Database Methods Project 2007
B20028
This project is worth 50% of the total Module marks.

Outline:

The Town Council in Fubarstown is examining it's methods of generating revenue. In the process of doing this it has examined it's parking schemes.

Broadly, it finds that the general on-street parking system with discs is working satisfactorily. As such it has no intention of changing this scheme in ways that it might others.

However, within the general scheme is a sub-scheme to enable residents in the town centre to have parking permits which exclude them from the requirement to display per-occasion parking disks; residents are able to acquire a permit disk for their own street.

Like many local authorities, the desire to earn income has seemingly become an executive function of the unelected 'public servants' who work in the council. While many right and good thinking people feel that there is an increasing void between the electorate and the officials that the elected representatives fail to address, these thoughts have not led to action so the void remains. A 'them and us' attitude seems to exist where higher unelected officials constantly drive to earn more money from 'them' (the public) to facilitate 'us' (the unelected officials) in their projects.

This project is about the creation of a database to enable tracking of on-street parking permits in electronic form.

You should refer to the Database Methods general project guidelines, to ensure you are meeting them, at all times.

The traveling Database Project...

Greetings from Cork!

Greetings from Dublin!

Greetings from Dundrum!

Greetings from Limerick!

Greetings from Treviso!

Greetings from Venice!

Greetings from Clonmel!

You will be required to generate your own data, sufficient for 20 records. Creation guidelines for this data will follow. **See updates at end of document.**

Basic Requirements:

The Council wishes to record the:

1. Name

2. Address

3. Registration Number

4. Vehicle Type

5. Renewal Date

6. Notes

for each Resident/Address/Vehicle combination. Create a table to store this data.

When there is a new applicant for a permit they are required to fill out a form providing the details above. You are required to provide an A4 form that can be given by the office staff to the applicant for them to complete. By this means the staff can easily collect the details to be entered into the database if the applicant is entitled to the permit.

NOTES:

1. The form should have a suitable title/heading.

2. The applicant should be instructed to complete the form in BLOCK CAPITALS.

3. 4 lines should be allowed for the address.

4. The 'Renewal Date' will appear on the form as 'Todays Date'.

5. The 'Vehicle type' will be a list of types, one of which will be circled by the applicant.

6. The 'Notes' section is an empty box, approximately ¼ of the A4 page in which the applicant can provide any other data which they consider relevant or want recorded.

7. Below the section that the applicant fills in and above the section outlined in the point below, provide a position for the applicants signature (and date if preferred).

8. At the bottom of the form approximately ¼ of the page should be marked as 'Office Use Only' and left blank.

This form should be created in OpenOffice Writer or Draw, as preferred. It is referred to in the project documentation as the *Data Capture Form*.

Input Form:

Upon completion of the DCF you will create a suitable OpenOffice Base input form tied to the previously created table. Extra data from the DCF may be ignored.

The Table:

The data table should have separate fields for First and Last Names, as well as a separate field for each of the address lines. The vehicle type should be a field able to contain any *one* of the normal vehicle licence categories. From a separate table (which is not your concern) the permit licence can be retrieved for each of the vehicle types. The Notes field should be a memo-type input box; often in database programs this is achieved by altering the behaviour of the return or enter key in a standard text field. The renewal date should be obvious.

All fields should be off appropriate type and size. Provide a WP table or an SS grid to list the Name of each field, the type of each field and size information for each field. This is part of the table design and proves you have thought about the data and its requirements.

Queries:

A query extracts a meaningful subset of data from the bulk of the data in the database (table).

A minimum of two queries are required. Normally one uses retrieval based on more than one field. This is outlined in the Database Methods general project guidelines.

1. Create a query that retrieves all vehicles in a given category (of your choice).

2. Create a query that retrieves all *car* permits due for renewal in the next month.

3. Create one other query of your choosing, with tutor approval.

Reports:

A report can either present all data in the database or a subset selected from a query. In either case, the purpose of the report is to list this data in presentable, attractive, format.

A minimum of two queries are required. Normally they are based on the queries but often a master listing (*organised one one field*) is included.

1. Create a master list of all data organised on vehicle type, and secondarily organised on surname.

2. Create a report of all cars which have permits expiring in the next month – this uses the ouput for the already created query (2) above. The report format is to be an addressed letter suitable for insertion into a DL sized windowed envelope reminding the vehicle owner of the need to renew their permit. Normal letter writing rules will apply.

3. Create one other report of your own choosing, with tutor approval.

The Data:

You are required to create your own data for the database (table):

1. There should be 20 or more records.

2. You may assume that Fubarstown is in Co. Clare. All addresses, even those in Fubarstown itself should have the Co. name as the last line of the address.

3. If you wish, use real Clonmel (or your own) Town addresses to save exerting your imagination; however the town name is Fubarstown!

4. Likewise, use the names of friends and relatives.

5. Renewal dates should be spread throughout the year; note that for the query #2 above only *some* car details should be returned. The fact that some are not should show that the query can discriminate between relevant and irrelevant data.

6. Vehicle types are: A, A1, B, C, C1, D, D1, M, W. Though each has a different cost associated, you need not concern yourself with this for now.

7. Simply allocate each of the types using mainly A, A1, B, C1, D, M.

8. If you feel 20 records does not give you a representative sample of dates, vehicle types etc., add more. However, this will not yield extra marks unless it improves the performance/appearance of queries and/or reports.

9. After your 20 records are entered, print that table. Then delete at least one record and replace it, as well as modifying at least on other. This shows the ability to modify the table contents.

10. Registrations should be from a selection of not less than 4 counties.

11. Provide extra Notes from the Notes section of the DCF for 3 records. Make entries of any nature you like.

12. Provide one handwritten completed DCF in your project. Scan and include.

FAQ:

Q: Where do I post a question?

A: To fachtna@fachtnaroe.servehttp.com, of course. But think as you write, and double check before you send, that you don't know the answer already. If you think the question may be beneficial for others, go ahead and ask it anyway.

Q: When is this project due?

A: Check the calendar.

Q: Is the database exam hard?

A: No, far from it. Everything you do in the project you'll do in the exam, except make up the data. It's a 2 hour exam which should be quite sufficient. Follow the instructions and it's even easier.