

Подупити

SQL дозвољава да се пишу упити који су саставни део других упита и тада се називају угњеждени упити или подупити и увек се пишу унутар малих заграда.

Подупит је SELECT наредба уграђена у другу SELECT, INSERT, UPDATE или DELETE наредбу.

Угњеждени упит се често назива и унутрашњи упит а упит у којем се он налази се назива спољашњи упит.

Подупит увек враћа резултат спољашњем упиту.

Према врсти резултата које враћају подупити могу бити:

- Скаларни – враћају једну вредност
- Вишевредносни – враћа резултат сличан табели са једном колоном
- Табеларни – враћа резултат у виду табеле и тада се подупит пише иза FROM

Подупити могу бити самостални (извршавају се независно од спољног упита) или зависни (реферишу на једну или више колона из спољног упита).

Скаларни подупити

Могу се користити било где у спољном упиту (у SELECT, WHERE, HAVING клаузулама).

Ако подупит враћа празан скуп, тај резултат се враћа као NULL.

Пример:

```
SELECT  NarudzbinaID,
        ProizvodMagacinID,
        Kolicina,
        CenaProizvoda
FROM Prodaja.Posiljke
WHERE narudzbenicaid =
(
    SELECT MAX(NarudzbenicaID) AS poslednja_narudzbenica FROM Prodaja.Posiljke
);
```

Пример даје подупит који је самосталан и враћа једну вредност: највећи број наруџбенице.

Враћени број из подупита се користи да филтрира податке који ће се приказати.

Спољашњи упит ће приказати редове из табеле Продаја.Пошиљке али само за ону наруџбеницу чији је број једнак највећем броју креиране наруџбенице.

У овом случају, резултат извршавања упита враћа само један ред, зато што последња креирана наруџбеница има само једну ставку:

NarudzbenicaID	ProizvodMagacinID	Kolicina	CenaProizvoda
73595	80	120	18.00

Пример:

```
SELECT  MusterijaID,
        NarudzbinaID,
        DatumNarucivanja
FROM Prodaja.Posiljke AS spoljasnji
WHERE datumnarucivanja =
(
    SELECT MAX(DatumNarucivanja)
    FROM Prodaja.Posiljke AS unutrasnji
```

```

    WHERE unutrasnji.MusterijaID = spoljasnji.MusterijaID
)
ORDER BY MusterijaID;

```

На примеру је скаларни подупит који зависи од вредности из спољашњег упита.

Упит приказује податке о последњој креираној наруџбеници за сваког купца.

Пример користи зависни подупит да врати најновије наруџбенице за сваког купца.

Подупи обухвата вредност MušterijaID из спољног упита, користи га у WHERE услови у подупиту и враћа резултат спољашњом упиту који служи за филтрирање.

Зависни подупити се обрађују више пута.

Разлог за то је тај што се прво извршавају спољни упити који прослеђује вредност унутрашњем упиту, који се тек тада извршава.

И за сваки ред враћен из спољног упита, једном се извршава унутрашњи упит.

Пример даје резултат:

MusterijaID	NarudzbenicaID	DatumNarucivanja
1	7390	2016-05-27
2	72475	2016-05-14
3	73465	2016-05-30

Пример:

```

SELECT MusterijaID,
       MusterijaIme,
       (
         SELECT MAX(DatumNarucivanja)
         FROM Posiljke AS unutrasnji
         WHERE unutrasnji.MusterijaID = spoljasnji.MusterijaID
       ) AS poslednja_narudzbenica
FROM Musterije AS spoljasnji
ORDER BY MusterijaID;

```

У примеру се користи скаларни, завистан подупит у оквиру SELECT листе.

SELECT наредба враћа колоне: MusterijaID, MusterijaIme.

Упит, на основу подупита, враћа и израчунату колону (израз) последња_наруџбеница, која садржи последњи датум поруџбенице за сваког купца.

Извршавање упита враћа онолико редова колико има купаца у табели пошто је сваки купац макар једном нешто купио.

MusterijaID	MusterijaIme	poslednja_narudzbenica
1	Miki Mikic	2020-05-27
2	Ana Anic	2020-05-14
3	Dragan Dragic	2021-02-18

Вишеверносни подупити

Вишеверносни подупит враћа спољњем упиту више вредности у форми табеле која има једну колону и један или више редова.

Овакав тип подупита може се користити на местима где се очекује резултат са више вредности.

```
SELECT      so.MusterijaID,
            so.NarudzbenicaID,
            so.DatumNarucivanja
FROM Narudzbenice AS so
      INNER JOIN Musterije AS sc
      ON so.MusterijaID = sc.MusterijaID
WHERE so.MusterijaID IN
(
    SELECT MusterijaID
    FROM Musterije AS sc
    INNER JOIN Gradovi AS ac
    ON sc.PostanskiKodID = ac.GradID
    WHERE ac.GradIme = 'Beograd'
)
ORDER BY so.MusterijaID, NarudzbenicaID;
```

Упит треба да врати све наруџбенице купаца из града Београд.

Шифре купаца из Београда могу се добити помоћу самосталног подупита у којем су спојене табеле Musterije i Gradovi.

Услов за филтрирање у оквиру WHERE клаузуле садржи оператор IN, јер из овог града може бити више купаца.

MusterijaID	NarudzbenicaID	DatumNarucivanja
345	102	2021-03-14
345	105	2021-04-10
567	100	2021-05-10
3456	607	2021-06-11

Табеларни подупити

Постоји два типа табеларних подупита: изведене табеле и Common Table Expressions (CTE).

Изведена табела је именовани подупит, угњежђен унутар FROM клаузуле спољашњег упита.

Изведен табеле се користе да би скратиле дуге упите и оне су привремене тј не памте се у базама података.

Код њих се морају користити алијаси, не сме се користити ORDER BY.

Општа синтакса изведених табела:

```
SELECT lista_kolona_u_spoljnom_upitu
FROM
(
    SELECT lista_kolona_u_unutrasnjem_upitu
    FROM tabela_ili_pogled
)AS alias_za_izvedenu_tabelu;
```

Пример:

```
SELECT godina,
        COUNT(DISTINCT kupac) AS broj_kupca
FROM
(
    SELECT YEAR(datumnarucivanja) AS godina,
           MusterijaID AS kupac
    FROM Prodaja.Narudzbine
) AS godine_kupci
GROUP BY godina
ORDER BY godina;
```

Пример приказује број јединствених купаца по годинама.

Иза FROM клаузуле се користи изведена табела, којој је преко алијаса додељено име godine_kupci.

Називи колона се дефинисани у оквиру SELECT листе изведене табеле.

Ови називи су доступни у наредним корацима код извршавања упита.

Због тога, GROUP BY клаузула и SELECT листа у спољном упиту не морају да користе функцију YEAR(datumnarucivanja) већ могу да користе називе колона, дефинисане за изведену табелу.

godina	broj_kupaca
2013	625
2014	640
2015	663

Задачи за самосталан рад

1. Написати упит који ће вратити шифре и имена купаца који нису обавили ниједну куповину.
2. Написати SELECT наредбу која ће вратити колону MusterijaID, просечну и укупну куповину за сваког купца. Филтрирати податке, тако да буду укључени само купци, чије су појединачне просечне куповине веће од просечне вредности свих куповина (користити скаларни упит, самостални подупит за филтрирање)