

# 2o Project, Σχεδιασμός βάσεων Δεδομένων

p3170122 | Κων/νος Νικολούτσος

## Άσκηση 1

1.

Φτιάχνουμε μια βάση δεδομένων με όνομα LIBDW και θέτοντας collation ως GREEK\_CI\_AS όπως στο πρώτο project.

Η διαδικασία αυτή έγινε από το GUI του MS Server Management 2018

2.

Ας φτιάξουμε το fact table μας και τα dimension tables.

Διαβάζοντας προσεκτικά τις απαιτήσεις της βιβλιοθήκης προκύπτει:

```
CREATE TABLE Dim_date
(
    id date,
    year int,
    month int,
    dayofmonth int,
    quarter int,
    week int,
    dayofweek int,
    Dayofyear int,
    PRIMARY KEY(id)
);
```

```
CREATE TABLE Dim_location
(
    id int IDENTITY(1, 1),
    copyloc varchar(100) UNIQUE,
    PRIMARY KEY(id)
);
```

```
CREATE TABLE Dim_material
```

```
(
    id int IDENTITY(1, 1),
    material varchar(100) UNIQUE,
    PRIMARY KEY(id)
);
```

```
CREATE TABLE Dim_department
(
    id int IDENTITY(1, 1),
    dep_name varchar(200) UNIQUE,
    PRIMARY KEY(id)
);
```

```
CREATE TABLE Dim_sex
(
    id int IDENTITY(1, 1),
    sex varchar(1) UNIQUE,
    PRIMARY KEY(id)
);
```

```
CREATE TABLE Fact_loans
(
    key_date date,
    key_location int,
    key_material int,
    key_depcode int,
    key_sex int,
    loans_total int

    FOREIGN KEY (key_date) REFERENCES Dim_date(id),
    FOREIGN KEY (key_location) REFERENCES Dim_location(id),
    FOREIGN KEY (key_material) REFERENCES Dim_material(id),
    FOREIGN KEY (key_depcode) REFERENCES Dim_department(id),
    FOREIGN KEY (key_sex) REFERENCES Dim_sex(id)
);
```

### 3.

Εισάγουμε στο DataWarehouse τα δεδομένα:

```
INSERT INTO LIBDW.dbo.Dim_location  
    SELECT DISTINCT copyloc FROM LIBRARY.dbo.copies
```

```
INSERT INTO LIBDW.dbo.Dim_material  
    SELECT DISTINCT material FROM LIBRARY.dbo.bibreca
```

```
INSERT INTO LIBDW.dbo.Dim_sex  
    SELECT DISTINCT sex FROM LIBRARY.dbo.borrowers
```

```
INSERT INTO LIBDW.dbo.Dim_date  
    SELECT DISTINCT loandate,  
        datepart(year, loandate),  
        datepart(month, loandate),  
        datepart(day, loandate),  
        datepart(quarter, loandate),  
        datepart(week, loandate),  
        datepart(WEEKDAY, loandate),  
        datepart(dayofyear, loandate)  
    FROM LIBRARY.dbo.loanstats
```

```

INSERT INTO LIBDW.dbo.Fact_loans
    SELECT d.loandate AS key_date,
           dim_loc.id AS key_location,
           dim_mat.id AS key_material,
           dim_dep.id AS key_depcode,
           dim_sex.id AS key_sex,
           count(*) AS loans_total
    FROM (SELECT loandate, copyloc, material, sex, depname FROM
LIBRARY.dbo.loanstats ls
        JOIN LIBRARY.dbo.copies c ON ls.copyno = c.copyno
        JOIN LIBRARY.dbo.bibrecs bi ON c.bibno = bi.bibno
        JOIN LIBRARY.dbo.borrowers bo ON ls.bid = bo.bid
        JOIN LIBRARY.dbo.departments d ON d.depcode = bo.depcode) AS d
    JOIN LIBDW.dbo.Dim_location dim_loc ON d.copyloc = dim_loc.copyloc
    JOIN LIBDW.dbo.Dim_material dim_mat ON d.material = dim_mat.material
    JOIN LIBDW.dbo.Dim_sex dim_sex ON d.sex = dim_sex.sex
    JOIN LIBDW.dbo.Dim_department dim_dep ON d.depname = dim_dep.dep_name
    GROUP BY d.loandate, dim_loc.id, dim_mat.id, dim_sex.id, dim_dep.id

```

4.



Άσκηση 2

1.

```
SELECT year, key_depcode, SUM(loans_total)
FROM Fact_loans as fl
JOIN Dim_date as dim_d ON dim_d.id = fl.key_date
GROUP BY year, key_depcode
```

2.

```
SELECT key_location, key_material, SUM(loans_total) AS total_loans
FROM Fact_loans
GROUP BY key_location, key_material
```

3.

```
SELECT month, key_sex, SUM(loans_total) FROM Fact_loans
JOIN Dim_date ON key_date = Dim_date.id
WHERE year = '2000'
GROUP BY month, key_sex
```

4.

```
SELECT year, month, SUM(loans_total) FROM Fact_loans
JOIN Dim_date ON key_date = Dim_date.id
GROUP BY year, month
HAVING SUM(loans_total) > 800
```

5.

```
SELECT year, key_depcode, key_sex, SUM(loans_total) FROM Fact_loans
JOIN Dim_date ON key_date = Dim_date.id
GROUP BY ROLLUP( year, key_depcode, key_sex)
```

6.

```
CREATE VIEW [Temp] AS
SELECT key_depcode, key_sex, SUM(loans_total) as loans_total FROM Fact_loans
JOIN Dim_date ON key_date = Dim_date.id
GROUP BY key_depcode, key_sex

SELECT females.key_depcode
FROM Temp as females, Temp AS males
WHERE females.key_depcode = males.key_depcode
AND females.key_sex = '1'
```

```
AND males.key_sex = '2'  
AND females.loans_total > males.loans_total
```

## Άσκηση 3

1.

```
SELECT year, key_location, key_sex, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY CUBE(year, key_location, key_sex)
```

2.

Παρατηρούμε οτι χωρίς το CUBE keyword θα επρεπε να γράφαμε όλους τους συνδιασμούς

```
(SELECT year, key_location, key_sex, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY year, key_location, key_sex)
```

UNION

```
(SELECT NULL, key_location, key_sex, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY key_location, key_sex)
```

UNION

```
(SELECT year, NULL, key_sex, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY year, key_sex)
```

UNION

```
(SELECT year, key_location, NULL, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY year, key_location)
```

UNION

```
(SELECT year, NULL, NULL, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY year)
```

UNION

```
(SELECT NULL, NULL, key_sex, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY key_sex)
```

UNION

```
(SELECT NULL, key_location, NULL, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY key_location)
```

UNION

```
(SELECT NULL, NULL, NULL, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id )
```

## Άσκηση 4

Για να λύσουμε αυτό το ερώτημα αρκεί να φτιάξουμε ένα table στο οποίο θα αποθηκευτούν όλα τα queries της μορφής GROUP BY που γράψαμε στο ερώτημα 3β.

Έτσι λοιπόν θα έχουμε το cube αποθηκευμένο στην βάση. Φυσικά κάθε φορά που θα θελήσουμε να τον ανανεώσουμε είτε θα πρέπει να θέσουμε triggers είτε να το κάνουμε manually.



Ακολουθεί ο κώδικας για το load του κύβου:

```
(SELECT year, key_location, key_sex, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY year, key_location, key_sex)
```

UNION

```
(SELECT NULL, key_location, key_sex, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY key_location, key_sex)
```

UNION

```
(SELECT year, NULL, key_sex, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY year, key_sex)
```

UNION

```
(SELECT year, key_location, NULL, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY year, key_location)
```

UNION

```
(SELECT year, NULL, NULL, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY year)
```

UNION

```
(SELECT NULL, NULL, key_sex, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY key_sex)
```

UNION

```
(SELECT NULL, key_location, NULL, SUM(loans_total) FROM Fact_loans  
JOIN Dim_date ON key_date = Dim_date.id  
GROUP BY key_location)
```

UNION

```
(SELECT NULL, NULL, NULL, SUM(loans_total) FROM Fact_loans
```

```
JOIN Dim_date ON key_date = Dim_date.id )
```