Advanced Databases Systems

Date: Monday 3rd June 2013
Time: 14:00 - 16:00

Please answer any THREE questions from the FIVE Questions provided

This is a CLOSED book examination

The use of electronic calculators is NOT permitted
1. This question requires knowledge of two of the main approaches for overcoming the computational limitations of SQL: combining SQL with a programming language, such as Java, as in JDBC; and extending the syntax of SQL with procedural language features, as in PL/SQL. To answer this question, consider the following relational database, composed of three relations. Note that no assumptions are made about the types for the attributes of these relations.

Property (prop#, address, staff#, categ#, price);
Category (categ#, name, description);
Staff (staff#, name, DoB, NSS, contact);

a) Considering the following PL/SQL code fragment, which accesses the relations described above, write an equivalent JDBC code fragment, listing all your assumptions regarding any schema information which may be absent from the description of the database above. You do not need to show connection to the database.

(10 marks)

```sql
declare
  cursor PropertyCursor is
    select Property.prop#, Property.price,
           Category.description, Staff.contact
    from Property, Category, Staff
    where Property.categ# = Category.categ# and
          Property.staff# = Staff.staff#;

  Num Property.prop#%type;
  Price Property.price%type;
  Desc Category.description%type;
  Contact Staff.contact%type;

begin
  open PropertyCursor;
  fetch PropertyCursor into Num, Price, Desc, Contact;
  while PropertyCursor%found loop
    dbms_output.put_line(' ' || Num || ' ' || Price || ' ' || Desc || ' ' || Contact);
    fetch PropertyCursor into Num, Price, Desc, Contact;
  end loop;
  close PropertyCursor;
end;
```

b) Considering the PL/SQL code fragment above, name two features of PL/SQL used in this code which have facilitated its development in terms programmer’s time and effort, resulting in higher productivity. Explain your answer.

(2 marks)
c) Considering the JDBC program fragment that you have written to answer Part a) of this question, name two features of JDBC used in this code which have made its development harder with regards to programmer’s time and effort, possibly resulting in lower productivity. Explain your answer.

(2 marks)

d) Considering issues such as network traffic, server load, client load, security and code sharing, discuss the pros and cons of having the application described in this question as a JDBC program instead of a PL/SQL one.

(6 marks)
2. This question requires knowledge of PL/SQL and its use in the implementation of Triggers on the Server machine, considering a Client-Server architecture. To answer the question, consider the following relational tables.

\begin{align*}
&\text{SoldProperty (prop\#, address, staff\#, price)} \\
&\text{StaffPerformance (staff\#, numOfProperties, sumOfAll, profit)} 
\end{align*}

The \text{StaffPerformance} table stores the identifier of each staff member of a State Agent Company (staff\#), the overall number of properties this member has sold in his/her career in the Company (numOfProperties) and the amount of profit this member has brought to the Company (profit), which represents 40\% of the sum of the prices of all properties this member has sold (sumOfAll).

a) Write row triggers that propagate only \textbf{updates} made to the \text{SoldProperty} table to the \text{StaffPerformance} table, stating any assumptions you make. (10 marks)

b) Discuss the pros and cons of implementing automatic reactive behaviour on the Server machine rather than on the Client. In your discussion, use at least 5 arguments supporting or refuting the implementation of this behaviour on the Server. (5 marks)

c) Discuss the differences between row triggers and statement triggers and explain why or why not you would choose to implement a statement trigger for the application described in this question, instead of row triggers. (5 marks)
3. This question requires knowledge of object database technology that does not represent an extension to relational technology, such as JDO. Consider the following class definitions, which are a part of a JDO database application that describes the branches of a restaurant chain:

```java
class Branch {
    protected Staff owner;
    protected String address;
    protected Vector kitchenStaff;
    public Branch();
    public Branch(String o, String ad, Vector k);
    public Staff getOwner();
    public Staff getAddress();
    public void addKitchenStaff(Staff s);
    public void printAddress();
}

class Staff {
    protected String NSS;
    protected String name;
    protected String address;
    protected int age;
    public Staff();
    public Staff(String nss, String n, String ad, int ag);
    public void print();
}
```

a) Write a JDO program that retrieves the address of all branches which have kitchenStaff members over the age of 50. Assume that attribute kitchenStaff of class Branch is a Vector containing references to Staff objects. You do NOT need to write the statements related to database connection, and you can assume that you have access to an initialised PersistenceManager object. You DO need to include transactions and exception handling.

(8 marks)

b) Explain the concept of “Persistance By Reachability” and illustrate it with a code-based example involving this concept being applied in the context of the application described in this question. A code-based example requires you to write any JDO code related to the description of the example, e.g., creation of new objects, the saving of objects in the database, etc.

(6 marks)

c) Explain why or why not you have used a JDOQL query expression in your solution to Part a) of this question, discussing the advantages or disadvantages of the use of JDOQL in your solution compared to alternative approaches.

(6 marks)
4. This question requires knowledge of the management of semi-structured data, with a focus on the XML data model. To answer it, consider the following XML Schema:

```xml
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://xmlschemaexamples.org/this.xsd"
  xmlns="http://xmlschemaexamples.org/this.xsd"
  elementFormDefault="qualified">
  <xs:element name="purchaseOrder" type="PurchaseOrderType"/>
  <xs:element name="comment" type="xs:string"/>
  <xs:complexType name="PurchaseOrderType">
    <xs:sequence>
      <xs:element name="shipTo" type="Address"/>
      <xs:element name="billTo" type="Address"/>
      <xs:element name="items" type="Items"/>
    </xs:sequence>
    <xs:attribute name="orderDate" type="xs:date"/>
  </xs:complexType>
  <xs:complexType name="Address">
    <xs:sequence>
      <xs:element name="name" type="xs:string"/>
      <xs:element name="street" type="xs:string"/>
      <xs:element name="city" type="xs:string"/>
      <xs:element name="postcode" type="xs:decimal"/>
    </xs:sequence>
    <xs:attribute name="country" type="xs:NMTOKEN" fixed="UK"/>
  </xs:complexType>
  <xs:complexType name="Items">
    <xs:sequence>
      <xs:element name="item" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="productName" type="xs:string"/>
            <xs:element name="quantity">
              <xs:element name="price" type="xs:decimal"/>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="itemNum" type="xs:string" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

a) Construct a relational schema that contains the same information as the above XML Schema. The schema should state keys and constraints where relevant, and types for all attributes. State any assumptions you make. (10 marks)
b) Discuss the differences between the provided XML schema and your relational schema, emphasising advantages and disadvantages of the both models. Illustrate your answer with examples taken from both schemas.

(10 marks)
5. Consider the following schemas, independently developed for two databases that record information about properties for rent and sale, as well as contracts between property owners and clients. In all your answers, state any assumptions you make:

**Database 1:**

PropertyForRent(propertyNo, street, city, postcode, type, rooms, rent, ownerNo)
PropertyForSale(propertyNo, street, city, postcode, type, rooms, salePrice, ownerNo)
PrivateOwner(ownerNo, fName, LName, address, telNo)
Client(clientNo, fName, LName, address, telNo)
Lease(leaseNo, propertyNo, clientNo, rent, paymentMethod, deposit, paid, rentStart, rentFinish, duration)

**Database 2:**

Property(prop_id, address, details_id, price, owner_id)
Details_Property(details_id, type, rooms, features, details_furniture, view, facilities, details_area)
Owner(owner_id, name, address, tel)
Client(client_id, name, address, tel, constraints_id)
Client_Constraints(constraints_id, client_id, prop_type, minPrice, maxPrice, num_rooms, carParking)
Contract(contract_id, prop_id, client_id, rent, paymentMethod, dateStart, durationPeriod)

a) Suppose that the estate agent companies that own the above databases will merge, and, therefore, their database schemas need to be integrated. During the integration process, schema conflicts have to be reconciled.

Answer the questions below, using the following notation: Database_i.TableName.AttributeName, to refer to an attribute in any of the two schemas; Database_i.TableName, to refer to a table in any of the two schemas.

i) Describe two different one-to-one table name conflicts. (2 marks)

ii) Describe two different one-to-one table structure conflicts, one of them being the case of a missing, but implicit attribute. (2 marks)

iii) Describe one table inclusion conflict. (1 mark)

iv) Describe one many-to-many table conflict. (1 mark)

v) Describe two different many-to-many attribute conflicts. (2 marks)

b) Views are often used to bridge across independently developed schemas. They protect users from the schematic differences that may arise as a result of that independent development. You are using Oracle ManagementDB to provide a database that brings together the two state agent databases.
i) Give the two Oracle distributed database commands that would be required to link the ManagementDB with Database_1 and Database_2. Explain why you have chosen public or private, and connected user or fixed user links in your answer. (6 marks)

ii) Produce a SQL view in ManagementDB that derives a table `AllPropertiesDetails(propertyNo, address, details, price, ownerNo)`, where attribute details include information on property type (`type`) and number of rooms (`rooms`), and which contains data derived from both Database_1 and Database_2. (6 marks)