UNIVERSITY OF LONDON

Goldsmiths College

BSc Examination 2005

COMPUTING AND INFORMATION SYSTEMS

CIS209

Database Systems

Eastern

Duration: 3 hours

This paper consists of **5** questions. Each question carries **25** marks. Answer only **4** of them. You may choose **any 4** questions. Full marks will be awarded for **complete** answers to **4** questions.

The mark carried by each part is printed within square brackets. **Gauge the time to be spent on each part by the number of marks awarded**.

THIS EXAMINATION PAPER MUST NOT BE REMOVED FROM THE EXAMINATION ROOM.

QUESTION 1

Part 1 List up to five features that you would expect to be provided by a multi-user DBMS. [5]

Part 2 a) What is the purpose of NULLs and what do they represent? [2]

b) Write an SQL query that lists all the tuples containing a NULL value in a table T with attributes A, B, and C. [2]

c) Is a component of a primary key allowed to accept NULL values? Why? [3]

d) What is the term for the primary key constraint? [1]

Part 3 Explain each of the terms 'Candidate Key', 'Primary Key' and 'Foreign Key', in the context of the relational data model. Use only one sentence per term. Illustrate the term 'Foreign Key' with an example. [5]

Part 4 Define, in brief, the Data Manipulation Language (DML), and provide three examples of commands, of different types, in DML. Also, state their meaning in English. [7]

QUESTION 2

The following tables form part of a database held in a Relational Database Management System:

Employee	(**empID**, fName, lName, address, DOB, sex, position, salary, deptNo)
Department	(**deptNo**, deptName, mgrEmpID)
Project	(**projNo**, projName, budget, deptNo)
WorksOn	(**empID**, **projNo**, hoursWorked)

where	Employee	contains employee details and **empID** is the key.
	Department	contains department details and **deptNo** is the key. mgrEmpID identifies the employee who is the manager of the department. There is only one manager for each department.
	Project	contains details of the projects in each department and the key is **projNo** (no two departments can run the same project).
and	WorksOn	contains details of the hours worked by employees on each project, and **empID/projNo** form the key.

Part 1 Translate the following queries in SQL.

(1) List all the employees in the alphabetical order of their last names and, within the same last name, the order of their first names. [2]

(2) List all the details of employees who are female. [2]

(3) List the maximum salary of Programmers. [2]

(4) List the names and addresses of all employees who are Managers. [2]

(5) Produce a list of the names and addresses of all employees who work for the 'IT' department. [2]

(6) Find out how many employees are managed by 'James Adams'. [3]

(7) For each project on which more than two employees work, list the project number, project name and the number of employees who work on that project. [3]

(8) List the departments having the average project budget above £2 million. [3]

(9) List the total number of employees in each department for those departments with more than 10 employees. Create an appropriate heading for the columns of the results table. [3]

Part 2 Create a view of employee details, excluding department number, for all the employees who work on the project 'MIS Development',. [3]

QUESTION 3

Part 1 Draw an ER diagram for the following requirements. [20]

A company called "Neat Cars" has a number of garages. A garage has many members of staff; one of them is the garage manager (each garage has a manager, but there are many staff that aren't managers). Each garage has a unique code (garageCode) by which it is identified in the database. Each member of staff has a unique staff number (staffNo).
"Neat Cars" has clients (car owners). A client can own one or more cars. When a client (car owner) contacts a garage (for a problem that their car may have), both the client and the car are registered with the garage, if they have not already been previously registered, either with this garage or with others. Therefore, each client has a unique registration with the company in the database, namely with the first garage they go to. Similarly, each car has a unique registration. namely with the garage where it was first time brought to. Each client has a unique identification number (clientNo) and so has each car (carNo).
When a car is brought to a garage, it undergoes a number of tests, carried out by certain members of the garage staff. For a car, each test is carried out by only one member of staff, and is given a unique identification number (testNo). Each test may result in the recommendation of several repairs. The company has a number of standard types of repairs, each being identified by a code (repairTypeCode). Each prescribed repair must be of one and only one of the standard types.

Part 2 Choose a part of your diagram consisting of two entity types linked through a one-to-many relationship type. Consider two or more attributes per entity type and translate this part into the relational model using SQL commands. [5]

QUESTION 4

Part 1 Consider the following relation.

album	deadline	track	band	track-budget	completion-date	cover

Express the following statements as irreducible functional dependencies:

a) each album has a unique deadline; [1]

b) any track of any album is done by one and only one band; [1]

c) each band negotiates a (unique) budget for each individual track they record on each album (for example, "Scissor Sisters" may receive two different track-budgets for the same song/track "Marie" played on two different albums); [1]

d) 'completion-date' represents the date on which a band completes a track for a particular album; [1]

e) each band can propose one cover only for any album. [1]

Advice: at points c and d you may be tempted to propose a reducible functional dependency; the answer to point b will help you to resolve this possible problem.

Part 2 Consider the table/relation described above. Show the table with a few inserted tuples in order to illustrate some data redundancy; show explicitly where redundancy occurs. [3]

Part 3 Consider the table/relation described above. Using explicit data values, give an example of an update anomaly. Accompany the example by a brief explanation. [5]

Part 4 Consider the following relation. It stores information about cross-country races. 'race-id' represents a particular race; 'type' is the type of a race (e.g., "5km" or "marathon"); 'date' and 'venue' represent the date and the venue of a particular race.

runner	race-id	type	date	venue	result

Consider the following functional dependencies:

runner, race-id → result
race-id → type, date
race-id → venue
type, date → race-id

Assume they completely express all the functional dependencies existing in the given relation (i.e., the others are either trivial or can be deduced from the given ones).

The given relation is not in BCNF. Decompose/transform it (non-loss) into a set of relations in BCNF. The normalisation/decomposition process should be carried out through the application of Heath's theorem. In this process, **you must consider the functional dependencies in the order in which they are listed above**. For each decomposition state:

- the relation that is to be decomposed;
- the functional dependency (or dependencies) on which the decomposition is based;
- the resulting relations;
- the candidate keys for each resulting relation;
- for each resulting relation, whether it is or it isn't in BCNF.

When the normalisation process is finished, state the end result clearly. [12]

QUESTION 5

Consider the following two tables. They are used in all of the following parts.

Competitions2005

Code	Name	Available
MP	Masters Paris	yes
W	Wimbledon	yes
OA	Open Australian	yes
AS	All Stars	no

IndividualResults

PlayerId	Username	Name	Ranking-2004	*Competition-05*	Points
001	a01aa	Andre Agassi	A-Group	MP	20
002	a01th	Tim Henman	A-Group	MP	6
003	b01ii	Igor Ivan	B-Group	OA	null
004	c01ms	Maria Smith	C-Level	OA	null

'Competitions2005' lists all the tennis competitions available in 2005. 'IndividualResults' lists all the players entered for each competition and the number of points achieved. The primary keys are printed in italics ('Individual Results' has a composite primary key). The column 'Competition05' in 'IndividualResults' is a foreign key referencing 'Code' in 'Competitions2005'. 'Username' is unique per 'PlayerId'.

Part 1 The following insert operation is attempted:

```
INSERT INTO IndividualResults
        VALUES ('010', 'b01ak', 'Anna Kornikova', 'B-Group', 'ODD', null);
```

The database management system generates an error.

a) Explain the reason why an error was generated. [2]

b) Explain what should be done in order for the insert operation to succeed. [2]

Part 2 Express the following constraints in SQL.

a) No player with the 'D-Level' ranking in 2004 is allowed to register for any competition in 2005. [3]

b) The 2004 ranking 'C-Level' is not acceptable (sufficient) for entering (registering for) Wimbledon. [3]

c) No player should be allowed to register for competitions that are not available in 2005. [4]

d) Each player can register for only one competition in 2005. [4]

Part 3 Express the following security rules in SQL.

a) The results for the competitions that already took place (i.e., 'Points' is not null) – i.e., names and points – can be viewed by anyone. [3]

b) Each player should be allowed to see all his/her records from 'IndividualResults', but should not be allowed to see any other records. [4]