

## ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

### PROJECT B

ΑΠΟΣΤΟΛΟΠΟΥΛΟΣ ΘΕΜΙΣΤΟΚΛΗΣ p3180013

Ζήτημα πρώτο:

1) Δημιούργησα την τοπική βάση CAMPDW στον sql manager studio και έκανα insert τα δεδομένα που μας δόθηκαν. Η διαδικασία αυτή φαίνεται στον κώδικα:

```
CREATE TABLE campdata(  
    custID int,  
    fname varchar(30),  
    lname varchar(30),  
    cID int,  
    country varchar(30),  
    bookID int,  
    bookDate date,  
    campCode char(3),  
    campName Varchar(50),  
    empno int,  
    catCode char(1),  
    category varchar(20),  
    unitCost numeric(4,2),  
    startDate date,  
    overnights int,  
    persons int  
);
```

```
BULK INSERT campdata  
FROM 'C:\data\CAMPDATA.TXT'  
WITH (FIRSTROW =2, FIELDTERMINATOR='|', ROWTERMINATOR = '\n');
```

2,3) Διαβάζοντας την εκφώνηση (και βλέποντας ότι μας ενδιαφέρουν αναφορές ανα χώρα, τουριστικό γραφείο, κατασκήνωση και θέση κατασκήνωσης) αποφάσισα στην δημιουργία των εξής «περιφερειακών» tables που θα απαρτίσουν τον fact table(bookings):

Table customer που είναι πίνακας με στοιχεία των γραφείων και των υπεύθυνων τους (στον κώδικα φαίνεται και η εισαγωγή των δεδομένων στον πίνακα)

```
CREATE TABLE customer(  
    custID int primary key,
```

```

    fname varchar(30),
    lname varchar(30)
)

```

```

insert into customer
select distinct custID, fname, lname from campdata;

```

Table country με εγγραφές τις χώρες των τουριστικών γραφείων

```

CREATE TABLE country(
    cID int primary key,
    country varchar(30)
)

```

```

insert into country
select distinct cID, country from campdata;

```

Table Camp με εγγραφές τις κατασκηνώσεις

```

CREATE TABLE Camp(
    campCode char(3) primary key,
    campName Varchar(50),
);

```

```

insert into Camp
select distinct campCode, campName from campdata;

```

Table seatCategory με εγγραφές τις κατηγορίες θέσεων κατασκήνωσης:

```

CREATE TABLE seatCategory(
    catCode char(1) primary key,
    category varchar(20) ,
    unitCost numeric(4,2)
)

```

```

insert into seatCategory
select distinct catCode, category, unitCost from campdata;

```

Table timeInfo με εγγραφές που έχουν σχέση με ημερομηνίες(χρόνο) startDate

```

CREATE TABLE timeInfo(
    time_key datetime primary key,
    t_year int,
    t_month int,
    t_dayofmonth int,
    t_quarter int,
    t_week int,
    t_dayofyear int,
    t_dayofweek int
);

```

```

insert into timeInfo
select distinct startDate, datepart(year, startDate), datepart(month,startDate),
datepart(day,startDate),
datepart(quarter,startDate), datepart(week,startDate),

```

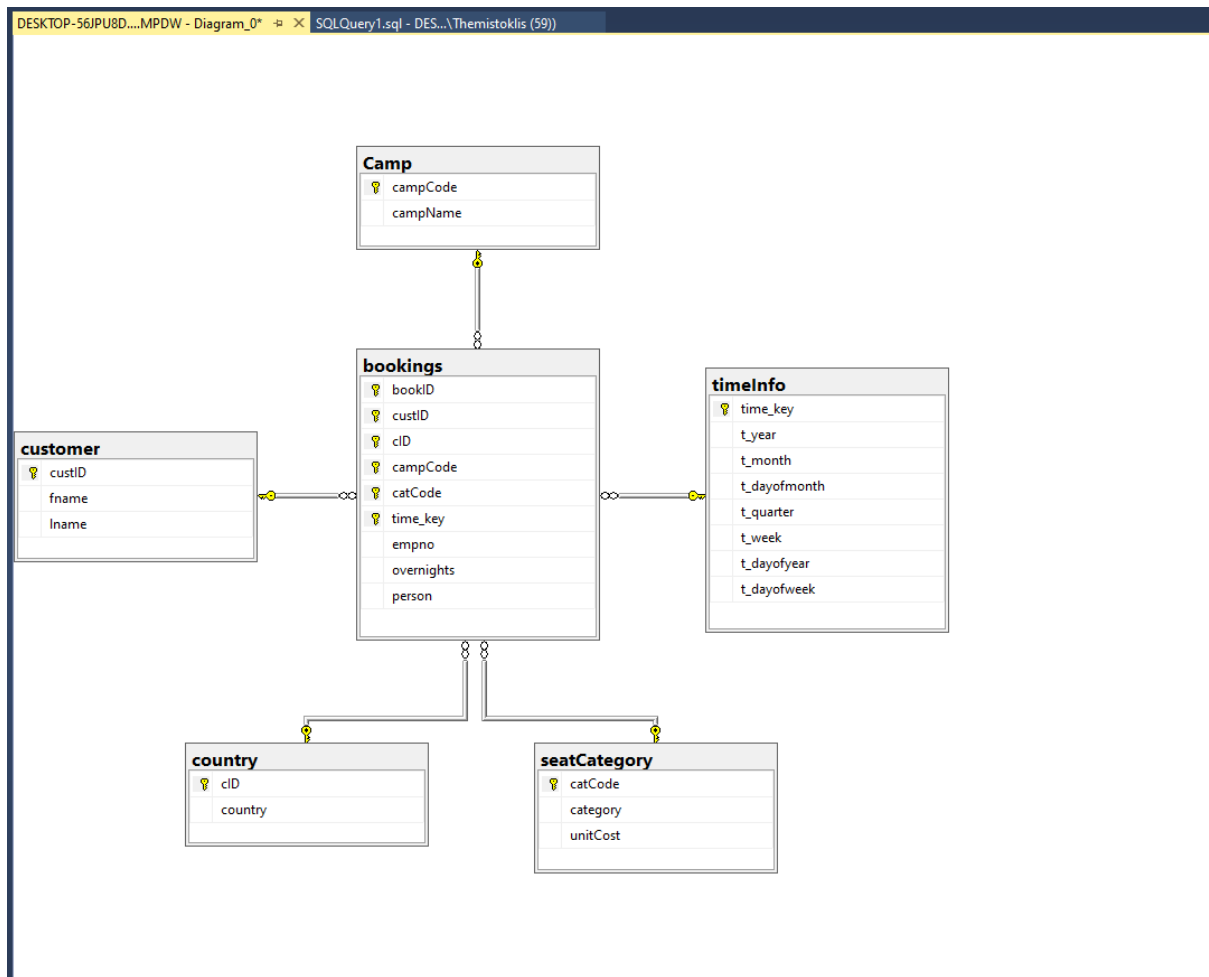
```
datepart(dayofyear,startDate),datepart(dw,startDate)
from campdata;
```

Τώρα είμαι έτοιμος να δημιουργήσω τον fact table bookings με εγγραφές που δείχνουν κρατήσεις.

```
CREATE TABLE bookings(
    bookID int,
    custID int,
    cID int,
    campCode char(3),
    catCode char(1),
    time_key datetime,
    empno int,
    overnights int,
    person int,
    foreign key(time_key) references timeInfo(time_key),
    foreign key(catCode) references seatCategory(catCode),
    foreign key(campCode) references Camp(campCode),
    foreign key(cID) references country(cID),
    foreign key(custID) references customer(custID),
    primary key(bookID,custID,cID,campCode,catCode,time_key)
);

insert into bookings
select bookID,custID, cID, campCode, catCode, startDate, sum(empno),
sum(overnights), sum(persons) from campdata group by
bookID,custID,cID,campCode,catCode,startDate
```

4)Το διάγραμμα που παράχθηκε από τον sql manager studio είναι το ακόλουθο:



## Ζήτημα 2

Με την αποθήκη δεδομένων που δημιούργησα θα απαντήσω στα επόμενα ερωτήματα ως εξής:

1. Εμφανίστε έναν κατάλογο με τους πρώτους εκατό καλύτερους πελάτες της εταιρείας. Ο κατάλογος θα περιέχει την χώρα, το ονοματεπώνυμο του υπεύθυνου του τουριστικού γραφείου και την συνολική αξία των κρατήσεων που έχουν γίνει από το συγκεκριμένο τουριστικό γραφείο.

```
select top 100 country, fname, lname, sum(unitCost*overnights*person) as  
total_cost  
from bookings, country, customer, seatCategory
```

```

where bookings.custID = customer.custID and bookings.cID = country.cID and
bookings.catCode = seatCategory.catCode
group by bookings.custID, country, fname, lname

```

Results

Messages

```

(100 rows affected)

Completion time: 2021-06-06T13:05:19.9448663+03:00
|

```

2) Εμφανίστε ένα κατάλογο με την συνολική αξία των κρατήσεων κάθε κατασκήνωσης ανά κατηγορία θέσης για το έτος 2000 (Όνομα Κατασκήνωσης, Κατηγορία Θέσης, Αξία κρατήσεων).

```

select campName, category, sum(unitCost*overnights*person) as total_cost
from bookings, Camp, seatCategory, timeInfo
where bookings.campCode = Camp.campCode and bookings.catCode = seatCategory.catCode
and bookings.time_key = timeInfo.time_key and timeInfo.t_year = 2000
group by campName, category

```

	campName	category	total_cost
1	Apollonia	Camper Van	175440.00
2	Dionysus	Camper Van	550620.00
3	Elbianos	Camper Van	381705.00
4	Kissamos	Camper Van	66120.00
5	Rosibos	Camper Van	1979520.00
6	Apollonia	Caravan	53190.00
7	Dionysus	Caravan	87580.00
8	Elbianos	Caravan	69800.00
9	Kissamos	Caravan	33880.00
10	Rosibos	Caravan	130270.00
11	Apollonia	Tent	25090.00
12	Dionysus	Tent	31115.00
13	Elbianos	Tent	34970.00
14	Kissamos	Tent	17690.00
15	Rosibos	Tent	55335.00

```

(15 rows affected)

Completion time: 2021-06-06T13:07:55.7843924+03:00
|

```

3) Εμφανίστε έναν κατάλογο με την συνολική αξία των κρατήσεων ανά κατασκήνωση σε μηνιαία βάση για το έτος 2018

```
select campName, timeInfo.t_month, sum(unitCost * overnights * person) as total_cost
from Camp, timeInfo, bookings, seatCategory
where bookings.campCode = Camp.campCode and bookings.catCode = seatCategory.catCode
and bookings.time_key = timeInfo.time_key and timeInfo.t_year = 2018
group by campName, timeInfo.t_month
```

(60 rows affected)

Completion time: 2021-06-06T13:11:00.9301536+03:00

4) Η διοίκηση της εταιρείας θέλει μία αναφορά που θα περιέχει τις ακόλουθες πληροφορίες. α. Τον συνολικό αριθμό των ενοικιαστών β. Τον αριθμό των ενοικιαστών ανά έτος γ. Τον αριθμό των ενοικιαστών ανά έτος και κατασκήνωση δ. Τον αριθμό των ενοικιαστών ανά έτος, κατασκήνωση και κατηγορία θέσης. Γράψτε μια επερώτηση σε γλώσσα SQL η οποία να παράγει την παραπάνω αναφορά.  
(Ο τελεστής COALESCE χρησιμοποιείται για να μην φαίνεται στο query που δεν υπάρχει η κατηγορία στο group by null. δηλαδή αν έχω στο rollup group by t\_year, campName η Τρίτη στήλη δεν θα δείχνει numm αλλά All categories)

```
select t_year as year_, COALESCE(campName, 'All camps') as camp_name,
COALESCE(category, 'All categories') as category_name, count(bookings.custID) as
num_of_customers
from bookings, timeInfo, Camp, seatCategory
where bookings.campCode = Camp.campCode and bookings.catCode = seatCategory.catCode
and timeInfo.time_key = bookings.time_key
group by rollup (timeInfo.t_year, campName, category)
```

(1044 rows affected)

Completion time: 2021-06-06T13:14:37.9984695+03:00

5) Εμφανίστε τα ονόματα των κατασκηνώσεων που δέχθηκαν περισσότερους ενοικιαστές το έτος 2018 σε σχέση με το 2017. Για το συγκεκριμένο ερώτημα επιτρέπεται η δημιουργία όψεων.(Δημιουργία 2 όψεων.Μία με αριθμό ενοικιαστών ανα κατασκήνωση το 2017 και μία το 2018.Μετά απλά επιλέγω της κατασκήνωσης όπου το 2018 είχαν περισσότερους ενοικιαστές)

```
create view view2017 as
select campName, count(custID) as num_of_customer
from bookings, Camp, timeInfo
where bookings.campCode = Camp.campCode and bookings.time_key = timeInfo.time_key and
timeInfo.t_year = '2017'
group by campName
```

```
create view view2018 as
select campName, count(custID) as num_of_customer
from bookings, Camp, timeInfo
where bookings.campCode = Camp.campCode and bookings.time_key = timeInfo.time_key and
timeInfo.t_year = '2018'
group by campName
```

```
select view2018.campName, view2018.num_of_customer, view2017.num_of_customer
from view2017, view2018
where view2018.num_of_customer > view2017.num_of_customer and view2018.campName =
view2017.campName
```

	campName	num_of_customer	num_of_customer
1	Kissamos	1070	1065
2	Elbianos	4770	4624

✓ Query executed successfully.

(Στην τελευταία στήλη φαίνονται οι ενοικιαστές στην συγκεκριμένη κατασκήνωση για το 2017)

### Ζήτημα 3

```
select COALESCE(country.country, 'All countries') as country_, COALESCE(campName, 'All camps') as camp_name, COALESCE(category, 'All categories') as category_name,
count(bookings.custID) as num_of_customers
from bookings, country, Camp, seatCategory
where bookings.campCode = Camp.campCode and bookings.catCode = seatCategory.catCode
and country.cID = bookings.cID
group by cube (country, campName, category);
```

Η παραπάνω ερώτηση δημιουργεί ένα κύβο με διαστάσεις που έχουν σχέση με χώρα, κατασκήνωση και κατηγορία θέσης κατασκήνωσης. Κάποιο κελί του συγκεκριμένου κύβου μας δείχνει σε μία συγκεκριμένη χώρα, σε συγκεκριμένη κατασκήνωση και συγκεκριμένη θέση πόσοι ενοικιαστές υπάρχουν.

	country_	camp_name	category_name	num_of_customers
1	Austria	Apollonia	Camper Van	1807

(παραδειγμα για χώρα=αυστρία, κατασκήνωση = απολώνια, κατηγορία θέσης = Camper van.

Τα group by που γίνονται στον κύβο είναι τα ακόλουθα:

Country, campName, catCode (group by country, campName, catCode)

Country, campName, null (group by country, campName)

Country, null, catCode (group by country, catCode)

null, campName, catCode (group by campName, catCode)

Country, null, null (group by country)

Null, campName, null (group by campName)

Null, null, catCode (group by catCode)

Null, null, null (group by None)

## Ζήτημα 4

Το ευρετήριο bitmap πάνω στις χώρες:



RowId	1	2	3	4	5	6	7	8	9	10
69	0	0	0	1	0	0	0	0	0	0
70	0	0	0	1	0	0	0	0	0	0
71	0	0	0	0	0	1	0	0	0	0
72	0	0	0	0	0	0	0	1	0	0
73	0	0	0	0	0	0	0	1	0	0
74	1	0	0	0	0	0	0	0	0	0
75	1	0	0	0	0	0	0	0	0	0
76	1	0	0	0	0	0	0	0	0	0

Όπου οι αριθμοί 1 έως 10 στην πρώτη γραμμή είναι τα cID των countries

	cID	country
1	1	Germany
2	2	Italy
3	3	France
4	4	Spain
5	5	Sweeden
6	6	Denmark
7	7	Netherlands
8	8	Austria
9	9	Belgium
10	10	Finland