

1 Dokumentace temporálních dotazů

1.1 SELECT

Nalezení volných pokojů v daném období.

TSQL2:

```
SELECT * FROM pokoje
WHERE id NOT IN (SELECT pokoj FROM rezervace)
VALIDTIME ['2013-12-24' - '2013-12-26']
```

SQL:

```
SELECT * FROM pokoje
WHERE id NOT IN (
    SELECT pokoj FROM rezervace WHERE (
        (od BETWEEN
            TO_DATE('2013-12-24', 'yyyy-mm-dd') AND
            TO_DATE('2013-12-26', 'yyyy-mm-dd'))
        OR
        (do BETWEEN
            TO_DATE('2013-12-24', 'yyyy-mm-dd') AND
            TO_DATE('2013-12-26', 'yyyy-mm-dd'))
    )
    OR
    (TO_DATE('2013-12-24', 'yyyy-mm-dd') BETWEEN od AND do) OR
    (TO_DATE('2013-12-26', 'yyyy-mm-dd') BETWEEN od AND do)
);
```

Zjištění průměrného počtu doposud využitých služeb na každého zákazníka, který byl kdy ubytován ubytován.

TSQL2:

```
SELECT nvl(avg(nvl2(sluzby_rezervace.id, 1, 0)),0) as prumer
FROM zakaznik LEFT JOIN sluzby_rezervace
ON sluzby_rezervace.zakaznik = zakaznik.id
WHERE VALID(rezervace_sluzeb) PRECEDES DATE NOW
```

SQL:

```
SELECT nvl(avg(nvl2(sluzby_rezervace.id, 1, 0)),0) as prumer
FROM zakaznik LEFT JOIN sluzby_rezervace
ON sluzby_rezervace.zakaznik = zakaznik.id
WHERE sluzby_rezervace.den <= trunc(sysdate) OR
sluzby_rezervace.den IS NULL
```

Nalezení zákazníka s doposud nejdelším pobytem v hotelu.

TSQL2:

```
SELECT zakaznik(jmeno), zakaznik(prijmeni), CAST(VALID(L) AS INTERVAL DAY) AS pocet_dnu
FROM rezervace JOIN zakaznik AS L ON (rezervace(zakaznik) = zakaznik(id))
WHERE CAST(VALID(L) AS INTERVAL DAY) > ALL (
    SELECT CAST(VALID(L2) AS INTERVAL DAY)
    FROM zakaznik(jmeno) L2 WHERE L.jmeno != L2.jmeno
) AND VALID(zakaznik) PRECEDES DATE NOW
```

SQL:

```
SELECT zakaznik.jmeno, zakaznik.prijmeni, trunc (do - od) AS pocet_dnu
FROM rezervace JOIN zakaznik ON (rezervace.zakaznik = zakaznik.id)
WHERE trunc (do - od) IN (
  SELECT MAX (pocet) AS maximum FROM (
    SELECT x.*, trunc (do - od) AS pocet
    FROM rezervace x
  ) r
);
```

1.2 UPDATE

Změna pokoje na jiný volný pokoj.

TSQL2:

```
UPDATE rezervace
  SET pokoj = 9
  WHERE pokoj = 1
VALID TIME ['2013-12-24' - '2013-12-26']
```

SQL:

```
UPDATE rezervace
  SET pokoj = 9
  WHERE pokoj = 1 AND
    od = TO_DATE('2013-12-24', 'yyyy-mm-dd') AND
    do TO_DATE('2013-12-26', 'yyyy-mm-dd')
```

1.3 DELETE

Smazání všech rezervací v daném období (v aplikaci při vyhlášení karantény).

TSQL2:

```
DELETE FROM rezervace  
VALIDTIME ['2013-12-24' - '2013-12-26']
```

SQL:

```
CREATE OR REPLACE PROCEDURE rezervace_smaz_v_obdobi (  
    datum_od DATE, datum_do DATE ) IS  
    new_end DATE;  
    new_start DATE;  
    tmp_zakaznik NUMBER;  
    tmp_pokoj NUMBER;  
    tmp_od DATE;  
    tmp_do DATE;  
    tmp_id NUMBER;  
    CURSOR cursor1 IS  
        SELECT id, zakaznik, pokoj, od, do  
        FROM rezervace WHERE od < datum_od AND do > datum_do;  
    CURSOR cursor2 IS  
        SELECT id, zakaznik, pokoj, od, do  
        FROM rezervace WHERE od < datum_do AND do > datum_od;  
BEGIN  
    IF datum_od <= datum_do THEN  
        new_end := datum_od - 1;  
        new_start := datum_do + 1;  
        DELETE FROM rezervace WHERE od >= datum_od AND do <= datum_do;  
        UPDATE rezervace SET do = new_end  
            WHERE od < datum_od AND (do BETWEEN datum_od AND datum_do);  
        UPDATE rezervace SET od = new_start  
            WHERE do > datum_do AND (od BETWEEN datum_od AND datum_do);  
        OPEN cursor1;  
        LOOP  
            FETCH cursor1 INTO tmp_id, tmp_zakaznik, tmp_pokoj, tmp_od, tmp_do;  
            EXIT WHEN cursor1%NOTFOUND;  
            INSERT INTO rezervace (zakaznik, pokoj, od, do)  
                VALUES(tmp_zakaznik, tmp_pokoj, new_start, tmp_do);  
            UPDATE rezervace  
                SET do = new_end WHERE id = tmp_id;  
        END LOOP;  
        CLOSE cursor1;  
    ELSE  
        new_end := datum_do - 1;  
        new_start := datum_od + 1;  
        DELETE FROM rezervace WHERE od >= datum_do AND do <= datum_od;  
        UPDATE rezervace SET do = new_end  
            WHERE od < datum_do AND (do BETWEEN datum_do AND datum_od);  
        UPDATE rezervace SET od = new_start  
            WHERE do > datum_od AND (od BETWEEN datum_do AND datum_od);  
        OPEN cursor2;  
        LOOP  
            FETCH cursor2 INTO tmp_id, tmp_zakaznik, tmp_pokoj, tmp_od, tmp_do;
```

```
EXIT WHEN cursor2%NOTFOUND;
INSERT INTO rezervace (zakaznik, pokoj, od, do)
    VALUES(tmp_zakaznik, tmp_pokoj, new_start, tmp_do);
UPDATE rezervace SET do = new_end WHERE id = tmp_id;
END LOOP;
CLOSE cursor2;
END IF;
END;
/
```