COP 4703 Applied Database Systems

Credits: 3 credits


(2) Detailed documents posted on Blackboard

Reference materials: None

Specific course information

Catalog description: Use Oracle 11g to study the Java EE approach to information technology, with emphasis on Web programming. The Oracle JDeveloper 11g IDE will be used for all applications, including SQL, PL/SQL, Java, Web and XML applications

Prerequisites: Prerequisites: COP 3530 Data Structures. COT 4703/6726 satisfies the Advanced Content requirement for the BSCS program and can be used as a technical elective in all CS and CE programs.

Specific goals for the course: Investigate Java EE as a modern IT platform.

Brief list of topics to be covered:

1. Using Oracle 11g to master the industry standard relational database language SQL.
2. Using Oracle 11g to master PL/SQL, Oracle's procedural extension of SQL. Using PL/SQL to implement stored procedures, stored functions, packages, triggers, and Web programs. The latter is done using the PL/SQL Web Toolkit and PL/SQL Server Pages.
3. Using Java to gain familiarity with object-oriented concepts such as class, inheritance, and polymorphism. Java 1.5 generics and iterators will be covered.
4. Using the object-oriented features of Oracle 11g to gain an understanding of how relational features and object-oriented features can be effectively combined in SQL. Such object-oriented features include user-defined types, row types, methods, collection types (nested tables and arrays), OID references, and type inheritance.
5. Using Oracle 11g to study the invocation of SQL from Java provided by JDBC. Using JDBC to implement stored subprograms, triggers, and user-defined type member functions, as well as to implement client-side applications.
6. Java EE Web programming with: Java servlets, JavaServer Pages (JSP), JavaServer Faces (JSF), and Ajax-enabled ADF Faces.
7. Using the Oracle XML parsers, the Oracle XML-SQL utility, Oracle XSLT, and the Oracle XML data type to study XML processing in the object-relational environment.
8. The SQLJ (embedded SQL for Java) and TopLink (an object-relational mapping for Java) alternatives to JDBC.
9. Using JPublisher to "one-click map" SQL user-defined types into Java classes, instead of mapping SQL tables into Java Classes as is done with object-relational mappings (ORMs) like TopLink, and then using SQL with these JPublisher generated classes, so as to provide an improved alternative to such ORMs.