

2.6. Kelių lentelių jungimas

Viena iš svarbiausių **SELECT** sakinio galimybių yra dviejų ar daugiau lentelių jungimas (angl. *join*).

Pavardės vykdytojų, vykdančių projektą Nr. 1:

Vykdytojai

Nr	Pavardė	Kvalifikacija	Kategorija	Išsilavinimas
1	Jonaitis	Informatikas	2	VU
2	Petraitis	Statistikas	3	VU
3	Gražulytė	Inžinierius	1	NULL
4	Onaitytė	Vadybininkas	5	VDU
5	Antanaitis	Informatikas	3	VU

Vykdymas

Projektas	Vykdytojas	Statusas	Valandos
1	1	Programuotojas	30
1	2	Dokumentuotojas	100
1	3	Testuotojas	100
1	4	Vadovas	100
2	1	Programuotojas	300
2	2	Analitikas	250
2	4	Vadovas	100
3	1	Programuotojas	250
3	2	Vadovas	400
3	3	Dizaineris	150

SELECT Pavardė

FROM Vykdytojai, Vykdymas

WHERE Vykdytojas = Nr **AND** Projektas = 1

Sąlyga $Vykdytojas = Nr$ - loginis ryšys tarp dviejų lentelių.

Bendruoju atveju, jei užklausoje dviem lentelėms nėra jokios sąlygos ir vienoje iš lentelių yra n eilučių, o kitoje - m eilučių, tai rezultata sudarys $m \times n$ eilučių

LentelėA

A1	B1
a1	b1
a2	b1
a2	b2

LentelėB

A2	B2	C2
a1	b1	c1
a2	b2	c2

SELECT A1, B1, A2, B2, C2 **FROM** LentelėA, LentelėB

A1	B1	A2	B2	C2
a1	b1	a1	b1	c1
a2	b1	a1	b1	c1
a2	b2	a1	b1	c1
a1	b1	a2	b2	c2
a2	b1	a2	b2	c2
a2	b2	a2	b2	c2

SELECT A1, B1, C2

FROM LentelėA, LentelėB

WHERE A1 = A2 **AND** B1 = B2

A1	B1	C2
a1	b1	c1
a2	b2	c2

Stulpelių porų (A1, A2) ir (B1, B2) reikšmės turi būti tarpusavyje palyginamos - jų reikšmių aibės neturi konfliktuoti.

Vietoje

SELECT A1, B1, A2, B2, C2

FROM LentelėA, LentelėB

galima rašyti

SELECT LentelėA.*, LentelėB.*

FROM LentelėA, LentelėB

ar

SELECT * **FROM** LentelėA, LentelėB

Informacija apie tai, kokie vykdytojai kokius projektus vykdo ir kiek kiekvienam projektui skiria valandų:

SELECT Pavardė, Pavadinimas, Valandos

FROM Vykdytojai, Projektai, Vykdymas

WHERE Projektas = Projektai.Nr **AND**
Vykdytojas = Vykdytojai.Nr

Stulpelio Nr patikslinimas lentelės vardu yra būtinas.

Lentelę galima **jungti ir su ja pačia**.

Poros vykdytojų, turinčių tą pačią kvalifikaciją:

Nr	Pavardė	Kvalifikacija	...
1	Jonaitis	Informatikas	...
2	Petraitis	Statistikas	...
3	Gražulytė	Inžinierius	...
4	Onaitytė	Vadybininkas	...
5	Antanaitis	Informatikas	...

1	2
Jonaitis	Antanaitis

SELECT *A.Pavardė, B.Pavardė*
FROM *Vykdytojai A, Vykdytojai B*
WHERE *A.Kvalifikacija = B. Kvalifikacija*
AND *A.Nr < B.Nr*

9-37

Laikome, kad yra 2 vienos lentelės egzemplioriai
 Jiems nurodyti apibrėžiame sinonimus: *A* ir *B*.

A.Kvalifikacija = B. Kvalifikacija - jungimo sąlyga.

A.Nr < B.Nr - išvengti vykdytojo poros su juo pačiu ir porų pasikartojimo,

t.y. vietoje 2-jų porų (*x, y*) ir (*y, x*) pakanka turėti 1.

DISTINCT nepadedą, nes (*x, y*) ≠ (*y, x*), jei tik *x*≠*y*.

SELECT DISTINCT *A.Pavardė, B.Pavardė*
FROM *Vykdytojai A, Vykdytojai B*
WHERE *A.Kvalifikacija = B. Kvalifikacija*

10-37

1	2
Jonaitis	Jonaitis
Jonaitis	Antanaitis
Petraitis	Petraitis
Gražulytė	Gražulytė
Onaitytė	Onaitytė
Antanaitis	Jonaitis
Antanaitis	Antanaitis

SELECT *A.Pavardė, B.Pavardė*
FROM *Vykdytojai A, Vykdytojai B*
WHERE *A.Kvalifikacija = B. Kvalifikacija*
AND *A.Nr <> B.Nr*

11-37

1	2
Jonaitis	Antanaitis
Antanaitis	Jonaitis

Vykdytojai, vykdančys didelės svarbos projektus, ir kiek kiekvienam tokiam projektui skiria valandų:

SELECT *Pavardė, Pavadinimas, Valandos*
FROM *Vykdytojai, Projektai, Vykdymas*
WHERE *Projektas = Projektai.Nr AND*
Vykdytojas = Vykdytojai.Nr AND
Svarba = 'Didelė'

12-37

Projektai, kuriuos vykdo informatikai, ir kiek kiekvienam projektui skiria valandų:

SELECT *Pavardė, Pavadinimas, Valandos*
FROM *Vykdytojai, Projektai, Vykdymas*
WHERE *Projektas = Projektai.Nr AND*
Vykdytojas = Vykdytojai.Nr AND
Kvalifikacija = 'Informatikas'

13-37

Visų vykdytojų pavardės:

SELECT *Pavardė* **FROM** *Vykdytojai*
 - teisingiausia

SELECT *Pavardė* **FROM** *Vykdytojai, Projektai*
 - **kartoja**si tiek kartų kiek yra projektų

SELECT DISTINCT *Pavardė*
FROM *Vykdytojai, Projektai*
 - teisinga, bet **neefektyvi**,
 o ypatingu atveju ir neteisinga

14-37

Užklausa, jungianti 2 lenteles:

SELECT <stulpeliai> **FROM** <lentelė1>, <lentelė2>
WHERE <jungimo sąlyga> [**AND** <paieškos sąlyga>]

15-37

Daugelyje DBVS yra užtikrinamas lentelių jungimas **SELECT** sakinio fraze **JOIN**:

SELECT <stulpeliai>
FROM <lentelė1> **JOIN** <lentelė 2>
ON <jungimo sąlyga>
[**WHERE** <paieškos sąlyga>]

Projekto Nr. 1 vykdytojų pavardės:

SELECT *Pavardė*
FROM *Vykdytojai JOIN Vykdymas*
ON *Vykdytojas = Nr*
WHERE *Projektas = 1*

16-37

Užklausa be **JOIN**:

SELECT *Pavardė*
FROM *Vykdytojai, Vykdymas*
WHERE *Vykdytojas = Nr AND Projektas = 1*

Galima jungti ir daugiau lentelių, pvz.,

```
SELECT Pavardė, Pavadinimas, Valandos
FROM (Vykdytojai JOIN Vykdymas ON
      Vykdytojas = Vykdytojai.Nr)
JOIN Projektai
ON Projektas = Projektai.Nr
```

Užklausa be **JOIN**:

```
SELECT Pavardė, Pavadinimas, Valandos
FROM Vykdytojai, Projektai, Vykdymas
WHERE Projektas = Projektai.Nr AND
      Vykdytojas = Vykdytojai.Nr
```

Visų projekto Nr. 1 vykdytojų pavardės, statusai ir valandos,

```
SELECT Pavardė, Statusas, Valandos
FROM Vykdytojai JOIN Vykdymas
      ON Vykdytojas = Nr
WHERE Projektas = 1
```

Rezultate bus tik tų autorių pavardės, kurie dalyvauja projekte Nr. 1, vykdytojo Nr. 5 nebus.

Vykdytojas Nr. 5 nedalyvauja jokiame projekte – jo nėra lentelėje *Vykdymas*.

Visi vykdytojai bus pateikiami taikant - **išorinį** (anglų k. *outer*) **jungimą**.

LEFT OUTER JOIN - jungimo rezultatas yra papildomas kairiosios (pirmosios) lentelės nesujungiamomis eilutėmis.

RIGHT OUTER JOIN - papildoma dešinėsios lentelės nesujungiamomis eilutėmis

FULL OUTER JOIN - papildoma abiejų lentelių eilutėmis.

Operacija **JOIN** yra tapati operacijai **INNER JOIN**.

Darbuotojai ir jų dalyvavimas projekte Nr. 1:

```
SELECT Pavardė, Statusas, Valandos
FROM Vykdytojai LEFT OUTER JOIN Vykdymas
      ON Vykdytojas = Nr
WHERE Projektas = 1 OR Projektas IS NULL
```

Pavardė	Statusas	Valandos
Jonaitis	Programuotojas	30
Petraitis	Dokumentuotojas	100
Gražulytė	Testuotojas	100
Onaitytė	Vadovas	100
Antanaitis	NULL	NULL

Rezultatas nesikeičia taikant **RIGHT OUTER JOIN** ir sukeičiant lenteles vietomis,

```
SELECT Pavardė, Statusas, Valandos
FROM Vykdymas RIGHT OUTER JOIN
      Vykdytojai ON Vykdytojas = Nr
WHERE Projektas = 1 OR Projektas IS NULL
```

2.7. Struktūrinės užklauskos

Vienoje užklausoje gali būti ir kita užklausa, t.y. galimos struktūrinės užklauskos – „Structured QL“.

- Keli **SELECT** sakiniai yra griežtoje hierarchinėje priklausomybėje.

Pavardės vykdytojų, dalyvaujančių projekte Nr. 1:

```
SELECT Pavardė FROM Vykdytojai
WHERE Nr IN (SELECT Vykdytojas
             FROM Vykdymas
             WHERE Projektas = 1)
```

Vykdytojai, dalyvaujantys bent viename didelės svarbos projekte:

```
SELECT Pavardė FROM Vykdytojai
WHERE Nr IN
      (SELECT Vykdytojas FROM Vykdymas
       WHERE Projektas IN
         (SELECT Nr FROM Projektai
          WHERE Svarba = 'Didelė'))
```

- Didelis dalinių užklauskų skaičius, nereiškia gerą stilių.

Vykdytojai, dalyvaujantys bent viename didelės svarbos projekte:

```
SELECT DISTINCT Pavardė
FROM Vykdytojai, Vykdymas, Projektai
WHERE Projektas = Projektai.Nr AND
      Vykdytojas = Vykdytojai.Nr AND
      Svarba = 'Didelė'
```

- Ta pati lentelė gali būti vidinėje ir išorinėje užklausoje.

Numeriai vykdytojų, kurie dalyvauja bent viename projekte, kuriame dalyvauja vykdytojas Nr. 1:

```
SELECT DISTINCT Vykdytojas FROM Vykdymas
WHERE Vykdytojas <> 1 AND
      Projektas IN (SELECT Projektas
                   FROM Vykdymas
                   WHERE Vykdytojas = 1)
```

•Priklausomosios (koreliuotos) užklauso.

Vykdytojai, dalyvaujantys projekte Nr. 1:

```
SELECT Pavardė FROM Vykdytojai
WHERE 1 IN (SELECT Projektas FROM Vykdymas
           WHERE Vykdytojas = Nr)
```

Priklausoma užklausa - tai užklausa, kurios vidinės užklauso rezultatas priklauso nuo išorinės užklauso rezultato - abi užklauso negali būti atliekamos nuosekliai.

2.8. Laikinosios lentelės

Frazėje **FROM** lentelės vietoje gali būti užklausa.

Vykdytojai, dalyvaujantys bent viename projekte, kuriame dalyvauja vykdytojas Nr. 1 :

```
SELECT DISTINCT Vykdytojas FROM Vykdymas,
              (SELECT Projektas FROM Vykdymas
               WHERE Vykdytojas = 1) AS Projektai1
WHERE Projektai1.Projektas = Vykdymas.Projektas
AND Vykdymas.Vykdytojas <> 1
```

Projektai1 – laikinoji lentelė.

Ši laikinoji lentelė egzistuoja tik vykdant užklausą.

Laikinosios lentelės apibrėžimas konstrukcija **WITH** :

```
WITH Projektai1 AS
      (SELECT Projektas FROM Vykdymas
       WHERE Vykdytojas = 1)
SELECT DISTINCT Vykdytojas
FROM Vykdymas, Projektai1
WHERE Vykdymas.Vykdytojas <> 1 AND
      Projektai1.Projektas = Vykdymas.Projektas
```

Viena fraze **WITH** galima apibrėžti kelias laikinas lenteles, atskiriant jas tarpusavyje **kableliais**.

Apibrėžiant laikinąją lentelę, po jos pavadinimo, tarp skliaustų galima nurodyti stulpelių vardus.

```
WITH Projektai1 (Projektas) AS
      (SELECT Projektas FROM Vykdymas
       WHERE Vykdytojas = 1)
SELECT DISTINCT Vykdytojas
FROM Vykdymas, Projektai1
WHERE Vykdymas.Vykdytojas <> 1 AND
      Projektai1.Projektas = Vykdymas.Projektas
```

2.9. Bendrumo ir egzistavimo kvantoriai užklauso

Vykdytojai, dalyvaujančių projekte Nr. 1:

```
SELECT Pavardė FROM Vykdytojai
WHERE EXISTS (SELECT * FROM Vykdymas
             WHERE Vykdytojas = Nr
             AND Projektas = 1)
```

EXISTS (SELECT * FROM...) – predikatas, kurio reikšmė yra „tiesa“ tik tuomet, kai užklauso rezultatas - netuščioji aibė.

Ši užklausa yra priklausomoji – nėra efektyvi.

Paieškos sąlyga

```
EXISTS (SELECT * FROM...)
```

yra ekvivalenti sąlygai

```
(SELECT COUNT(*) FROM...) > 0
```

Predikatų logikos kvantorių atitikmenys naudojami ir tokiuose predikatuose

<reiškiny> <palyginimo operacija>

```
<ALL | ANY | SOME> (<užklausa> )
```

Predikatas **SOME** yra predikato **ANY** sinonimas.

<reiškiny> <palyginimas> **ALL** (<užklausa>)

Reiškinio reikšmė yra lyginama su visomis užklauso rezultato reikšmėmis.

Predikato **ALL** reikšmė yra „tiesa“ tuomet ir tik tuomet, kai užklauso rezultatas yra tuščioji aibė arba palyginimo reikšmė yra „tiesa“ visoms užklauso rezultato reikšmėms.

Predikato **ANY** reikšmė yra „tiesa“ tik tuomet, kai palyginimo reikšmė yra „tiesa“ bent vienai užklauso rezultato reikšmei.

Darbuotojų kvalifikacijos, kuriose visi darbuotojai yra ne mažesnės negu 2-os kategorijos:

```
SELECT DISTINCT A.Kvalifikacija
FROM Vykdytojai A
WHERE 2 <= ALL (SELECT B.Kategorija
                FROM Vykdytojai B
                WHERE A.Kvalifikacija =
                    B.Kvalifikacija)
```

Vidinė užklausa yra priklausoma - priklauso nuo parametro *A.Kvalifikacija*

Tarus, kad skirtingų kvalifikacijų yra žymiai mažiau negu darbuotojų, galima sudaryti efektyvesnį uždavinio sprendinį:

```
SELECT A.Kvalifikacija
FROM (SELECT DISTINCT Kvalifikacija
      FROM Vykdytojai) AS A
WHERE 2 <= ALL
      (SELECT B.Kategorija FROM Vykdytojai B
      WHERE A.Kvalifikacija = B.Kvalifikacija)
```

Vykdytojų, dalyvaujančių projekte Nr. 1, pavardės:

```
SELECT Pavardė FROM Vykdytojai
WHERE Nr = ANY (SELECT Vykdytojas
                FROM Vykdymas
                WHERE Projektas = 1)
```

Ši užklausa yra panaši į ankstesnę užklausą su predikatu **IN**.

Predikatą **IN** galima išreikšti per predikatą **ANY**.

<reiškinys> **IN** (<užklausa>)

yra ekvivalentiškas užrašui

<reiškinys> = **ANY** (<užklausa>)

Predikatas **NOT IN** išreiškiamas per predikatą **ALL**.

<reiškinys> **NOT IN** (<užklausa>)

yra ekvivalentiškas užrašui

<reiškinys> <> **ALL** (<užklausa>)

Darbuotojų kvalifikacijos, kuriose visi darbuotojai yra ne mažesnės negu 2-os kategorijos:

```
SELECT DISTINCT A.Kvalifikacija
FROM Vykdytojai A
WHERE (SELECT COUNT(*)
      FROM Vykdytojai B
      WHERE A.Kvalifikacija = B.Kvalifikacija
      AND B.Kategorija < 2) = 0
```

Tai kvalifikacijos, kuriose nėra mažesnės negu antros kategorijos darbuotojų.