

5. Duomenų bazės sukūrimas ir užpildymas duomenimis

5.3a. Duomenų tipų suderinamumas

Praktikoje dažnai pasitaiko, kad atliekant operacijas su keliais argumentais, kreipiantis į funkcijas ir pan. tenka suderinti duomenų tipus. Taip yra, pvz., prie sveiko skaičiaus pridant slankaus kablelio (realųjį) skaičių, taip pat lyginant fiksuoto ilgio simbolių eilutę su kintamojo ilgio simbolių eilute. Šiais, dažnai pasitaikančiais atvejais duomenų tipai yra suderinami automatiškai pagal plačiai taikomas taisykles, pvz., prieš sudedant sveikąjį ir realųjį skaičius, sveikasis yra pervedamas į realųjį. Konkretios DBVS dokumentacijoje galima rasti tikslias tipų suderinamumo lenteles.

Tačiau automatinis tipų suderinamumas ne visada yra tikslingas ir įmanomas. Pvz., jei reiškinys *Vykdytojai.Kategorija* || 'kategorija' yra panaudotas užklausoje duomenų bazei *Darbai*, tai tikslinga jį laikyti klaidingu sintaksės požyriu, nes vienas simbolių jungimo operacijos operandas yra skaičius. Norint užklaustos rezultate šalia skaičiaus, atitinkančio darbuotojo kategoriją, turėti žodį „kategorija“, reikia reiškinyje išreikštai nurodyti skaičiaus transformavimą į simbolių eilutę. Vieno tipo reikšmės transformavimas į kito tipo reikšmę SQL kalboje išreiškiamas specifikacija CAST

CAST (<reiškinys> AS <duomenų tipas>)

Šia fraze SQL sakinyje nurodoma apskaičiuotąjį reiškinio rezultatą transformuoti į pasirinkto tipo reikšmę, pvz. jau suformuluotą užklausą informatikų pavardėms ir kategorijoms gauti, galima suformuluoti ir taip

```
SELECT Pavardė, CAST(Kategorija AS CHAR(1)) || 'kategorija' AS Kategorija
FROM Vykdytojai WHERE Vykdytojai.Kvalifikacija = 'Informatikas'.
```

Jei duomenų bazėje yra antrajame skyriuje pateiktieji duomenys, tai šios užklaustos rezultate bus dvi eilutės ir du stulpeliai

<i>Pavardė</i>	<i>Kategorija</i>
Jonaitis	2 kategorija
Antanaitis	3 kategorija

Tarkime, mums reikia apskaičiuoti visų vykdytojų kategorijų vidurkį. Paprasčiausiai tai galima išreikšti sakiniu

```
SELECT AVG(Kategorija) AS "Kategorijų vidurkis" FROM Vykdytojai.
```

Nesunku apskaičiuoti mintinai, kad vidurkis yra 2,8, tačiau šios užklaustos rezultatas yra

<i>Kategorijų vidurkis</i>
2

Rezultate gavome sveikąjį skaičių, nes funkcijos AVG argumentas yra sveikojo skaičiaus tipo. Kad rezultate gauti trupmeninį skaičių, reikia argumentą *Kategorija* transformuoti į trupmeninį skaičių, pvz. CAST(*Kategorija* AS FLOAT). Tačiau taip gausime vidurkį, kuriame po dešimtainio taško bus daug skaitmenų. Žinoma, didelis tikslumas yra svarbu, bet, kalbant apie kategorijas, tikriausiai pakanka vieno ar kelių dešimtainės trupmenos skaitmenų. Apskaičiuotasis vidurkis bus pavaizduotas penkiais dešimtainiais skaitmenimis, iš kurių du po kablelio, jei vykdysime tokią užklausą

```
SELECT CAST(AVG(CAST(Kategorija AS FLOAT)) AS DECIMAL(5,2))
AS "Kategorijų vidurkis" FROM Vykdytojai.
```

Pastebėsime, kad specifikacija CAST galimas ne bet koks tipų transformavimas. Pavyzdžiui, negalima skaitinius duomenis transformuoti į dvejetainius ir datas bei laiko duomenis. Leistinių transformacijų rinkinį galima rasti konkrečios DBVS dokumentacijoje.

Daugelyje plačiai naudojamų DBVS tipų reikšmės taip pat galima transformuoti skaliarinėmis funkcijomis. Pvz., DBVS DB2 darbuotojų kategorijų vidurkį šimtųjų tikslumu galima apskaičiuoti tokia užklausa

```
SELECT DECIMAL(AVG(FLOAT(Kategorija)), 5, 2)) AS "Kategorijų vidurkis"  
FROM Vykdytojai.
```

Kai standartinės DBVS galimybės vartotoją netenkina, SQL2 standartą užtikrinančiose sistemose vartotojas gali pats apibrėžti reikiamas tipų transformavimo funkcijas. Tokių funkcijų naudojimą aptarsime tolimesniuose skyreliuose.