**„EU peníze školám“**

**Projekt DIGIT – digitalizace výuky na ISŠTE Sokolov**

**reg. č. CZ.1.07/1.5.00/34.0496**

|  |  |
| --- | --- |
| **III/2 Inovace a zkvalitnění výuky prostřednictvím ICT** | **VY\_32\_INOVACE\_1\_6\_07** |
| Název vzdělávacího materiálu | Základy SQL – přejmenování sloupce v tabulce |
| Jméno autora | Ing. Bohuslava ČEŽÍKOVÁ |
| Tématická oblast | Databázové systémy |
| Vzdělávací obor | 68-46-M/01 Veřejnosprávní činnost |
| Předmět | Informační a komunikační technologie |
| Ročník | 2., 3. |
| Rozvíjené klíčové kompetence | Kompetence k učení* posoudí vlastní pokrok a určí překážky či problémy bránící učení;
* kriticky zhodnotí výsledky;

Kompetence k řešení problémů* osvědčené postupy aplikuje při řešení obdobných problémových situací;
* sleduje vlastní pokrok při zdolávání problémů.
 |
| Průřezové téma | Informační a komunikační technologie |
| Časový harmonogram | 1 vyučovací hodina |
| Použitá literatura a zdroje | Databáze, modely dat, relační algebra, SŘBD, SQL, normální formy [online]. [cit. 2013-02-12]. Dostupné z: http://is.muni.cz/th/143327/fi\_b/bc\_prace\_vitxx.txtEntity-relationship model. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2013-02-12]. Dostupné z: http://cs.wikipedia.org/wiki/Entity-relationship\_modelSKŘIVAN, Jaromír. SQL - tvorba tabulek. In: [online]. [cit. 2013-02-12]. Dostupné z: http://interval.cz/clanky/sql-tvorba-tabulek/ |
| Pomůcky a prostředky | pc, internet, sw |
| Anotace | Databázové systémy, základy jazyka SQL – vazby mezi tabulkami. |
| Způsob využití výukového materiálu ve výuce | Výklad, pracovní list |
| Datum (období) vytvoření vzdělávacího materiálu | 2/2013 |

*Tento výukový materiál je plně v souladu s Autorským zákonem (jsou zde dodržována všechna autorská práva). Pokud není uvedeno jinak, autorem textů
a obrázků je Ing. Bohuslava Čežíková.*

**Přejmenování sloupce**

Pro přejmenování sloupce nemáme v SQL žádný konkrétní příkaz. Abychom přejmenovali sloupec, je potřeba provést několik kroků: Nejprve musíme vytvořit sloupec nového názvu, pak zkopírovat hodnoty ze starého sloupce do nového, a nakonec smažeme ten sloupec se starým názvem. Uvedu příklad. Vezměme si naši tabulku ZAMĚSTNANEC včetně sloupce FUNKCE. Rozhodli jsme se, že sloupec FUNKCE přejmenujeme na ZAŘAZENÍ. Posloupnost příkazů SQL, jak toho docílit je následující:

ALTER TABLE zamestnanec

 ADD (zarazeni VARCHAR(5) DEFAULT 'žádná')

UPDATE zamestnanec

 SET zarazeni = funkce

ALTER TABLE zamestnanec

 DROP funkce

První příkaz vytvoří nový sloupec ZARAZENI, druhý příkaz slouží pro aktualizaci hodnot ve sloupcích. Příkaz UPDATE budu probírat až v příštím díle, zatím si vystačíme s tím, že budeme vědět, že tento příkaz zkopíruje hodnoty ze sloupce FUNKCE do sloupce ZARAZENI. Poslední příkaz pak smaže sloupec FUNKCE. Tímto postupem docílíme přejmenování sloupce. Jenom malá poznámka na závěr. Nový sloupec se bude vypisovat až jako poslední, tzn., že se neuchová původní pořadí sloupců, jaké bylo, což z hlediska vyhledávání informací v databázi vůbec nevadí.

**Pracovní list**

Zapište SQL příkaz pro vytvoření tabulky ZAMESTNANEC. Tabulka bude obsahovat sloupce:

*cislo* typu celé číslo, počet cifer nastavte na 4,

*jmeno* typu řetězec znaků s pohyblivou délkou, nastavte délku 10 znaků,

*prijmeni* typu řetězec znaků s pohyblivou délkou, nastavte délku 20 znaků,

*rodne*\_*cislo* typu řetězec znaků s pohyblivou délkou, nastavte délku 11 znaků,

*funkce* typu celé číslo, počet cifer nastavte na 5,

*vzdelani* typu řetězec znaků s pohyblivou délkou, nastavte délku 30 znaků,

*ridicke*\_*opravneni* typu řetězec znaků s pohyblivou délkou, nastavte délku 3 znaky a bude mít výchozí hodnotu „NE“.

Sloupec „*rodne*\_*cislo*“ při vkládání dat nesmí zůstat prázdný, primární klíč je nastaven na sloupec „*cislo*“.

Přejmenujte sloupec „*ridicske*\_*opravneni*“ na „*ridicak*“.