

10. Heslá, LIMIT, funkcie v MySQL, potlačenie nežiadúceho výstupu

24.4.2006

Obsah

- Heslá a používatelia
- Obmedzenie výsledku dotazu – LIMIT
- Funkcie v MySQL
- Potlačenie nežiadúceho výstupu

Heslá

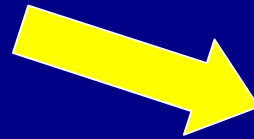
- Autentifikácia používateľov
- Tabuľka používateľov – podobná iným tabuľkám
- Ako ukladať heslá v databáze?
 - Ako čistý text
 - Šifrované (**PASSWORD, MD5**) – z hesla sa vytvorí hexadecimálny reťazec, jednosmerné šifrovanie

Úprava systému prihlasovania

- Vytvoríme tabuľku používateľov, ktorí môžu pracovať v systéme (**skuska_pouzivatelia**)
- V pôvodnej tabuľke **skuska** zrušíme stĺpce **meno**, **kruzok**, ktoré budú v novej tabuľke **skuska_pouzivatelia**
- Pri prihlasovaní budeme kontrolovať používateľov podľa novej tabuľky

Tabuľka používateľov

Pole	Typ
<u>uid</u>	smallint(6)
prihlasmeno	varchar(20)
heslo	varchar(50)
meno	varchar(20)
priezvisko	varchar(30)
kruzok	varchar(10)



Pole	Typ
<u>id</u>	smallint(6)
uid	smallint(6)
id_termin	tinyint(4)

Pole	Typ	Funkcia	Nulový	Hodnota
uid	smallint(6)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
prihlasmeno	varchar(20)	<input type="text"/>	<input type="checkbox"/>	roman
heslo	varchar(50)	MD5	<input type="checkbox"/>	roman
meno	varchar(20)	<input type="text"/>	<input type="checkbox"/>	Roman
priezvisko	varchar(30)	<input type="text"/>	<input type="checkbox"/>	Hrušecký
kruzok	varchar(10)	<input type="text"/>	<input type="checkbox"/>	4m-i

Vytváranie a kontrola používateľov

- Zatiaľ bude realizované len cez prostredie phpMyAdmin
- INSERT INTO skuska_pouzivatelia (prihlasmeno, heslo) VALUES ('roman', **MD5**('roman'));
- SELECT * FROM skuska_pouzivatelia WHERE prihlasmeno='meno' AND heslo=**MD5**('heslo');
- \$sql = "SELECT * FROM skuska_pouzivatelia WHERE prihlasmeno='\" . \$_POST['meno'] . \"' AND heslo=**MD5**('\" . \$_POST['heslo'] . \"')";

uid	prihlasmeno	heslo	meno	priezvisko	kruzok
1	admin	21232f297a57a5a743894a0e4a801fc3	Administrátor	systemu	fmfi
2	roman	b179a9ec0777eae19382c14319872e1b	Roman	Hrušecký	4m-i

Zmeny prihlasovania

- Overovanie cez databázu
- Výpis celého mena prihláseného používateľa (pri každom zobrazení stránky s prihlásením/odhlásením)
- Zapamätáme si nielen prihlasovacie meno, ale kompletne údaje o používateľovi do SESSION, aby sme ich nemuseli vždy zisťovať z databázy
- **Zmena** prihlášky, výpisu, úpravy, zmazania

Zmena pri výpise

- Zmena dotazu – údaje vyberáme z 3 tabuliek
- `SELECT * FROM skuska, skuska_terminy, skuska_pouzivatelia WHERE skuska.id_termin = skuska_terminy.id_termin AND skuska.uid = skuska_pouzivatelia.uid`

Zmena prihlášky

- Môže sa prihlásiť len prihlásený používateľ
- Meno a krúžok sa nebudú zadávať, ale sa vypíšu podľa údajov v SESSION
- Meno a krúžok sa nemusia vkladať do tabuľky, lebo sa vkladá len **UID** prihláseného používateľa

Zmena zrušenia prihlásenia

- V prvom kroku netreba nič meniť – dotaz sa nemení.
- **Problém:** Prihlásený používateľ môže zmazať prihlášky iných používateľov
- **Zmena:** pri výpise sa odkaz na zrušenie prihlášky zobrazí len pri prihláške, ktorá patrí prihlásenému používateľovi (**+kontrola** podľa `$_SESSION['uid']` – či **zadané id prihlášky v adrese** patrí prihlásenému používateľovi)

Zmena úpravy prihlásenia

- V prvom kroku: drobná zmena kontroly, dotazu a formulára (vyplývajúce z predchádzajúcich zmien)
- **Skrytý problém:** Prihlásený používateľ môže upravovať prihlášky iných používateľov (**ak zmení parameter id v adrese**)
- **Zmena:** pri výpise sa odkaz na úpravu prihlášky zobrazí len pri prihláške, ktorá patrí prihlásenému používateľovi (**+kontrola** podľa `$_SESSION['uid']`) – podobne ako pri zrušení prihlášky

Nevyriešený problém

- **Môžeme odoslať prihlášku viackrát**
- **Riešenie 1:** zistíme, či existuje v databáze záznam (prihláška) pre prihláseného používateľa
- **Riešenie 2:** pri (prvom) odoslaní prihlášky si napr. do SESSION zapíšeme nejakú hodnotu, ktorú budeme kontrolovať pri zobrazení prihlášky

Obmedzenie výsledku dotazu – **LIMIT**

- Pri výbere údajov z databázy môžeme niekedy potrebovať obmedziť počet výsledkov (riadkov), resp. obmedziť počet riadkov, ktorých sa má daný dotaz dotýkať (napr. pri UPDATE)
- Môžeme použiť klauzulu **LIMIT**
- `SELECT ... LIMIT max_pocet_riadkov`
- `UPDATE ... LIMIT max_pocet_riadkov`
- `SELECT ... LIMIT zaciatok, max_pocet_riadkov`
- **Začiatok** sa čísluje od 0
- Využíva sa často pri vyhľadávaní

LIMIT – příklady

- `SELECT * FROM skuska LIMIT 10;`
vyberie max. 10 záznamov z tabuľky
- `SELECT * FROM skuska LIMIT 10, 15;`
vyberie max. 15 záznamov z tabuľky od riadku č. 10 (začína sa číslovať **od 0**)
- `UPDATE skuska_pouzivatelia SET kruzok='1-ai' LIMIT 10;`
zmení prvých max. 10 záznamov v tabuľke na **1-ai**

Funkcie na stĺpcoch

- **count** (počet), **min**, **max**, **avg** (priemer), **sum** (suma)
- `SELECT max(datum) FROM skuska_terminy;`
- `SELECT count(kruzok) FROM skuska_pouzivatelia WHERE kruzok='4m-i';`
- `SELECT count(kruzok) as pocet FROM skuska_pouzivatelia WHERE kruzok='4m-i';`



<code>count(kruzok)</code>
2



<code>pocet</code>
2

Ako zabrániť nežiadúcim výpisom/výstupom?

- Ľubovoľný príkaz v PHP môže generovať chybovú správu, ktorú nechceme vypísať.
- Potlačíme ju znakom @.
- **Napr.:** vyskúšajte si vypnúť MySQL server a následne spustiť niektorý php skript.
- `$link = @mysql_connect('localhost', 'uwa', 'uwa');`

Cvičenia 27.4.2006

- Bude test
- Obsah: témy z prednášok 8 a 9 + cvičenia

Prednáška 1. mája

- Nebude – voľno (štátny sviatok)
- BUDE on-line prednáška namiesto 22. mája
- Prednáška 22. mája nebude.

**Ďakujem za
pozornosť 😊**

Priestor na vaše otázky