DATABASE MANAGEMENT SYSTEMS
PROJECT- COURSE WORK

GROUP WORK:	3 Students
CONTRIBUTION:	25 MARKS (62.5% OF COURSE WORK)
SUBMISSION DATE:	Week 11
DEMONSTRATION DATE:	Computer Lab, Time: Week 11

Introduction to Coursework

You and your group members are part of a consultancy company that specialises in the provision of database applications. The Director of *FastCabs* has recently approached your company to undertake a project to design and partially implement a database management system for the company.

Notes

1. You are in the initial stages of user requirements collection and analysis and are required to read the *FastCabs* case study.
2. The information presented in the case study is an overview of how business is conducted at the *FastCabs*. As the information described in the case study is an overview and ambiguous in places, it will be necessary for you to make your own assumptions about certain aspects of the case study. Your assumptions should be clearly stated in your coursework submission.

Coursework Requirements

Part 1 – Design the Database

1. Create an Entity–Relationship (ER) model of the data requirements for the *FastCabs* case study using the UML notation. Note: if necessary, use the additional concepts of the Enhanced Entity–Relationship (EER) model. State any assumptions necessary to support your design.

   (Submit hardcopy)

2. Derive relational schema from your ER model that represents the entities and relationships. Identify primary, alternate and foreign keys. Note: use the following notation to describe your relational schema, as shown in the example of a Staff relation given below.

   Staff (staffNo, fName, lName, address, NIN, sex, DOB, deptNo)
   Primary Key staffNo
   Alternate Key lName, DOB
   Alternate Key NIN
   Foreign Key deptNo **references** Department(deptNo) **On Delete No Action On Update Cascade**

   (Submit hardcopy)

3. Use the technique of normalization to validate the structure of your relational schema. Demonstrate that each of your relations is in third normal form (3NF) by displaying the functional dependencies between attributes in each relation. Note, if any of your relations are not in 3NF, this may indicate that your ER model is structurally incorrect or that you have introduced errors in the process of deriving relations from your model.

   (Submit hardcopy)

4. To further demonstrate your knowledge of normalization, assume that a proposed (badly structured) relation for the *FastCabs* database has the following structure.

jobID	pickupDateTime	driverID	dFName	dLName	clientID	cFName	cLName	cAddress
1001	25/07/00.10.00	I456	Jane	Watt	C034	Anne	Way	111 Storrie Road, Paisley
1102	29/07/00.10.00	I456	Jane	Watt	C034	Anne	Way	111 Storrie Road, Paisley
1203	30/07/00.11.00	I344	Tom	Jones	C034	Anne	Way	111 Storrie Road, Paisley
1334	2/08/00.13.00	I666	Karen	Black	C089	Mark	Fields	120 Lady Lane, Paisley
1455	2/08/00.13.00	I957	Steven	Smith	C019	John	Brown	13 Renfrew Road, Paisley
1676	25/08/00.10.00	I344	Tom	Jones	C039	Karen	Worth	34 High Street, Paisley

Identify the functional dependencies represented in this relation and demonstrate the process of normalizing this relation into 3NF relations.

(Submit hardcopy)

Part 2 – Implement the Database

1. Create the tables for the *FastCabs* database. Where appropriate set field and table properties, including any required indexes. Ensure that referential integrity is established between related tables.

2. Create customised forms for data entry.

3. Enter some test data (approximately 5 – 10 rows) into each table.

Part 3 – Query the Database

Before starting this section, please ensure that your tables contain sufficient data to enable you to test the query transactions described in the *FastCabs* case study.

1. Create and save the query transactions.

2. Create a customised form or a report for each saved query.

3. Provide 10 additional examples of queries, which retrieve useful data from the *FastCabs* database. State the purpose of each query and attempt to use each example to demonstrate the breadth of your knowledge of Access QBE/SQL.

   (Only submit hardcopy for Part 3, point 3)

Part 4 – Implement Database Application

Implement a prototype database application for the *FastCabs*. The purpose of this prototype is to allow the Director to provide feedback on your proposed design.

The prototype should facilitate the creation, maintenance and querying of records and where appropriate automate various tasks for the user.

Part 5 – Document Database Application

1. Create a user manual that describes your prototype for the *FastCabs* database application.

2. The user manual should introduce the database application to the users, describe the functionality of the database application and clearly demonstrate how to use the application.

   Note: The user manual should be a maximum of 15 pages in length (including screen dumps).

   (Submit hardcopy)

Part 6 – Demonstrate Database Application

You and the members of your group are required to demonstrate your database application in Week 11.

Part 7 – Individual Critical Evaluation

Each student should submit his or her own critical assessment of the coursework. The evaluation should include a discussion on how the coursework has reinforced (or otherwise) his or her appreciation of the techniques and processes employed in undertaking a database project. In addition the evaluation may include a wider discussion on topics such as:

- How the Database Systems module relates to the other modules on your course.
- How the knowledge and skills taught on the module and/or course, relates to your previous experience as a student and/or employee.
- The appropriateness of the knowledge and skills taught on the module for future employment.

This component of the coursework can be submitted as a separate document from the main part of the coursework.

(Submit hardcopy)

Marking Scheme

The assessment of this coursework will be carried out on the following components of the work. Please note that each student should submit his or her own critical evaluation of the coursework and will receive an individual mark for this component (out of 10%). This individual student mark will be combined with the mark for the groupwork component (out of 90%) for the coursework. The total score shall be converted to be out of 25.

Part 1 – Design the Database	(30)
Part 2 – Implement the Database	(15)
Part 3 – Query the Database	(15)
Part 4 – Implement Prototype Database Application	(15)
Part 5 – Document Database Application	(10)
Part 6 – Demonstrate Database Application	(5)
Part 7 – Individual Critical Evaluation	(10)

The *FastCabs* Case Study

A private taxi company called *FastCabs* was established in Glasgow in 1992. Since then, the company has grown steadily and now has offices in most of the main cities of Scotland. However, the company is now so large that more and more administrative staff are being employed to cope with the ever-increasing amount of paperwork. Furthermore, the communication and sharing of information within the company is poor. The Director of the company, Paddy MacKay feels that too many mistakes are being made and that the success of his company will be short-lived if he does not do something to remedy the situation. He knows that a database could help in part to solve the problem and has approached you and your team to help in creating a database application to support the running of *FastCabs*.

The Director has provided the following brief description of how *FastCabs* operates.

Each office has a Manager; several taxi owners, drivers and administrative staff. The Manager is responsible for the day-to-day running of the office. An owner provides one or more taxis to *FastCabs* and each taxi is allocated for use to a number of drivers. The majority of owners are also drivers.

FastCab taxis are not available for hire by the public hailing a taxi in the street but must be requested by first phoning the company to attend a given address.

There are two kinds of clients, namely private and business. The business provided by private clients is on an *ad hoc* basis. The details of private clients are collected on the first booking of a taxi. However, the business provided by business clients is more formal and involves agreeing a contract of work with the business. A contract stipulates the number of jobs that *FastCabs* will undertake for a fixed fee.

When a job comes into *FastCabs* the name, phone number and contract number (when appropriate) of the client is taken and then the pick-up date/time and pick-up/drop-off addresses are noted. Each job is allocated a unique jobID. The nearest driver to the pick-up address is called by radio and is informed of the details of the job.

When a job is completed the driver should note the mileage used and the charge made (for private clients only). If a job is not complete, the reason for the failed job should be noted.

The Director has provided some examples of typical queries that the database application for *FastCabs* must support.

(a) The names and phone numbers of the Managers at each office.
(b) The names of all female drivers based in the Glasgow office.
(c) The total number of staff at each office.
(d) The details of all taxis at the Glasgow office.
(e) The total number of W registered taxis.
(f) The number of drivers allocated to each taxi.
(g) The name and number of owners with more than one taxi.
(h) The full address of all business clients in Glasgow.

(i) The details of the current contracts with business clients in Glasgow.
(j) The total number of private clients in each city.
(k) The details of jobs undertaken by a driver on a given day.
(l) The names of drivers who are over 55 years old.
(m) The names and numbers of private clients who hired a taxi in November 2000.
(n) The names and addresses of private clients who have hired a taxi more than three times.
(o) The average number of miles driven during a job.
(p) The total number of jobs allocated to each car.
(q) The total number of jobs allocated to each driver.
(r) The total amount charged for each car in November 2000.
(s) The total number of jobs and miles driven for a given contract.

SUCCESS