spojí pomocí jednoznačných polí vztahem typu `1:N`. Vztah `1:1` – jeden záznam z jedné tabulky a jeden k němu příslušející záznam z druhé tabulky – by umožnil fakturovat vždy jen jeden výrobek na jednu fakturu. Tento typ vztahu se zřídka v databázích vyskytuje, i když je dost omezující.

Nejvhodnější vztah z pohledu návrhu většiny databází je `1:N` – jeden záznam z jedné tabulky a více k němu příslušejících záznamů z druhé tabulky. Tímto je možné na jednu fakturu přidat více výrobků na fakturaci. Stejně může mít jeden zákazník více faktur.

V návrhu databáze je potřeba ve struktuře tabulek zabezpečit, abyste související údaje mohli spojit vztahem `1:N`. Pokud v návrhu vidíte vztah `M:N`, je zapotřebí návrh vhodně změnit pomocí propojovací tabulky. Když budou jasné všechny vazby mezi tabulkami, nakreslete si je kvůli přehlednosti buď ručně nebo v aplikaci `Visio 2013`. Následně podle tohoto návrhu budete po vytvoření tabulek vytvářet vazby v okně **Relace**.