UOW AUSTRALIA

School of Computing and Information Technology

Student to complete:

Family name	
Other names	
Student number	
Table number	

CSCI235 Database Systems Wollongong Campus

Final Examination Paper Spring Session 2021

Exam duration	3 hours + 30 minutes of submission time
Weighting	60 %
Items permitted by examiner	*Open Book*
Aids supplied	None
Directions to students	6 questions to be answered. Submit an answer to each question in a separate `txt` or `doc`, or `docx` or `pdf` file through Moodle. If an answer is hand-written then take a picture or scan it and submit a file in either `pdf, jpeg, jpg, gif, bmp, png,` or `tiff` format through Moodle. In all cases it is your responsibility to make sure that the submitted files are readable. The files that cannot be read will not be evaluated.

Marks will be deducted for the late submissions at a rate of 1 mark per 1 minute late.

Question 1 (Total 10 marks)

Consider the relational schemas given below and the respective sets of functional dependencies valid in the schemas. Assume, that each schema is in 1NF.

For each one of the relational schemas, determine the highest normal form, which is valid for a schema. **Justify your answer.** Justification must include the derivations of minimal keys from the functional dependencies and testing the validity of all normal forms (2NF, 3NF, BCNF) against the relational schemas, minimal keys, and functional dependencies.

If a schema is not in BCNF, then decompose it into a *minimum number of relational schemas* such that each one of them is in BCNF.

(1) R = (A, B, C, D)
$B \rightarrow AD$
$A \rightarrow B$

Here, A can get B, and B can get A and D.
Thus A -> ABD, C is independent. AC -> ABCD,
AC are candidates and prime attributes. BD are non-prime attributes.
2NF Test:
B, as a non-prime, is partially dependant on AC, only dependant on A.
Thus this relational schema is not 2NF, not to mention 3NF, BCNF.
To become a BCNF, We can decompose R=(A,B,C,D) into:
1.R1=(B,A,D)
R2=(A,C)
2.R1=(B,A,D)
R2=(B,C)

(2) R = (A, B, C, D)
$BC \rightarrow A$
$B \rightarrow A$

Here, A is an non-prime attribute,
B->A, BC->A
BCD+ -> ABCD
Thus BCD are candidate and prime.
In this situation, A is partial dependant on B, not BC, which violates 2NF.
So this relational schema is 1NF.

---

(3) R = (A, B, C, D)
$A \rightarrow BCD$
$D \rightarrow A$

A -> BCD
Thus a is a candidate and a prime attribute.
BCD are non-prime.
There is no relation indicates non-prime attributes partially dependant on candidates.
So it satisfy 2NF.
3NF Testing: There is no transitive dependency between A and D, so satisfy 3NF
Now BCNF,
A is a super key, while D -> A, but here it does not violate BCNF
So BCNF.

(4) R = (A, B, C, D)
No functional dependencies are valid in a schema R.

R=(ABCD) satisfies all normal forms, including BCNF, because it does not have functional dependancies. By default we assume that R=(ABCD) without functional dependancies satisfy BCNF.

THE QUESTIONS 2, 3 and 4 REFER TO THE RELATIONAL TABLES LISTED BELOW

The schemas of relational tables, specifications of primary, candidate, foreign keys and check constraints are given below.

```
CREATE TABLE CUSTOMER(
 CUSTOMER_CODE      VARCHAR(5)   NOT NULL,     /* Customer code                        */
 COMPANY_NAME       VARCHAR(40)  NOT NULL,     /* Company name                         */
 COUNTRY            VARCHAR(15)  NOT NULL,     /* Country                              */
 PHONE              VARCHAR(24)  NOT NULL,     /* Phone number                         */
  CONSTRAINT PK_CUSTOMER PRIMARY KEY (CUSTOMER_CODE) );

CREATE TABLE SUPPLIER(
 COMPANY_NAME       VARCHAR(40)  NOT NULL,     /* Company name                         */
 COUNTRY            VARCHAR(15)  NOT NULL,     /* Country                              */
 PHONE              VARCHAR(24)  NOT NULL,     /* Phone number                         */
    CONSTRAINT PK_SUPPLIER PRIMARY KEY (COMPANY_NAME) );

CREATE TABLE PRODUCT(
 PRODUCT_NAME       VARCHAR(40)  NOT NULL,     /* Product name                         */
 SUPPLIER_NAME      VARCHAR(40)  NOT NULL,     /* Supplier name                        */
 CATEGORY_NAME      VARCHAR(30)  NOT NULL,     /* Category name                        */
 UNIT_PRICE         NUMBER(10,2) DEFAULT 0,    /* Price per unit                       */
 DISCONTINUED       CHAR(1)      DEFAULT 'N',  /* Either discontinued or available     */
  CONSTRAINT PK_PRODUCT PRIMARY KEY (PRODUCT_NAME),
  CONSTRAINT FK_SUPPLIER_NAME FOREIGN KEY (SUPPLIER_NAME)
                             REFERENCES SUPPLIER(COMPANY_NAME),
  CONSTRAINT CK_PRODUCT_UNIT_PRICE CHECK (UNIT_PRICE >= 0),
  CONSTRAINT CK_PRODUCT_DISCONTINUED CHECK (DISCONTINUED in ('Y','N')) );

CREATE TABLE ORDERS(
 ORDER_NUM          NUMBER(9)    NOT NULL,     /* Order number                         */
 CUSTOMER_CODE      VARCHAR(5)   NOT NULL,     /* Customer code                        */
 ORDER_DATE         DATE         NOT NULL,     /* Order date                           */
  CONSTRAINT PK_ORDERS PRIMARY KEY (ORDER_NUM),
  CONSTRAINT FK_CUSTOMER_CODE FOREIGN KEY (CUSTOMER_CODE)
                             REFERENCES CUSTOMER(CUSTOMER_CODE) );

CREATE TABLE ORDER_DETAIL(
 ORDER_NUM         NUMBER(9)     NOT NULL,    /* Order number                         */
 PRODUCT_NAME      VARCHAR(40)   NOT NULL,    /* Product name                         */
 QUANTITY          NUMBER(9)     NOT NULL,    /* Quantity ordered                     */
  CONSTRAINT PK_ORDER_DETAIL PRIMARY KEY (ORDER_NUM, PRODUCT_NAME),
  CONSTRAINT FK_ORDER_ID FOREIGN KEY (ORDER_ID)
                        REFERENCES ORDERS (ORDER_ID),
  CONSTRAINT FK_PRODUCT_NAME FOREIGN KEY (PRODUCT_NAME)
                            REFERENCES PRODUCT (PRODUCT_NAME),
  CONSTRAINT CK_ORDER_DETAIL_QUANTITY CHECK (QUANTITY > 0) );
```

Question 2 **(Total 10 marks)**

This question is related to a sample database created through processing of `CREATE TABLE` statements listed on a page 3 of the examination paper.

(1) Write an implementation of a **statement trigger** that verifies in a sample database the following consistency constraint:

"All products supplied by a supplier must belong to at most 3 different categories".

Your implementation must consider both `INSERT` and `UPDATE` events.

`Use a procedure RAISE_APPLICATION_ERROR` to abort a transaction that attempts to violate the consistency constraint listed above.

(4 marks)

(2) Write SQL statements that extend a sample database with information about the total number of products supplied in each category. SQL statements must create additional data structures and must fill the new structure with data consistent with the present contents of a sample database

(2 marks)

(3) Write an implementation of a **row trigger** that automatically updates information about the total number of products supplied in each category.

Your implementation must consider both `INSERT` and `DELETE` events.

(4 marks)

Question 3 **(Total 10 marks)**

This question is related to a sample database created through processing of `CREATE TABLE` statements listed on a page 3 of the examination paper.

The following PL/SQL stored function `AVGPROD` finds an average number of products supplied by each supplier located in a given country.

```
CREATE OR REPLACE FUNCTION AVGPROD(supplier_country IN VARCHAR ) RETURN NUMBER
IS
total_products  NUMBER;
total_suppliers NUMBER;

BEGIN
 SELECT COUNT(*)
 INTO total_suppliers
 FROM SUPPLIER
 WHERE COUNTRY = supplier_country;

 SELECT COUNT(*)
 INTO total_products
 FROM PRODUCT JOIN SUPPLIER
           ON PRODUCT.SUPPLIER_NAME = SUPPLIER.COMPANY_NAME
 WHERE COUNTRY = supplier_country;

  IF total_suppliers = 0 THEN
   RETURN 0;
  ELSE
   RETURN total_products/total_suppliers;
  END IF;

END AVGROD;
```

Assume that a stored function `AVGPROD` is processed as a database transaction running at `READ COMMITTED` isolation level, Assume, that a database system uses the scheduler implemented by Oracle 19c database server (snapshot isolation protocol explained to you during the lecture classes).

Show a sample concurrent execution of the stored function listed above, such that it interleaves its operations with the other transaction and such that the results returned by the function are incorrect. The other transaction is up to you.

When visualizing the concurrent executions use a technique of two-dimensional diagrams presented to you during the lecture classes, for example, see a presentation 14 Transaction Processing in Oracle DBMS slide 16 and the following slides.

(8 marks)

Write the comprehensive explanations on why the concurrent execution proposed in the previous step returns the incorrect results.

(2 marks)

Question 4 **(Total 10 marks)**

This question is related to a sample database created through processing of `CREATE TABLE` statements listed on a page 3 of the examination paper.

Assume, that the sample database has been created and loaded with data on a "host server" at `data-pc01.adeis.uow.edu.au`. Assume, that SQL scripts `dbcreate.sql` and `dbload.sql` have been used to create the relational tables and to load data into the tables.

(1) Explain how the same and **empty relational tables** can be created on a "remote server" at `data-pc02.adeis.uow.edu.au`. Apply your user name and password for a "remote server". Assume, that system identifier on a "remote server" is `db` and that a "remote server" listens to a port `1521`.

(2 marks)

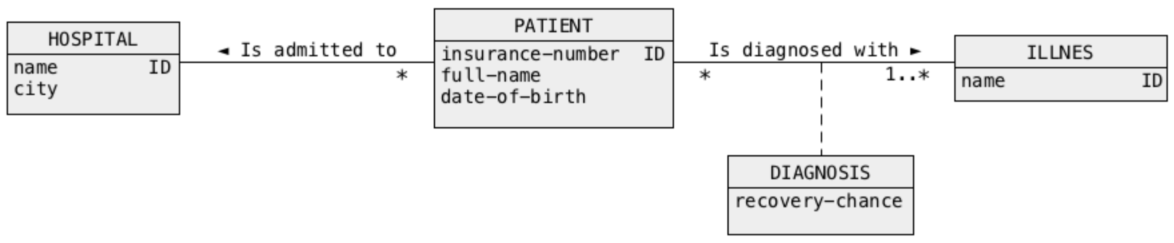
(2) Write SQL statements to move information about the orders submitted in 2021 and the customers who submitted such orders from a "host server" to a "remote server". Assume, that all SQL statements must be processed by a "host server". It means, that you must create and you must use database links.

Hint
Note, that some customers submitted the orders before 2021 and that the orders submitted in 2021 can be related to any product listed in a database. It means that some of the relational tables must be replicated on both "host server" and "remote server" and some of the tables must be partially moved from a "host server" to a "remote server".

(8 marks)

Question 5 (Total 10 marks)

Consider a conceptual schema given below. The schema represents a sample database domain where the patients are admitted to the hospitals and the patients are diagnosed with the illnesses.



(1) Transform a conceptual schema given above into a logical schema of BSON documents. For a sample logical schema of BSON documents see presentation 23 BSON Design, slide 53.

(7 marks)

(2) For a logical schema created in the previous step write the sample BSON documents whose contents are consistent the logical schema. Your documents must contain information about at least one hospital, two patients, each diagnosed with two illnesses, and three illnesses. The values associated with the keys in BSON documents are up to you.

(3 marks)

Question 6 **(Total 10 marks)**

Consider a sample BSON document given below. Assume that all documents in a BSON collection `drivers` have the same structure as the document listed below.

```
db.driver.insert(
  { "first_name":"James",
    "last_name":"Bond",
    "licence":007,
    "address":{"street":"Northfields Ave",
               "bldg":3,
               "city":"Wollongong",
               "country":"Australia"},
   "trips":[ {"number":5,
              "truck_rego":"PKR856",
              "date":"12-DEC-2017",
              "legs": [ {"number":1,
                        "departure":"Sydney",
                        "destination":"Melbourne" },
                       {"number":2,
                        "departure":"Melbourne",
                        "destination":"Sydney" } ] },
             {"number":25,
              "truck_rego":"AL08UK",
              "date":"03-JUN-2018",
              "legs": [ {"number":1,
                        "departure":"Sydney",
                        "destination":"Melbourne" } ] }
           ]
    }
);
```

Use a method `aggregate()` available in MongoDB to write the implementations of the following queries.

(1) List the names of countries where the drivers are living in. List the results as a sequence of pairs `{"country": string, "city":string}`.

For example,
```
{"country":"Australia", "city":"Dapto"},
{"country":"UK","city":"Liverpool"),
    ...     ...     ...     ...
```

(2 marks)

(2) Find the registration number of trucks used by a driver `James Bond`. List the results as a sequence of pairs `{"registration": string}`.

For example,
```
{"registration":"PKR856"},
{"registration":"AL08UK"),
    ...     ...     ...     ...
```

(2 marks)

(3) Find the first and the last names of all drivers together with the total number trips performed by each driver.

(2 marks)

Use a method a method `update()` available in MongoDB to write the implementations of the following modifications to the documents in a collection `drivers`.

(4) Change a date of a trip number `5` performed by `James Bond` to `28-SEP-2021`.

(2 marks)

(5) Remove information about a trip number `25` performed by `James Bond`.

(2 marks)

End of examination paper