**COMP 3610 Review for Midterm II**

* **Exam Duration: 2 lab hours:** 
  + **30 minutes (more or less as needed by the student) for a paper-based and closed book exam (Part 1)**
  + **1 hour and 30 minutes for the practical part with access to all materials**

**(Part II)**

**Coverage:**

Chapter 8 (PL/SQL) function, trigger, stored procedure (with explicit cursors),

Chapter 10 (Database Design) Conceptual/Logical/Physical Model General understanding,

Chapter 11 (Database Analysis) General understanding,

Chapter 12 (ERD) practical approach,

Chapter 14 (Normalization) practical approach 1NF, 2NF, 3NF

Students’ presentations on Oracle functions and regular expressions

Exam will cover all chapters, but it will focus on four main topics: using PL/SQL to create a function, a trigger, and a stored procedure, conceptual modeling (ERD), normalization (1NF, 2NF, 3NF, functional dependencies), the process of database design (conceptual, logical, and physical models). The material presented by the students (including regular expressions) will be included in the exam.

**The following are examples of questions from previous years:**

1. Draw an ERD (UML notation) for TRU courses. Include entity(ies), relationship(s), and cardinalities with min and max: **(1 mark)**

*TRU offers a number of courses. Each COURSE has* ***Course\_Number*** *(for example: COMP361),* ***Course\_Name****, and* ***Credits****. Each course may be offered several times as a separate offering (for example FALL 2016, section 1). Each offering has an instructor, but instructors may teach none or many offerings. Some courses have co-requisites (other courses to be taken together or prior to the course). A course may have many co-requisites.*

1. Normalize the following relations (entities). Write down any assumptions that you have made:

Pet (PetID, OwnerID, OwnerName, OwnerAddress, PetName, Breed)

Visit(PetID, DateTime, TypeofVisit, Comments, PetName)

1. Assuming that the following relation is in 2NF, decompose this relation into a set of equivalent normalized (3NF) relations. Underline primary keys. **(2 marks)**

Book (ISBN, author name, author address, author e-mail, book title, total number of pages, price)

1. A relation that has no multivalued attributes is in \_\_\_\_\_\_\_\_\_\_NF. A relation that is in 1NF and has no partial dependencies is in \_\_\_\_\_\_\_\_NF. A relation that is in 3NF is also in \_\_\_\_\_\_\_NF and \_\_\_\_\_\_\_NF. **(1 mark for all)**
2. Indicate the normal form (highest) for ORDER\_LINE. If the relation is not in the 3NF, decompose it into 3NF relations. Functional dependencies (other than those implied by the primary key) are shown where appropriate.

ORDER\_LINE (Order\_No, Line\_No, Product\_Id, Order\_Date)

FD: Order\_No 🡪 Order\_Date

1. Why dates should be stored using special data types for dates? Why dates should not be stored as a string of characters?
2. What is the difference between 1NF and **generalized 1NF**?
3. What is the difference between 2NF and **Generalized 2NF**?
4. What is the difference between a candidate key and a primary key.
5. Use PowerDesigner to create a CDM for vehicle rental database.
6. Write a function to calculate the number of reservations starting today. If there are no reservations return 0
7. Write a function CHECK\_25 to check it the customer (given C\_ID) is older than 25 (for the insurance purposes). PL/SQL has a BOOLEAN data type. Return TRUE when older than 25; otherwise return FALSE.

RESERVATION(R\_ID, START\_DATE, END\_DATE, C\_ID,V\_ID)

CUSTOMER(C\_ID, FIRST\_NAME, LAST\_NAME, DOB)

VEHICLE(V\_ID, VIN, MAKE, MODEL, YEAR, REGISTRATION\_NUMBER)

1. The database for ski rentals has two tables:

CUSTOMER(customer\_id, first\_name, last\_name)

BOOKING(booking\_id, b\_from\_date, b\_to\_date, equipment\_id, pickup\_time, status, description)

Status has the following domain = {booked, equipment assigned, cancelled, taken, returned, equipment lost}

Write a stored procedure to list the customers who have already taken or will be taking the equipment today. You can modify the following procedure which prints the customers who are using currently the equipment. current bookings means that SYSDATE is between from and to dates and the status is “taken”.

CREATE OR REPLACE PROCEDURE list\_bookings

-- this procedure list all the customers

-- who have currenty the equipment out

AS

CURSOR c\_data

IS

SELECT customer\_id, first\_name, last\_name

FROM customer

WHERE customer\_id IN

(SELECT DISTINCT customer\_id

FROM booking

WHERE SYSDATE BETWEEN b\_from\_date AND b\_to\_date

AND status = 'taken');

BEGIN

dbms\_output.put\_line ('Customers with bookings' || SYSDATE);

FOR v\_rec IN c\_data

loop

dbms\_output.put\_line (

v\_rec.customer\_id || ' '|| v\_rec.first\_name || ' '|| v\_rec.last\_name);

END LOOP;

END;

1. Draw a conceptual model using UML notation for the following requirements. Indicate primary keys and specify multiplicities for the relationships (associations). Note (chapter 12 p.377) multiplicity consists of two constraints: participation (minimum) and cardinality (maximum).

Kamloops Antiques buys and sells one-of-a-kind antiques of all kinds (furniture, watches, jewelry, china, books, etc.) Each item has a unique item number, a description, asking price, condition, and remarks. Stillwater works with its clients who sell items to the store and buy items from the store (some clients sell and buy, some just sell, some only buy). Clients are identified by unique numbers. Stillwater keeps the following information about clients: first and last name, street address, city, province, postal code, home phone number, work phone number, and remarks. When Stillwater sells an item to a client, the owners want to record the commission paid, the actual selling price, sales tax (if applicable), and date sold. When Stillwater buys an item from a client, the owners want to record the purchase cost, date purchased, and condition at time of purchase.

1. Use the following relations

Hotel (hotelNo, hotelName, city)

Room (roomNo, hotelNo, type, price)

Booking (hotelNo, guestNo, roomNo, dateFrom, dateTo, guestName, )

Guest (guestNo, guestName, guestAddress)

1. The relation **Booking** is in the
   1. UNF
   2. 1NF
   3. 2NF
   4. 3NF

Create a relational schema for the ER diagram shown below.

Dependent

dName {PPK}

dAddress

relationship

RelatedTo

0..\*

1..1

Company

cName {PK}

cAddress

cTelNo

cFaxNo

Operates

1..1

1..\*

Department

dNo {PK}

dName

dLoc

Employs

1..1

Employee

empNo {PK}

name

fName

lName

address

Has

0..\*

1..1

Employment History

compOrganName (PPK}

address

position

sDate

fDate

1..\*