

Pr

Informační systémy

2008/2009



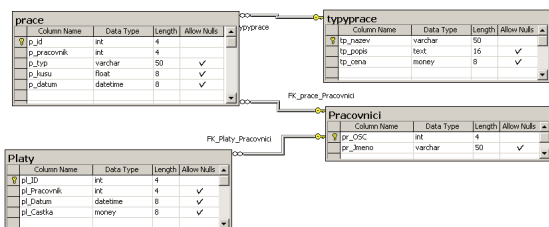
Radim Farana

Řešený příklad

- Vytvořte aplikaci pro sledování pracovních výkonů jednotlivých pracovníků a výpočet jejich denní odměny na základě stanovených cen za výkon jednotlivých typů práce.



Struktura navržené databáze



Vytvoření databáze

```
CREATE TABLE [dbo].[Platy] WITH NOC  
CONSTRAINT [PK_Platy]  
(  
    [pl_id]  
    ) ON [PRIMARY]  
GO  
CREATE TABLE [dbo].[Pracovnici] WITH NOC  
CONSTRAINT [PK_Pracovnici]  
(  
    [pr_id]  
    ) ON [PRIMARY]  
GO  
CREATE TABLE [dbo].[prace] WITH NOC  
CONSTRAINT [PK_prace]  
(  
    [p_id]  
    [pr_id]  
    ) ON [PRIMARY]  
GO  
CREATE TABLE [dbo].[typprace] WITH NOC  
CONSTRAINT [PK_typprace]  
(  
    [tp_id]  
    ) ON [PRIMARY]  
GO  
ALTER TABLE [dbo].[Platy] ADD  
CONSTRAINT [CK_Platy] CHECK ([pl_Castka] > 10000)  
GO  
ALTER TABLE [dbo].[Platy] ADD  
CONSTRAINT [FK_Platy_Pracovnici] FOREIGN KEY  
(  
    [pl_Pracovnik]  
    ) REFERENCES [dbo].[Pracovnici] (  
    [pr_OSC]  
    )  
GO  
ALTER TABLE [dbo].[prace] ADD  
CONSTRAINT [FK_prace_Pracovnici] FOREIGN KEY  
(  
    [p_pracovnik]  
    ) REFERENCES [dbo].[Pracovnici] (  
    [pr_OSC]  
    )  
CONSTRAINT [FK_prace_typprace] FOREIGN KEY  
(  
    [p_tp]  
    ) REFERENCES [dbo].[typprace] (  
    [tp_nazev]  
    ) ON UPDATE CASCADE  
GO
```



Informační systémy

4

Testovací data

```
INSERT INTO typprace (tp_nazev, tp_popis, tp_cena) VALUES ('Kvákání', 'skupinové', 1000);  
INSERT INTO typprace (tp_nazev, tp_popis, tp_cena) VALUES ('praní', 'každý kousek zvlášť', 10);  
INSERT INTO typprace (tp_nazev, tp_popis, tp_cena) VALUES ('spánek', 'celou noc', 120);  
INSERT INTO typprace (tp_nazev, tp_popis, tp_cena) VALUES ('skouzení', 'každý skančík', 15);  
INSERT INTO typprace (tp_nazev, tp_popis, tp_cena) VALUES ('trdění', 'jen mezi příteluznými', 25);  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (122, 'Jan Kvákál');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (123, 'Petr Skančík');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (124, 'Ivanka Desátá-Jedenáctá');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (125, 'Jan-Dáan Škrančík');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (126, 'Patricia de Souza y Cachal de Brito');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (127, 'Jan Pávek I');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (128, 'Kája Ošedl');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (130, 'Orgej Chorchg');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (131, 'Piotr Ivanovič Uštinovskij');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (133, 'Jana Rádona Kasátková');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (140, 'Jan Evangelista Svatovlásky');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (141, 'Tan z Gberjanu');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (152, 'Pavla Ošepál');  
INSERT INTO Pracovnici (pr_OSC, pr_jmeno) VALUES (153, 'Karel Ošepál');  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (122, 'praní', 4, CAST (' 03/12/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (122, 'praní', 5, CAST (' 03/13/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (123, 'škouzení', 12, CAST (' 03/14/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (131, 'praní', 1, CAST (' 03/13/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (131, 'praní', 5, CAST (' 03/14/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (131, 'škouzení', 18, CAST (' 03/15/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (140, 'praní', 20, CAST (' 03/12/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (140, 'praní', 2, CAST (' 03/13/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (140, 'praní', 31, CAST (' 03/14/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (128, 'praní', 2, CAST (' 03/15/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (128, 'škouzení', 2, CAST (' 03/12/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (128, 'praní', 6, CAST (' 03/13/2006' AS DateTime));  
INSERT INTO prace (p_pracovnik, p_tp, p_kusu, p_datum) VALUES (128, 'praní', 7, CAST (' 03/14/2006' AS DateTime));
```



Informační systémy

5

Agregace

- Určete počet záznamů o práci pro jednotlivé pracovníky
- SELECT p_pracovnik,
COUNT(p_id) as Pocet
FROM prace
GROUP BY p_pracovnik;

```
p_pracovnik Pocet  
-----  
122          2  
123          1  
128          4  
131          3  
140          3  
  
(5 row(s) affected)
```



Informační systémy

6

Agregace

- Určete průměrný počet kusů ze všech záznamů práce
- `SELECT AVG(p_kusu) FROM prace;`

```
8.9230769230769234
(1 row(s) affected)
```



Agregace

- Určete průměrný počet kusů pro jednotlivé druhy práce
- `SELECT p_typ, AVG(p_kusu) FROM prace GROUP BY p_typ;`

```
p_typ
-----
prani      8.4000000000000004
škudlení  10.666666666666666
(2 row(s) affected)
```



Vnořený dotaz vracející skalární hodnotu

- Určete záznamy o práci s nadprůměrným počtem kusů
- `SELECT * FROM prace WHERE p_kusu > (SELECT AVG(p_kusu) FROM prace);`

```
p_id  p_pracovnik  p_typ  p_kusu  p_datum
-----
6     123          škudlení  12.0    2006-03-14 00:00:00.000
9     131          škudlení  18.0    2006-03-15 00:00:00.000
10    140          prani     20.0    2006-03-12 00:00:00.000
12    140          prani     31.0    2006-03-14 00:00:00.000
(4 row(s) affected)
```



Vnořený dotaz s propojením

- Určete záznamy o práci s nadprůměrným počtem kusů v rámci stejného druhu práce

```
SELECT * FROM prace  
WHERE p_kusu >
```

```
(SELECT AVG(p_kusu)  
FROM prace  
WHERE p_datum = '2006-03-14 00:00:00.000')  
FROM prace  
WHERE p_kusu > (SELECT AVG(p_kusu)  
FROM prace  
WHERE p_datum = '2006-03-14 00:00:00.000')
```

p_id	p_pracovnik	p_typ	p_kusu	p_datum
10	140	prani	20.0	2006-03-12 00:00:00.000
12	140	prani	31.0	2006-03-14 00:00:00.000
6	123	škudlení	12.0	2006-03-14 00:00:00.000
9	131	škudlení	18.0	2006-03-15 00:00:00.000

(4 row(s) affected)



Vnořený dotaz vracející vektor hodnot

- Určete všechny práce, kterých bylo v jednom dni více než tři (vnořený dotaz)

```
SELECT * FROM prace  
WHERE p_datum In
```

```
(SELECT p_datum  
FROM prace  
GROUP BY p_datum  
HAVING COUNT(*) > 3)  
ORDER BY p_datum
```

p_id	p_pracovnik	p_typ	p_kusu	p_datum
15	128	prani	6.0	2006-03-13 00:00:00.000
11	140	prani	2.0	2006-03-13 00:00:00.000
5	122	prani	5.0	2006-03-13 00:00:00.000
7	131	prani	1.0	2006-03-13 00:00:00.000
8	131	prani	5.0	2006-03-14 00:00:00.000
6	123	škudlení	12.0	2006-03-14 00:00:00.000
12	140	prani	31.0	2006-03-14 00:00:00.000
16	128	prani	7.0	2006-03-14 00:00:00.000

(8 row(s) affected)



Vnořený dotaz s propojením vracející vektor hodnot

- Určete všechny práce, kterých bylo v jednom dni více než dvě stejného druhu

```
SELECT * FROM prace  
WHERE p_datum In
```

```
(SELECT p_datum  
FROM prace  
GROUP BY p_datum  
HAVING COUNT(*) > 2)  
ORDER BY p_datum
```

p_id	p_pracovnik	p_typ	p_kusu	p_datum
5	122	prani	5.0	2006-03-13 00:00:00.000
7	131	prani	1.0	2006-03-13 00:00:00.000
11	140	prani	2.0	2006-03-13 00:00:00.000
15	128	prani	6.0	2006-03-13 00:00:00.000
16	128	prani	7.0	2006-03-14 00:00:00.000
12	140	prani	31.0	2006-03-14 00:00:00.000
8	131	prani	5.0	2006-03-14 00:00:00.000

(7 row(s) affected)



Spojení tabulek

- Určete pro každou osobu a den výdělek za jednotlivé práce (počet kusů krát cena)
- `SELECT p_pracovnik p_pracovnik p_datum Plat`
`p_kusu*tp_cena A`
`FROM prace INNE`
`p_typ=tp_nazev;`

p_pracovnik	p_datum	Plat
122	2006-03-12 00:00:00.000	40.0
122	2006-03-13 00:00:00.000	50.0
123	2006-03-14 00:00:00.000	12.0
131	2006-03-13 00:00:00.000	10.0
131	2006-03-14 00:00:00.000	50.0
131	2006-03-15 00:00:00.000	18.0
140	2006-03-12 00:00:00.000	200.0
140	2006-03-13 00:00:00.000	20.0
140	2006-03-14 00:00:00.000	310.0
128	2006-03-15 00:00:00.000	30.0
128	2006-03-12 00:00:00.000	2.0
128	2006-03-13 00:00:00.000	60.0
128	2006-03-14 00:00:00.000	70.0

(13 row(s) affected)



Spojení dat spolu s agregací

- Určete pro každou osobu a den celkový výdělek `Suma(počet kusů krát cena)`
- `SELECT p_pracovnik p_pracovnik p_datum Plat`
`Sum(p_kusu*tp_cena`
`FROM prace INNE`
`p_typ=tp_nazev`
`GROUP BY p_pracovnik`

p_pracovnik	p_datum	Plat
122	2006-03-12 00:00:00.000	40.0
128	2006-03-12 00:00:00.000	2.0
140	2006-03-12 00:00:00.000	200.0
122	2006-03-13 00:00:00.000	50.0
128	2006-03-13 00:00:00.000	60.0
131	2006-03-13 00:00:00.000	10.0
140	2006-03-13 00:00:00.000	20.0
123	2006-03-14 00:00:00.000	12.0
128	2006-03-14 00:00:00.000	70.0
131	2006-03-14 00:00:00.000	50.0
140	2006-03-14 00:00:00.000	310.0
128	2006-03-15 00:00:00.000	30.0
131	2006-03-15 00:00:00.000	18.0

(13 row(s) affected)



Odstranění dat z tabulky

- Odstraňte všechny záznamy v tabulce `Platy`
- `DELETE FROM Platy;`

(0 row(s) affected)



Vložení dat do tabulky

- Přidejte do tabulky platy pro každého pracovníka za každý den plat ve výši jeho celkového výdělku Suma(počet kusů krát cena)
- INSERT INTO Platy
(pl_Pracovnik, pl_Datum, pl_Castka)
SELECT p_pracovnik, p_datum,
Sum(p_kusu*tp_cena) As Plat
FROM prace
INNER JOIN typyprace
GROUP BY p_pracovnik, p_datum

```
Server: Msg 547, Level 16, State 1, Line 1  
INSERT statement conflicted with COLUMN CHECK  
constraint 'CK_Platy'. The conflict occurred in  
database 'prace', table 'Platy', column 'pl_Castka'.  
The statement has been terminated.
```



Vložení dat do tabulky – oprava

- Zjistěte, proč nebyly platy vloženy
Nyní je vložení platů možné:
- (Je n INSERT INTO Platy
([pl_Pracovnik, pl_Datum, pl_Castka]
SELECT p_pracovnik, p_datum,
Sum(p_kusu*tp_cena) As Plat
FROM prace INNER JOIN typyprace
On p_typ=tp_...
GROUP BY p_...

```
(13 row(s) affected)
```



Určení záznamů bez podřízených

- Určete seznam pracovníků, kteří nemají žádnou práci
- SELECT Pracovnici.*
FROM Pracovnici LEFT JOIN prace
ON pr_osc=...
WHERE p_id = ...

```
pr_osc pr_jmeno  
-----  
124 Ivanka Desátá-Ježenčtá  
125 Jean-Dean Sikorski  
126 Patricia de Souza y Cachal de Brito  
127 Jan Pavel I  
130 Orgoij Chorchoj  
133 Jana Růžena Hostičkovitá  
141 Ian z Glenyanu  
152 Pavla Ospalá  
153 Karel Ospalý  
(9 row(s) affected)
```



Vložený dotaz v definici sloupce

- Pro každého pracovníka určete počet prací

```

SELECT
(
SELECT
FROM p
WHERE
AS Počet
FROM P

```

pr_osc	pr_jmeno	Počet
122	Jan Kvákal	2
123	Patrik Svátý	1
124	Ivanka Desátá-Jedenáctá	0
125	Jean-Dean Sikorski	0
126	Patricia de Souza y Cachal de Brito	0
127	Jan Pavel I	0
128	Karla Osmá	4
130	Orgej Chorchoj	0
131	Piotr Ivanovič Ustinovskij	3
133	Jana Růžena Kostickovitá	0
140	Jan Evangelista Svatovitský	3
141	Ian z Glenyamu	0
152	Pavla Ospalá	0
153	Karel Ospalý	0

(14 row(s) affected)



Informační systémy

19

Vložený dotaz v definici sloupce

- Pro každého pracovníka určete počet dnů, kdy pracoval

```

SELECT
(
SELECT
FROM p
WHERE
AS Počet
FROM P

```

pr_osc	pr_jmeno	Počet
122	Jan Kvákal	2
123	Patrik Svátý	1
124	Ivanka Desátá-Jedenáctá	0
125	Jean-Dean Sikorski	0
126	Patricia de Souza y Cachal de Brito	0
127	Jan Pavel I	0
128	Karla Osmá	4
130	Orgej Chorchoj	0
131	Piotr Ivanovič Ustinovskij	3
133	Jana Růžena Kostickovitá	0
140	Jan Evangelista Svatovitský	3
141	Ian z Glenyamu	0
152	Pavla Ospalá	0
153	Karel Ospalý	0

(14 row(s) affected)



Informační systémy

20

Konverze dat – eliminace NULL

- Pro každého pracovníka určete počet kusů z tabulky prace, pokud dosud nepracoval, bude počet 0

```

SELECT
COALESCE
FROM p
LEFT JOIN
GROUP

```

pr_osc	pr_jmeno	Počet
122	Jan Kvákal	9.0
123	Patrik Svátý	12.0
124	Ivanka Desátá-Jedenáctá	0.0
125	Jean-Dean Sikorski	0.0
126	Patricia de Souza y Cachal de Brito	0.0
127	Jan Pavel I	0.0
128	Karla Osmá	18.0
130	Orgej Chorchoj	0.0
131	Piotr Ivanovič Ustinovskij	24.0
133	Jana Růžena Kostickovitá	0.0
140	Jan Evangelista Svatovitský	53.0
141	Ian z Glenyamu	0.0
152	Pavla Ospalá	0.0
153	Karel Ospalý	0.0

(14 row(s) affected)



Informační systémy

21

Agregace - opakovaná

- Pro každého pracovníka zobrazte jeho osobní číslo, jméno a první a poslední den, ve kterém pracoval
- SELECT pr_osc, pr_jmeno,
Min(p_datum) As prvni,
Max(p_datum) As posledni

```
FROM pracovnici INNER JOIN prace
pr_osc pr_jmeno prvni posledni
-----
140 Jan Evangelista Svatovitsky 2006-03-12 00:00:00.000 2006-03-14 00:00:00.000
122 Jan Kvákal 2006-03-12 00:00:00.000 2006-03-13 00:00:00.000
128 Karla Osámá 2006-03-12 00:00:00.000 2006-03-15 00:00:00.000
123 Patrik Svátý 2006-03-14 00:00:00.000 2006-03-14 00:00:00.000
131 Piotr Ivanovič Ustinovskij 2006-03-13 00:00:00.000 2006-03-15 00:00:00.000
(5 row(s) affected)
```



Konverze dat

- Pro každého pracovníka zobrazte jeho osobní číslo, jméno a počet dnů mezi první a poslední prací
- SELECT pr_osc, pr_jmeno,
CAST(Max(p_datum)-Min(p_datum) AS Integer)

```
FROM pracovnici INNER JOIN prace
ON pr_osc = pracovnik
pr_osc pr_jmeno
-----
140 Jan Evangelista Svatovitsky 2
122 Jan Kvákal 1
128 Karla Osámá 3
123 Patrik Svátý 0
131 Piotr Ivanovič Ustinovskij 2
(5 row(s) affected)
```



Dávka – využití kurzoru

- Vytvořte dávku, která vygeneruje pro tabulku typyprace dávku definic dat tabulky

```
DECLARE @nazev varchar(50), @popis varchar(800), @cena money
DECLARE listtypyprace CURSOR FORWARD_ONLY
FOR
SELECT (tp_nazev), (tp_popis), (tp_cena) FROM [prace].[dbo].[typyprace]
OPEN listtypyprace
FETCH NEXT FROM listtypyprace
INTO @nazev, @popis, @cena
WHILE @@FETCH_STATUS = 0
BEGIN
PRINT 'INSERT INTO typyprace ((tp_nazev), (tp_popis), (tp_cena))
VALUES ('' + @nazev + '' + @popis, '' + @nazev + '' + @popis, ' + @cena);'
INSERT INTO typyprace ((tp_nazev), (tp_popis), (tp_cena))
VALUES ('prani', 'každý kousek zvlášť', 10.00);
INSERT INTO typyprace ((tp_nazev), (tp_popis), (tp_cena))
VALUES ('spaní', 'celou noc', 120.00);
INSERT INTO typyprace ((tp_nazev), (tp_popis), (tp_cena))
VALUES ('škudlení', 'každý okamžik', 1.00);
END
CLOSE listtypyprace
DEALLOCATE listtypyprace
GO
```



Uložená procedura

- Vytvořte uloženou proceduru ,VlozPraci', která převezme údaje o pracovníkovi, typu práce, počtu kusů a nepovinné datum (v tom případě vloží aktuální datum), zkontroluje existenci pracovníka a typu práce a vloží nový záznam do databáze a vrátí jeho ID

```
(1 row(s) affected)
18
pr_osc   pr_Imeno
-----
122     Jan Kvákal
123     Patrik Svаты
124     Ivanka Desátá-Jedenáctá
125     Jean-Dean Sikorski
126     Patricia de Souza y Cachal de Brito
127     Jan Pavel I
128     Karla Ombá
130     Orgoј Chorrohoј
131     Piotr Ivanovič Ustinovskij
133     Jana Růžena Kostičková
140     Jan Evangelista Svatovítaky
141     Ian z Glenyanu
152     Pavla Ospalá
153     Karel Ospalý
154     Karel Ospalý
(14 row(s) affected)
5
```

Spoušť

```
Zápis:
ALTER TRIGGER pr_osc ON (db) [prace]
FOR INSERT, UPDATE, DELETE
AS
-- zpracujeme vloženou práci
-- zpracujeme existující práci
BEGIN
-- zjistíme kolik záznamů je již v tabulce
-- zpracujeme vložení nové práce
FOR
SELECT p_id, p_nazev
FROM p_prace
OPEN cur
-- vezmeme první záznam
FETCH NEXT FROM cur INTO @id, @nazev
WHILE @@FETCH_STATUS = 0
BEGIN
-- vezmeme první nový záznam
OPEN cur2
FETCH NEXT FROM cur2 INTO @id2, @nazev2, @p_kusu, @p_datum
WHILE @@FETCH_STATUS = 0
BEGIN
-- Upríme cenu vložené práce
SELECT (@cena = MAX(p_cena) FROM typyprace WHERE tp_nazev=@nazev)
-- Pokud již tento pracovník v daném dni pracoval, budeme záznam upravovat, jinak musíme přidat nový
IF EXISTS (SELECT * FROM platy WHERE pl_pracovnik=@pracovnik AND pl_datum=@datum)
UPDATE platy SET pl_casikari=@casika, @pl_casika/@pl_casika AS FLOAT) WHERE pl_pracovnik=@pracovnik AND pl_datum=@datum
ELSE
INSERT INTO platy(pl_pracovnik, pl_datum, pl_casika) VALUES (@pracovnik, @datum, @pl_casika AS FLOAT)
END
CLOSE cur2
DELETE FROM platy WHERE pl_casika=0
END
END
GO
Info
```

Kontrolní otázka

- Co se stane, pokud se změní cena některého typu práce?
- Odměny za tento typ práce nebudou odpovídat nové ceně.
- Důvodem je chyba v koncepci databáze – tabulky **typyprace** obsahuje aktuální hodnoty, zatímco tabulky **prace** a **platy** obsahují historická data.
- Řešením je uložení aktuální ceny typu práce přímo k nově vložené práci, kde zůstane zachována i do budoucna.
