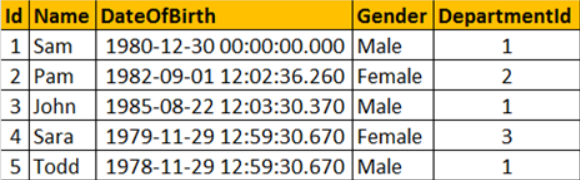
32. Mitme avaldisega tabeliväärtusega funktsioonid

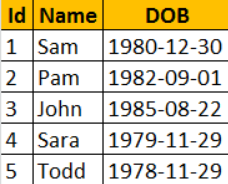
Mitme avaldisega tabeliväärtusega funktsioonid on väga sarnased tabelisiseste väärtustega funktsioonidele.

Selline peaks välja nägema Employees tabel:



Pilt 32.1

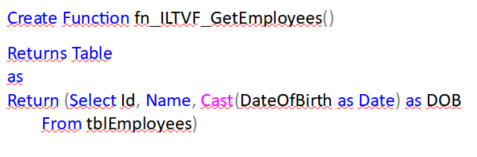
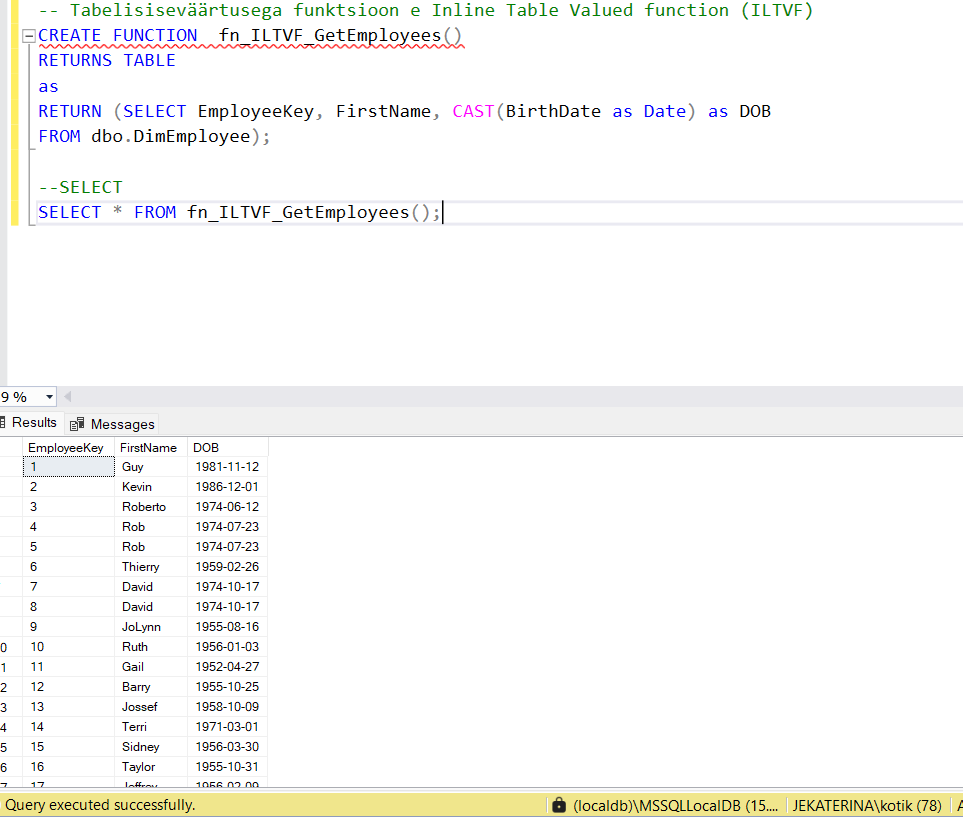
Nüüd teeme tabelisisese ja mitme avaldisega tabeliväärtusega funktsiooni, mis mõlemal juhul annab sama tulemuse:



Pilt 32.2

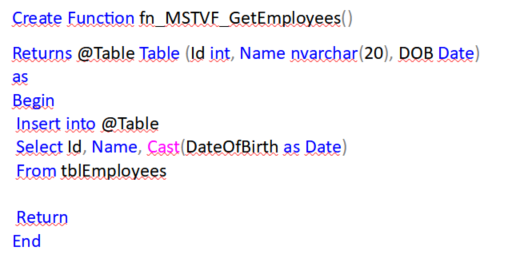
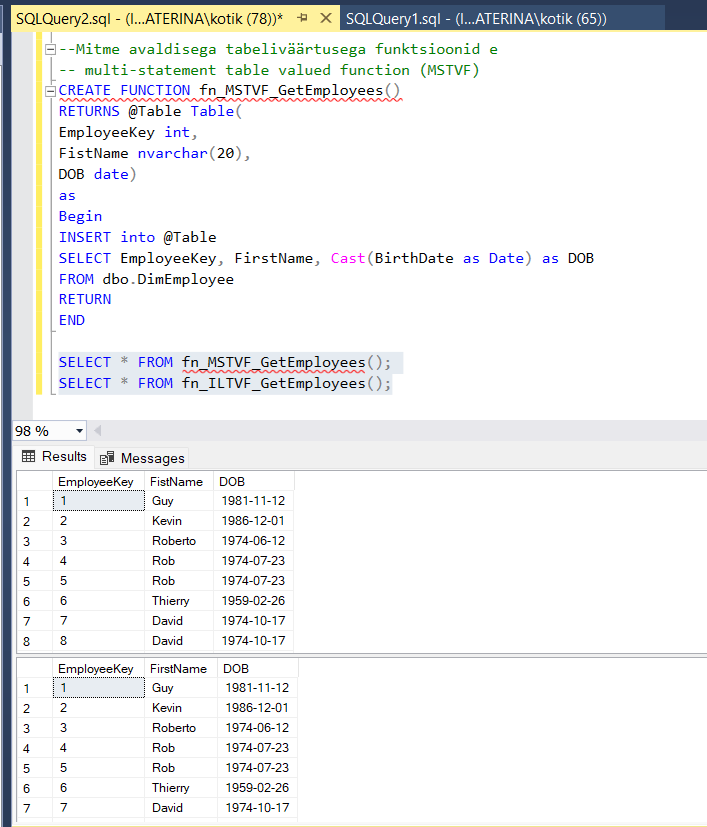
Tabelisiseväärtusega funktsioon e Inline Table Valued function (ILTVF) koodinäide:

**Õpilane teeb päringu ja näitab koodi kirja ning tulemust pildi kujul.**

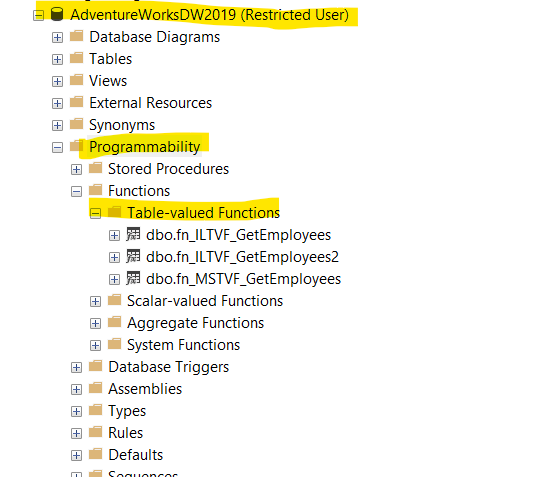
Mitme avaldisega tabeliväärtusega funktsioonid e multi-statement table valued function (MSTVF):

**Õpilane teeb päringu ja näitab koodi kirja ning tulemust pildi kujul.**

Kui nüüd soovid mõlemat funktsiooni esile kutsuda, siis kasutad koodi:

**Õpilane teeb päringu ja näitab koodi kirja ning tulemust pildi kujul.**



Select \* from fn\_ILTVF\_GetEmployees()

Select \* from fn\_MSTVF\_GetEmployees()

Nüüd võrdleme eelnevalt käsitletud kahte erinevat funktsiooni omavahel.

1. Tabelisiseväärtusega funktsioonis RETURNS klausel ei saa sisaldada tabeli struktuuri, see vaid tagastab. Samas mitme avaldisega tabeliväärtusega täpsustatakse tabeli struktuur ja see saab tagastatud.

2. ILTVF ei saa olla BEGIN ja END plokki, aga MSTVF on olemas.

3. ILTVF on parema jõudlusega. Kui on valida, et kumba kasutada, siis soovitav oleks alati ILTVF kasutada.

4. ILTVF juures on selle alla kuuluvat tabelit võimalik muuta, aga MSTVF puhul ei saa.

Uuendame allasuvat tabelit ja kasutame selleks ILTVF funktsiooni.

See päring uuendab Sam-i Sam1 peale tblEmployee allasuvas tabelis. Kui proovid seda teha MSTVF funktsiooniga, siis saad veateate 'Object 'fn\_MSTVF\_GetEmployees' cannot be modified.'.

75

ILTVF parema jõudluse põhjus on see, et SQL Server käsitleb seda, kui view-d ja MSTVF stored procedureina.

