Faculty of Engineering, Environment and Computing
EEC 7086CEM

Assignment Brief 2022/2023

Module Title Data Management Systems	Individual **or** Group of 2	Cohort (May-Sep)	Module Code 7086CEM
Coursework Title (e.g. CWK1) Database management			Hand out date: 9 June 2022
Lecturer Dr Rachid Anane			Due date: 7 July 2022
Estimated Time (hrs): 20 Word Limit*: up to 800 applicable to Part D only	Coursework type: Assignment		% of Module Mark 100
Submission arrangement online via Aula: File types and method of recording: Mark and Feedback date: Mark and Feedback method: feedback file			

Module Learning Outcomes Assessed:

1. Demonstrate a sound understanding of the theoretical and practical issues relevant to data management systems
2. Critically evaluate a range of conceptual and technical tools and to apply them selectively in the design and implementation of an effective data management system
3. Assess and evaluate the theoretical and technological underpinnings of distributed frameworks
4. Review and comment critically on the current trends in distributed processing

Task and Mark distribution:

A. ER model and table generation (25%)
B. SQL programming (30%)
C. Sequential and Distributed processing (25%)
D. Research report (20%)

Notes:

1. You are expected to use the CUHarvard referencing format. For support and advice on how this students can contact Centre for Academic Writing (CAW).
2. Please notify your registry course support team and module leader for disability support.
3. Any student requiring an extension or deferral should follow the university process as outlined here.

4. The University cannot take responsibility for any coursework lost or corrupted on disks, laptops or personal computer. Students should therefore regularly back-up any work and are advised to save it on the University system.
5. If there are technical or performance issues that prevent students submitting coursework through the online coursework submission system on the day of a coursework deadline, an appropriate extension to the coursework submission deadline will be agreed. This extension will normally be 24 hours or the next working day if the deadline falls on a Friday or over the weekend period. This will be communicated via email and as a CUMoodle announcement.

7086CEM – Data Management Systems

This assignment is made up of three parts:

- Part A deals with conceptual database design, using E-R modelling
- Part B concerns database creation and querying, using SQL
- Part C covers distributed processing frameworks
- Part D involves a research report

Please note that plagiarism (copying and pasting from sources) and collusion (submitting the same content as a fellow student) are detected automatically by Turnitin and flagged. You should be able to check for similarity.
You are expected to submit your own work. The report must be self-contained. Links to external datasets are not acceptable.

Submission process

If the CW is submitted as a group of two students, then

a) **one student will submit individually the whole report, with both names on the front page, as a singe pdf file, and**
b) **the other student must submit individually one page only, with both names, as a pdf file, where he/she presents reflections on the collaboration.**

A. Human Resources

A human resource management (HRM) department wishes to create a database to monitor its employees. The company is divided into a number of departments, and employees are assigned to one department. A department is identified by department id and title, whereas an employee is identified by employee id and name. Two types of worker have been identified: shop floor workers and office workers. Each shop floor worker has a specific skill and performs a specific task. Office workers on the other hand, are identified by their role and the department to which they belong.

The department has a designated Manager who has overall responsibility for the department and the employees in the department. However, to help manage the department, a number of employees are nominated to supervise groups of staff. When a new employee joins the company, information on previous work history and qualifications is required. On a regular basis, each employee is required to undergo a review on a specific date, which is normally carried out by the Manager, but may be delegated to a nominated representative.

The company has defined a number of position types, such as Manager, Business Analyst, Salesperson, Secretary, and each position type has a number of grades associated with it, which for most non-senior positions determines the employee's salary. Positions are allocated to a department depending on its workload. For example, a department may be allocated two new Business Analyst positions. A position will be filled by one employee, although over time, employees will fill a number of different positions.

1. Create an ER diagram for the above scenario and indicate the cardinality of relationships and the nature of the associations (mandatory or optional). You should allocate adequate attributes to the entities of interest, especially the identifiers.

   (20%)

2. Generate, with justification, relational tables from the ER diagram. Indicate clearly the names of the tables, the attributes, the primary keys and the foreign keys.

   (5%)

Guidance: *i) Create the ER diagram and clearly identify any identifiers, and indicate the cardinality of relationships and the nature of the associations (mandatory or optional). ii) Generate tables and include primary and foreign keys. Use the schema notation; <u>you do not have to produce SQL statements.</u>*

Example of table generation in schema form:

Course(<u>courseId,</u> courseName)
Student (<u>studentId</u>, name, courseId)*

Part B. SQL programming

Consider the following Employee database and sample data. You may wish to add more data records.

Employee (empId, name, address, DOB, job, salaryCode, deptId, manager, schemeId)
Department (deptId, name)
SalaryGrade (salaryCode, startSalary, finishSalary)
PensionScheme (schemeId, name, rate)

TABLE: Employee

empId	name	address	DOB	job	salaryCode	deptId	manager	schemeId
E101	Keita, J.	1 high street	06/03/76	Clerk	S1	D10	E110	S116
E301	Wang, F.	22 railway road	11/04/80	Sales person	S2	D30	E310	S124
E310	Flavel, K.	14 crescent road	25/11/69	Manager	S5	D30		S121
E501	Payne, J.	7 heap street	09/02/72	Analyst	S5	D50		S121
E102	Patel R.	16 glade close	13/07/74	Clerk	S1	D10	E110	S116
E110	Smith, B.	199 London road	22/05/70	Manager	S5	D10		S121

TABLE: Department

deptId	name
D10	Administration
D20	Finance
D30	Sales
D40	Maintenance
D50	IT Support

TABLE: SalaryGrade

salaryCode	startSalary	finishSalary
S1	17000	19000
S2	19001	24000
S3	24001	26000
S4	26001	30000
S5	30001	39000

TABLE: PensionScheme

schemeId	name	rate
S110	AXA	0.5
S121	Premier	0.6
S124	Stakeholder	0.4
S116	Standard	0.4

1. Use appropriate data types and write the SQL statements to create the tables defined in the schema above.

   (10%)

2. Write SQL Statements to return the following data from the Employee database:

   a) The name (in ascending order), the starting salary and department id of each employee within a descending order of department ids.

   (5%)

   b) Give the number of employees for each of the pension schemes offered by the company. Result listing should include the name of each scheme and its corresponding number of employees who join the scheme.

   (5%)

   c) Give the total number of employees who are not managers but currently receive an annual salary of over £35,000.

   (5%)

   d) List the id and name of each employee along with his/her manager's name.

   (5%)

Guidance: *Please use standard SQL. Indicate clearly the primary keys and the foreign keys. State the SQL statements and give the results.*

The presentation of each query should have a text summary which includes i) the query itself, ii) the corresponding SQL statement solution, iii) the result of the execution of the statement and iv) evidence that you have used standard SQL and implemented each statement on a database (use screenshots or spool facility).

A small data sample is given. When appropriate, you can create and insert additional data records in order to make sure that the queries return results.

C. Sequential and parallel processing

Consider a sales data store with the following data structure, where all values are either integer or real. Each record consists of eight attributes; the set of allowable values of the attributes and format are specified in the description (metadata).

OrderNo	Integer
ProductNo	Integer
Price	Price of each product (Real/Float)
Quantity	Integer
Sales	Real/Float
MonthId	Integer (1-12)
YearId	Integer

Sample records are included in the following table. You can include additional records to illustrate the solution.

OrderNo	**ProductNo**	**Price**	**Quantity**	**Sales**	**MonthId**	**YearId**
10107	2	95.7	30	2871	2	2003
10107	5	99.91	39	3896.49	2	2003
10110	9	86.13	29	2497.77	2	2003
10121	5	81.35	34	2765.9	11	2003
10134	2	94.74	41	3884.34	7	2004
10134	5	100	27	3307.77	7	2004
10159	14	100	49	5205.27	10	2005
10161	9	86.13	29	2497.77	10	2005
10163	14	100	20	2000	10	2005
10168	1	96.66	36	3479.76	10	2006
10180	12	100	42	4695.6	11	2006
...	...	...	...	...	...	...

1. Assuming that the data is stored in a relational database produce, with justification

   a) the SQL statement to create the corresponding table

   b) the SQL code to determine, for each product, the number of products which were sold in each month of each year. (You do not need to implement the SQL statements on Oracle).

(5%)

2. Assuming that the data is too large to be processed in a centralised manner in a relational database, and that it is stored in an ordinary file, produce a decentralised solution which applies MapReduce to the data processing. Justify your decisions and all the steps of your solution. Use diagrams if required.

(20%)

Guidance: *You should study carefully the examples of mapReduce covered in the lecture notes. You should consider the structure of the key in the (key, value) pair in the original record and in the mapping stage. This is not a programming exercise. The solution should follow the structure given in the lecture notes.*

Part D. Research report

Consider the following quote from an online article:

“With more than 1.5 million new ads posting every day, Craigslist (a site for classified listings) users have generated over a billion records – some might even consider that ‘big data.’ What’s more, legislation demands that after a 60 day retention period in the live portion of the site, records must be migrated over to an archival space for legislative compliance.

Craigslist faced several challenges due to the nature and volume of data being stored in their relational MySQL servers. For example, the structure of their data had changed several times over the years. This alone made any change to the database schema a costly, prolonged nightmare, as changes often meant downtime. And if database alterations were a challenge, just imagine how difficult introducing entirely new features became? What’s more, each change to the live database schema required a corresponding change to the entire archive – a process that took months every time“.

In 2011 Craigslist decided to migrate the data and the processing from the relational MySQL servers to NoSQL MongoDB servers.

Refer to the strengths and limitations of relational databases and of NoSQL databases and explain **in no more than 800 words** why Craigslist moved their operations from MySQL to MongoDB .

(20%)

Guidance: *Use your own words for the report.* ***Copying and pasting is plagiarism.*** *You should include relevant references. The maximum length of the report is 700 words. Longer reports will be penalised.*

Marking Rubric

		Marking Scheme		
Grade	**Part A**	**Part B**	**Part C**	**Part D**
<40	• Incorrect interpretation of scenario and Incomplete formulation of solution • Limited identification of entities and poor annotation of relationships • Incorrect generation of relational tables • Limited or absent rationale	• Poor interpretation of requirements and of queries • DDL and DML SQL statements limited in scope • Incomplete and incorrect SQL statements • Absence of rationale	• Partial understanding of requirements and partially correct SQL formulation • Partial understanding of context and relevance of parallel processing • Incomplete steps in the application of MapReduce • Partial justification of design decisions	• Lack of understanding of requirements • Inadequate identification of issues • Incompetent understanding of structural and processing components • Poorly written essay
40-49	• Partial interpretation of scenario and formulation of solution • Partially correct ER diagram with relevant entities and relationships • Partial consistency in the generation of the relational tables • Partial justification of design decisions	• Basic understanding of requirements and partially correct interpretation • Relevant use of DDL and DML statements in solution formulation • Partially complete SQL statements • Limited justification of solution	• Partial understanding of requirements and partially correct SQL formulation • Partial understanding of context and relevance of parallel processing • Incomplete steps in the application of MapReduce • Partial justification of design decisions	• Partial interpretation of requirements • Limited presentation of key issues • Relevant description of structural and processing components • Mostly descriptive essay
50-59	• Adequate and consistent interpretation of scenario and satisfactory conceptual modelling • Mostly correct generation of ER diagram with relevant entities and relationships • Relatively competent generation of the relational tables • Adequately justified design decisions	• Adequate understanding of requirements and mostly correct interpretation • Adequate use of DDL and DML SQL statements in solution formulation • Mostly complete SQL statements • Adequate justification of solution	• Adequate understanding of requirements and correct SQL formulation • Adequate presentation of context of application of parallel processing • Mostly correct application of the different steps of MapReduce • Adequately justified solution	• Adequate interpretation of requirements • Key issues well identified and partially addressed • Adequate presentation of key structural and processing components • Adequately written essay
60-69	• Good and clear interpretation of scenario and good solution formulation of the two parts • Correct and complete identification of entities and relationships • Consistent generation of relational tables	• Good interpretation of requirements and good formulation of solution • Correct use of DDL and DML statements in solution formulation • Complete and relevant treatment of queries • Good justification of decisions	• Good solution to the initial query in terms of SQL statements • Focused presentation of context of application of parallel processing • Clearly stated and correct sequential steps of MapReduce • Well expressed rationale	• Good understanding and statement of requirements • Well focused presentation of main issues • Good description of structural and processing components • Well presented

	• Well expressed rationale			essay with some reflection
70+	• Very good and clear interpretation of scenario and excellent solution formulation of the two parts • Correct and coherent identification of well annotated entities and types of relationships • Logical and consistent generation of relational tables • Excellent justification of design decisions	• Excellent interpretation of requirements and very good formulation of solution • Excellent use of DDL and DML statements in solution formulation • Complete treatment of queries and SQL formulation • Very good rationale	• Very good solution to the initial query in terms of SQL statements • Excellent presentation on the need for an overall parallel solution • Clear deployment and annotation of the sequential steps of MapReduce • Excellent rationale	• Excellent understanding and interpretation of requirements • Very good identification and formulation of key issues • Relevant and specific presentation of structural and processing components • Reflective writing supported by an excellent structure