

Ďalšie vzdelávanie učiteľov základných škôl a stredných škôl v predmete informatika

Úvod do databáz

Predmet: Úvod do databáz

Línia: Vlastný odborový kontext informatiky a informatickej výchovy





Ďalšie vzdelávanie učiteľov
základných škôl a stredných škôl
v predmete *informatika*

Úvod do databáz

Identifikácia modulu

Aktivita projektu: 1.2 Vzdelávanie nekvalifikovaných učiteľov informatiky na 2. stupni ZŠ a na SŠ

Línia aktivity: Vlastný odborový kontext informatiky a informatickej výchovy

Predmet: Úvod do databáz

Zaradenie modulu



Modul je jediným modulom rovnomenného predmetu, ktorý je súčasťou línie Vlastný odborový kontext informatiky a informatickej výchovy.

Abstrakt modulu

Účastníci sa informatívne oboznámia s databázovou technológiou ukladania a spracovania údajov, s relačnou databázou a súvisiacimi pojmami – tabuľka, záznam, atribút, primárny kľúč a pod.

Na riešení konkrétneho príkladu si účastníci precvičia tvorbu malej databázy s niekoľkými prepojenými tabuľkami a konštruovanie dotazov.

Garant predmetu:

RNDr. Zuzana Kubincová,
PhD.
KZVI FMFI UK, Bratislava
kubincova@fmph.uniba.sk

Autori modulu:

RNDr. Zuzana Kubincová,
PhD.
KZVI FMFI UK, Bratislava
kubincova@fmph.uniba.sk

Mgr. Martin Cápaj, PhD.
KI UKF, Nitra
mcapaj@ukf.sk

PaedDr. Andrea Hrušecká,
PhD.
KZVI FMFI UK, Bratislava
hrusecka@fmph.uniba.sk

RNDr. Ľubomír Salanci, PhD.
KZVI FMFI UK, Bratislava
salanci@fmph.uniba.sk



| | |
|---|----|
| Úvod | 3 |
| Cieľ modulu..... | 3 |
| Vstupné vedomosti | 3 |
| Požadované prerekvizity | 3 |
| Predpokladané vstupné vedomosti a zručnosti | 3 |
| Preverenie vstupných vedomostí..... | 3 |
| 1. Databázy okolo nás | 4 |
| 2. Tabulkový kalkulátor a databázy | 7 |
| 3. Databázy a databázový softvér | 17 |
| 3.1 Vytváranie tabuliek..... | 17 |
| 3.2 Napĺňanie tabuliek údajmi a úprava údajov | 19 |
| 3.3 Formuláre | 21 |
| 3.4 Vyhľadávanie a filtrovanie v tabuľke | 25 |
| 3.5 Prepájanie tabuliek..... | 27 |
| 3.6 Vyhľadávanie v databáze | 32 |
| 4. Informačné systémy | 36 |
| Čo sme sa naučili v tomto module..... | 39 |
| Preverenie výstupných vedomostí | 39 |
| Literatúra a použité zdroje | 39 |

Úvod

V dnešnej dobe, keď sa na nás neustále valia informácie zo všetkých strán, je veľmi dôležité vedieť ich vhodným spôsobom analyzovať, potrebné údaje ukladať a spracovávať. Údajov je obrovské množstvo a často je potrebné uchovávať ich tak, aby boli k dispozícii aj pre ďalšie použitie. Je dôležité robiť to v čo najvhodnejšej forme, ktorá by umožňovala rýchlo a efektívne vyhľadávať potrebné údaje, vkladať nové, prípadne aktualizovať existujúce, zatriediť ich, ale aj vymazať nepotrebné.

Spracovávať údaje, aj keď možno nie až v takom množstve, bolo potrebné aj pred tým, ako sa na to začali využívať počítače. Na evidenciu sa používali rôzne papierové kartotéky a katalógy, do ktorých sa údaje vkladali na kartách či listoch. Usporiadávali sa podľa niekoľkých základných kritérií, čo umožňovalo rýchle vyhľadanie, ale len podľa tých istých kritérií. Všetky operácie sa robili manuálne.

V súčasnosti sa veľké množstvá údajov uchovávajú elektronicky, na rôznych médiách, na počítačoch a v počítačových sieťach. Spomedzi širokej ponuky počítačových programov je vyššie spomenutému princípu spracovania údajov najbližšie skupina programov nazývaná databázové systémy. Tieto pracujú nad databázou – akýmsi dátovým skladoom, v ktorom sa uchovávajú údaje o určitej skupine objektov, napr. databáza telefónnych čísel, databáza produktov, databáza kníh, videopožičovňa, stravovací systém, evidencia zamestnancov a pod.

V tomto module sa budeme venovať základným princípom tvorby databázy a práce s údajmi, ktoré sú v nej uložené.

Cieľ modulu

Cieľom tohto modulu je zoznámiť účastníkov vzdelávania s databázovou technológiou ukladania a spracovania údajov.

V prvej časti modulu im túto technológiu podáme z pohľadu, ktorý im je známy, a ukážeme im, že za mnohými počítačovými aplikáciami, ktoré bežne používajú, je databáza.

V druhej časti modulu sa prostredníctvom riešenia príkladu presvedčia, že na spracovanie niektorých úloh, pri ktorých potrebujeme evidovať a spracovávať rôzne typy dát, nám nestačí tabuľkový kalkulátor, ale potrebujeme na to databázový softvér.

V tretej časti modulu sa cez konkrétny databázový softvér zoznámia so všeobecnými princípmi tvorby a používania databázy.

V poslednej časti účastníci získajú základné znalosti o informačných systémoch v súvislosti s databázami.

Vstupné vedomosti

Požadované prerekvizity

Modul 2DG1 – Základná digitálna gramotnosť (práca s tabuľkami, internet).

Predpokladané vstupné vedomosti a zručnosti

Účastník vzdelávania dokáže pracovať s tabuľkovým kalkulátorom (napr. MS Office Excel) a s webovým prehliadačom.

Preverenie vstupných vedomostí

Úspešné absolvovanie modulu-prerekvizity, blesková úvodná diagnostika vstupných vedomostí lektorom.

1. Databázy okolo nás

Určite sme sa už všetci stretli so zbieraním a ukladaním údajov. Ved' aj naše mobilné telefóny sú plné rôznych nami uložených údajov.

Úloha 1.1

Prezrite si svoj mobil a zistite, aké údaje si v ňom uchováate.



Kontakty v mobilnom telefóne

Najčastejšie si v mobiloch uchováame kontakty. Aké rôzne údaje (okrem mena a telefónneho čísla) si môžeme ku kontaktu priradiť?



Obr. 1.1 Rôzne údaje, ktoré môžeme ku kontaktu priradiť

Môžeme si všimnúť, že okrem **textových** a **číselných** môžeme pridať aj iné typy údajov, napr. **obrázok**, **zvuk**, **dátum**.

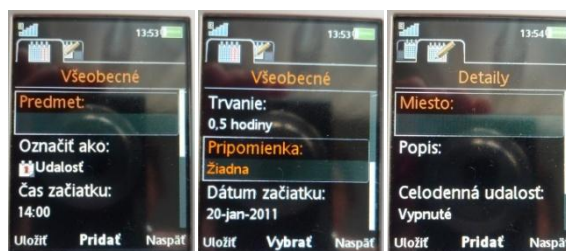
Otázka

Ktoré údaje si pri kontaktoch uchováate vy?

Často používaným aplikačným programom mobilu, v ktorom môžeme uchovávať údaje, je kalendár. Môžeme si doňho pridávať rôzne udalosti. Ak sme si v kontaktoch zaznamenali aj narodeniny, zvyčajne sa vložia ako udalosť aj do kalendára.



Kalendár v mobilnom telefóne



Obr. 1.2 Údaje, ktoré môžeme priradiť k udalosti

Otázka

Aké ďalšie informácie si v mobile zaznamenáate? Aké typy údajov to sú?

Je potrebné si uvedomiť, že pojmy **údaj** a **informácia** neznamenajú v informatike úplne to isté. Údaje sú fakty, ktoré sú výsledkom pozorovaní alebo meraní. Reprezentujú sa formálnym spôsobom tak, aby mohli byť spracovávané napr. počítačom. Sú nositeľmi informácií, ale samy osebe nemajú informačnú hodnotu. Informáciou sa údaj stáva, až keď k nemu pridáme interpretáciu. Napríklad číselné údaje 90-60-90 sa pre nás stanú informáciou, až keď zistíme, že sú to napr. rozmery skrinky, ktorú potrebujeme kúpiť, aby sme mali kde archivovať písomky našich študentov ☺

S usporiadanými údajmi sa často stretávame aj na internete. Sú to napríklad:

- internetové obchody, napr. internetové kníhkupectvá,

- stránky s cestovnými poriadkami,
- stránky denníkov,
- stránky s televíznym programom,
- archívy softvérových aplikácií,
- vzdelávacie portály,
- internetbanking,
- rezervácia a predaj vstupeniek,
- mapy,
- a ďalšie.

Pozrime sa napríklad na stránku internetového kníhkupectva, na nejaký konkrétny produkt.

The screenshot shows a product page for the book "Svět je plochý - Stručné dějiny 21.století" by Thomas L. Friedman. The page includes the book cover, the author's name, the publisher (Academia, 2007), and pricing information. The regular price is 22,30 €, and the current price is 21,19 €. There is a 5% discount (1,11 €) and a coupon code "+ ZĽAVA 1SIC". The page also lists details such as 560 pages, hardcover binding, and Czech language. Shipping information indicates that the book is usually delivered within 13 days. There are links for "Informácie", "O knihe", "Recenzie", and "Odporúčania". A yellow box highlights that shipping is free for orders over 39 €.

Obr. 1.3 Ukážka konkrétneho produktu v internetovom kníhkupectve

O každej knihe musia byť vo vhodnej forme uchované rôzne údaje zachytávajúce informácie napr. o názve, autorovi (autoroch), vydavateľstve, roku vydania, cene, zľave, dodacej lehote, vzhľade obalu (obrázok); niekedy sa ukladajú aj ukážky z knihy a ďalšie detailnejšie informácie. Tieto údaje potom internetové kníhkupectvo – webová aplikácia – používa pri zobrazovaní informácií, ktoré hľadáme. Každú knihu možno komentovať a hodnotiť, a aj tieto informácie sa musia uchovať. Ak si chceme pozrieť aj ďalšie knihy od daného autora alebo vydavateľstva, stačí kliknúť na meno autora či názov vydavateľstva. Každá kniha tiež patrí do nejakých kategórií (Spoločenské vedy, Sociológia, Politika, a pod). Cez tieto kategórie sa môžeme rýchlo dostať ku knihám s podobným zameraním.

Ak by sme si mali priblížiť rozdiel medzi údajom a informáciou na príklade internetového kníhkupectva, tak všetky zoznamy kníh a ich popis môžeme dať do kategórie údaj. Keď si ale necháme vyhľadať knihy na základe nejakého kritéria, získali sme informácie o dostupnosti, cene apod.



Internetové kníhkupectvo www.martinus.sk



Nákupný košík

Úloha 1.2

Otvorte si v prehliadači stránku internetového kníhkupectva a určte, aké všetky akcie môžeme s knihami spraviť. Ako môžeme upravovať výber zobrazených kníh?

Úloha 1.3

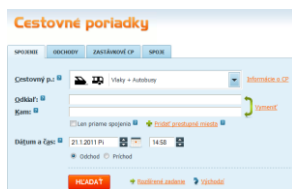
Rozmýšľajte a diskutujte o tom, aké ďalšie údaje musí mať internetové kníhkupectvo uložené, aby sme si mohli objednávať knihy.

Samozrejmosťou v každom internetovom obchode je vyhľadávanie a hľadanie v kategóriách.

Úloha 1.4

Porozmýšľajte a diskutujte o tom, ako sa bude líšiť ukladanie údajov o knihách v knižnici.

Ďalšou často používanou internetovou aplikáciou, ktorá si vo vhodnej forme musí uchovávať údaje, je vyhľadávanie cestovných spojení.



www.cp.sk

Otázka:

Ako sa vám pracovalo s predchádzajúcimi webovými aplikáciami?

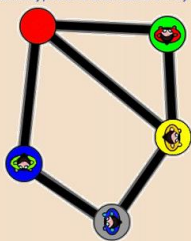
Aby sa nám s aplikáciami pracovalo dobre, musia mať jednoduché a ľahko pochopiteľné ovládanie, musia však poskytovať aj silné funkcie, aby dokázali vyhovieť našim požiadavkám.



iBobor – informatická súťaž

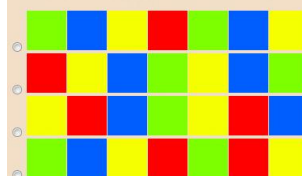
13. Farebná hra

Na hracej ploche sú takto rozostavené štyri farebné figúrky:



V jednom ťahu sa niektorá figúrka presunie na prázdne susedné políčko. Hra sa končí, keď už každá figúrka stojí na políčku svojej farby. Sedmé políčko ostane prázdne. Hru môžeme zakresliť ako postupnosť farebných štvorcíkov. Každý štvorček svojou farbou vyjadruje, ktorá figúrka prešla na voľné políčko.

Po ktorej z týchto postupností sa hra na obrázku skončí?



Úloha zo súťaže iBobor

| | |
|------------------|--|
| Úloha 1.5 | Otvorte si v prehliadači stránku www.cp.sk a nájdite všetky priame spoje, ktoré pôjdu 1. 5. 2011 medzi 9:00 a 11:00 z Bratislavy do Košíc. |
| Úloha 1.6 | Diskutujte o vašich skúsenostiach s vyhľadávaním spojení. Čo všetko môžete na tejto stránke vyhľadať? Aké rôzne parametre môžete zadávať? Aké informácie môžete/nemôžete na tejto stránke získať? |
| Úloha 1.7 | Rozmyšľajte, aké údaje musí mať aplikácia uložené, aby mohla zodpovedať na vaše požiadavky, ako ich má mať uložené a akým spôsobom pri ich vyhľadávaní postupuje. |
| Úloha 1.8 | Povedzte, ktoré údaje o vlakoch sú dôležité pre tieto osoby alebo povolania: <ul style="list-style-type: none"> dizajnér vlakov, výrobca a konštruktér vlakov, opravár vlakov, zberateľ vlakov, vláčikový modelár, tvorca cestovného poriadku. |

Obe webové aplikácie si údaje ukladajú do takzvanej **databázy**. Databáza je vlastne súbor štruktúrovaných údajov uložených v počítačovom systéme. Tieto údaje sa z databázy získavajú pomocou **dotazovacieho jazyka**, ktorý je však pre bežného používateľa skrytý. Databáza spolu s programami, ktoré nám umožňujú s ňou pracovať, tvorí **databázový systém**.

Databázy sa využívajú na rôzne účely – záznamy u lekára by sa mali postupne digitalizovať, na čo sa určite využije databáza, databázy sa využívajú v predajniach, v knižniciach, v skladoch, zamestnávateľia si vedú databázu zamestnancov, atď. Aj školské systémy (napr. elektronické žiacke knižky – aScAgenda, www.eziacka.sk) využívajú databázy. Možno ich využiť napríklad aj na archiváciu príkladov zo súťaže. Takýto archív nájdeme napr. na stránke www.ibobor.sk, kde po zvolení ročníka súťaže a kategórie je možné riešiť úlohy.

| | |
|-------------------|---|
| Úloha 1.9 | Pozrite si úlohy v archíve súťaže iBobor a porozmyšľajte, aké údaje musia byť v databáze uložené. Aké typy údajov sú použité? |
| Úloha 1.10 | Diskutujte o tom, aké údaje sa musia v databáze uchovávať, keď sú to úlohy v práve prebiehajúcej súťaži. |

Čo sme sa naučili

V tejto časti sme sa naučili, že mnohé (nie len webové) aplikácie využívané na poskytovanie **informácií**, si potrebujú vhodne uchovávať **údaje** rôznych typov, aby ich vedeli, keď je to potrebné, rýchlo nájsť. Tieto údaje sa ukladajú v **databázach**. Databáza spolu s funkciami, ktoré zabezpečujú prácu s ňou, tvoria **databázový systém**. Ten musí byť navrhnutý tak, aby bola manipulácia s údajmi jednoduchá. Databázový systém však musí poskytovať také funkcie, aby ľudia dokázali efektívne získať všetky potrebné informácie. Podľa toho, na čo databázu plánujeme používať, je potrebné navrhnuť jej štruktúru tak, aby v nej boli uložené všetky dôležité údaje o objektoch a vzťahoch.

2. Tabuľkový kalkulátor a databázy

Diskutujúc o tom, aké rozličné informácie dokážeme získať napríklad vyhľadávaním v rôznych systémoch na internete, sme si uvedomili, aké rôzne údaje potrebujú mať tieto systémy uložené, aby dokázali zodpovedať na otázky používateľa. Je tu však ešte jeden dôležitý aspekt, ktorý treba zdôrazniť. Údajov, ktoré sú takto uložené, je veľké množstvo, a pokiaľ v nich chceme vyhľadávať efektívne, musia byť uložené takým spôsobom, ktorý toto efektívne vyhľadávanie podporuje.

Predstavte si, že sa napríklad niekam veľmi ponáhľate a potrebujete veľmi rýchlo zistiť, ktorými linkami MHD sa tam dostanete a kedy je odchod najbližšieho spoja. Našťastie máte k dispozícii internet a viete, že tam sa potrebné informácie dajú zistiť. Iste by vás však nepotešilo, keby ste po zadaní všetkých potrebných parametrov na odpoveď čakali niekoľko minút.

V predchádzajúcej kapitole sme si povedali, že v systémoch, ktoré používame na vyhľadávanie rôznych informácií, sú údaje uložené v databáze. Skúsme sa teraz bližšie pozrieť na to, akým spôsobom sú v databáze údaje organizované a pochopiť, prečo práve tak.

V súčasnosti najčastejšie používanou štruktúrou na ukladanie údajov v databáze je *tabuľka*. Je to jednoduchá štruktúra, s ktorou sme už pracovali napr. v programovaní (dvojrôzmerne pole) ale aj v tabuľkovom kalkulátore. Využime teda práve tabuľkový kalkulátor na to, aby sme si ukázali, akým spôsobom by mali, či nemali byť navrhnuté tabuľky, keď s nimi chceme pracovať efektívne. Keďže súčasťou balíčka MS Office, ktorý sa na školách používa, je aj tabuľkový kalkulátor Excel, budeme pri riešení úloh používať práve ten. Rovnako dobre by sme však mohli použiť ktorýkoľvek iný.

Postupne budeme vyvíjať riešenie úlohy o ukladaní údajov potrebných ako podklady pre rozvrh. Našou snahou bude ukladať údaje tak, aby sa z nich dali efektívne získať rôzne informácie a aby sa dali pohodlne upravovať, dopĺňať, či vymazávať.

Na stránke imhd.sk možno nájsť informácie o MHD v rozličných mestách celého Slovenska.

Určite ste sa už stretli s pojmom *relačná databáza*. Je to v súčasnosti najčastejšie využívaný typ databáz. Práve relačné databázy ukladajú svoje údaje do tabuliek.

Úloha 2.1

Stiahnite si excelovský súbor `ZoznamPredmetov.xlsx` a preštudujte si jeho obsah. Zamyslite sa nad tým, na aké typy otázok sa dá odpovedať použiv údajov z tabuľky.

V súbore `ZoznamPredmetov.xlsx` je uložená tabuľka s údajmi o učiteľoch, predmetoch a triedach, ktoré majú slúžiť rozvrhárovi ako podklady pre tvorbu rozvrhu.

| | A | B | C |
|----|---|---|---------------------------|
| 1 | | Zoznam predmetov, ktoré vyučujú jednotliví učitelia | |
| 2 | | | |
| 3 | | Meno | Predmet |
| 4 | | Janko Hraško | Matematika IV., III.C |
| 5 | | Ruženka Šípková | Matematický seminár, IV.A |
| 6 | | Kubko Bača | Matematika VII. IV.C |
| 7 | | Maťko Valach | Informatika, IV.A |
| | | ... | ... |
| 26 | | Kubko Bača | Matematický seminár, IV.C |

Obr. 2.1 Základná tabuľka

Všimnime si, ako je tabuľka navrhnutá. Má dva stĺpce pomenované *meno* a *predmet*. V stĺpci *meno* sú uložené mená a priezviská učiteľov a v stĺpci *predmet* zase názvy predmetov, ktoré príslušní učitelia učia, a aj údaj o tom,

v ktorej triede. V databázovej terminológii stĺpcom hovoríme **atribúty**. Každý riadok tabuľky zodpovedá jednému **záznamu** o učiteľovi a predmete, ktorý tento učiteľ učí v príslušnej triede.



Z údajov v tejto tabuľke vieme vyčítať odpoveď nielen na otázku, aké predmety a v ktorých triedach ten-ktorý učiteľ vyučuje, ale napr. aj na otázky, ktorí učitelia vyučujú daný predmet, ktorí učitelia učia v danej triede alebo ročníku a podobne. Pokiaľ je údajov v tabuľke pomenej, hľadajú sa nám odpovede na takéto otázky pomerne jednoducho, aj keď v nej nie sú uložené spôsobom, ktorý by nám bol pri vyhľadávaní nápomocný. Pokiaľ by sme však mali v takejto tabuľke uložených veľa údajov napríklad o učiteľoch z celej školy a všetkých predmetoch vyučovaných vo všetkých triedach, ručné vyhľadávanie by už zrejme nebolo až také jednoduché.

Jednotlivé riadky v tabuľke sú na sebe navzájom nezávislé a môžu byť usporiadané ľubovoľným spôsobom. Pri vyhľadávaní informácií v tabuľke, najmä ak ide o vyhľadávanie nie iba jedného konkrétneho záznamu, ale viacerých spĺňajúcich nejaké kritériá, môže pomôcť **usporiadanie** riadkov tabuľky podľa príslušných stĺpcov. Tabuľkový kalkulátor nám spravidla poskytuje nástroj na usporadúvanie záznamov v tabuľke.

Nástroj na zoraďovanie záznamov nájdete pod ikonkou:



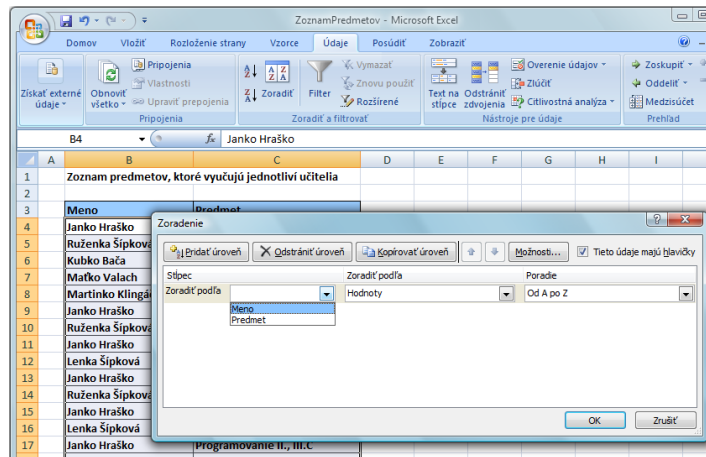
Pri označovaní tabuľky kvôli zoraďovaniu jej záznamov je výhodné označiť tabuľku aj s hlavičkou, pretože pri zadávaní kritérií pre zoraďovanie možno potom používať názvy stĺpcov z hlavičky. Je to názornejšie, ako označenia stĺpcov v Exceli.

Pri zoraďovaní záznamov v tabuľke možno použiť aj ikony  a  (vedľa hlavnej ikony **Zoraďiť**), ktoré umožňujú rýchle usporiadanie riadkov tabuľky podľa označeného stĺpca, ale aj usporiadanie len vo vyznačenej časti tabuľky.

Úloha 2.2

Usporiadajte údaje v tabuľke podľa mena a priezviska učiteľa.

V Exceli sa nástroj na zoraďovanie záznamov nachádza na záložke **Údaje** v časti **Zoraďiť a filtrovať** (vid' obr. 2.2). Pred použitím tohto nástroja je potrebné označiť všetky údaje, ktoré chceme usporiadať. Na záložke **Údaje** klikneme na ikonu **Zoraďiť** a v dialógovom okne vyberieme kritériá, podľa ktorých chceme údaje v tabuľke usporiadať. Ak sme označili celú tabuľku aj s hlavičkou, treba skontrolovať, či je v dialógovom okne zaškrtnuté políčko **Tieto údaje majú hlavičku**.



Obr. 2.2 Dialógové okno pre zoraďovanie záznamov v tabuľke

Pre vyriešenie úlohy 2.2 teda ako kritériá v dialógovom okne **Zoradenie** uvedieme, že chceme zoraďovať podľa stĺpca **Meno**, v ňom podľa **Hodnoty** a to v poradí **Od A po Z**.

Pri väčšom množstve údajov v tabuľke usporiadanie záznamov nemusí pri vyhľadávaní postačovať. Pre takýto prípad tabuľkový kalkulátor poskytuje ďalší nástroj, ktorým je **filtrovanie** údajov. Pomocou neho dokážeme z tabuľky odfiltrovať nechcené záznamy a nechať v nej len tie, ktoré spĺňajú zadané kritériá. Pôvodný obsah tabuľky sa pritom zachová, odfiltrované riadky budú iba skryté a kedykoľvek sa dajú opäť zobrazit' vypnutím filtra.

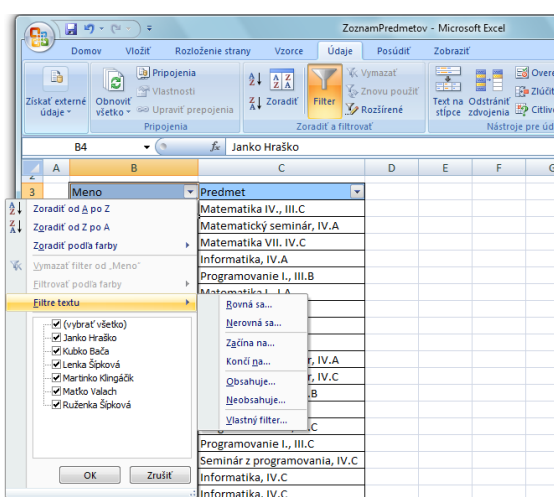
Úloha 2.3

Pomocou filtra nájdite záznamy o všetkých učiteľoch, ktorí učia informatiku v IV.C triede.

V MS Excel nájdeme filter na rovnakej záložke, ako nástroj na zoradovanie záznamov. Ak ho použijeme (klik na jeho ikonu), v našej tabuľke sa objaví v hlavičke každého stĺpca šípka. Po kliknutí na ňu sa objaví zoznam rôznych možností filtrovacích kritérií pre daný stĺpec. Patria medzi ne:

- Filtrovanie podľa farby – podľa farby bunky alebo podľa farby písma
- Filtre textu, filtre čísel, filtre dátumu a pod. – podľa toho, aké hodnoty sa v stĺpci nachádzajú
- Filtrovanie podľa obsahu bunky – pričom sa dá zo všetkých hodnôt, ktoré sa v danom stĺpci nachádzajú vybrať jedna hodnota alebo viac hodnôt, prípadne všetky

Súčasťou výberu kritérií filtrovania je aj možnosť zoradiť riadky, a to vzostupne, zostupne alebo podľa farby.



Obr. 2.3 Rôzne kritériá filtrovania

Na získanie záznamov požadovaných v úlohe 2.3 stačí zapnúť filter a v dialógu pre stĺpec *Predmet* nastaviť filter textu na hodnotu *Informatika, IV.C*. Výsledkom bude tabuľka s dvoma riadkami:

| Meno | Predmet |
|---------------|-------------------|
| Lenka Šípková | Informatika, IV.C |
| Janko Hraško | Informatika, IV.C |

Obr. 2.4 Výsledok filtrovania

Okrem základného filtrovania poskytuje MS Excel možnosť použiť *Rozšírené filtrovanie*, ktoré dovoľuje nielen filtrovať priamo v tabuľke, ale aj kopírovať údaje na iné miesto v hárku, nastaviť oblasť pre filtrovanie inú, ako je celá tabuľka a pod.

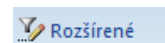
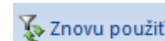
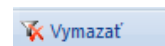
Pozrime sa teraz bližšie na to, či je tabuľka, s ktorou pracujeme, navrhnutá tak, aby nám umožňovala vyhľadávať v nej údaje efektívne. Pri zoradovaní záznamov v tabuľke podľa mena učiteľa sme sa museli uspokojiť s usporiadaním, ktoré síce bolo abecedné, ale iba podľa krstného mena učiteľa. Takéto zoradenie mien je trochu neštandardné a zrejme by nám lepšie vyhovovalo zoradenie podľa priezviska. Žiaľ, toto usporiadanie sa nám nepodarí dosiahnuť, keďže nástroj na zoradovanie záznamov nie je schopný usporiadať záznamy podľa údajov, ktoré sú len časťou hodnoty niektorého atribútu (stĺpca).



Filter

Pred kliknutím na ikonku pre základné filtrovanie treba kliknúť niekam do tabuľky, v ktorej chceme filtrovanie použiť. Inak sa objaví upozornenie, že príkaz nie je možné vykonať.

Ďalšie nástroje filtrovania:



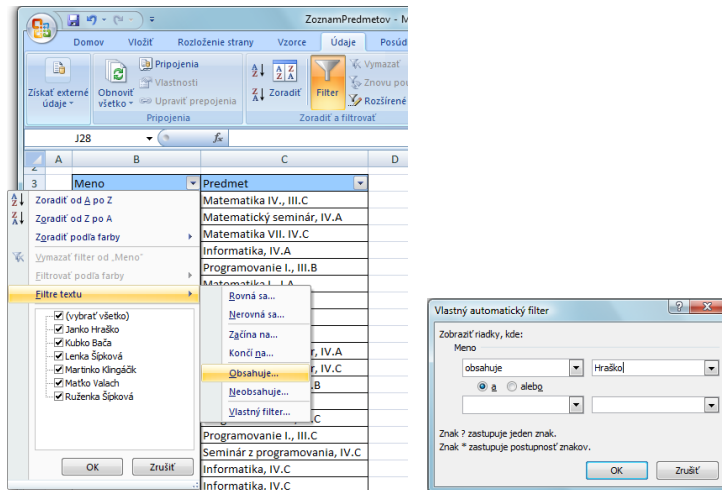
Pokiaľ však ide o filtrovanie záznamov, tu už dokážeme nastaviť kritériá tak, aby sme z tabuľky vedeli získať aj niektoré údaje, ktoré súvisia nielen s celou hodnotou zapísanou v príslušnom stĺpci, ale iba s jej časťou, napr. záznamy o konkrétnom učiteľovi vieme vyfiltrovať len podľa jeho priezviska. Stačí, ak zapneme základný filter a po kliknutí na šípku v hlavičke stĺpca **Meno** v dialógu vyberieme *Filtre textu* a v nich možnosť *Obsahuje*. V dialógovom okne *Vlastný automatický filter* potom zadáme priezvisko hľadaného učiteľa (Obr.).

Kritériá filtrovania možno kombinovať logickými spojkami A a ALEBO.

Pri zadávaní reťazcov do kritérií filtrovania sa používajú zástupné znaky:

? – zastupuje jeden znak
* – zastupuje postupnosť znakov

Základný filter poskytuje veľa možností, ktoré sú odlišné pre rôzne typy hodnôt stĺpca. Napr. ak ide o čísla – filtrovanie podľa hodnôt, ktoré sú rovné, či nerovné zadanej hodnote, menšie, väčšie, nadpriemerné a pod. V texte zasa možno filtrovať začiatok, koniec, podreťazec a iné.



Obr. 2.5 Vľavo: Nastavenie filtra na možnosť 'Obsahuje'; Vpravo: Zadávanie hodnoty do vlastného filtra


Vyskúšajme teraz rôzne iné možnosti filtrovania v našej tabuľke.

Samostatná úloha

| | |
|------------------|---|
| Úloha 2.4 | Pomocou filtra nájdite všetky záznamy, v ktorých sa priezvisko končí na „ová“. Potom vytvorte nový filter a nájdite všetky záznamy o učiteľoch Hraškoví a Klingáčikovi. |
| Úloha 2.5 | Vyfiltrujte v tabuľke všetky záznamy o učiteľoch, ktorí učia v IV. ročníku. |

Preštudujte si, aké rôzne možnosti filtrovania poskytuje základný filter.

Riešenie úlohy 2.5 sa zdá byť jednoduché a podobné riešeniu druhej časti predchádzajúcej úlohy. Pozrime sa ale, aký výsledok dostaneme, keď budeme filtrovať tak, že v stĺpci **Predmet** budeme vyhľadávať hodnoty obsahujúce reťazec 'IV.'

Ikonka  na pravom okraji stĺpca naznačuje, podľa ktorého stĺpca sa filtrovalo.

| Meno | Predmet |
|-----------------|---------------------------|
| Janko Hraško | Matematika IV., III.C |
| Ruženka Šípková | Matematický seminár, IV.A |
| Kubko Bača | Matematika VII. IV.C |
| Maťko Valach | Informatika, IV.A |
| Janko Hraško | Matematika IV., III.A |
| Lenka Šípková | Informatika, IV.A |
| ... | ... |
| Kubko Bača | Matematický seminár, IV.C |

Obr. 2.6 Výsledná tabuľka pri snahe vyfiltrovať záznamy o IV. ročníku

Na obrázku 2.6 vidno výsek z výslednej tabuľky. Riadky označené červenou farbou síce spĺňajú kritérium, ktoré sme zadali do filtra, nie sú však súčasťou správneho riešenia. Problém nastal preto, lebo vyhľadávaný reťazec 'IV.' nemusí byť v stĺpci

Predmet len súčasťou označenia triedy, ale tak, ako tomu bolo v tomto prípade, môže byť súčasťou názvu predmetu.

Podobne ako v prípade, keď sme chceli zoradiť záznamy v tabuľke podľa priezviska učiteľa, aj pri filtrovaní sme narazili na problém, ktorý súvisí s nesprávnym návrhom tabuľky. Pokiaľ vieme, že k hodnotám v bunkách tabuľky budeme potrebovať pristupovať nielen ako k celku, ale aj k ich častiam, je lepšie navrhnuť tabuľku tak, aby jednotlivé stĺpce obsahovali radšej len také hodnoty, ktoré už nepotrebujeme ďalej deliť. Hodnotám takéhoto typu hovoríme *atomické hodnoty*.

Aby sme v našej tabuľke mohli efektívne zoradovať, resp. vyhľadávať záznamy aj podľa priezviska alebo podľa ročníka, potrebujeme tabuľku preusporiadať tak, aby tieto hodnoty boli uložené ako hodnoty samostatných atribútov.

Na tabuľku, ktorá je výsledkom filtrovania, možno použiť opätovné filtrovanie, prípadne usporiadanie. Na použitý nástroj vždy upozorňujú ikonky na pravom okraji stĺpcov, pre ktoré bol daný nástroj použitý.

Informácie pre pohodlnejšie rozdelenie obsahu jednej bunky do viacerých možno nájsť napr. na

<http://office.microsoft.com/sk-sk/excel-help>.

Prepisovanie údajov v tabuľke je pre vás iste práčne. Práve pri tom si možno najlepšie uvedomiť, aké je dôležité navrhnuť tabuľku správne hneď na začiatku ;-)

Úloha 2.6

Rozdelte stĺpec *Meno* na dva stĺpce: *Meno* a *Priezvisko* a stĺpec *Predmet* na dva stĺpce: *Predmet* a *Trieda*. Naplňte novú tabuľku príslušnými údajmi.

Po úprave tabuľky tak, aby obsahovala len atomické hodnoty, bude výsledok takýto:

| Meno | Priezvisko | Predmet | Trieda |
|---------|------------|---------------------|--------|
| Janko | Hraško | Matematika IV. | III.C |
| Ruženka | Šípková | Matematický seminár | IV.A |
| ... | ... | ... | ... |

Obr. 2.7 Upravená tabuľka so štyrmi stĺpcami

Práca s údajmi uloženými v tabuľke zahŕňa nielen operácie vyhľadávania údajov za účelom získania nejakej informácie, ale aj vkladanie nových údajov, úpravu a vymazávanie existujúcich údajov. Skúsme sa teraz pozrieť na našu tabuľku z tohto hľadiska. Je navrhnutá tak, aby sa údaje v nej ľahko upravovali? Aké komplikované bude napríklad zmeniť názov predmetu, zmeniť priezvisko niektorého učiteľa, prípadne doplniť do tabuľky ďalšie (nové) údaje?

Úloha 2.7

Predpokladajme, že predmet „Matematický seminár“ sa premenuje na „Seminár z matematiky“. Spravte túto zmenu v tabuľke.

Úloha 2.8

Predpokladajme, že učiteľka Ruženka Šípková sa vydala a zmenila svoje priezvisko na Princová. Upravte údaje v tabuľke tak, aby zodpovedali novej skutočnosti.

Úloha 2.9

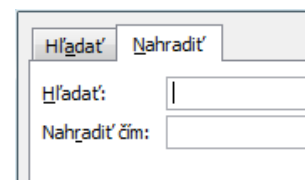
Prediskutujte s kolegami, čo všetko ste museli pri riešení predchádzajúcich dvoch úloh v tabuľke zmeniť. Ako inak by sa dali uchovávať údaje z tabuľky, aby nebolo potrebné robiť zmeny na toľkých miestach?

Úlohu 2.7 vieme vyriešiť pomerne jednoduchým spôsobom. Keďže potrebujeme nahradiť jeden reťazec („Matematický seminár“), ktorý sa môže v tabuľke vyskytovať na viacerých miestach, druhým reťazcom („Seminár z matematiky“), môžeme to urobiť pomocou dialógového okna *Hľadanie a nahradenie*.

Dostaneme sa k nemu, ak na záložke *Domov* v časti *Úpravy* vyberieme z ponuky *Nájsť & vybrať* položku *Nahradiť*. Do dialógu zadáme oba reťazce. Potom môžeme striedavým používaním tlačidiel *Hľadať ďalej* a *Nahradiť* postupne prechádzať po jednotlivých nájdených reťazcoch a nahrádzať ich druhým reťazcom. Ešte



Nájsť & vybrať



Tip:
Rýchly spôsob, ako sa dostanete k dialógovému oknu *Hľadanie a nahradenie*, je klávesová skratka *Ctrl+H*.

jednoduchším riešením je použiť v tomto dialógovom okne tlačidlo *Nahradiť všetky*, pomocou ktorého sa nájdu všetky výskyty prvého zadaného reťazca a nahradia sa druhým zadaným reťazcom naraz.

Keď si lepšie prezrieme obsah tabuľky, zistíme, že úlohu 2.8 už takto rýchlo – pomocou tlačidla *Nahradiť všetky* – riešiť nemôžeme. Je to preto, lebo priezvisko Šípková nie je v tabuľke jedinečné, ale patrí dvom osobám a použitie tohto tlačidla by spôsobilo, že nahradíme priezvisko aj tam, kde nechceme. Zostávajú nám dve možnosti (samozrejme, okrem vlastného ručného prehládavania a prepisovania), ako danú úlohu vyriešiť.

Prvou z nich je postupné vyhľadávanie záznamov, ktoré obsahujú priezvisko Šípková a jeho nahradenie novým priezviskom v prípade, že ide o záznam učiteľky Ruženky Šípkovej. Ak tabuľka obsahuje veľa takýchto záznamov, pôjde o pomerne prácnu operáciu.

Druhý spôsob riešenia využije kombináciu filtrovania a nahrádzania. Najprv z tabuľky vyfiltrujeme všetky záznamy s priezviskom Šípková, potom zo získanej tabuľky vyfiltrujeme všetky záznamy s menom Ruženka a na výslednú tabuľku aplikujeme operáciu nahrádzania. Tentokrát už môžeme použiť tlačidlo *Nahradiť všetky*.

Úlohy 2.7 a 2.8 mali jednoduché zadania. A aj keď pracujeme s tabuľkou, ktorá je pomerne malá, riešenie úlohy 2.8 bolo prácne. Predstavme si teraz, že by tabuľka, v ktorej potrebujeme robiť úpravy, mala omnoho viac atribútov súvisiacich s učiteľom a niekoľko desiatok alebo stoviek záznamov. Spraviť zmenu nejakej hodnoty pre konkrétneho učiteľa by mohlo znamenať buď nastavovanie veľkého množstva filtrov, alebo postupné prehládávanie veľkého množstva riadkov tabuľky.

Podobne ako pri programovaní si aj tu zvykneme nepoužívať v názvoch atribútov diakritiku a medzery.

Samostatná úloha

Samostatná úloha

| | |
|-------------------|--|
| Úloha 2.10 | Rozvrhár, ktorému slúži tabuľka v súbore <i>ZoznamPredmetov.xlsx</i> ako podklad pre rozvrh, potrebuje mať kontakty na učiteľov. Doplňte do tabuľky nový atribút <i>Telefon</i> a podľa pokynov lektora naplňte príslušný stĺpec údajmi. |
| Úloha 2.11 | Učiteľ Kubko Bača je dlhodobo chorý, preto by už nemal figurovať v rozvrhárovej tabuľke. Vymažte jeho záznamy z tabuľky. |
| Úloha 2.12 | Prediskutujte s kolegami, na aké problémy ste narazili pri riešení predchádzajúcich dvoch úloh. Aká iná organizácia údajov by pomohla tieto problémy vyriešiť? |

Zamyslime sa nad tým, či sú riešenia úloh 2.10 a 2.11 v niečom problematické. Vložiť nový stĺpec do tabuľky je jednoduché, no naplniť ho údajmi môže byť prácne. Navyše sa pritom robí v istom zmysle „robotu navyše“, pretože v prípade, že učiteľ učí viac predmetov, resp. vo viacerých triedach, a teda sa vo viacerých záznamoch nachádzajú o ňom údaje, tak treba jeho telefónne číslo dopisovať do všetkých týchto záznamov. Prácu si síce možno uľahčiť použitím filtrovania a kopírovania, ale napriek tomu býva často zdrojom chýb.

Problém, ktorý sa skrýva za úlohou 2.11, nesúvisí ani tak s postupom pri jej riešení, ako s výsledkom. Keďže z tabuľky boli vymazané všetky záznamy, ktoré súvisia s daným učiteľom a predmetmi, ktoré vyučuje, stratili sa aj všetky údaje o ňom. V našom prípade šlo o meno, priezvisko a telefonický kontakt, ale vo všeobecnosti by mohlo ísť aj o väčšie množstvo údajov.

Čo teda spôsobuje ťažkosti pri úprave údajov v tabuľke? Ako sme mali možnosť vidieť pri riešení niekoľkých posledných úloh, zdrojom problémov sú najčastejšie

také údaje, ktoré sa v tabuľke na viacerých miestach opakujú – napríklad všetky údaje o učiteľovi sa opakujú v tabuľke toľkokrát, koľko rôznych predmetov v rôznych triedach tento učiteľ učí. Takáto **duplicita údajov** spôsobuje nielen to, že sa s nimi pracuje neefektívne, ale pri ich úprave či pri ich vkladaní na viaceré miesta sa do tabuľky často zanášajú aj chyby (napr. keď sa omylom neupravia všetky výskyty danej hodnoty, alebo sa v niektorej úprave urobí preklep a pod.). Vzniknutá tabuľka potom obsahuje **nekonzistentné údaje**.

Ďalší problém sa ukázal pri vymazávaní záznamov, pri ktorom došlo ku **strate údajov**, a teda aj informácií, a to aj takých, ktoré by bolo výhodné si ponechať. Tu už ťažkosť nespôsobila duplicita údajov, ale ich uchovávanie na „nesprávnom mieste“.

Riešením vyššie popísaných problémov môže byť reorganizácia tabuľky. Bolo by efektívnejšie udržiavať údaje, ktoré navzájom spolu úzko súvisia a popisujú jeden objekt (napr. učiteľa), spolu, nie však spolu s ostatnými údajmi. Táto úvaha nás teda vedie k rozdeleniu pôvodnej tabuľky na dve tabuľky alebo na viac tabuliek.

Úloha 2.13

Prediskutujte v skupine, aké by bolo čo najúčelnejšie rozdelenie tabuľky. Koľko nových tabuliek by ste navrhli? Aké atribúty by mali a aké údaje by obsahovali?

Ako najproblematickejšie sa v našom príklade o podkladoch pre rozvrhára javia údaje, ktoré popisujú jednotlivých učiteľov. Pozostávajú z troch hodnôt a opakujú sa vždy v niekoľkých záznamoch v tabuľke. Určite bude preto vhodné vytvoriť pre učiteľa samostatnú tabuľku.

Úloha 2.14

Vytvorte samostatnú tabuľku pre učiteľa s príslušnými atribútmi (označme ju UCITEL) a naplňte ju údajmi z pôvodnej tabuľky.

Úloha 2.15

Zamyslite sa nad tým, v akom vzťahu sú údaje v tabuľke UCITEL k pôvodnej tabuľke a jej údajom. Prediskutujte s kolegami, čo a ako treba upraviť v pôvodnej tabuľke, aby tieto dve tabuľky spolu zachytávali tú istú situáciu ako pôvodná tabuľka.

Keďže nová tabuľka má zachytávať informácie bezprostredne súvisiace s učiteľom, bude mať zrejme tie atribúty z pôvodnej tabuľky, ktoré jednoznačne patria iba k učiteľovi. Mohli by sme povedať, že sú to jeho vlastnosti, resp. že ho popisujú.

Novovzniknutá tabuľka UCITEL bude mať teda atribúty Meno, Priezvisko a Telefon. Pre každého učiteľa, ktorý sa vyskytol v pôvodnej tabuľke, bude v tabuľke UCITEL vyčlenený jeden záznam s údajmi o ňom, bez ohľadu na to, v koľkých záznamoch sa vyskytol v pôvodnej tabuľke.

Dôvodom pre presun údajov z pôvodnej tabuľky boli problémy pri vkladaní, úprave a vymazávaní niektorých údajov. Po tejto reorganizácii údajov sa všetky zmeny týkajúce sa údajov o nejakom konkrétnom učiteľovi (zmena priezviska, telefónu, prípadne vymazanie, ak sú už údaje nepotrebné) udejú len na jednom mieste – v jedinom zázname v tabuľke UCITEL. Aj vytvorenie nového atribútu a jeho naplnenie bude znamenať pre každého učiteľa zmenu len v jednom zázname. V prípade, že je potrebné z pôvodnej tabuľky odstrániť záznam o tom, aké predmety vyučuje niektorý konkrétny učiteľ, neovplyvní to záznam o učiteľovi, ktorý môže teraz naďalej zostať uložený v tabuľke UCITEL a môže byť naďalej v budúcnosti použitý.

Predpokladajme, že vďaka uloženiu údajov do novej tabuľky sme pri odstraňovaní záznamov o predmetoch, ktoré učil dlhodobo chorý učiteľ, nemuseli vymazať aj

Excel umožňuje v rámci jedného hárka vytvárať viacero nezávislých tabuliek. Stačí, aby boli medzi sebou oddelené prázdny riadok/stĺpcom.



Nájsť & vybrať

| | |
|---------------|----------------------|
| Hľadať | Nahradiť |
| Hľadať: | <input type="text"/> |
| Nahradiť čím: | <input type="text"/> |

údaje o ňom. Tabuľka naplnená údajmi a usporiadaná podľa priezviska učiteľa bude vyzerat' takto:

| Meno | Priezvisko | Telefon |
|----------|------------|------------|
| Kubko | Bača | 0907171717 |
| Janko | Hraško | 0905112233 |
| Martinko | Klingáčik | 0915149712 |
| Ruženka | Princová | 0949491194 |
| Lenka | Šípková | 0903033000 |
| Maťko | Valach | 0949134134 |

Obr. 2.8 Tabuľka UCITEL usporiadaná podľa priezviska

Pozrime sa teraz na ďalšie zmeny, ktoré je ešte potrebné spraviť po presune údajov do tabuľky UCITEL. Pôjde o zmeny v pôvodnej tabuľke, ale nielen v nej.

Keďže všetky údaje týkajúce sa učiteľov sa majú udržiavať iba v tabuľke UCITEL, musia sa tieto údaje odstrániť z pôvodnej tabuľky. Keby z nej boli len jednoducho vymazané, tabuľka stratí svoj pôvodný význam, pretože nebude zachytávať súvis medzi učiteľom a predmetmi, resp. triedami. Aby sa tento súvis – v databázovej terminológii **vzt'ah** – zachoval, je potrebné sa vedieť v každom zázname z pôvodnej tabuľky odvolať na príslušný záznam v tabuľke UCITEL. Toto však zatiaľ nevieme spraviť, pretože jednotlivé záznamy v tabuľke UCITEL nevieme identifikovať.

Aké máme možnosti na rozlíšenie jednotlivých záznamov a ako sa na ne odvolávať v inej tabuľke? Jednou z možností by bolo odvolávať sa na záznam prostredníctvom jeho atribútu. Muselo by však byť zaručené, že jeho hodnota bude jedinečná pre každý záznam. Dalo by sa v našom prípade odvolávať na konkrétneho učiteľa napríklad prostredníctvom jeho mena alebo priezviska? Žiadny z týchto atribútov nemusí byť jednoznačný – určite máte aj vy vo svojej škole aspoň dvoch kolegov s rovnakým krstným menom a dve rovnaké priezviská tiež nie sú úplnou výnimkou. Atribút Telefon taktiež nemusí vyhovovať, stačí si predstaviť napr. situáciu, že na škole učia manželia a ako kontakt na seba uvedú číslo na ich domácu pevnú linku.

Druhou možnosťou by teoreticky mohla byť kombinácia viacerých atribútov. Pravdepodobnosť, že sa pomocou nej budú dať jednotlivé záznamy identifikovať, je síce väčšia, ale tiež nie je stopercentná. Navyše, keby sme sa chceli prostredníctvom nich v jednej tabuľke odvolávať na záznam z druhej tabuľky, mohli by sme sa dostať naspäť k duplicitne údajov, ktorej sme práve vytvorením druhej tabuľky chceli zabrániť.

Riešením preto bude tretia možnosť – zavedenie ďalšieho atribútu, ktorý bude jednoznačný. Jeho hodnotou môže byť napríklad poradové číslo záznamu v tabuľke. Nazvime tento atribút ID_ucitela.

| ID_ucitela | Meno | Priezvisko | Telefon |
|------------|----------|------------|------------|
| 1 | Kubko | Bača | 0907171717 |
| 2 | Janko | Hraško | 0905112233 |
| 3 | Martinko | Klingáčik | 0915149712 |
| 4 | Ruženka | Princová | 0949491194 |
| 5 | Lenka | Šípková | 0903033000 |
| 6 | Maťko | Valach | 0949134134 |

Obr. 2.9 Tabuľka UCITEL s atribútom ID_ucitela

Vo všetkých vyššie uvedených možnostiach sme hľadali spôsob, ktorým sa dá s určitou istotou identifikovať konkrétny záznam. V databázovej terminológii atribút alebo skupinu atribútov, ktoré jednoznačne určujú záznam, nazývame **primárny kľúč** (niekedy len *klúč*).

Primárny kľúč z tabuľky UCITEL teraz využijeme na to, aby sme vytvorili vzťah medzi pôvodnou tabuľkou a tabuľkou UCITEL.

Úloha 2.16

Upravte pôvodnú tabuľku tak, aby neobsahovala duplicitné údaje o učiteľovi a aby spolu s tabuľkou UCITEL zachytávala pôvodné vzťahy medzi učiteľmi, predmetmi a triedami.

Z pôvodnej tabuľky treba odstrániť údaje, ktoré sme presunuli do tabuľky UCITEL. Aby sme nestratili súvis medzi učiteľom a údajmi o predmete a triede, vložíme v prvom rade do tabuľky nový stĺpec pre atribút ID_ucitela. Potom podľa hodnôt atribútov Meno a Priezvisko doplníme do jednotlivých záznamov príslušné ID_ucitela z tabuľky UCITEL. Nakoniec vymažeme z tabuľky (označujeme ju ZOZNAM_PREDM) teraz už nepotrebné stĺpce s atribútmi Meno, Priezvisko a Telefon. Výsledná tabuľka by mohla vyzerat' takto:

| ID_ucitela | Predmet | Trieda |
|------------|----------------------|--------|
| 2 | Matematika IV. | III.C |
| 4 | Seminár z matematiky | IV.A |
| 6 | Informatika | IV.A |
| 3 | Programovanie I. | III.B |
| ... | ... | ... |

Obr. 2.10 Tabuľka ZOZNAM_PREDM po odstránení duplicitných údajov o učiteľoch

Vráťme sa teraz k úlohe 2.7 o premenovaní predmetu, v ktorej sme po prvý raz narazili na neefektívne uchovávanie údajov a kde bolo pri ich upravovaní treba vykonať „prácu navyše“. Podobne, ako v prípade údajov o učiteľovi, aj tu bol problém spôsobený duplicitou údajov. Preto ho aj v tomto prípade (aj keď ide len o jeden atribút) možno riešiť rovnakým spôsobom, teda vytvorením samostatnej tabuľky PREDMET, presunutím údajov o predmetoch do tejto tabuľky a vytvorením jej primárneho kľúča. Potom už stačí len odstrániť nepotrebné údaje z tabuľky ZOZNAM_PREDM a odvolávať sa v nej na primárny kľúč tabuľky PREDMET.

Úloha 2.17

Vytvorte samostatnú tabuľku PREDMET s príslušnými atribútmi a naplňte ju údajmi z pôvodnej tabuľky.

Samostatná úloha

Keďže v tabuľke ZOZNAM_PREDM sa hodnoty v stĺpci Trieda tiež opakujú, na základe rovnakých úvah možno dospieť k tomu, že aj pre triedu treba vytvoriť samostatnú tabuľku a prostredníctvom jej primárneho kľúča ju prepojiť s tabuľkou ZOZNAM_PREDM. (Ako argument pre tento krok možno uviesť napríklad nedávne premenovávanie tried osemročných gymnázií. ☺)

Úloha 2.18

Vytvorte samostatnú tabuľku TRIEDA s príslušnými atribútmi a naplňte ju údajmi z pôvodnej tabuľky.

Samostatná úloha

Posledným krokom, ktorý ešte treba spraviť, je úprava tabuľky ZOZNAM_PREDM – odstránenie údajov zo stĺpca Trieda a jeho nahradenie stĺpcom, v ktorom bude odvolávka na primárny kľúč tabuľky TRIEDA.

Úloha 2.19

Upravte tabuľku ZOZNAM_PREDM tak, aby neobsahovala duplicitné údaje a aby spolu s tabuľkami UCITEL, PREDMET a TRIEDA poskytovala presne tie informácie o učiteľoch, predmetoch a triedach, ktoré bolo možné zistiť z pôvodnej tabuľky.

Novovytvorené tabuľky `PREDMET` a `TRIEDA` budú po usporiadaní údajov a priradení primárnych kľúčov vyzerat' nasledovne:

Alternatívna úloha:

Vytvorte *evidenčný systém pre videopožičovňu*.

Navrhnite štruktúru tabuliek tak, aby spĺňali všetky vlastnosti, ktoré sme ukázali na ukázkovom príklade *rozvrh*. Chceme evidovať nasledujúce údaje o DVD, klientoch a výpožičkách:

o DVD:

- názov filmu,
- žáner

o klientoch:

- meno
- priezvisko
- číslo preukazu
- kontakt

o výpožičkách:

- kto si požičiaval,
- čo si požičiaval,
- kedy,
- na ako dlho,
- koľko platí?

Otázka:

Ako by sme mohli evidovať viacžánrové DVD?

| ID_predm | Nazov_predmetu | ID_triedy | Trieda |
|----------|-------------------------|-----------|--------|
| 1 | Informatika | 1 | I.A |
| 2 | Matematika I. | 2 | I.B |
| 3 | Matematika II. | 3 | III.A |
| ... | | ... | |
| 9 | Seminár z programovania | 9 | IV.D |

Obr. 2.11 Vľavo: Tabuľka `PREDMET`; Vpravo: Tabuľka `TRIEDA`

Po vykonaní všetkých požadovaných úprav v tabuľke `ZOZNAM_PREDM` dostaneme tabuľku s takýmito údajmi:

| ID_ucitela | ID_predm | ID_triedy |
|------------|----------|-----------|
| 2 | 4 | 5 |
| 4 | 8 | 6 |
| 6 | 1 | 6 |
| ... | ... | ... |
| 6 | 1 | 4 |
| 4 | 8 | 9 |

Obr. 2.12 Výsledná tabuľka `ZOZNAM_PREDM`

Všetky údaje o učiteľoch, predmetoch a triedach, ktoré sme si potrebovali pamätať, boli pôvodne uložené spolu v jednej tabuľke. Ako sme videli na konkrétnych príkladoch, s takto navrhnutou tabuľkou sa pri úprave údajov, ich vkladaní, resp. vymazávaní pracovalo neefektívne. Preto sme postupne pôvodnú tabuľku rozdelili a pretransformovali na štyri tabuľky. Tieto už neobsahujú duplicitné údaje, preto je práca s nimi efektívnejšia, nezanášajú sa do nich chyby a umožňujú zamedziť istež strate informácie.

Na druhej strane, štruktúra štyroch tabuliek navzájom prepojených kľúčmi nie je natoľko prehľadná a čitateľná pre bežného používateľa, aby sa v nej dali v tabuľkovom kalkulátore pohodlne vyhľadávať informácie.

Takže ani s jednou zo štruktúr, ktoré sme tu zatiaľ prezentovali – či už všetky údaje uložené spolu v jednej tabuľke, alebo sada niekoľkých navzájom prepojených tabuliek – sa v tabuľkovom kalkulátore nepracuje ľahko a efektívne. Je to preto, lebo tento typ softvéru bol primárne navrhnutý na spracovanie údajov iného charakteru a našim požiadavkám na ukladanie a vyhľadávanie údajov úplne nevyhovuje.

Ak chceme udržiavať množinu údajov tak, aby sme mohli do nej nové údaje priebežne vkladať, vymazávať z nej, prípadne uložené údaje upravovať a chceme v nej vedieť vyhľadávať údaje, spĺňajúce často zložité logické podmienky, potrebujeme na to použiť databázový softvér.

Čo sme sa naučili

V tejto časti sme si vysvetlili pojmy ako **atribút** a **záznam** a dali sme ich do súvisu s už známou štruktúrou – **tabuľkou**. Precvičili sme si prácu s ňou v tabuľkovom kalkulátore a použili sme **zoraďovanie** a **filtrovanie** ako pomocné nástroje pri vyhľadávaní údajov v tabuľke. Ukázali sme, že pre efektívnu prácu s údajmi veľmi záleží na tom, ako sú tabuľky navrhnuté. V niektorých prípadoch možno problematickú situáciu riešiť **rozdelením** tabuľky. Lepším riešením však je vedieť hneď na začiatku **navrhnuť správnu štruktúru viacerých tabuliek**. Naš príklad ale ukázal aj to, že tabuľkový kalkulátor nie je postačujúcim nástrojom na databázové spracovanie údajov.

3. Databázy a databázový softvér

Riešenie úloh v predchádzajúcej časti ukázalo, že tabuľkový kalkulátor (napr. MS Office Excel) nie je dostatočne silným nástrojom na riešenie nášho problému. Na prácu s databázou preto používame špecializovaný databázový softvér, ktorý dokáže okrem základnej práce s tabuľkami zabezpečiť aj vzájomné prepojenie tabuliek, kontrolovať jednoznačnosť záznamov v tabuľkách, uľahčuje vyhľadávanie záznamov spĺňajúcich aj komplikované kritériá a pod. Už v prvej kapitole sme sa dozvedeli, že takémuto softvéru hovoríme databázový systém. Jeho úlohou je spravovať údaje uložené v databáze, určovať spôsob ich uloženia i prístupu k nim, možnosti ich spracovania a podobne.

V praxi ste sa mohli stretnúť s rôznymi databázovými systémami. Medzi najznámejšie patria dBase, Access, FoxPro, Paradox, Oracle, Interbase, MS SQL, My SQL a pod.

Pripomeňme si rozdiel medzi databázou a databázovým systémom. Databáza – to sú len samotné údaje, zatiaľ čo databázový systém pozostáva z databázy a softvéru, ktorý zabezpečuje správu údajov v databáze od spôsobu ich uloženia až po zodpovedanie komplikovaných požiadaviek.

Úloha 3.1

Pracovali ste už niekedy s niektorým z vyššie uvedených alebo iným databázovým systémom, resp. s viacerými? Popíšte svoje skúsenosti, prípadne porovnajte prácu v dvoch rôznych databázových systémoch.

Keďže základný balíček MS Office, ktorý školy dostávajú zadarmo, neobsahuje databázový softvér (MS Access), zvolili sme pre náš kurz voľne šíriteľný databázový softvér OpenOffice.org Base (ďalej označujeme len OO Base).

3.1 Vytváranie tabuliek

V predchádzajúcej kapitole sme si na príkladoch vysvetlili, že údaje pôvodne uložené v jednej tabuľke neboli správne zorganizované a pôvodnú tabuľku sme rozložili (dekomponovali) na niekoľko tabuliek. Pozrime sa teraz na to, ako budeme údaje udržiavať v databáze a ako s nimi budeme pracovať. Začneme tým, že údaje z tabuliek, ku ktorým sme dospeli v tabuľkovom kalkulátore, preniesieme do databázového softvéru.

Úloha 3.1.1

V databázovom systéme OO Base vytvorte novú databázu s názvom Rozvrh_podklady.odt.

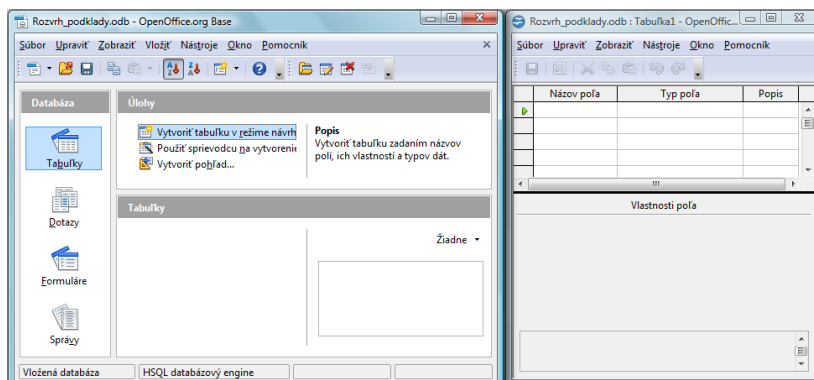
Úloha 3.1.2

V novovytvorenej databáze vytvorte tabuľku s názvom UCITEL a atribútmi Meno, Priezvisko a Telefon.

Aj napriek tomu, že takmer všetky súčasné databázové systémy už dovoľujú diakritiku, názvy tabuliek a atribútov je lepšie voliť bez diakritiky. Môže to pomôcť vyhnúť sa napr. problémom s rozdielnym kódovaním.

Po spustení programu OO Base a vytvorení novej databázy ju otvoríme a v hlavnej časti okna sa nastavíme na panel *Databáza* a jeho ikonu *Tabuľky*. Na paneli *Úlohy* sa objaví niekoľko možností vytvorenia tabuľky. Pomocou voľby *Vytvoriť tabuľku v režime návrhu...* vytvoríme novú tabuľku.

Ďalšia možnosť, ako vytvoriť novú tabuľku je *Použiť sprievodcu na vytvorenie tabuľky*. Sprievodca ponúka širokú paletu vzorových tabuliek (napr. Úlohy, Objednávky, Kontakty, Zamestnanci a pod.), z ktorých si možno vybrať a tabuľku upraviť podľa svojich potrieb.



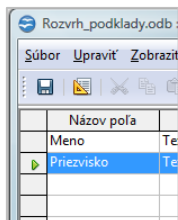
Obr. 3.1 Vytvorenie novej tabuľky v OO Base

V režime návrhu tabuľky zapisujeme názvy atribútov do jednotlivých riadkov tabuľky, aj keď potom pri zobrazovaní tabuľky budú atribúty tvoriť jej stĺpce.

V OO Base sa namiesto pojmu *atribút* používa pojem *pole*.

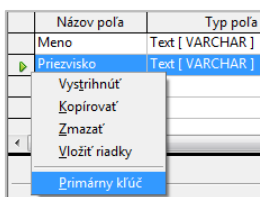
Niektoré atribúty v tabuľke (spravidla tie, ktoré neobsahujú tie najdôležitejšie informácie) môžu mať nastavený atribút tak, že zadanie ich hodnoty je nepovinné. V takom prípade táto hodnota nemusí byť v tabuľke vyplnená.

Riadok v tabuľke možno označiť kliknutím ľavým tlačidlom myši na ľavý šedý okraj v príslušnom riadku tabuľky.



Ak chceme označiť viac po sebe idúcich riadkov, označíme prvý z nich a so stlačeným klávesom *Shift* klikneme do posledného z nich.

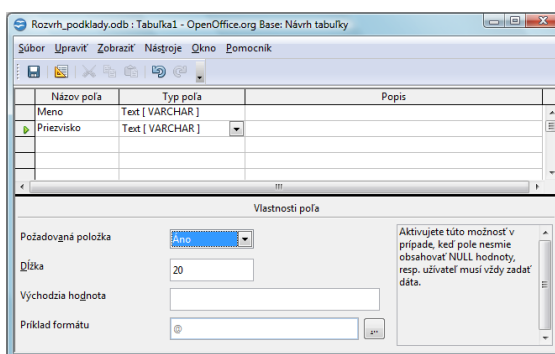
Ak chceme označiť viac riadkov, ktoré nenasledujú bezprostredne po sebe, označujeme ich klikaním so stlačeným klávesom *Ctrl*.



Pre novovytvorenú tabuľku treba zadať, aké atribúty budú obsahovať jej záznamy a aké budú typy a vlastnosti týchto atribútov. V OO Base sa na označenie stĺpca, resp. atribútu používa pojem *pole*, preto ho budeme aj my používať ako ekvivalent.

Názvy polí zapisujeme v režime návrhu do jednotlivých riadkov tabuľky. Typ údajov, ktoré budú tvoriť hodnoty poľa, treba vybrať z ponuky. V prípade atribútov *Meno*, *Priezvisko* aj *Telefon* je vhodné nastaviť typ *Text [VARCHAR]*, ktorý povolí pri napĺňaní tabuľky údajmi vkladať ako hodnoty tomuto poľu znakové reťazce rôznych dĺžok.

Pre každé pole možno v dolnej časti okna, označenej ako *Vlastnosti poľa*, urobiť rôzne nastavenia – napríklad východziu hodnotu, príklad formátu, pre čísla počet desatinných miest, prípadne, či ide o hodnotu, ktorá sa generuje automaticky. Vo vlastnosti *Dĺžka* tu možno nastaviť aj maximálnu povolenú dĺžku textových položiek (napr. poľu *Priezvisko* sme nastavili dĺžku na 20). Dôležitým nastavením je *Požadovaná položka*, kde sa nastavuje, či je zadávanie hodnoty v tomto poli povinné, alebo nepovinné. Poliam *Meno* a *Priezvisko* nastavíme túto položku na povinnú, poľu *Telefon* ju môžeme nastaviť na nepovinnú.



Obr. 3.2 Zadávanie atribútov novej tabuľky

Pripomeňme si teraz, ako vyzeral výsledný návrh tabuľky UCITEL v tabuľkovom kalkulátore. Keďže sme potrebovali vedieť jednotlivých učiteľov navzájom rozlíšiť, resp. daného učiteľa jednoznačne identifikovať a vedieť sa naňho odvolať, zaviedli sme do tabuľky nový atribút *ID_ucitel* s celočíselnou hodnotou. Tento atribút sa stal primárnym kľúčom, čiže identifikátorom jednotlivých záznamov v tabuľke.

Pri vytváraní tabuľky UCITEL v OO Base sme zatiaľ tento atribút nevytvorili. Nie je nutné tak spraviť, pretože databázový systém dokáže vytvoriť primárny kľúč (s vopred nastaveným číselným typom) aj sám, a to pri ukladaní tabuľky. Objaví sa dialógové okno, v ktorom treba zadať názov ukladanej tabuľky a za ním okno s upozornením, že tabuľka zatiaľ nemá primárny kľúč. Ponúkne sa možnosť vytvoriť ho a po jej potvrdení sa do tabuľky vsunie nový prvý riadok s názvom poľa *ID* celočíselného typu. Pole možno v prípade potreby premenovať. Vhodné je nastaviť mu aj položku *Automatická hodnota* na *Áno* a zabezpečiť tak automatické generovanie jednoznačnej hodnoty tohto poľa pri každom novom dopĺňanom zázname.

Primárny kľúč možno do tabuľky zaviesť aj iným spôsobom. Stačí označiť riadok s príslušným atribútom, prípadne niekoľko riadkov a po kliknutí pravým tlačidlom myši na ľavý šedý okraj označeného riadku vybrať z kontextovej ponuky možnosť *Primárny kľúč*.

Úloha 3.1.3

Podobným spôsobom, ako sa vytvárala tabuľka UCITEL vytvorte tabuľku s názvom PREDMET s poľom *Nazov* typu znakový reťazec s premenlivou dĺžkou. Pri ukladaní tabuľky dajte vytvoriť celočíselný primárny kľúč *ID* a zapnite mu automatické generovanie hodnôt.

Samostatná úloha

Úloha 3.1.4

Vytvorte tabuľku s názvom TRIEDA a poľami (stĺpcami):

- *ID* typu celé číslo (INTEGER),
- *Trieda* typu znakový reťazec s premenlivou dĺžkou.

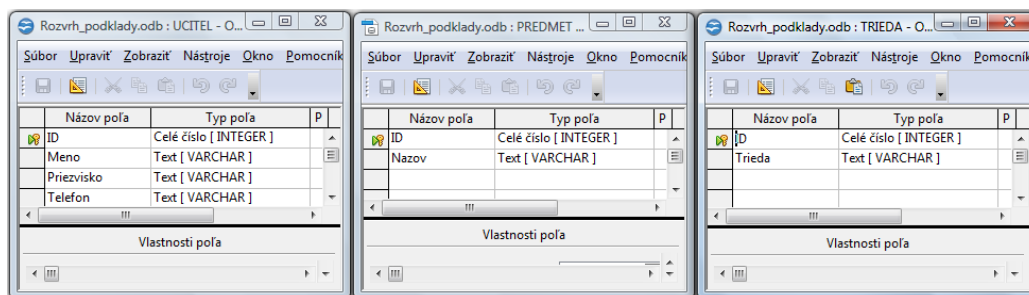
Pole *ID* označte ako primárny kľúč tabuľky. Nič iné mu nenastavujte.

Samostatná úloha

3.2 Napíňanie tabuliek údajmi a úprava údajov

Výsledkom riešenia úloh 3.3 až 3.6 sú tri tabuľky (viď Obr. 3.3), ktoré sú zatiaľ prázdne. Aby sme mohli ukázať ďalšiu prácu s nimi, potrebujeme ich najprv naplniť údajmi.

Uvedomme si, že ak by sme tieto tri tabuľky aj naplnili rovnakými údajmi, aké obsahovali ich predlohy v tabuľkovom kalkulátore, nebudú ešte zachytávať stav, keď boli všetky údaje uložené spolu v jednej tabuľke a ani zatiaľ nebude možné získať z nich rovnaké informácie. Nevytvorili sme totiž tabuľku, ktorá zachytáva vzťahy medzi učiteľom, predmetom a triedou.



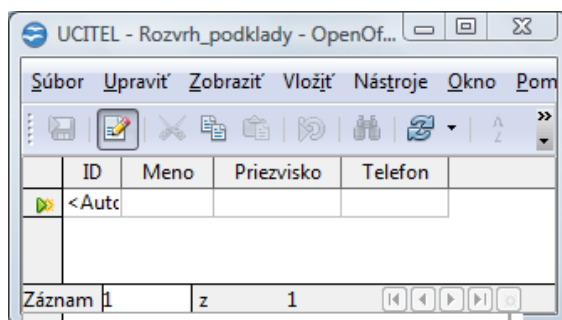
Obr. 3.3 Návrh troch tabuliek databázy

Úloha 3.2.1

Naplňte tabuľku UCITEL údajmi napr. podľa excelovskej tabuľky v stĺpci.

| Meno | Priezvisko | Telefon |
|-------|------------|------------|
| Kubko | Bača | 0907171717 |
| Janko | Hraško | |
| | Klingáčik | 0915149712 |

Dvojkliknutím na tabuľku v hlavnom okne OO Base sa tabuľka otvorí. V tomto móde je možné nielen zobrazit', resp. prezerat' jej obsah, ale údaje aj upravovat'. Keďže zatiaľ sme do tabuliek neukladali žiadne údaje, po otvorení tabuľky UCITEL sa nám ukáže prázdna tabuľka:



Obr. 3.4 Prázdna tabuľka UCITEL

Všimnime si, že v riadku, do ktorého ideme vkladat' údaje, je prvé pole (*ID*) vopred vyplnené hodnotou *<Auto>*, ktorú nie je možné zmenit' – do príslušného poľa sa počas vkladania údajov v danom riadku ani nemožno nastaviť kurzorom. Je to preto, že sme tomuto poľu nastavili vlastnosť *Automatická hodnota* na *Áno*. Systém teda

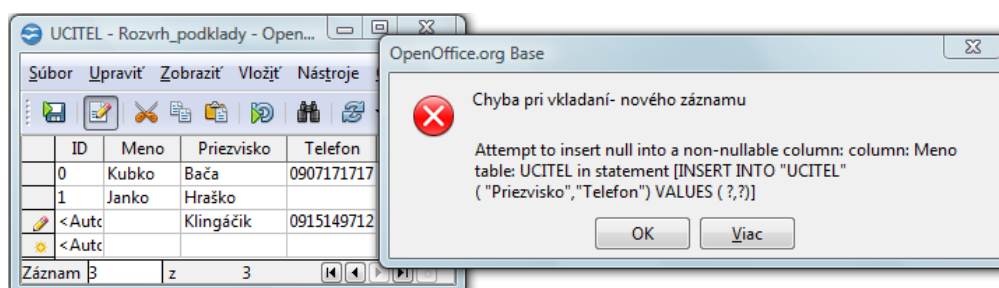
Ak by sme v budúcnosti niektorý z riadkov zmazali, tak ID žiadneho iného neskôr vytvoreného záznamu nenadobudne hodnotu vymazaného ID. Vznikne akási diera v číslovaní, zachová sa však unikátnosť hodnoty tohto poľa.

generuje unikátne celočíselné hodnoty do tohto poľa sám. Spravidla začína číslovanie od nuly a každý ďalší riadok dostáva číslo o jedna väčšie (pozri obr. 3.5). Po vyplnení údajov v jednom riadku sa stlačením klávesu Enter, Tab, alebo šípka (vpravo, resp. dolu) presúvame do ďalšieho riadku.

| ID | Meno | Priezvisko | Telefon |
|----|-------|------------|------------|
| 0 | Kubko | Bača | 0907171717 |
| 1 | Janko | Hraško | |
| | <Auto | | |

Obr. 3.5 Vkladanie údajov do tabuľky UCITEL

Údaje, ktoré vkladáme podľa zadania v úlohe 3.2.1 do tabuľky, nie sú úplné. V zázname učiteľa Janka Hraška chýba telefónne číslo a v zázname učiteľa Klingáčika chýba krstné meno. Pri zadávaní týchto údajov sa systém nespráva rovnako. Ako ilustruje obrázok 3.5, vynechanie telefónneho čísla nie je problematické, systém nám povolí prejsť na nový riadok tabuľky a zadávať ďalší záznam. Na rozdiel od toho však pri snahe prejsť na ďalší riadok po zadaní záznamu, v ktorom chýba krstné meno, systém ohlásí chybu (obr. 3.6).



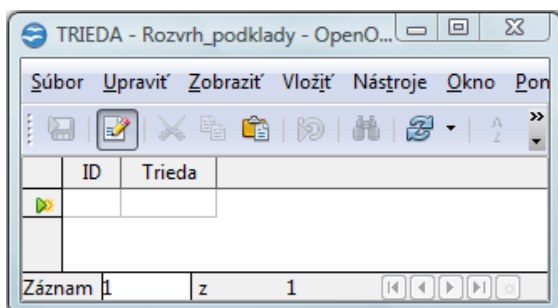
Obr. 3.6 Hlásenie chyby pri nesprávnom zápise údajov

Dôvodom sú opäť nastavenia, ktoré sme pri návrhu tabuľky spravili pre jednotlivé polia (stĺpce). Pole Meno má nastavenú vlastnosť *Požadovaná položka* na Áno, zatiaľ čo pole Telefon má túto vlastnosť nastavenú na Nie. Po zatvorení chybového hlásenia treba chýbajúci údaj v poli Meno doplniť, inak nám systém nedovolí prejsť k ďalšiemu záznamu ani uložiť tabuľku.

| | |
|--------------------|--|
| Úloha 3.2.2 | <p>Doplňte do tabuľky UCITEL</p> <ul style="list-style-type: none"> do záznamu o učiteľovi Klingáčikovi krstné meno Martin, nový záznam o učiteľovi Jožkovi Mrkvičkovi s telefónnym číslom 0949555333. |
| Úloha 3.2.3 | Naplňte tabuľku TRIEDA údajmi podľa tabuľky v stĺpci. |

| |
|--------|
| Trieda |
| I.A |
| I.B |
| III.A |
| III.B |

Po otvorení tabuľky TRIEDA vidíme, že tentokrát máme možnosť vkladat' údaje aj do poľa ID, pretože sme mu pri návrhu tabuľky vlastnosť *Automatická hodnota* nechali nastavenú na hodnotu NIE.



Obr. 3.7 Prázdna tabuľka TRIEDA s možnosťou zadávať údaje poľa ID.

Ak by sme chceli zmeniť nastavenie poľa ID na automatické generovanie hodnoty, môžeme tak spraviť ešte aj teraz. V hlavnom okne OO Base označíme tabuľku TRIEDA a prostredníctvom ponuky *Upraviť/Upraviť* ju otvoríme v režime návrhu. Ak sme popri režime návrhu mali tabuľku otvorenú súčasne aj v režime vkladania a úpravy údajov, tak sa zmeny v nastaveniach vlastností polí prejavia až po zatvorení a opätovnom otvorení tabuľky.

Ak zadávame údaje do poľa ID sami (pretože nie je zapnuté automatické zadávanie hodnoty), je potrebné si uvedomiť, že toto pole je označené ako primárny kľúč a jeho hodnoty teda musia byť jedinečné. Pri pokuse zadať rovnakú hodnotu, akú už obsahuje niektorý zo zadaných záznamov, systém vyhlási podobnú chybu, ako na obr. 3.6.

Aby sme boli konzistentní s tabuľkou UCITEL, ktorú sme už vyplňali, začneme číslovanie v poli ID aj v tabuľke TRIEDA od nuly.

Tip:
Tabuľku možno otvoriť v režime návrhu aj cez kontextovú ponuku: klikneme na ňu pravým tlačidlom myši a z kontextovej ponuky vyberieme *Upraviť*.

Úloha 3.2.4

Zmeňte nastavenie poľa ID v tabuľke TRIEDA tak, aby sa hodnoty dopĺňali automaticky.

Úloha 3.2.5

Naplňte tabuľku PREDMET údajmi podľa tabuľky v stĺpci.

| Nazov |
|----------------|
| Informatika |
| Matematika I. |
| Matematika II. |
| Matematika IV. |

V prípade, že zadané údaje treba z nejakého dôvodu zmeniť (napr. ak sme sa pri zadávaní údajov pomýlili, alebo došlo k zmene, či doplneniu údajov), príslušnú tabuľku stačí opäť otvoriť (dvojklikom v hlavnom okne OO Base) a údaje je možné upravovať.

Úloha 3.2.6

Upravte údaje v tabuľke UCITEL nasledovne:

- doplňte učiteľovi Jankovi Hraškovi telefónne číslo 0905112233,
- zmeňte krstné meno učiteľovi Klingáčikovi z „Martin“ na „Martinko“,
- zmažte záznam o učiteľovi Jožkovi Mrkvičkovi,
- doplňte ďalší záznam – pre učiteľku Ruženu Princovú, zatiaľ bez telefónneho čísla.

Tabuľku opäť uložte.

Všimnime si, že pri pridávaní ďalšieho záznamu po tom, ako bol posledný záznam zmazaný, nedostane nový záznam ID zmazaného záznamu, ale systém pokračuje v číslovaní nasledujúcim číslom.

Alternatívna úloha:

V OO Base vytvorte tabuľky pre evidenčný systém videopožičovne. Uvažujte o možnej dekompozícii tabuliek navrhnutých na strane 16. Uvažujte o nastaveniach pre primárne a cudzie kľúče. Čo zvolíme za primárny kľúč v tabuľke klient?

3.3 Formuláre

V moderných databázových systémoch sa údaje do tabuliek nevkladajú tak, ako sme to robili zatiaľ my – priamo v editovacom režime tabuľky. Mnohé systémy takúto možnosť dokonca vôbec neponúkajú. Aj kvôli pohodlnosti práce pri vkladaní údajov aj kvôli tomu, aby údaje v tabuľke nemohol meniť hocikto ľubovoľným spôsobom. Iná možnosť vkladania údajov je prostredníctvom formulárov.

OO Base dovoľuje vytvárať formuláre, pomocou ktorých možno údaje do tabuľky vkladať, prípadne pomocou ktorých možno uložené údaje zobrazovať či upravovať. Tieto formuláre nemusia byť totožné ani vzhľadom ani funkčnosťou. Formulár, ktorý bude zadaný ako formulár len na vkladanie, sa nebude dať použiť na úpravu už uložených údajov. Alebo môžeme povoliť prezerat' len niektoré údaje a iné vôbec nezobrazovať.

Možnosti tvorby formulára:



Vytvoriť formulár v režime návrhu...



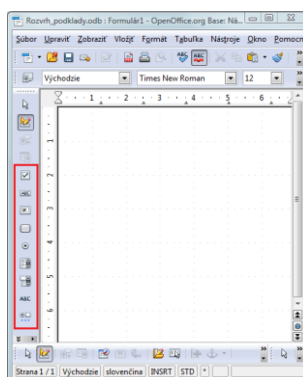
Vytvoriť formulár použitím sprievodcu...

Úloha 3.3.1 Vytvorte formulár na vkladanie údajov do tabuľky PREDMET. Formulár by mal už uložené údaje zobrazovať, ale nemal by povoliť ich meniť ani vymazávať.

Postup pri vytváraní formulára v OO Base

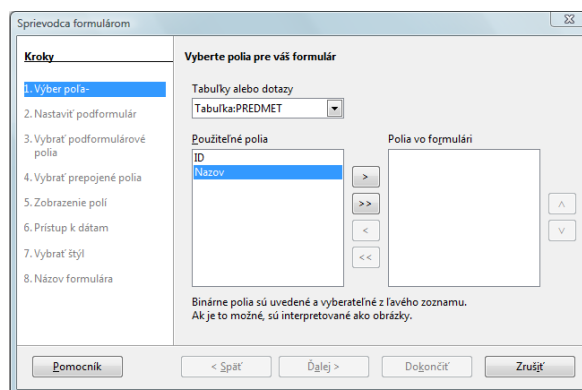
1. V hlavnom okne OO Base na paneli *Databáza* klikneme na *Formuláre*.
2. Na paneli *Úlohy* sa objavia dve možnosti vytvárania formulára. Zvolíme si jednu z nich. (Pre začiatníkov je vhodnejšia druhá možnosť – *Vytvoriť formulár použitím sprievodcu*. Zvolíme teda tú.)
3. Otvorí sa nový prázdny a nepomenovaný formulár a pred ním sa zobrazí dialógové okno *Sprievodca formulárom*, ktoré nám v 8 krokoch pomôže s vytvorením formulára. V prvom kroku – *Výber poľa* – vyberieme tabuľku, pre ktorú ideme formulár vytvárať, a polia, ktoré majú byť v ňom použité. Medzi jednotlivými krokmi sa presúvame stlačením tlačidla *Ďalej*, resp. *Späť*.

Druhý spôsob vytvorenia formulára je manuálne – v režime návrhu. V takom prípade sa nám otvorí prázdny formulár, do ktorého môžeme ukladať komponenty z ponuky.



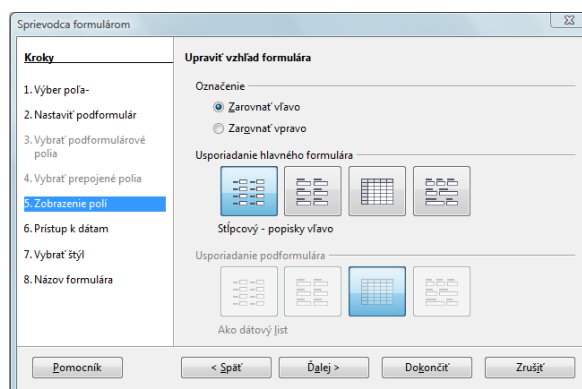
Niektoré ponúkané komponenty:

- Zaškrtávacie políčko
- Textové pole
- Formátované pole
- Tlačidlo
- Tlačidlo možností
- Pole zoznamu
- Popisné pole
- Výberové pole



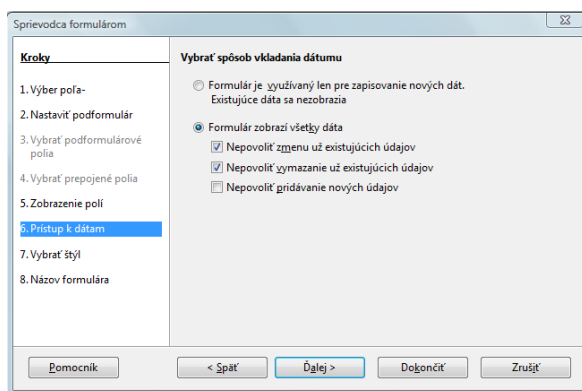
3.8 Návrh formulára pomocou sprievodcu – výber tabuľky a polí

4. Keďže v druhom kroku – *Nastaviť podformulár* – nebudeme pri jednoduchých formulároch nič nastavovať, iba stlačíme tlačidlo *Ďalej*. Tým preskočíme ďalšie dva kroky, ktoré tiež súviseli s podformulárom.



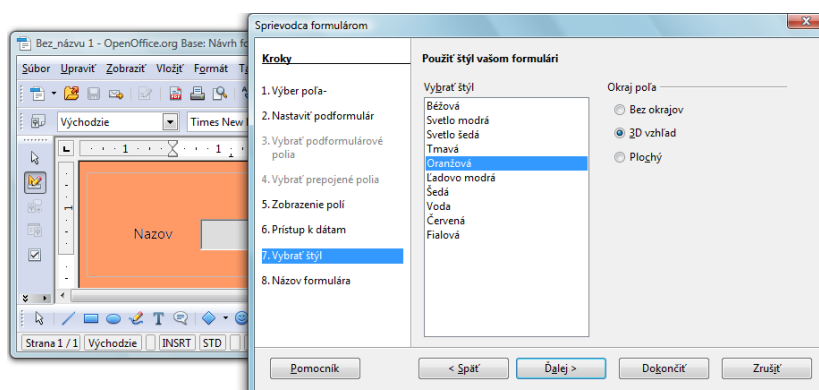
3.9 Návrh formulára – zobrazenie polí

5. V kroku päť – *Zobrazenie polí* – upravujeme vzhľad formulára. Označíme, či má byť dodržané zarovnanie vľavo alebo vpravo a kliknutím na ukážku vyberieme usporiadanie polí a ich popisov (Obr. 3.9). Tlačidlom *Ďalej* sa presunieme na ďalší krok.
6. V ďalšom kroku – *Pristup k dátam* – volíme spôsob, akým sa budú vkladať dáta, ako sa budú zobrazovať a čo bude povolené s nimi cez formulár robiť.
 - a. Ak zvolíme prvú možnosť, vznikne formulár, ktorý bude možné použiť len na vkladanie nových údajov, pričom tie, ktoré sú už v tabuľke uložené, sa vo formulári vôbec nebudú zobrazovať.
 - b. Druhú možnosť zvolíme, ak chceme, aby sa vo formulári zobrazovali všetky údaje z tabuľky. Označíme pritom, ak chceme zakázať zmenu či vymazávanie už existujúcich údajov, resp. pridávanie nových údajov.



Obr. 3.10 Návrh formulára – prístup k dátam

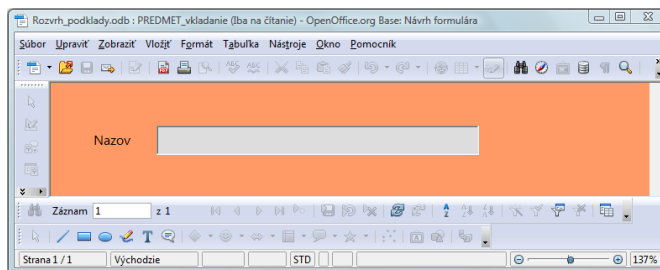
7. V siedmom kroku – *Vybrať štýl* – sa dokončuje dizajn formulára. Vyberieme farbu pozadia a spôsob zobrazovania poľa. Výber sa okamžite prejaví na vytváranom formulári.



Obr. 3.11 Tvorba formulára – výber štýlu

8. V poslednom kroku uvedieme názov formulára a označíme, či s ním chceme hneď pracovať, alebo ho chceme ďalej upravovať.

Výsledkom je nasledujúci formulár, pomocou ktorého možno vkladať nové údaje do tabuľky PREDMET.



Obr. 3.12 Formulár na vkladanie údajov o predmetoch

| Nazov |
|-------------------------|
| Informatika |
| Matematika I. |
| Matematika II. |
| Matematika IV. |
| Matematika VII. |
| Programovanie I. |
| Programovanie II. |
| Seminár z matematiky |
| Seminár z programovania |

| Trieda |
|--------|
| I.A |
| I.B |
| III.A |
| III.B |
| III.C |
| IV.A |
| IV.B |
| IV.C |
| IV.D |

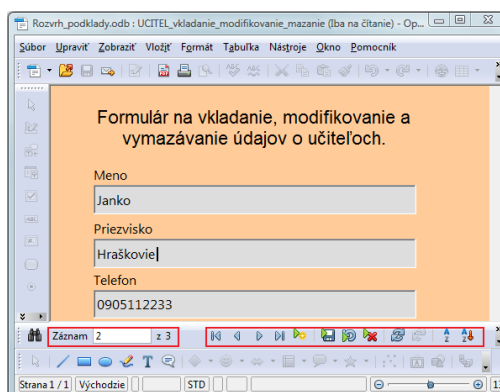
Formuláru sme zámerne nezobrazili pole ID, keďže vďaka nastaveniu automatického dopĺňania hodnôt doňho údaje nekladáme.

| | |
|--------------------|---|
| Úloha 3.3.2 | Doplňte pomocou práve vytvoreného formulára do tabuľky PREDMET chýbajúce údaje o predmetoch podľa tabuľky v stĺpci. |
| Úloha 3.3.3 | Vytvorte formulár na vkladanie údajov do tabuľky TRIEDA. Pouvažujte, či je vhodné formulár navrhnuť tak, aby bolo cezeň možné iba vkladat' nové údaje, ale nie prezerat' údaje, ktoré už sú v tabuľke uložené. Ako by sa asi pracovalo s takýmto formulárom? Prediskutujte s kolegami, kedy je vhodné použiť aký typ formulára. |
| Úloha 3.3.4 | Doplňte cez formulár do tabuľky TRIEDA chýbajúce údaje o triedach podľa excelovskej tabuľky v stĺpci. |
| Úloha 3.3.5 | Vytvorte formulár na vkladanie údajov do tabuľky UCITEL. Formulár navrhnete tak, aby bolo cezeň možné vkladat' nové údaje, prezerat' údaje, ktoré už sú v tabuľke uložené, aj ich upravovat', prípadne vymazavat'. |

Po otvorení tohto formulára sa zobrazia údaje, ktoré boli doteraz vložené do tabuľky UCITEL. V ľavej spodnej časti formulára (pozri obr. 3.11) je zachytené, koľký záznam z koľkých uložených je práve zobrazený. Po jednotlivých záznamoch sa možno pohybovať pomocou navigácie v dolnej časti okna formuláru v strede a vpravo. Nachádzajú sa tu aj ikony pre vytvorenie nového záznamu, jeho uloženie, vymazanie záznamu a podobne.

Alternatívna úloha:

Vytvorte formuláre pre *evidenčný systém videopožičovne* a naplňte ho vhodnými údajmi. Pri naplňaní môžete využiť online databázu na www.csfd.cz.



Obr. 3.13 Formulár pre tabuľku UCITEL

Úloha 3.3.6

Doplňte cez formulár do tabuľky UCITEL chýbajúce údaje o učiteľoch podľa tabuľky v stĺpci.

3.4 Vyhľadávanie a filtrovanie v tabuľke

V tejto časti sa vrátíme k vyhľadávaniu, triedeniu a filtrovaniu informácií, ktoré sme si principiálne predstavili už v druhej kapitole. Tentoraz si ukážeme, ako tieto činnosti realizovať v prostredí OO Base.

Vyhľadávanie

Úloha 3.4.1

Uvažujme najskôr o možnosti „manuálneho“ vyhľadávania. Vyhľadajte v tabuľke UCITEL všetkých učiteľov s krstným menom Ján alebo Janko. Nájdite tých, ktorí majú telefónnu predvoľbu 0949. Ako by ste postupovali pri vyhľadávaní v usporiadanej tabuľke?

Tabuľka UCITEL obsahuje malé množstvo údajov, preto je aj neautomatizované vyhľadávanie jednoduchšie. V prípade väčších databáz je však praktickejšie údaje si utriediť. OO Base nám umožňuje dva druhy triedenia:

- *podľa stĺpca* – údaje v tabuľke sa usporiadajú podľa toho poľa, kde je nastavený kurzor, ide vlastne o triedenie podľa jedného kritéria,
- *podľa kritérií* – údaje v tabuľke sa usporiadajú podľa tých polí, ktoré boli zvolené v okne *Poradie triedenia* (k dispozícii sú tri úrovne triedenia).

Zásadným rozdielom medzi usporadúvaním v tabuľkovom kalkulátore a v databázovom systéme je, že pred použitím toho nástroja nie je nutné označovať žiadne údaje, zmeny sa automaticky prejavujú v celej tabuľke.

Úloha 3.4.2

Usporiadajte tabuľku UCITEL podľa stĺpcov *Priezvisko* a *Telefón*, porovnajte, ako sa zmenilo poradie záznamov po výmene poradia kritérií.

Usporiadanie záznamov však nemusí byť jediná možnosť, ako vyhľadať konkrétny záznam. Automatizované vyhľadávanie spustíme pomocou nástroja *Nájsť záznam*. V zobrazenom okne musíme špecifikovať hľadaný výraz, oblasť ako i spôsob vyhľadávania (čo a kde hľadať). Pri vyhľadávaní ostávajú zobrazené všetky riadky bez zmeny ich usporiadania, kurzor sa len presunie na prvý výskyt hľadaného kritéria. Opätovným spustením ponuky sa kurzor presunie na ďalší výskyt.

Systém ponúka možnosť zúžiť priestor hľadania len na vybraný stĺpec (pole) alebo umiestnenie v položke záznamu (na konci, na začiatku, celé alebo hocikde).

Úloha 3.4.3

Pomocou nástroja *Nájsť záznam* vyhľadajte v tabuľke UCITEL učiteľa:


- s krstným menom Ján alebo Janko,
- ktorého meno alebo priezvisko končí na „ko“,
- ktorého telefónne číslo začína predvoľbou 0949.


Samostatná úloha

| Meno | Priezvisko | Telefon |
|----------|------------|------------|
| Kubko | Bača | 0907171717 |
| Janko | Hraško | 0905112233 |
| Martinko | Klingáčik | 0915149712 |
| Ruženka | Princová | 0949491194 |
| Lenka | Šípková | 0903033000 |
| Maťko | Valach | 0949134134 |

Možnosti usporadúvania:

 triediť podľa kritérií

 vzostupné triedenie podľa aktuálneho stĺpca


 zostupné triedenie podľa aktuálneho stĺpca


 nástroj *Nájsť záznam*

Preštudujte si, aké rôzne možnosti hľadania poskytuje okno *Hľadanie záznamu*.


Okno nám ponúka napr. aj možnosť hľadania *Obsah poľa je NULL*, čo znamená nájsť polia s nevyplnenou (prázdnu) hodnotou.

Nástroje filtrovania:

 Automatický filter

 Všeobecný filter

 Použiť filter

 Zrušiť filter

Na vyfiltrované údaje môžeme aplikovať nástroj triedenia.

Ikony informujúce o stave filtrovania:



filter nemá zvolené kritéria



filter má nastavené kritéria ale je vypnutý



filter je zapnutý

Operátory známe z počítačového prostredia (=, <, >, <=, >=, <>) sa inak správajú, ak je v podmienke testovaný číselný, a inak, ak je testovaný textový dátový typ. Textové premenné sa vyhodnocujú na základe lexikografického usporiadania.

'10' < '2'

'Martin' > 'Maroš'

'Martin' < 'Martinko'

Filtrovanie

Posledným spôsobom vyhľadávania v záznamoch tabuľky je ich filtrácia. Z hľadiska prehľadnosti je toto pravdepodobne najvhodnejší spôsob, keďže zobrazené zostanú len tie záznamy, ktoré spĺňajú zvolené kritéria a nedôjde len k nastaveniu kurzora na prvý výskyt.

Najjednoduchšou formou je nástroj *Automatický filter*. Jeho spustením dôjde k filtrácii údajov, pričom za kritérium sa berie obsah bunky s pozíciou kurzora. Ak sme v tabuľke UCITEL boli nastavení v poli Priezvisko na zázname Hraško, tak sa vyfiltrujú všetci s týmto priezviskom. *Automatický filter* sa nedá kombinovať s ďalšími kritériami. Zmena stavu ikony filtra nás informuje, či je zvolené kritérium filtrovania a či je filter zapnutý alebo vypnutý. Filter môžeme kedykoľvek vypnúť a zobraziť kompletný obsah tabuľky.

Väčšie možnosti v nastavovaní kritérií nám ponúka *Všeobecný filter* skladajúci sa z troch základných nastavení:

- **Názov poľa** – určuje, ktoré pole záznamu je pri filtrovaní porovnávané.
- **Podmienka** – určuje, aký operátor sa použije pri porovnaní obsahu záznamu s požadovanou hodnotou.
- **Hodnota** – určuje hľadaný obsah.

Pre každé filtrovanie môžeme nastaviť hlavné kritérium a dve vedľajšie, spojené pomocou logickej spojky AND alebo OR.

- **AND (a)** – hlavné a vedľajšie kritérium musia platiť súčasne,
- **OR (alebo)** – musí platiť aspoň jedno z kritérií.

Samostatná úloha

Obr. 3.14 Všeobecný filter pre tabuľku UCITEL.

Úloha 3.4.4

Pomocou všeobecného filtra zobrazte záznamy, kde:

- meno učiteľa je Janko,
- priezvisko učiteľa je Hraško alebo meno je Matko,
- priezviská učiteľov sú v abecede pred priezviskom Klingáčik,

Hodnoty nemusíme uzatvárať v apostrofoch, systém si ich doplní sám.

* zastupuje ľubovoľný počet znakov

? zastupuje jediný znak

Problém môže nastať pri požiadavkách typu: nájsť priezviská končiace na „ová“. Možnosť hľadať hodnotu na konci záznamu (začiatku, hocikde), ako nám ponúkal nástroj *Nájsť záznam* v prípade *Všeobecného filtra*, neexistuje. Miesto toho je tu však operátor *like*, ktorý v kombinácii so zástupnými znakmi predstavuje silný nástroj filtrovania. Filter by sme tak mohli nastaviť na Priezvisko like *ová.

| Meno | Priezvisko | Telefon |
|---------|------------|------------|
| Ruženka | Princová | 0949491194 |
| Lenka | Šípková | 0903033000 |

Obr. 3.15 Výsledok filtrovania podľa požiadavky Priezvisko like *ová

Ak by bola otázka položená záporne, kritérium by bolo `Priezvisko not like *ová`. `Not` predstavuje negáciu, respektíve doplnok k množine spĺňajúcej podmienku `Priezvisko like *ová`.

| Meno | Priezvisko | Telefon |
|----------|------------|------------|
| Kubko | Bača | 0907171717 |
| Janko | Hraško | 0905112233 |
| Martinko | Klingáčik | 0915149712 |
| Matko | Valach | 0949134134 |

Obr. 3.16 Výsledok filtrovania podľa požiadavky `Priezvisko not like *ová`

Úloha 3.4.5

Zistite, aké výstupy v tabuľke UCITEL získame s nasledujúcimi kritériami *Všeobecného filtra*:

- Meno LIKE *ko
- Meno LIKE ???ko
- Telefon LIKE 0949*
- Telefon NOT LIKE 0949* AND Priezvisko LIKE *a*

Nájdite všetkých učiteľov,

- ktorí nemajú uvedený telefónny kontakt,
- ktorých meno alebo priezvisko obsahuje samohlásku „a“.

Samostatná úloha

Pokiaľ neexistuje žiadny prirodzený primárny kľúč, prípadne by bol príliš zložitý, často sa používa ako primárny kľúč číslo automaticky generované databázovým systémom.

3.5 Prepájanie tabuliek

V tomto kroku si ukážeme, ako vytvoriť súvis medzi tabuľkami pomocou väzieb na základe spoločných atribútov. Intuitívne sme tento problém predstavili už v kapitole venovanej databázam v tabuľkovom kalkulátore, preto si len zhrňme, čo zatiaľ poznáme.

Každá tabuľka má mať pole (polia), v ktorom sa nachádzajú jedinečné, v rámci stĺpca sa neopakujúce, hodnoty. Už vieme, že toto pole voláme *primárny kľúč*. Typickým príkladom pre jedinečné údaje osôb sú rodné číslo, číslo občianskeho preukazu alebo odtlačok prsta, pre automobily je to ŠPZ/EČV, pre elektroniku sériové číslo.

Tento údaj sa môže využiť na identifikáciu v iných tabuľkách, kde ho nazývame *cudzí kľúč*. Zjednodušene povedané, na základe cudzieho kľúča môžeme vystopovať všetky informácie o jeho nositeľovi – cudzí kľúč odkazuje na svojho majiteľa. Pomocou cudzích kľúčov sa teda možno z jednej tabuľky odkazovať na inú tabuľku, resp. iné tabuľky a vytvoriť tak súvis – v databázovej terminológii *vzt'ah*, resp. *relácia* – medzi údajmi, ktoré sú v týchto tabuľkách uložené.

Vzhľadom k tomu, že tabuľky prepájame na základe relácií sa tento typ databáz nazýva *relačné databázy*.

Úloha 3.5.1

Diskutujte s akými jedinečnými údajmi sa stretávame napríklad v požičovniach DVD.

Porozmýšľajte o ďalších príkladoch, kde je používanie primárneho a cudzieho kľúča evidentné.

Diagramy rôznych typov, použiteľných aj pre výučbu môžeme navrhovať vo voľne dostupnom softvéri **DIA**
<http://live.gnome.org/Dia>.

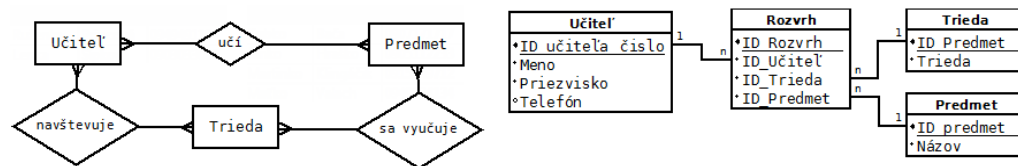
Profesionálni analytici navrhujú databázy postupne v modeloch:

- konceptuálny,
- logický,
- fyzický.

Zvládnuť tento proces si však vyžaduje množstvo nových informácií, poznať metodológiu modelovania, analytické myslenie, proces normalizácie a pod.

Strana pre pokročilých

V úvode druhej kapitoly sme popísali tabuľku, ktorú sme neskôr museli rozdeliť (dekomponovať). Vysvetlíme si preto aspoň základné princípy pokročilejšieho prístupu. V praxi sa väčšinou nepristupuje hneď k tvorbe tabuliek v konkrétnom softvéri, najskôr sa vytvára návrh štruktúry databáz v diagramoch. V modeloch nezávislých na použítom databázovom softvéri, čím sa snaží neskoršiemu rozdeľovaniu tabuliek predísť.



Obr. 3.17 Ukážka návrhu dátovej štruktúry v tzv. entito-relačnom modeli (vľavo) a v relačnej schéme (vpravo)

Uvedené diagramy, hlavne entito-relačný, vznikajú na základe diskusií a požiadaviek. Diagramy ich v prehľadnejšej forme popisujú. Zjednodušene povedané vizualizujú, čo by sa malo v systéme neskôr vytvoriť vo forme tabuliek. Na vyučovaní je vhodné diskutovať o tom so žiakmi a svoje nápady si zapisovať, možno aj priamo do diagramov.

Po určení entít – objektov, o ktorých má databáza uchovávať informácie (tie sa neskôr transformujú do základných tabuliek) sa pristupuje k určovaniu vzťahov. Treba vedieť, ktoré časti systému a ako sú navzájom prepojené. Napr. učiteľ učí niekoľko predmetov a ten istý predmet môže vyučovať niekoľko učiteľov. V triede sa vyučujú rôzne predmety a zároveň platí, že rovnaký predmet sa môže učiť v rôznych triedach. Učiteľ navštevuje viacero tried, pričom jedna trieda môže byť navštevovaná viacerými učiteľmi. Slovami jeden, každý, viacero, rôzne apod. určíme tzv. **kardinalitu** vzťahu. Poznáme tri základe vzťahu [1].

1:1

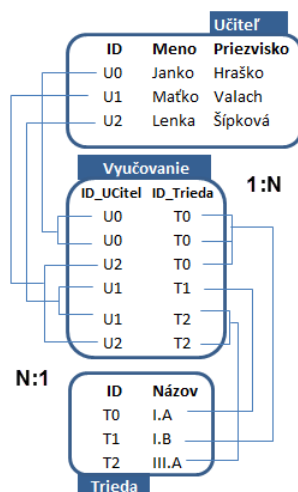
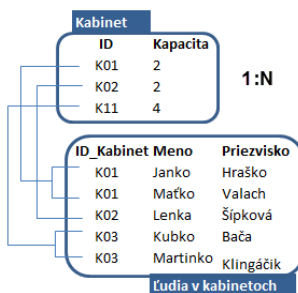
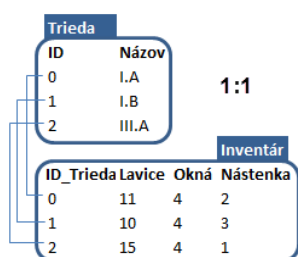
Jeden záznam v jednej tabuľke zodpovedá jednému záznamu v druhej tabuľke. Jednotlivé údaje sú vlastne spojené priamo. Predpokladajme, že by sme chceli o triede mať aj bližšie informácie ako len jej názov, napr. počet lavíc, okien a násteniek. Otázkou je, načo vytvárať novú tabuľku, ktorú previažeme pomocou atribútu ID_Trieda, keď všetky údaje môžu byť súčasťou tabuľky Trieda? Odpoveď? Druhá tabuľka môže slúžiť na odľahčenie prvej, najmä ak sa s údajmi v druhej tabuľke nepracuje často, prípadne v nej môžu byť skryté citlivé údaje.

1:N

Jednému záznamu v prvej tabuľke zodpovedá viacero záznamov v druhej tabuľke. Ide o najčastejší typ vzťahov. Platí, že v tabuľke na strane „1“ je definovaný primárny kľúč, ktorý sa v tabuľke na strane „N“ využíva ako cudzí kľúč. Ak by sme uvažovali, že jeden učiteľ môže mať pridelený len jeden (1) kabinet, v jednom kabinete však môže byť aj viac (N) učiteľov, tak sme práve pomenovali vzťah 1:N.

N:M

Niekoľkým záznamom v prvej tabuľke zodpovedá viacero záznamov v druhej tabuľke. Väzba typu M:N je však len logický vzťah, ktorý medzi fyzickými databázovými tabuľkami neexistuje, nie je možné ju zrealizovať. Preto je tento typ väzby nahradený dvoma väzbami typu 1:N, a to prostredníctvom novej tabuľky. V našom príklade napríklad platí, že učiteľ navštevuje viacero tried, pričom jedna trieda môže byť navštevovaná viacerými učiteľmi (teda typický vzťah N:M).



Ako riešenie predchádzajúceho problému doplníme do databázy *spojovaciú tabuľku* a prostredníctvom nej zavedieme väzby medzi tabuľkami na základe kľúčov. Podobne by sme mohli popísať aj ostatné vzťahy medzi učiteľmi, triedami a predmetmi.

Úloha 3.5.2

Vytvorte novú spojovaciú tabuľku ROZVRH, ktorá bude slúžiť na prepojenie tabuliek UCITEL, PREDMET a TRIEDA pomocou cudzích kľúčov.

Áké polia (stĺpce) bude teda obsahovať tabuľka ROZVRH? Každý jej záznam musí odkazovať na konkrétny (jedinečný) záznam v tabuľkách UCITEL, PREDMET a TRIEDA, ich primárne kľúče preto použijeme ako cudzie kľúče v novej tabuľke ROZVRH. Vytvoríme tu pre ne polia ID_Ucitel, ID_Predmet a ID_Trieda. Všetky tri budú povinnými položkami novej tabuľky. Typ poľa sa musí zhodovať s typom, ktorý majú kľúče nastavené v pôvodných tabuľkách, v našom prípade je to typ INTEGER.

Pri pokuse o uloženie nás však systém upozorní na chýbajúci primárny kľúč. Položme si otázku: Môže byť učiteľ na rozvrhu viackrát priradený k nejakej triede na ten istý predmet? Každá trojica učiteľ – predmet – trieda je jedinečná (nepotrebujeme a ani nechceme mať uložené dva rovnaké záznamy o tom, že napr. učiteľ Janko Hraško vyučuje predmet Matematika IV. v triede III.C) a je teda dobrým kandidátom na skupinový primárny kľúč. Navyše, na tabuľku ROZVRH sa nebudeme odvolávať v žiadnej inej tabuľke, čiže jej primárny kľúč nemusí byť nutne tvorený len jedným atribútom.

| | Názov poľa | Typ poľa |
|---|------------|------------------------|
| 🔑 | ID_Ucitel | Celé číslo [INTEGER] |
| 🔑 | ID_Predmet | Celé číslo [INTEGER] |
| 🔑 | ID_Trieda | Celé číslo [INTEGER] |

Obr.3.18 Návrh tabuľky ROZVRH obsahujúcej tri cudzie kľúče, ktoré zároveň tvoria skupinový primárny kľúč tabuľky.

Spojovacia tabuľka môže okrem atribútov prepojených tabuliek obsahovať ľubovoľný počet ďalších polí, ktoré bližšie špecifikujú záznam v novej tabuľke. V našom prípade to môže byť napr. deň a hodina vyučovania, miestnosť a pod.

Úloha 3.5.3

Doplňte do tabuľky ROZVRH vhodné údaje. V akej forme ich musíme zadávať?

Keďže všetky polia (stĺpce) sú číselného typu musíme tabuľku naplniť číslami.

| ID | Nazov | ID | Trieda | ID | Meno | Priezvisko | Telefon | ID_Ucitel | ID_Predmet | ID_Trieda |
|----|-------------------------|----|--------|----|----------|------------|------------|-----------|------------|-----------|
| 0 | Informatika | 0 | I.A | 0 | Kubko | Bača | 0907171717 | 1 | 3 | 4 |
| 1 | Matematika I. | 1 | I.B | 1 | Janko | Hraško | 0905112233 | 3 | 7 | 5 |
| 2 | Matematika II. | 2 | III.A | 2 | Martinko | Klingáčik | 0915149712 | 7 | 9 | 9 |
| 3 | Matematika IV. | 3 | III.B | 3 | Ruženka | Princová | 0949491194 | 1 | 3 | 4 |
| 4 | Matematika VII. | 4 | III.C | 4 | Lenka | Šípková | 0903033000 | 2 | 5 | 3 |
| 5 | Programovanie I. | 5 | IV.A | 5 | Matko | Valach | 0949134134 | | | |
| 6 | Programovanie II. | 6 | IV.B | | | | | | | |
| 7 | Seminár z matematiky | 7 | IV.C | | | | | 3 | 7 | 8 |
| 8 | Seminár z programovania | 8 | IV.D | | | | | 0 | 7 | 7 |

Obr. 3.19 Ukážka naplnených tabuliek PREDMET, TRIEDA, UCITEL a ROZVRH

Z tabuľky ROZVRH vieme kedykoľvek zistiť, kto, kde a čo vlastne učí. Podľa prvého riadku vieme, že Janko Hraško učí v III.C Matematiku IV. Z posledného záznamu vieme, že Kubko Bača učí v IV.C predmet Seminár z matematiky. Ako ale vyhodnotiť tretí a štvrtý riadok?

Primárny kľúč v relačnej databáze môže byť definovaný aj ako skupina atribútov. V každom prípade musí platiť, že žiadne dva záznamy sa nemôžu zhodovať vo všetkých týchto atribútoch.

V OO Base nastavíme viacerým poliam označenie primárny kľúč tak, že najprv označíme príslušné riadky, a potom im naraz pridáme príznak kľúča.

Alternatívna úloha:

Vytvorte spojovaciú tabuľku pre *evidenčný systém videopožičovne*.

Príklad spojovacej tabuľky pre výpožičku DVD.

| DVD | | |
|------|----------------|-----------|
| ID | Názov | Žáner |
| dvd0 | Drak a dračice | rozprávka |
| dvd1 | Ja a ty | komédia |
| dvd2 | Janko Hraško | rozprávka |

| Výpožička | | | | |
|-----------|-----------|------|------|------|
| ID_DVD | ID_Klient | Od | Do | Cena |
| dvd0 | 0 | 10.1 | 11.1 | 1 |
| dvd2 | 0 | 10.1 | 11.1 | 1 |
| dvd1 | 1 | 10.1 | 11.1 | 1 |
| dvd1 | 0 | 11.1 | 13.1 | 2 |
| dvd2 | 2 | 11.1 | 12.1 | 1 |

| Klient | | |
|--------|---------|------------|
| ID | Meno | Priezvisko |
| 0 | Edko | Guľočka |
| 1 | Helenka | Krásna |
| 2 | Ferko | Mrkvička |

| ID_uciteľa | ID_predm | ID_triedy |
|------------|----------|-----------|
| 1 | 3 | 4 |
| 3 | 7 | 5 |
| 0 | 4 | 7 |
| 5 | 0 | 5 |
| 2 | 5 | 3 |
| 1 | 1 | 0 |
| 3 | 2 | 1 |
| 1 | 3 | 2 |
| 4 | 0 | 5 |
| 1 | 7 | 5 |
| 3 | 7 | 7 |
| 1 | 6 | 3 |
| 4 | 0 | 6 |
| 1 | 6 | 4 |
| 2 | 5 | 4 |
| 2 | 8 | 7 |
| 4 | 0 | 7 |
| 1 | 0 | 7 |
| 3 | 7 | 6 |
| 5 | 0 | 2 |
| 5 | 0 | 3 |
| 3 | 7 | 8 |
| 0 | 7 | 7 |

Úloha 3.5.4

Na obrázku 3.15 sú v tabuľke ROZVRH nesprávne vyplnené údaje, nájdite ich. Dá sa predísť takejto situácii? Naplňte tabuľku korektnými údajmi, napr. podľa obrázka vpravo.

Tabuľka ROZVRH je naplnená, stále však nie je prepojená so zdrojovými tabuľkami. To nám v predchádzajúcom príklade umožnilo zadávať aj také údaje, ktoré neodkazujú na žiadny záznam, teda v základných tabuľkách k nim neexistuje hodnota primárneho kľúča.

Zadefinovaním vzťahu – *relácie* – zabezpečíme, aby správnosť vkladáných alebo editovaných dát kontroloval systém automaticky. Hovoríme, že systém zabezpečuje *referenčnú integritu*. Referenčná integrita je vlastnosť databázy, ktorá zaručuje, že pri používaní systému budú cudzie kľúče vždy odkazovať iba na existujúce riadky tabuliek alebo nebudú ukazovať nikam (budú mať hodnotu null), ak je to prípustné [5].

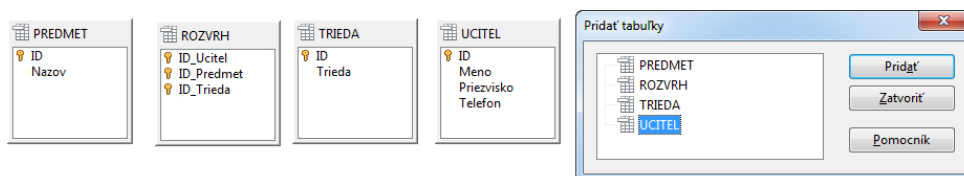
Úloha 3.5.5

Vytvorte vzťahy medzi tabuľkou ROZVRH, UCITEL, PREDMET a TRIEDA.

Pri vytváraní vzťahov medzi tabuľkami nie je nutné, aby polia, ktoré sú vo vzťahu mali rovnaké pomenovanie, musia však mať rovnaké dátové typy.

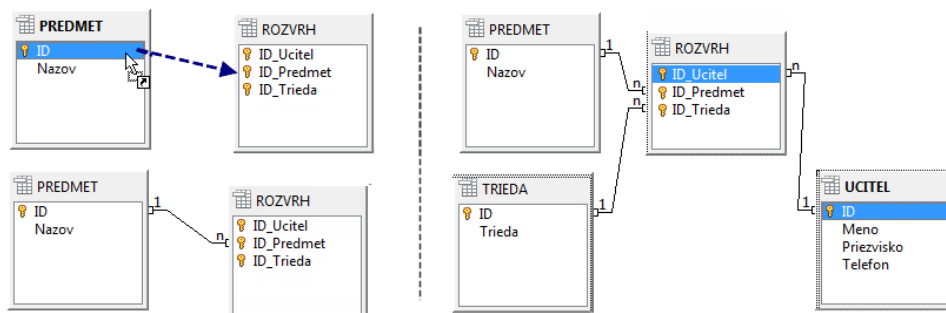
Postup vytvorenia vzťahu medzi tabuľkami v OO Base

1. V hlavnom okne OO Base vyberieme ponuku *Nástroje/Vzťahy*. Pokiaľ sme ešte v databáze nedefinovali žiadny vzťah, automaticky sa nám zobrazí dialógové okno *Pridať tabuľku*, v ktorom pridáme tabuľky určené na prepájanie. Tie sa nám zobrazia len ako zoznam polí bez konkrétnych hodnôt.



Obr. 3.20 Pridanie tabuliek v okne Vzťahy

2. Rozmiestnime ich tak, aby spojovacia tabuľka, v našom prípade ROZVRH, bola v strede, čím zvýšime prehľadnosť.
3. Samotné relácie sa definujú tak, že pretiahneme požadované pole z jednej tabuľky do druhej. Polia, ktoré majú byť prepojené, týmto navzájom „previažeme“. Po previazaní vznikne spojnica medzi poľami. Automaticky sa pre vzťah nastaví referenčná integrita, ktorá je naznačená číslom 1 na strane primárneho a písmenom N na strane cudzieho kľúča (1:N).



Obr. 3.21 Vytvorenie vzťahu ťahaním polí

Okno *Pridať tabuľku* zobrazíme cez ponuku *Vložiť/ Pridať tabuľku* alebo aj cez ikonu



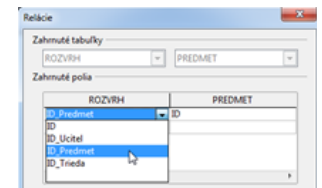
Krok 3 zopakujeme pre všetky požadované prepojenia.

Alternatívna úloha:

V evidenčnom systéme videopožičovne vytvorte vzťahy medzi tabuľkami.

4. Pri ťahaní polí sa občas stane, že dôjde k nesprávnemu prepojeniu. Vtedy vzťah buď odstránime (označíme ho a stlačíme kláves *Delete*) a opäť vytvoríme, alebo ho upravíme (dvojklikom na vzťah otvoríme okno *Relácie*, v ktorom správne určíme zahrnuté polia). Cez okno *Relácie* môžeme vytvárať aj nový vzťah. Vtedy musíme zvoliť zahrnuté tabuľky aj polia.

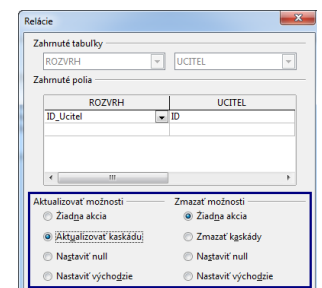
Okno *Relácie* zobrazíme cez ponuku *Vložiť/Nová relácia* alebo cez ikonu



Doménová integrita je taká vlastnosť systému, ktorá zaručuje, že všetky polia databázy budú nadobúdať len prípustné hodnoty. Doménová integrita kontroluje aj podmienky, ktoré musí hodnota spĺňať, napr. či môže ostať pole prázdne.

Excel umožňuje meniť typ hodnoty priebežne, dokonca môžeme v jednom stĺpci evidovať hodnoty rôznych typov. V databázových systémoch má každé pole, stĺpec, definovaný presný typ.

Účelom referenčnej integrity je nielen zabrániť vzniku osamotených záznamov, ale aj zabezpečiť synchronizáciu odkazov tak, aby neexistovali položky, ktoré odkazujú na neexistujúce záznamy.



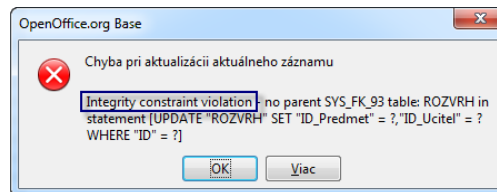
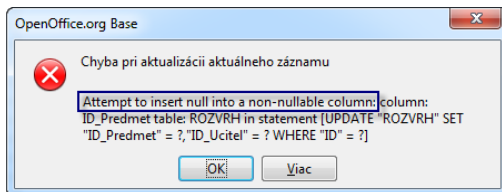
Úloha 3.5.6

Skúste do tabuľky ROZVRH vložiť nasledujúce údaje.

| ID_Ucitel | ID_Predmet | ID_Trieda |
|-----------|------------|-----------|
| 5 | 2 | 3 |
| 9 | | 5 |

Ako systém teraz reaguje na nesprávne vstupy?

Prvá chyba, na ktorú nás systém upozorní, je nevyplnenie poľa *ID_Predmet* v druhom riadku. S podobným problémom sme sa už stretli pri naplňaní tabuľky *UCITEL* neúplnými záznamami. Takže vieme, že ho spôsobuje to, že pri návrhu štruktúry tabuľky bolo vyplnenie všetkých troch polí určené ako povinné. Hovoríme, že systém stráži **doménovú integritu**. Keďže systém stráži aj referenčnú integritu databázy, tak pri zadávaní dát v druhom riadku zistí, že neexistuje učiteľ s *ID* = 9. Až keď budú všetky polia záznamu správne a platné, tak ich systém uloží.



Obr. 3.22 Upozornenie systému na porušenie doménovej a referenčnej integrity.

Máme teraz systém zabezpečený aj z druhej strany? Čo sa stane, ak zmením hodnotu primárneho kľúča pre učiteľa, o ktorom mám záznam v rozvrhu?

Úloha 3.5.7

V tabuľke ROZVRH máme uloženú informáciu o učiteľovi s *ID* = 5. Otvorte tabuľku UCITEL, vyhľadajte záznam s týmto *ID* a skúste ho zmeniť na inú, jedinečnú hodnotu. Ako systém reaguje na tieto zmeny?

Keď máme medzi tabuľkami vytvorené vzťahy, tak nám systém nedovolí meniť hodnoty primárnych kľúčov, pokiaľ sa na ne stále odkazuje v iných tabuľkách. Inak by bola porušená referenčná integrita. Je ale táto situácia správna? Čo ak budeme chcieť opodstatnene zmeniť hodnoty primárnych kľúčov? Riešenie je veľmi jednoduché. Systému vopred povieme ako má reagovať ak budeme hodnoty primárnych kľúčov meniť alebo mazať. Tieto nastavenia vykonáme v okne *Vzťahy – Relácie*, samostatne pre úpravy a samostatne pre mazanie. Nás zaujíma najmä možnosti tzv. **kaskádových úprav** [2].

Kaskádová aktualizácia súvisiacich polí – pri zmene hodnoty primárneho kľúča systém automaticky aktualizuje všetky polia, ktoré naň odkazujú. Databáza ostane v konzistentnom stave.

Kaskádové odstránenie súvisiacich záznamov – keď odstránime záznam obsahujúci primárny kľúč, systém automaticky odstráni všetky záznamy naň odkazujúce.

Každý vzťah v databáze môže mať inú kombináciu nastavení pre aktualizáciu a mazanie.

- Aktualizovať možnosti Zmazať možnosti
- Žiadna akcia Žiadna akcia
 - Aktualizovať kaskádu Zmazať kaskádu
 - Nastaviť null Nastaviť null
 - Nastaviť východzie Nastaviť východog

Úloha 3.5.8

Upravte vzťah medzi tabuľkami UCITEL a ROZVRH, tak aby sme mohli kaskádovo aktualizovať súvisiace polia, ale aby nebolo možné údaj o učiteľovi zmazať. Vyskúšajte funkčnosť tohto nastavenia.

Doteraz sme v tabuľke ROZVRH pracovali len s číslami. Napĺňanie takýmto spôsobom však môže byť veľmi prácne. Stále totiž musíme zisťovať hodnoty primárnych kľúčov pre jednotlivé polia. Ideálne by bolo, keby sme mohli pracovať v režime, kde miesto hodnoty ID sa bude zobrazovať konkrétne priezvisko, názov triedy, alebo názov predmetu. Riešením by mohlo byť vytvorenie formulára s podformulárom, kde by sme miesto priameho vkladania vyberali hodnoty zo zoznamu.

Obdobne môžeme spraviť formulár, kde pre každú triedu zobrazíme učiteľa a predmet, ktorý v nej učí. Prípadne pre každý predmet zobrazí triedu a učiteľa, ktorý ho učí.

Tip:
V systéme Moodle nájdete návod, ako takýto formulár vytvoriť.

Úloha 3.5.9

pokročilá

Vytvorte nový formulár, ktorý bude slúžiť na zobrazovanie informácií o učiteľoch. Vložte do neho podformulár, v ktorom sa zobrazí zoznam tried a predmetov, ktoré v nich daný učiteľ učí.

3.6 Vyhľadávanie v databáze

V kapitole 3.4 sme si ukázali ako usporiadať a filtrovať záznamy v jednej tabuľke. Získané zručnosti nám však nepomôžu, pokiaľ potrebujeme vyhľadávať údaje, ktoré sú uložené nie iba v jednej, ale vo viacerých tabuľkách, prípadne ak vyhľadáваме podľa kritérií, ktoré si overujeme vo viacerých tabuľkách.

V českej literatúre sa *query* prekladá ako *dotaz*. Tento výraz je často prevzatý aj v slovenskej literatúre. Aj v slovenskej lokalizácii OO Base sa používa výraz *dotaz*, preto ho budeme v materiáli používať aj my (spisovný výraz pre query je však *dopyt*).

SQL (Structured Query Language)

Poznáme tri základné typy dotazov:

- výberové,
- agregáčne (suma, min, max, priemer, zoskupenie),
- aktualizáčne (insert, delete, update).

V takomto prípade použijeme nástroj databázového softvéru, ktorý dokáže podľa zadaných kritérií hľadané informácie z databázy získať. Úlohu vyhľadať príslušné informácie mu predkladáme vo forme *dotazu*, v ktorom zachytíme všetky kritériá, podľa ktorých chceme vyhľadávať. Na tvorbu dotazov sa používa tzv. *dotazovací jazyk SQL*. Dotazy si môžeme predstaviť ako požiadavky (otázky), ktoré kladieme systému, a na ktoré on odpovedá. Preto ich musíme konštruovať tak, aby nám systém rozumel, a aby nám jeho odpoveď skutočne poskytla riešenie.

Dotaz môžeme zo začiatku formulovať slovne:

- a. Aký je menný zoznam učiteľského zboru?
- b. Ktoré predmety vyučuje Ruženka v IV.C?
- c. Aké predmety má v rozvrhu III.A?
- d. Ktorí učitelia vyučujú v III.A?
- e. Koľko predmetov sa učí v jednotlivých triedach?
- f. Aký veľký je učiteľský zbor?
- g. Zmeň názov Seminár z programovania na Programovanie v Delphi.
- h. Vlož záznam o novom učiteľovi s menom Ferdinand Múdry.
- i. Zmaž záznamy o učiteľovi Martinkovi Klingáčikovi.

Ako si môžeme všimnúť, štruktúra dotazov je rôzna. Niektoré sa pýtajú na istú podmnožinu všetkých záznamov (a. – d.), iné sa pýtajú na početnosť (e. – f.), no a tie posledné sú skôr príkazmi ako otázkami (g. – i.). Najčastejšie používané sú však tie, pomocou ktorých údaje filtrujeme. Nazývame ich *výberové dotazy*. Skôr ako vytvoríme prvý dotaz, vysvetlíme si ako taký výberový dotaz funguje.

Dotaz: Aký je kontakt na učiteľa Janka Hraška?

Postup hľadania odpovede:

1. V tabuľke sa „označia“ polia, ktoré chceme zahrnúť do odpovede, v našom prípade nás zaujíma **Telefon**.
2. V tabuľke sa „označia“ záznamy, v ktorých je **Meno = 'Janko'** a **Priezvisko = 'Hraško'**, v našom prípade je to jeden záznam.
3. Vytvorí sa prienik medzi označenými poľami a záznamami. Prienik sa vráti ako výsledok.

Všimnime si, že sme dostali len jeden údaj – telefónne číslo, bez mena aj priezviska. Dôvod zistíme, keď sa pozrieme na to, ako sme formulovali dotaz. Chceli sme vedieť len kontakt. Dotaz by sme preto mohli upraviť nasledovne:

Dotaz: Vyber z tabuľky UCITEL meno, priezvisko a telefón učiteľa s menom Janko a priezviskom Hraško.

1

| Meno | Priezvisko | Telefon |
|----------|------------|------------|
| Kubko | Bača | 0907171717 |
| Janko | Hraško | 0905112233 |
| Martinko | Klingáčik | 0915149712 |
| Ruženka | Princová | 0949491194 |
| Lenka | Šípková | 0903033000 |
| Maťko | Valach | 0949134134 |

2

| Meno | Priezvisko | Telefon |
|----------|------------|------------|
| Kubko | Bača | 0907171717 |
| Janko | Hraško | 0905112233 |
| Martinko | Klingáčik | 0915149712 |
| Ruženka | Princová | 0949491194 |
| Lenka | Šípková | 0903033000 |
| Maťko | Valach | 0949134134 |

3

| Meno | Priezvisko | Telefon |
|------|------------|------------|
| | | 0905112233 |
| | | |
| | | |
| | | |
| | | |

Úloha 3.6.1

Diskutujte, ako by ste preformulovali dotazy na predchádzajúcej strane do takej podoby, aby bolo jasné **odkiaľ** sa majú údaje vyberať, **aké polia** chceme zobrazit' a **aké sú kritériá** na filtrovanie záznamov.

Pouvažujte, čo je vlastne výsledkom dotazu.

Výsledkom výberového dotazu je nová tabuľka, ktorá však **nie je súčasťou databázy**. Vznikla „vyskladáním“ záznamov z pôvodných databázových tabuliek. Je preto jasné, že výsledky výberového dotazu, sa nedajú editovať, čo môže, ale nemusí byť vždy výhodou.

Skonštruujeme teda požiadavku tak, aby ju už vedel systém správne vyhodnotiť. Niektoré databázové softvéry (OO Base, MS Access) ponúkajú nástroj, v ktorom je možné dotazy tvoriť nie priamo v jazyku SQL (textový zápis), ale intuitívnejším spôsobom, v grafickom režime. Začnime dotazom s jednou tabuľkou.

Úloha 3.6.2

Skonštruujte nasledujúce dotazy:

1. Zobrazte všetky informácie o učiteľoch.
2. Zobrazte o učiteľoch všetky informácie okrem ID a usporiadajte ich podľa priezviska zostupne.
3. Zobrazte priezvisko a meno učiteľa s menom Maťko, Martin, alebo Martinko.
4. Zobrazte meno a priezvisko učiteľov, ktorých telefónny kontakt začína na 0949.
5. Zobrazte priezvisko a telefónne číslo učiteľov, ktorých priezvisko končí na „ová“ a telefónne číslo nezačína na 0949.
6. Zobrazte priezvisko a telefónne číslo učiteľov, ktorých číslo začína na 0949 a zároveň neobsahuje cifru 3.

V hlavnom okne databázy vyberieme ponuku **Dotazy** a v časti **Úlohy** zvolíme **Vytvorit' dotaz v režime návrhu...** Podobne ako pri návrhu vzťahov sa zobrazí ponuka so zoznamom tabuliek, na ktoré môžeme požiadavku smerovať. Keďže všetky dotazy v zadaní sú smerované na učiteľa, vyberieme tabuľku UCITEL.

V OO Base sa cez ponuku **Dotazy**, teda v grafickom režime, dajú vytvárať len dotazy výberové.

Samotný dotaz konštruujeme v spodnej časti okna. Skôr ako si podrobne jednotlivé

Alternatívna úloha:

V evidenčnom systéme pre videopôžičovňu zobrazte:

- všetky filmy žánru sci-fi alebo fantasy.
- všetky filmy, ktoré majú v názve „láska“.
- všetkých klientov na ktorých neexistuje kontakt.
- všetkých klientov zoradených podľa abecedy.



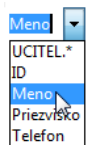
Dotazy

V OO Base sa cez ponuku **Dotazy** dajú vytvárať len dotazy výberové.

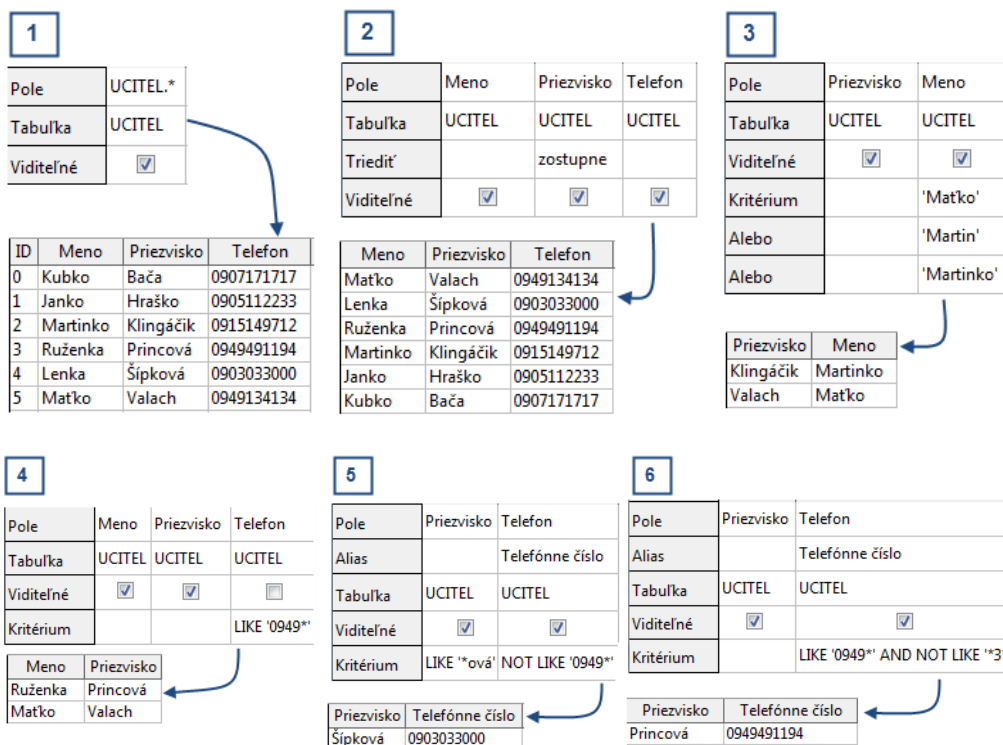
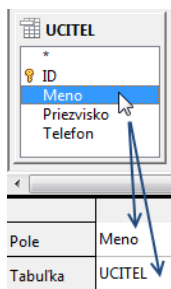
Pridať tabuľku môžeme aj neskôr cez ikonu

polia popíšeme, skúsme skonštruovať dotazy z úlohy 3.6.2. Dotazy uložíme. Kedykoľvek ich môžeme spustiť, upraviť alebo použiť pri ďalších úlohách.

Hodnoty polí môžeme vyberať zo zoznamu,



alebo pridávať kliknutím na konkrétne pole do tabuľky v okne s reláciami.



Obr. 3.23 Ukážka štruktúry dotazu v grafickom režime a výsledok vo forme tabuľky

Teraz si môžeme zhrnúť, aké atribúty môžeme pre dotaz v grafickom zobrazení OO Base špecifikovať:

Už pri návrhu dotazu si môžeme skontrolovať výsledok. Dotaz spustíme cez ikonu (F5). Zobrazí sa náhľad výsledku. Náhľad môžeme zavrieť kedykoľvek zobrazit' alebo skryt' *Zobrazit'/Náhľad* (F4).

- **Pole** – Určuje, ktoré pole sa má zahrnúť do výsledku. Ak chceme zahrnúť všetky polia, použijeme znak hviezdčky (*). Vo výsledku sa zachováva poradie, v akom sme polia definovali.
- **Alias** – Určuje, ako sa má pole do výsledku dotazu premenovať (pozri 4 a 5).
- **Tabuľka** – Určuje, z ktorej tabuľky je vybrané pole.
- **Triediť** – Určuje, či je pre pole zvolené kritérium usporiadania (pozri 2).
- **Viditeľné** – Určuje, či sa má pole aj zobrazit', alebo sa použije napríklad len ako kritérium výberu, bez zobrazenia poľa (pozri 4).
- **Kritérium** – Určuje podmienku filtrovania pre vybrané pole. Podmienky na jednom riadku musia platiť súčasne (AND), pri podmienkach pod sebou stačí platnosť jednej z nich (OR) (pozri 3, 4 a 5). V kritériách môžeme používať zástupné znaky (používali sme ich už pri filtrovaní dát v tabuľke, pozri úloha 3.4.5).

Podrobnejšiemu popisu základných príkazov jazyka SQL sa venuje doplnujúci materiál v elektronickej verzii umiestnený v kurze.

Všetky vytvorené dotazy (aj tie, ktoré boli skonštruované v grafickom režime) si možno nechať zobrazit' ako zápis v SQL. Dá sa tak urobiť cez ponuku *Zobrazit'/Pohľad na návrh prepínača*. Zápis je možné zobrazit' aj pomocou ikony



Teraz, keď už vieme ako konštruovať dotazy nad jednou tabuľkou, zamerajme sa na vyhľadávanie vo viacerých navzájom prepojených tabuľkách. V prostredí návrhu dotazu necháme zobrazit' všetky tabuľky vzťahu.

Úloha 3.6.3

Skonštruujte nasledujúce výberové dotazy:

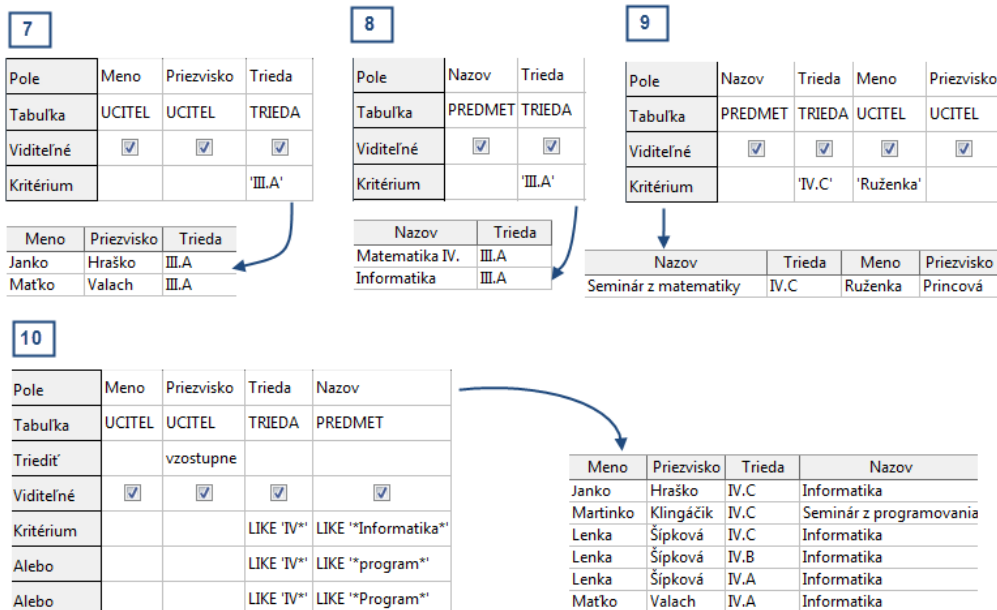
7. Ktorí učitelia vyučujú v III. A?
8. Ktoré predmety má v rozvrhu III. A?
9. Ktoré predmety vyučuje učiteľka Ruženka v IV. C?
10. Kto vyučuje informatické predmety vo IV. ročníku?

Použijeme tabuľku ROZVRH s údajmi z úlohy 3.5.4

Alternatívna úloha:

V evidenčnom systéme videopožičovne zistíte:

- ktoré médiá boli vypožičané v poslednom týždni kalendárneho roku 2010,
- kto si požičal ľubovoľnú komédiu alebo romantický film.



Obr. 3.24 Ukážka štruktúry dotazu v grafickom režime a výsledok vo forme tabuľky

Úloha 3.6.4

Samostatne skonštruujte nasledujúce dotazy:

- Ktoré predmety a v ktorých triedach učí učiteľ s menom Kubko?
- Zobrazte priezviská učiteľov, ktorých telefónne číslo končí párnou číslicou.
- Zobrazte informácie o učiteľoch, ktorí majú ako druhé písmeno mena samohlásku „a“.
- Zobrazte učiteľov, ktorí učia semináre a aj triedy, v ktorých tieto semináre učia.

Samostatná úloha

Poznámka:

Pokročilejšímu čitateľovi odporúčame doplňujúci materiál, v ktorom sa zaoberáme podrobnejším popisom jednotlivých príkazov jazyka SQL a vysvetlením významu a spôsobu konštrukcie rôznych typov dotazov. Tento materiál možno nájsť v elektronickej podobe v kurze 2DB - Úvod do databáz v systéme Moodle (<http://dvui.ccv.upjs.sk/kurzy/course/view.php?id=26>).

Čo sme sa naučili

Zopakovali sme si pojmy atribút, záznam, tabuľka a spojili sme ich s prostredím konkrétneho databázového systému, kde sme si precvičili **návrh** tabuliek, ich naplnenie údajmi, tvorbu **formulárov**, i prácu s údajmi v tabuľke. Naučili sme sa používať **primárny kľúč** i **cudzí kľúč**, definovať vzťahy a **prepájať tabuľky**. Vysvetlili sme si pojmy referenčná integrita a doménová integrita, zaviedli pojem SQL a naučili sme sa konštruovať **dotazy**.

4. Informačné systémy

Na záver nám ostáva nájsť odpoveď na otázku: „Na čo nám teda databázy slúžia?“. Samozrejme, otázku sme si mohli položiť už na začiatku, avšak teraz, keď už máme predstavu, ako to všetko funguje, lepšie pochopíme, čo sa deje okolo nás a kde máme databázy hľadať. Odpoveďou je, že v rôznych formách nájdeme databázy všade, ich využitie je rôzne.

Predchádzajúce časti nám priblížili, ako sa dajú vytvárať jednoduché databázy. Sú to väčšinou databázy určené pre tzv. **izolované počítače** [6], kde databáza ako aj program, v ktorom sa spracúvajú, sú na jednom počítači. Takéto riešenie však nie je vhodné, pokiaľ sú údaje, ktoré v nich uchováваме, zdieľané väčšou skupinou používateľov. Ak sú pri riešení istého problému informácie získané z databázy dôležité aj pri rozhodovaní o ďalších krokoch, tak je evidentné, že databáza musí byť súčasťou **informačných systémov**, ktoré si v skratke predstavíme.

Informačný systém [6] je súbor ľudí, metód a technických prostriedkov zabezpečujúcich zber, prenos, uchovávanie, analýzu, distribúciu a prezentáciu údajov určených pre potreby poskytovania informácií mnohým používateľom rôznych typov. Informačný systém je teda automatizovaný celok hardvéru, softvéru a dát, kde významnú úlohu hrá personál, ktorý s ním pracuje. Jeho základnou úlohou je zabezpečiť dostatok relevantných, správnych a presných informácií v požadovanej forme na prípravu rozhodnutí.

Informačný systém nemusí byť vždy nutne automatizovaný pomocou počítačov, napriek tomu si väčšina ľudí predstaví práve počítačový on-line systém. Každý z nás už v rôznej miere využil ich možnosti, ako napr. prihlásenie sa do informačných systémov bánk (internet banking), edukačných portálov (Moodle DVUi), elektronické objednávanie tovaru, vyhľadávanie dostupnosti služieb apod. Pojmom informačné systémy totiž môžeme zastrešiť rôzne formy dochádzkových, stravovacích, vzdelávacích, knižnično-informačných, účtovníckych, či podnikových systémov určených na riadenie vnútropodnikových procesov.

Cieľom informačného systému je zlepšiť výkonnosť ľudí (riadenia, vzdelávania, vyhľadávania a pod.) prostredníctvom využívania informačných technológií.

Z hľadiska obsahu automatizácie poznáme [3]:

- *Informačné systémy evidenčného typu* – Boli prvou vývojovou etapou informačných systémov s najjednoduchšími formami uplatnenia výpočtovej techniky. Takéto informačné systémy umožňujú napr. automatizovanú evidenciu zásob, hmotného investičného majetku, miezd, objednávok, faktúr, aktualizáciu evidenčných záznamov, tlačenie evidenčných a štatistických zostáv a pod.
- *Systémy na podporu rozhodovania* – Sú počítačovo realizované informačné systémy poskytujúce manažérom potrebné informácie vo vhodnej forme, využívajú okrem evidenčnej zložky aj rozličné modelové prostriedky a techniky umožňujúce automatizovať nielen informačné, ale aj niektoré rozhodovanie procesy.

Informačný systém je realizovaný nejakým aplikačným programom, väčšinou vytváraným alebo upravovaným podľa požiadaviek koncového zákazníka. Existujú rôzne firmy s vlastným softvérom stavaným na mieru organizácií. Firmy dohliadajú nielen na informačné systémy, ale aj dodržiavanie legislatívnych požiadaviek. Mnohé softvéry vznikajú len prekladom, či úpravou svetovo overených informačných systémov.

Budovanie ako i udržiavanie informačného systému nie je lacná záležitosť, preto sa väčšina organizácií rozhoduje pre budovanie systémov na základe webovej technológie. Čoraz viac sa však môžeme stretnúť s využívaním otvorených softvérov, personalizovaných vzhľadovo, ako i obsahovo. Medzi voľné softvéry, ktoré môžeme považovať za jednoduché informačné systémy pre riadenie vzdelávacieho procesu,

Informačný systém na riadenie výučby v projekte Ďalšie vzdelávanie učiteľov v predmete informatika
<http://dvuii.ccv.upjs.sk/kurz/>.

Informačné systémy Univerzity Konštantína filozofa v Nitre

INFORMAČNÉ SYSTÉMY

| |
|-----------------------------------|
| Vzdelávací portál (EDU) |
| Knižničný informačný systém (KIS) |
| Evidencia záverečných prác (EZP) |
| Evidencia projektov (Proj) |
| WebManager (WeM) |
| Správa dokumentov (DOC) |
| Objednávkový systém (oSys) |
| Stravovací systém (Str) |
| Dochádzkový systém (DIS) |

Prehľad knižnično-informačných systémov v knižniciach SR
<http://www.infolib.sk/index/podstranka.php?id=116>

Informačný systém Masarykovej univerzity
<http://is.muni.cz/>

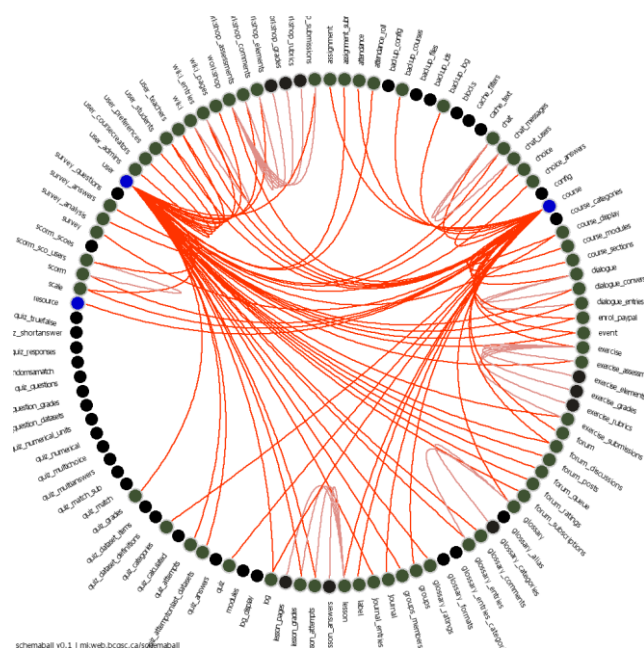
Informačný systém na odhalovanie plagiatov záverečných prác.
<http://theses.cz/>
<http://www.crzp.sk/>

Svetovými spoločnosťami poskytujúce služby v oblasti informačných systémov sú okrem iných aj SAP, ORACLE a IBM

patria rôzne LMS systémy (nám známy zástupca je MOODLE).

V súvislosti s informačnými systémami sa môžeme stretnúť aj s pojmom **intranet** [3]. Intranet je privátna sieť v rámci organizácie postavená na štandardoch internetu. Na rozdiel od internetu je však intranet prístupný len ohraňenej skupine interných používateľov, väčšinou zamestnancov jednej firmy. Vo všeobecnosti ním označujeme použitie internetovských služieb, pre vnútro podnikové siete a informačné systémy. Intranet môže byť realizovaný v rámci jednej budovy, v skutočnosti však môže pokrývať celú zemeguľu (interná je táto sieť iba logicky).

Údaje potrebné na svoju činnosť ukladajú informačné systémy do databáz, ktoré sú ich základným stavebným prvkom. Údaje musia byť zobraziteľné v zrozumiteľnej forme, a viditeľné len pre skupinu ľudí s príslušným stupňom oprávnenia (napr. v internetovej žiackej knižke má učiteľ sprístupnené iné ako žiak; v systéme elektronickej evidencie dochádzky zamestnanec vidí iné ako jeho nadriadený a pod.). Úrovne oprávnenia sú chránené heslom, opäť uloženým v databáze. Štruktúra databáz informačných systémov býva zložitá, navzájom prepojená množstvom relácií.



Obr. 4.1 Schema Ball – Vizualizácia štruktúry databázy Moodle

Predstavme si, že by sme chceli mať informácie o istom produkte, údaje o ňom však nie sú umiestnené na pracovnej stanici, ale niekde na vzdialenom serveri. Pošleme teda do systému požiadavku a systém nám vráti celú databázu, v ktorej si informáciu nájdeme. Po sieti by sa prenášalo množstvo zbytočných údajov, ktoré by ju mohli zahltiť (takýto model prenosu informácií sa nazýva *file server/pracovná stanica*) [6].

Predchádzajúci spôsob práce je neefektívny aj preto, že údaje sú uložené na inom mieste, ako sú spracúvané. Ideálne by bolo ponechať spracovanie údajov tam, kde sú uložené, a naopak výstup pre používateľa generovať tam, kde sa nachádza používateľ [6]. Aplikačný program (informačný systém) sa preto rozdeľuje na dve monolitické časti:

- **server** (host) – zaisťuje spracovanie údajov, ktoré sú na ňom uložené,
- **klient** – zaisťuje tzv. používateľské rozhranie.

Služby typu klient/server sú založené na vzťahu podriadený – nadriadený. Klient ako zákazník formuluje dotaz, resp. požiadavku, ktorú cez sieťové médium zasiela serveru [4]. Ten ako poskytovateľ služby ju spracuje a výsledok oznámi späť cez sieť

LMS je skratka pre anglické pomenovanie Learning Management System.

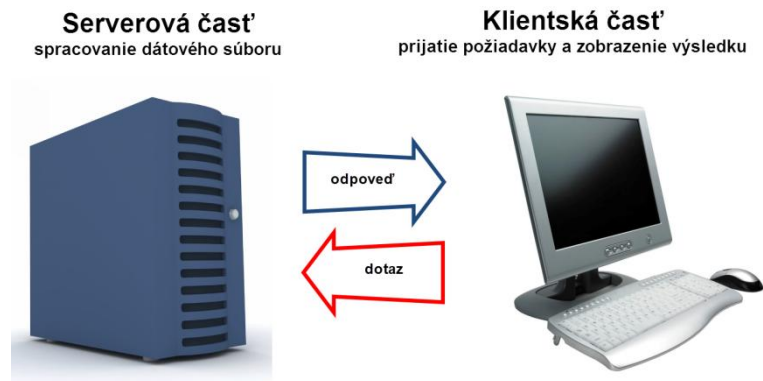
Rôzne typy LMS sa často využívajú na vysokých školách, najmä kvôli cenovej dostupnosti (niektoré sú zadarmo) a možnosti doprogramovať ich podľa vlastných predstáv. Open Source informačný systém však na druhú stranu pre väčšinu podnikových manažérov znamená bezpečnostné riziko, preto sa vo firemnej praxi používajú skôr komerčné produkty.

Na obrázku je vidieť rozsiahlu databázovú štruktúru LMS Moodle, ktorý môžeme zaradiť do kategórie Informačné systémy pre riadenie vzdelávacieho procesu.

Zdroj: <http://moodle.org/mod/forum/discuss.php?d=3935>

© Martin Langhoff

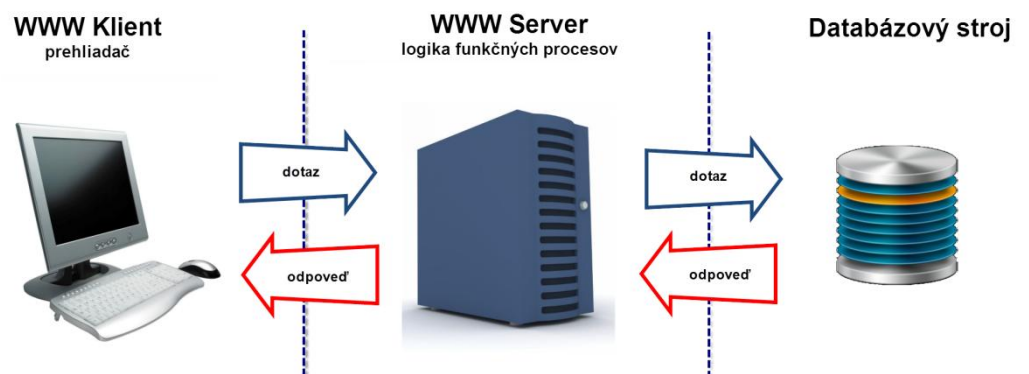
klientovi. Klient a server si teda posielajú údaje vo forme dotazov a odpovedí na ne. Čím lepšie postaví klient svoj dotaz, tým menej údajov bude potrebných na prenos. Práve príkladom z takéhoto prostredia sme náš modul začínali.



Obr. 4.2 Riešenie požiadavky v režime klient/server

Súčasnú riešenie klient/server rozdeľuje aplikačný program na viac vrstiev. Typickým príkladom je *trojvrstvová architektúra*, kde sú funkcie informačného systému rozdelené do troch častí [6].

- *prezentačné* (používateľské rozhranie, zber dotazov, prezentácia výsledkov),
- *aplikačné* (vlastná logika aplikácie),
- *správa údajov* (úložisko dát, databázové operácie).



Obr. 4.3 Trojvrstvová architektúra klient/server

Tieto tri vrstvy sú zvyčajne vyvíjané a udržiavané ako nezávislé moduly, väčšinou na rozdielnych platformách. Vďaka tomuto deleniu možno jednoducho vymeniť určitú časť systému, napr. úložisko údajov alebo používateľské rozhranie bez veľkých zásahov do celkovej štruktúry systému.

Informačné systémy sú už dnes všeobecne vnímané ako neoddeliteľná súčasť života. Musíme si však uvedomiť, že údaje, ktoré sú v ich databázach uchovávané, môžu byť veľmi osobné. Výmenou údajov medzi klientom a serverom môže taktiež dôjsť k výmene súkromných údajov. A tak je veľmi dôležité pri registráciách do takýchto systémov, najmä však verejných, prečítať si všetky podmienky registrácie, ako i sledovať, či systém komunikuje s používateľom v bezpečnom režime (protokol https).

Čo sme sa naučili

Naučili sme sa rozpoznať **informačný systém** ako pomocníka, ktorý používateľovi uľahčuje riadenie, distribúciu, či vyhodnotenie rôznych procesov. Vieme, že údaje, z ktorých v informačných systémoch získavame informácie, sú uložené v databázach. Používateľ (**klient**) komunikuje so systémom (**server**) cez požiadavky, na ktoré systém v príslušnej forme odpovedá. V skutočnosti systém v zrozumiteľnej forme zobrazí podmnožinu údajov uložených v inak často neprehľadných tabuľkách.

Čo sme sa naučili v tomto module

Zhrnutie

Účastník vzdelávania po úspešnom absolvovaní tohto modulu rozumie pojmom databáza, databázový systém, tabuľka, atribút, primárny kľúč, väzba, dotaz a pod. Rozumie princípom tvorby databáz a základnej práce s nimi, chápe dôležitosť správneho návrhu databázy a jeho vplyvu na efektívnosť spracovania údajov a získavania informácií na základe rôznych kritérií výberu. Dokáže v konkrétnom databázovom softvéri vhodne navrhnúť a vytvoriť tabuľky aj formuláre, naplniť tabuľky údajmi, prepojiť ich a skonštruovať dotazy podľa zadaných kritérií. Rozumie pojmu informačný systém, úlohe databázy v ňom a spôsobu komunikácie používateľa s ním.

Preverenie výstupných vedomostí

Preverenie vykonajú lektori priamo počas výučby na základe výsledkov realizovaných praktických cvičení.

Na základe preverenia výstupných vedomostí v zmysle predchádzajúceho odseku bude predmet hodnotený stupnicou absolvoval/neabsolvoval.

Literatúra a použité zdroje

- [1] Rachačová, H. (2005) *Access 2003 pro školy* 1. díl, Učebnice databázového programu Computer Media 2005, Kniha, čeština, 112 stran, ISBN 80-86686-44-2
- [2] Drlík, M., Skalka, J. (2007) *Tvorba databáz v prostredí MS Access* Nitra, UKF, 130 s., ISBN 978-80-8094-218-2.
- [3] <http://www.its4you.sk/informacne-systemy/informacne-systemy.php>
- [4] <http://fria.fri.uniza.sk/~vapo/vyuka/poc-siet.htm>
- [5] <http://hornad.fei.tuke.sk/predmety/ivpp/5datab.htm>
- [6] Klimeš, C., Drlík, M. (2007) *Informačné systémy*, Nitra, UKF, 223 s., (Prírodovedec, č. 246), ISBN 978-80-8094-115-4.

Tento študijný materiál vznikol ako súčasť národného projektu Ďalšie vzdelávanie učiteľov základných škôl a stredných škôl v predmete informatika v rámci Aktivity „Vzdelávanie nekvalifikovaných učiteľov informatiky na 2. stupni ZŠ a na SŠ“.

Autori © RNDr. Zuzana Kubincová, PhD.
Mgr. Martin Cápaj, PhD.
PaedDr. Andrea Hrušecká, PhD.
RNDr. Ľubomír Salanci, PhD.

Názov Ďalšie vzdelávanie učiteľov základných škôl a stredných škôl v predmete informatika

Podnázov Úvod do databáz

Študijný materiál prešiel recenzným pokračovaním.

Recenzenti doc. RNDr. Stanislav Krajčí, PhD.
PaedDr. Miloslava Sudolská, PhD.

Počet strán 40

Náklad 300 ks

Prvé vydanie, Bratislava 2011

Všetky práva vyhradené.

Toto dielo ani žiadnu jeho časť nemožno reprodukovat' bez súhlasu majiteľa práv.

Vydal Štátny pedagogický ústav, Pluhová 8, 830 00 Bratislava, v súčinnosti s Univerzitou Pavla Jozefa Šafárika v Košiciach, Univerzitou Komenského v Bratislave, Univerzitou Konštantína Filozofa v Nitre, Univerzitou Mateja Bela v Banskej Bystrici a Žilinskou univerzitou v Žiline

Vytlačil BRATIA SABOVCI, s r.o., Zvolen

ISBN 978-80-8118-093-4