

TUTORIAL 5: Normalisation (SOLUTION)

Before they had their system properly designed, GCUTours got their 'IT Guy' to put together a database. After all, how hard could it be to put a few tables together?

IT Guy didn't really know about database design, so he made a bit of a mess of it. In this tutorial you will look at some of his work. You'll think about what he did wrong and about how to fix it.



Question 1

Here's IT Guy's first attempt at a table to store data about holiday **packages**, shown with some sample data:

Packages(packageID, location, name, description, price, departureDates)

| packageID | location | name | description | price | departureDates |
|-----------|----------|-------------------|---------------------|-----------|---------------------------------------|
| 1 | USA | Western | A typical tour is o | £1,499.00 | 1/3/2008,5/6/2008,2/9/2008 |
| 2 | Asia | Roof of the World | New this year is c | £1,599.00 | 1/3/2008,5/6/2008 |
| 3 | Europe | Alpine Action | There is adventure | £899.00 | 1/3/2008,2/9/2008,2/10/2008,3/11/2008 |

1. Write an SQL query which will find all holidays with a departure on 5th June 2008.

```
SELECT name, location FROM Packages
WHERE departureDates LIKE '*5/6/2008*'
```

2. Try writing a query which will list all departure dates and locations – can you do it?
3. Summarise the problems that this table design can cause?

need to extract dates from text string, cannot store dates as correct data type, need to type dates in exactly the right format

IT Guy tried again:

Packages(packageID, location, name, description, price, departureDate1, departureDate2, departureDate3, departureDate4)

| packageID | location | name | description | price | departureDate1 | departureDate2 | departureDate3 | departureDate4 |
|-----------|----------|-------------------|---------------------|-----------|----------------|----------------|----------------|----------------|
| 1 | USA | Western Adventur | A typical tour is o | £1,499.00 | 01/03/2008 | 05/06/2008 | 02/09/2008 | |
| 2 | Asia | Roof of the World | New this year is c | £1,599.00 | 01/03/2008 | 05/06/2008 | | |
| 3 | Europe | Alpine Action | There is adventure | £899.00 | 01/03/2008 | 02/09/2008 | 02/10/2008 | 03/11/2008 |

1. Write an SQL query which will find all holidays with a departure on 5th June 2008

```
SELECT name, location FROM Packages3
WHERE departureDate1 = #6/5/2008#
```

OR departureDate2= #6/5/2008#
 OR departureDate3= #6/5/2008#
 OR departureDate4= #6/5/2008#

2. In what ways is this better? Is it worse in any ways?

Can use correct date type, one item of data per field
 multiple similar fields, required data could be in any of them, what if we need more than 4?

3. Write descriptions, in the **TableName(fieldnames...)** notation, of the tables required to get this into 1NF.

Packages(packageID, location, name, description, price)
DepartureDates(departureDate, packageID)

Question 2

IT Guy figured that he needed a table to store **bookings**, so he came up with this:

Bookings(departureDate, name, username, firstname, lastname, numberOfPersons)

| departureDate | name | username | firstname | lastname | persons |
|---------------|----------------|----------|-----------|----------|---------|
| 01/03/2008 | Western Adven | mpolo | Marco | Polo | 2 |
| 01/03/2008 | Western Adven | vdagama | Vasco | daGama | 4 |
| 05/06/2008 | Borneo Explore | mpolo | Marco | Polo | 1 |
| 05/06/2008 | Western Adven | ferdy | Ferdinand | Magellan | 5 |
| 05/02/2008 | Colorado Winte | ferdy | Ferdinand | Magellan | 3 |

1. List the functional dependencies in this table.

departureDate, name, username → firstname, lastname, persons
 username → firstname, lastname

2. Why is this table not in 2NF?

dependency on **username** - part of primary key

3. Write descriptions, in the **TableName(fieldnames...)** notation, of the tables required to get this into 2NF.

Bookings(departureDate, name, username, persons)
Users(username, firstname, lastname)

Question 3

IT Guy's solution to the problems in Question 1 was a table where each row in the table represents a **tour** departing on a specific date. GCUTours management then said that the database needed to store discount information for each departure date. This was OK - IT Guy just had to add a new field to the Tours table:

Tours(tourID, departureDate, location, name, description, price, offer)

| tourID | departureDate | location | name | description | price | offer |
|--------|---------------|----------|---------------------------|----------------------|-----------|-------|
| 1 | 01/03/2008 | USA | Western Adventure | A typical tour is ou | £1,499.00 | 15 |
| 2 | 05/06/2008 | USA | Western Adventure | A typical tour is ou | £1,499.00 | 0 |
| 3 | 01/03/2008 | Asia | Roof of the World Tour | New this year is ou | £1,599.00 | 20 |
| 4 | 01/03/2008 | Europe | Alpine Action | There is adventure | £899.00 | 0 |
| 5 | 05/02/2008 | USA | Colorado Winter Adventure | When winter calls, | £1,099.00 | 10 |
| 6 | 05/06/2008 | Asia | Borneo Explorer | A 15 day safari exp | £1,699.00 | 5 |
| 7 | 05/06/2008 | Asia | Roof of the World Tour | New this year is ou | £1,599.00 | 15 |

1. List the functional dependencies in this table.

$tourID \rightarrow departureDate, location, name, description, price, offer$
 $name \rightarrow description, price, location$
 $description \rightarrow name, price, location$

2. Is this table in 2NF? How do you know?

yes it is - no dependencies on part of the primary key (in fact, there couldn't be any, as the PK is a single field)

3. Why is this table not in 3NF?

dependency on **name** - non-key field

4. Write descriptions, in the **TableName(fieldnames...)** notation, of the tables required to get this into 3NF.

Tours(tourID, departureDate, offer, name)
Packages(name, location, description, price)

OR

Tours(tourID, departureDate, offer, description)
Packages(name, location, description, price)