

Lázně

Semestrální práce z předmětu DB1

Vypracovala: Eliška Vlčková
Studijní číslo: A12B0533P
Orion login: evlckova
Datum: 9.12.2013

Obsah

1	Úvod	3
2	Popis databázového modelu	4
3	SQL dotazy, pohledy	5
4	Ověření funkčnosti databáze	5
5	Závěr	6

1. Úvod

Tento databázový systém realizuje zjednodušený model lázeňského komplexu, kde uvažuji čtyři hlavní entity - *pracovník*, *kúra*, *klient* a *pokoj*. K jeho vytvoření jsem použila SQL nástroj phpMyAdmin na vlastním serveru. Cílem tohoto modelu je vytvoření databáze použitelné ve smyšleném lázeňském komplexu a následné ověření funkčnosti a integrity.

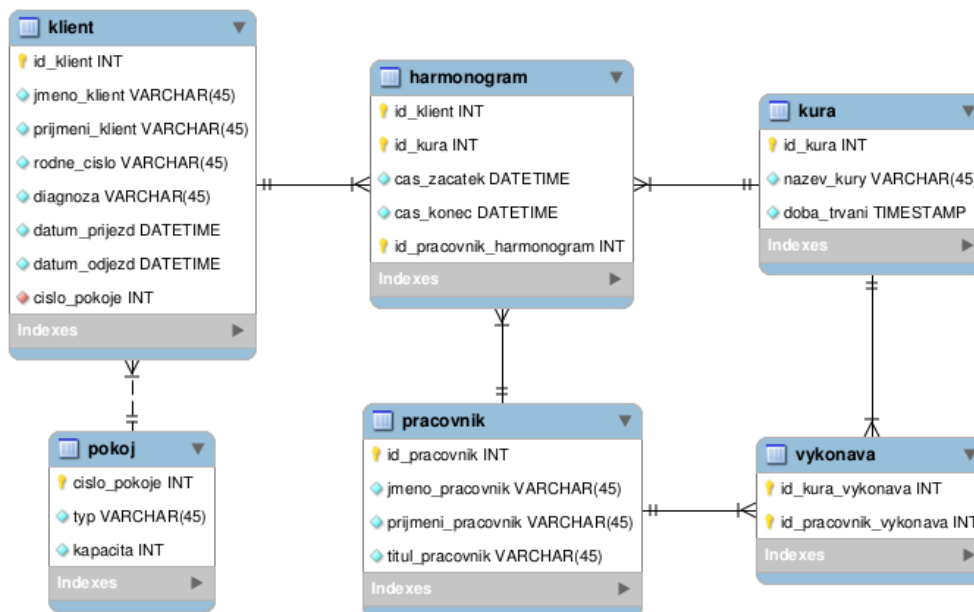
2. Popis databázového modelu

Pro výše zmíněné entity *pracovník*, *kúra*, *klient* a *pokoj* jsem uvažovala následující vztahy znázorněné na modelu databáze (viz obrázek).

Každý pracovník může vykonávat více kúr a každá kúra může být vykonávána více pracovníky. Tuto vazbu M:N rozkládá tabulka *vykonává* jednoduchým propojením *id_pracovník* a *id_kura*.

Dále každý klient může mít objednáno více kúr a jednu kúru může mít objednáno více klientů. Tento vztah M:N rozkládá tabulka *harmonogram*, do které kromě cizích klíčů tvořících vazby ukládám i údaje o čase začátku a skončení procedury. Navíc tabulku *harmonogram* propojuji s tabulkou *pracovník* pro rychlejší získání informace o vykonávajícím pracovníkovi.

Vztah mezi pokojem a klientem je typu 1:N, pouze přiřazuje jednomu každému klientovi, popř. více klientům, jedno číslo pokoje.



3. SQL dotazy, pohledy

Pro zpřehlednění a zjednodušení přístupu k datům v databázi jsem vytvořila několik pohledů a dotazů.

První pohled 'prijizdi_8_11' vyfiltruje z tabulky klient všechny klienty, kteří přijedou 8.11.2013, a vypíše jejich jména a číslo pokoje, tedy usnadňuje práci na recepci a registraci příjezdějících klientů.

```
CREATE VIEW prijizdi_8_11 AS SELECT prijmeni_klient AS 'Příjmení', jmeno_klient AS 'Jméno', cislo_pokoje AS 'Číslo pokoje' FROM klient WHERE datum_prijezd BETWEEN '2013-11-08 00:00:00' AND '2013-11-08 20:00:00' LIMIT 0 , 30;
```

Dále používám dotaz 'harmonogram_klienta_id=7' pro výpis harmonogramu klienta pod id = 7. Výslednou tabulku je vhodné klientovi po příjezdu vytisknout pro jednodušší orientaci.

```
SELECT k.prijmeni_klient AS 'Příjmení', k.jmeno_klient AS 'Jméno', u.nazev_kury AS 'Kúra', h.cas_zacatek AS 'Od', h.cas_konec AS 'Do' FROM klient k, kura u, harmonogram h WHERE k.id_klient = h.id_klient AND h.id_kura = u.id_kura AND h.id_klient = 7 ORDER BY h.cas_zacatek LIMIT 0 , 30;
```

Poslední dotaz 'vypis_kur_pracovnika_id=5' používám jako kontrolu funkčnosti databáze. Vypisuje názvy kúr, které je daný pracovník s id = 5 oprávněn provádět.

```
SELECT k.nazev_kury AS 'Názvy vykonávaných kúr:' FROM kura k, vykonava v WHERE k.id_kura = v.id_kura_vykonava AND v.id_pracovnik_vykonava = 5 LIMIT 0 , 30;
```

4. Ověření funkčnosti databáze

Funkčnost databáze ověřím pomocí série dotazů a příkazů nad databází a kontrolním výpisem před i po úpravě.

Nejprve zobrazím pohled 'prijizdi_8_11', který ukáže 3 stávající klienty. Poté přidám nového klienta do tabulky 'klient' a odeberu jednoho stávajícího klienta:

```
INSERT INTO klient ('id_klient', 'jmeno_klient', 'prijmeni_klient', 'rodne_cislo', 'diagnoza', 'datum_prijezd', 'datum_odjezd', 'cislo_pokoje') VALUES (NULL, 'Aleš', 'Nový', '590704/1514', 'po fraktuře pravé kyčle', '2013-11-08 16:00:00', '2013-11-20 10:00:00', 202);
DELETE FROM klient WHERE id_klient = 5;
```

Nyní mohu zavolat pohled 'prijizdi_8_11' znovu, ale tentokrát bude zobrazovat nového klienta a naopak nezobrazí klienta pod id = 5.

Dále mohu přidat nového pracovníka do tabulky 'pracovnik' a rovnou mu vytvořit záznam v tabulce vykonává:

```
INSERT INTO 'pracovnik' ('id_pracovnik', 'jmeno_pracovnik', 'prijmeni_pracovnik', 'titul_pracovnik') VALUES (NULL, 'Alexandr', 'Nový', 'MUDr.');
```

```
INSERT INTO 'vykonava' ('id_kura_vykonava', 'id_pracovnik_vykonava') VALUES (2, 6), (7, 6);
```

Malým pozměněním existujícího dotazu 'vypis_kur_pracovnika_id=5' (změním v.id_pracovnik_vykonava na hodnotu 6) ověřím existenci nového pracovníka v databázi a získám výpis jím vykonávaných kúr.

5. Závěr

Databázový model řeší zjednodušený lázeňský komplex, kde vztahy M:N mezi 'pracovník' a 'kura' a mezi 'klient' a 'kura' jsou rozloženy pomocí tabulek 'harmonogram' a 'vykonava'. Po úspěšném provedení ověřovacích příkazů a dotazů je zřejmé, že databáze je plně funkční. Pro zlepšení přehlednosti a zjednodušení užívání databáze by bylo nutné použít aplikaci k ovládání databáze. I přesto je databáze přehledná a zachovává referenční integritu díky propojení tabulek cizími klíči.