TOURO COLLEGE COURSE SYLLABUS
LANDER COLLEGE

DEPARTMENT: Computer Science
COURSE TITLE: Database Concepts and Design
COURSE NUMBER: MCO343
PREREQUISITES: MCO232
CREDIT HOURS: 3
DEVELOPER: Miriam Plonczak
LAST UPDATE: July, 2003

COURSE DESCRIPTION

This course provides students with an informed and critical perspective on current alternatives in the Database Management Systems (DBMS) area. Topics include: architecture; hierarchic, network and relational database approaches with respect to database structure; integrity; user view mechanisms; data storage and access techniques; data manipulation; normalization; file structures versus DBMS; SQL, and QBE.

COURSE/DEPARTMENTAL OBJECTIVES

The students will be able to:

Identify the deficiencies of typical file management systems

Explain how a well-designed database remedies the deficiencies of a file management system

Enumerate and describe the features and advantages of a good database management system

Describe, compare and contrast the three basic implementation models of DBMS : hierarchical, network, relational

Describe, compare and contrast the two basic conceptual models: Entity-Relationship Data Model and the Object Oriented Data Model

Use DDL to create a new database, add a new table, identify individual fields and their datatypes, identify constraints such as primary key and foreign key, remove a table from a database, add or remove or change a field in a specific table, add an index to a table

Use Microsoft Access interface to create, change and/or remove a table from a database, identify/enforce relationships between tables, design form/subforms that include more advanced controls such as listboxes and event procedures that perform additional data input validation,

design reports that include data from more than one Access table, calculated fields, sorting and grouping, running sums, headers, footers with summary data

Define certain basic relational database terms and explain their importance and significance in database design: primary key, functional dependency, candidate key, composite key, referential integrity, foreign key, secondary keys or indexes, linking table

Identify the relational operators and the subset of data they each yield.

Use either the Crow's Foot Model or the Chen model Entity-Relationship diagrams to represent the database design concepts such as connectivity, cardinality, relationship strength, relationship participation

Use ODBC to connect to an Access 2000 database

Use MFC and ODBC connection or DAO/ADO to issue DDL statements from within a C++ or VB program

Explain and give examples of what types of transactions will fail if a DBMS enforces referential integrity, data integrity

Use MFC, ODBC connection to issue DML statements from within a C++ problem

Use exception handling to identify failed transactions. Issue rollback or commit commands to complete the transactions properly.

Identify and describe the capabilities a complete DBMS should provide its users

Identify and describe the capabilities that a true relational DBMS should provide its users

Construct an SQL statement to retrieve data from one or more tables.

Identify whether to use Inner Join, Outer Join or Union in order to retrieve the data that has been requested

Construct an SQL statement that restricts rows, selects columns, creates calculated fields, and uses database functions such as count, sum and subqueries when and if necessary

Explain why the use of NULLs is potentially problematic

Explain each normalization rule and apply the set of normalization rules to design a normalized database

Demonstrate how a faulty design introduces update anomalies and how these anomalies are eliminated using normalization rules

Identify three problems with transaction processing that concurrency control must be able to prevent.

Explain when and why database recovery may be necessary

Describe, compare and contrast different recovery methods, how the transaction log is used and define the significant terms checkpoint, rollback, commit, undo and redo.

Identify what information a database designer must collect in order to determine existing and anticipated user requirements

List the techniques a database designer uses to ascertain user requirements and constraints

Explain how a database designer verifies and revises a conceptual model (ERD).

Explain why and give examples to illustrate situations that would necessitate compromise of a normalized conceptual database design to make the physical design more efficient or effective

List factors that a database designer considers when choosing a DBMS to implement the conceptual design.

Demonstrate how a conceptual design is mapped to a logical design for a particular relational DBMS

Explain why a physical database might have to be modified to tune the system and what type of changes are made to fine tune the system

Differentiate between senior, middle and operational management and how they use the DBMS

List the managerial and technical skills a successful DBA should have

Describe the DBA's managerial and technical role within a company

Describe some tools a DBA uses such as data dictionary and CASE tools

COURSE/INSTITUTIONAL OBJECTIVES

This course provides students with a fundamental understanding of database design and implementation issues. This understanding is increasingly necessary for becoming successful managers and technical specialists in our information driven society. The course also develops practical skills in information retrieval that will enable students to pursue careers in computer programming.

COURSE CONTENT

Database Concepts

File Systems versus Database Systems
Disadvantages of traditional file systems
Types of Database Management Systems
Basic Functions of DBMS