

Project 2

Σχεδιασμός Βάσεων Δεδομένων

Καρράς Κωνσταντίνος 3180076

❖ Ζήτημα Πρώτο

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

```
BULK INSERT campdata  
FROM 'C:\Users\Kostas\Desktop\Σχολή 6ο εξάμηνο\Project 2 Σχεδιασμός  
Βάσεων Δεδομένων\CAMPDATA\CAMPDATA.TXT'  
WITH (FIRSTROW =2, FIELDTERMINATOR='|', ROWTERMINATOR = '\n');
```

2. `create table` Dimension_Country(
 cID `integer`,
 country `varchar`(30),
 primary key(cID)
);

`create table` Dimension_Customer(
 custID `integer`,
 fname `varchar`(30),
 lname `varchar`(30),
 primary key(custID)
);

`create table` Dimension_Camp(
 campCode `char`(3),
 campName `Varchar`(50),
 primary key(campCode)
);

`create table` Dimension_Category(
 catCode `char`(1),
 category `varchar`(20),
 unitCost `numeric`(4,2),
 primary key(catCode)
);

`create table` Dimension_Time(
 id `date`,
 day `int`,
 month `int`,
 year `int`,
 primary key(id)
);

```

create table Fact(
    cID integer,

    custID integer,

    campCode char(3),

    catCode char(1),

    startDate date,

    empno integer,
    bookID integer,
    bookDate date,
    overnights integer,
    persons integer,

    foreign key (cID) references Dimension_Country (cID),
    foreign key (custID) references Dimension_Customer (custID),
    foreign key (campCode) references Dimension_Camp (campCode),
    foreign key (catCode) references Dimension_Category (catCode),
    foreign key (startDate) references Dimension_Time (id)
);

```

3. **insert into** Dimension_Country **select distinct** cID, country **from** campdata;

```

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

```

```

insert into Dimension_Camp select distinct campCode, campName from
campdata;

```

```

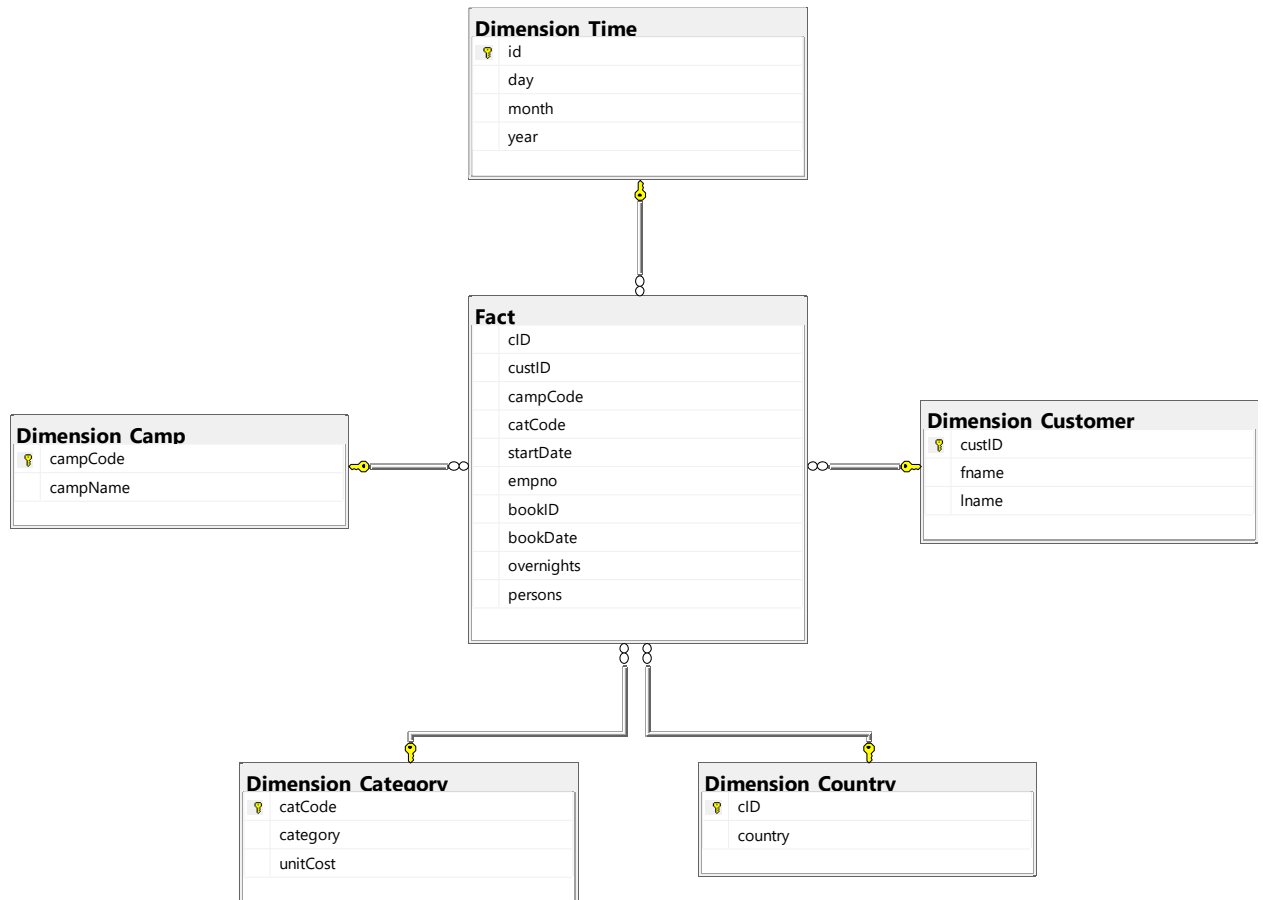
insert into Dimension_Category select distinct catCode, category, unitCost
from campdata;

```

```
insert into Dimension_Time select distinct startDate, datepart(day,
startDate), datepart(month, startDate), datepart(year, startDate) from
campdata;
```

```
insert into Fact select clD, custID, campCode, catCode, startDate, empno,
bookID, bookDate, overnights, persons from campdata;
```

4.



❖ Ζήτημα Δεύτερο

1. `select top 100 country, fname, lname, sum(overnights * unitCost * persons)
as earnings
from Dimension_Customer, Dimension_Country, Dimension_Category, Fact
where Dimension_Customer.custID = Fact.custID and
Dimension_Country.cID = Fact.cID and Dimension_Category.catCode =
Fact.catCode
group by country, fname, lname
order by earnings desc;`
2. `select Dimension_Camp.campName, category, sum(overnights * unitCost *
persons) as earnings
from Dimension_Camp, Dimension_Time, Dimension_Category, Fact
where Dimension_Camp.campCode = Fact.campCode and
Dimension_Time.id = Fact.startDate and Dimension_Category.catCode =
Fact.catCode and year = '2000'
group by category, Dimension_Camp.campName;`
3. `select Dimension_Camp.campName, month, sum(overnights * unitCost *
persons) as earnings
from Dimension_Camp, Dimension_Time, Dimension_Category, Fact
where Dimension_Camp.campCode = Fact.campCode and
Dimension_Time.id = Fact.startDate and Dimension_Category.catCode =
Fact.catCode and year = '2018'
group by month, Dimension_Camp.campName;`

4. `select year, Dimension_Camp.campName, category, count(bookID) as rents
from Dimension_Camp, Dimension_Time, Dimension_Category, Fact
where Dimension_Camp.campCode = Fact.campCode and
Dimension_Time.id = Fact.startDate and
Dimension_Category.catCode = Fact.catCode
group by rollup(year, Dimension_Camp.campName, category);`

5. `create view rentals2018 as
select Dimension_Camp.campName, count(bookID) as rents
from Dimension_Camp, Dimension_Time, Fact
where Dimension_Camp.campCode = Fact.campCode and
Dimension_Time.id = Fact.startDate and year = '2018'
group by Dimension_Camp.campName;`

`create view rentals2017 as
select Dimension_Camp.campName, count(bookID) as rents
from Dimension_Camp, Dimension_Time, Fact
where Dimension_Camp.campCode = Fact.campCode and
Dimension_Time.id = Fact.startDate and year = '2017'
group by Dimension_Camp.campName;`

`select distinct rentals2018.campName from rentals2017, rentals2018
where rentals2017.campName = rentals2018.campName and
rentals2018.rents > rentals2017.rents;`

❖ Ζήτημα Τρίτο

Οι 3 διαστάσεις που χρησιμοποιώ είναι το **όνομα της κατασκήνωσης**, ο **κωδικός κατηγορίας** και ο **κωδικός της χώρας**.

```
select Dimension_Camp.campName, Dimension_Category.catCode,  
Dimension_Country.clD, sum(overnights * unitCost * persons) as earnings  
from Dimension_Camp, Dimension_Category, Dimension_Country, Fact  
where Dimension_Camp.campCode = Fact.campCode and  
Dimension_Category.catCode = Fact.catCode and Dimension_Country.clD =  
Fact.clD  
group by cube (Dimension_Camp.campName, Dimension_Category.catCode,  
Dimension_Country.clD)
```

- a. Ο κύβος δείχνει όλους τους πιθανούς συνδυασμούς των 3 διαστάσεων. Αρχικά, ξεκινάει ανά όνομα, κωδικό κατηγορίας, και κωδικό χώρας. Για κάθε διαφορετικό όνομα (που είναι η πρώτη διάσταση) κρατάει σταθερή την κατηγορία και τη χώρα. Μόλις τελειώσουν τα ονόματα των κατασκηνώσεων που υπάρχουν το όνομα γίνεται NULL και παραμένουν σταθερά ο κωδικός κατηγορίας και ο κωδικός χώρας. Αυτό, ουσιαστικά, είναι το (ALL, κωδικός κατηγορίας, κωδικός χώρας). Ύστερα, και αφού έχουν περάσει όλα τα ονόματα των κατασκηνώσεων για τον ίδιο κωδικό κατηγορίας και χώρας, αλλάζει ο κωδικός κατηγορίας. Ακολουθείται η ίδια διαδικασία που περιγράφηκε και προηγουμένως. Για κάθε διαφορετικό όνομα κατασκήνωσης παραμένουν σταθερά ο κωδικός κατηγορίας και ο κωδικός χώρας (η διαφορά έγκειται στο γεγονός ότι τώρα η κατηγορία είναι διαφορετική, και παραμένει σταθερή μέχρι να αλλάξουν όλα τα ονόματα). Μόλις τελειώσουν τα ονόματα των κατασκηνώσεων που υπάρχουν το όνομα γίνεται NULL και παραμένουν σταθερά ο κωδικός κατηγορίας και ο κωδικός χώρας. Αυτό, ουσιαστικά, είναι το (ALL, **κωδικός κατηγορίας**, κωδικός χώρας). Έτσι θα συνεχίσουν να παράγονται αποτελέσματα και κάθε φορά θα αλλάζει και κάποιο στοιχείο της διάστασης ή συνδυασμός αυτών. Δεδομένου ότι υπάρχουν 5 ονόματα κατασκηνώσεων, 3 κωδικοί κατηγορίας και 10 κωδικοί χωρών το αποτέλεσμα που μου επιστρέφει ο SQL server είναι σωστό (μου

επιστρέφει 264 γραμμές), διότι $6 * 4 * 11 = 264$ (εννοείται πως κάθε μία από τις διαστάσεις θα παίρνει επιπλέον και την τιμή NULL).

b. `select` Dimension_Camp.campName, Dimension_Category.catCode,
Dimension_Country.cID,
`sum`(overnights * unitCost * persons) `as` earnings
`from` Dimension_Camp, Dimension_Category, Dimension_Country, Fact
`where` Dimension_Camp.campCode = Fact.campCode and
Dimension_Category.catCode = Fact.catCode and Dimension_Country.cID =
Fact.cID
`group by` Dimension_Camp.campName, Dimension_Category.catCode,
Dimension_Country.cID

`union`

`select` Dimension_Camp.campName, Dimension_Category.catCode, null,
`sum`(overnights * unitCost * persons) `as` earnings
`from` Dimension_Camp, Dimension_Category, Dimension_Country, Fact
`where` Dimension_Camp.campCode = Fact.campCode and
Dimension_Category.catCode = Fact.catCode and Dimension_Country.cID =
Fact.cID
`group by` Dimension_Camp.campName, Dimension_Category.catCode

`union`

`select` Dimension_Camp.campName, null, Dimension_Country.cID,
`sum`(overnights * unitCost * persons) `as` earnings
`from` Dimension_Camp, Dimension_Category, Dimension_Country, Fact
`where` Dimension_Camp.campCode = Fact.campCode and
Dimension_Category.catCode = Fact.catCode and Dimension_Country.cID =
Fact.cID
`group by` Dimension_Camp.campName, Dimension_Country.cID

`union`


```
select null, Dimension_Category.catCode, Dimension_Country.cID,  
sum(overnights * unitCost * persons) as earnings  
from Dimension_Camp, Dimension_Category, Dimension_Country, Fact  
where Dimension_Camp.campCode = Fact.campCode and  
Dimension_Category.catCode = Fact.catCode and Dimension_Country.cID =  
Fact.cID  
group by Dimension_Category.catCode, Dimension_Country.cID
```

union

```
select Dimension_Camp.campName, null, null,  
sum(overnights * unitCost * persons) as earnings  
from Dimension_Camp, Dimension_Category, Dimension_Country, Fact  
where Dimension_Camp.campCode = Fact.campCode and  
Dimension_Category.catCode = Fact.catCode and Dimension_Country.cID =  
Fact.cID  
group by Dimension_Camp.campName
```

union

```
select null, Dimension_Category.catCode, null,  
sum(overnights * unitCost * persons) as earnings  
from Dimension_Camp, Dimension_Category, Dimension_Country, Fact  
where Dimension_Camp.campCode = Fact.campCode and  
Dimension_Category.catCode = Fact.catCode and Dimension_Country.cID =  
Fact.cID  
group by Dimension_Category.catCode
```

union

```
select null, null, Dimension_Country.cID,  
sum(overnights * unitCost * persons) as earnings  
from Dimension_Camp, Dimension_Category, Dimension_Country, Fact  
where Dimension_Camp.campCode = Fact.campCode and  
Dimension_Category.catCode = Fact.catCode and Dimension_Country.cID =  
Fact.cID  
group by Dimension_Country.cID
```

