RS211206

DUBLIN INSTITUTE OF TECHNOLOGY

# BSc. (Honours) Degree in Computer Science (Infrastructure)

**Year 2**

**SUPPLEMENTAL EXAMINATIONS 2017**

## DATABASES 1 [CMPU2007]

PATRICIA O'BYRNE

DR. DEIRDRE LILLIS

TOM NOLAN

DATE AND TIME, e.g.

TBA
2 HOURS

ANSWER **QUESTION 1 AND 2 OTHER** QUESTIONS.

THERE IS A SYNTAX SHEET ON THE FINAL PAGE TO ASSIST YOU.

The Red Hill Rovers bicycle club is for cyclists who race competitively. Each cyclist has a unique id, a name and may give an address. The club owns bicycles (bikes) of all different models. Each bike is uniquely identified by its serial number. The club also keeps details on the model of the bike and sometimes the description of that model. There are a many bikes of each model. Each bike's weight and sometimes the wheel type and wheel diameter is also recorded.

When a race is advertised within the cycling fraternity, the club records it and records any of its cyclists that are going to take part in the race, along with the serial number of the bike they will ride. When the race is finished, the club updates the entry with the cyclists' finishing times.

**Case Study 1 Red Hill Rovers**

1. (a) List the primary entities for the Redhill Rovers bicycle club. (5 marks)

   (b) List the attribute(s) that you would choose to be the primary key for each entity. (5 marks)

   (c) Illustrate the relationship between your entities, in an Entity Relationship Diagram, using IE notation (also known as crow's foot notation). (5 marks)

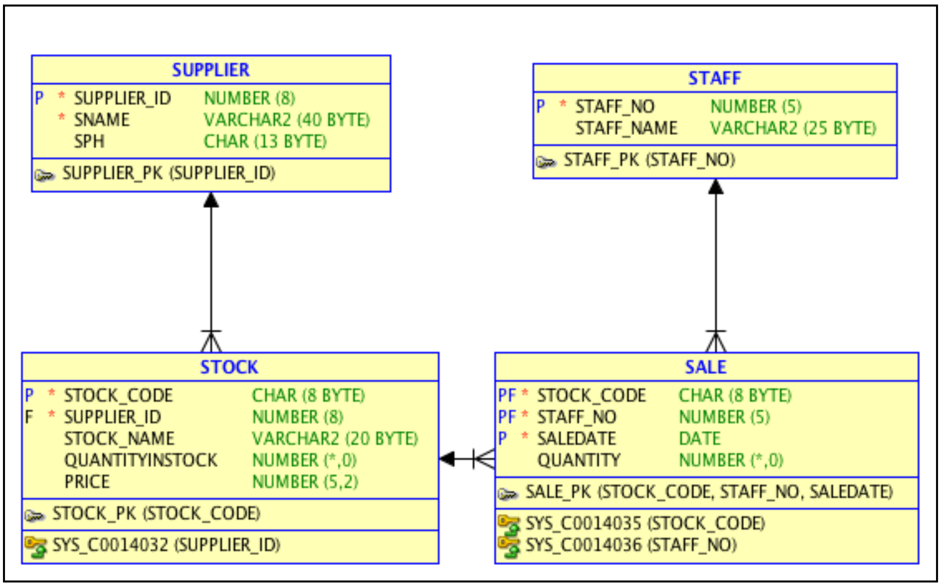
   (d) List the attributes for each of your entities. (5 marks)

   (e) Draw a full Entity Relationship Diagram, underlining primary keys and marking foreign keys with an asterisk. (10 marks)

   (f) What constraints would you add to ensure the following:

      (i) Every bike must weigh at least 6.8kg. (5 marks)

      (ii) When a race is added, the duration must be recorded, but the date may not be determined yet. (5 marks)


SUPPLIER
P * SUPPLIER_ID NUMBER (8)
* SNAME VARCHAR2 (40 BYTE)
SPH CHAR (13 BYTE)
SUPPLIER_PK (SUPPLIER_ID)
STAFF
P * STAFF_NO NUMBER (5)
STAFF_NAME VARCHAR2 (25 BYTE)
STAFF_PK (STAFF_NO)
STOCK
P * STOCK_CODE CHAR (8 BYTE)
F * SUPPLIER_ID NUMBER (8)
STOCK_NAME VARCHAR2 (20 BYTE)
QUANTITYINSTOCK NUMBER (*,0)
PRICE NUMBER (5,2)
STOCK_PK (STOCK_CODE)
SYS_C0014032 (SUPPLIER_ID)
SALE
PF * STOCK_CODE CHAR (8 BYTE)
PF * STAFF_NO NUMBER (5)
P * SALEDATE DATE
QUANTITY NUMBER (*,0)
SALE_PK (STOCK_CODE, STAFF_NO, SALEDATE)
SYS_C0014035 (STOCK_CODE)
SYS_C0014036 (STAFF_NO)

Description: Each stock item sold from the shop has a stock code, a name and a price. Usually, but not always, it comes from a registered supplier. When someone new comes to work in the shop, they are given a staff_no and their name is recorded. Whenever a sale is made, the stock code and quantity of stock sold is recorded, along with the date of sale, and the staff_no of whoever made the sale. The sale is given a unique sale_id number.

**Case Study 2 Campus Shop ERD and Description**

2. (a) Write a query to return the stock_code, quantity and staff_no of all sales made in the last ten days, in staff_no order. (10 marks)

(b) Write a query to return each staff member's staff_no and the number of sales he / she has made. (10 marks)

(c) Write a query to return the staff member's staff_no and the number of sales he / she has made for all staff members who have made more than 10 sales. (10 marks)

3. (a) Write a query to return the staff_name, stock_name and worth (i.e. quantity x price) of all sales made in the last 7 days. (10 marks)

(b) Write a query to return the staff_name of any staff member who has never made a sale. (10 marks).

(c) Write a query to return the staff_name, stock_name and worth (i.e. quantity x price) of all sales of stock supplied by the supplier named (sname) ‘Cadbury’. (10 marks)

4. (a) Write an SQL statement to add a new supplier to your database, where the supplier's name (sname) is 'Kevin Watson' and his phone number (sph) is 087789098. You may decide on how to allocate a Supplier_Id. (10 marks)

(b) Write a SQL statement to add a new stock item with Stock_Code 'CAFFBAR' to your database, using Kevin Watson as the supplier, whom you have added in part (a) (10 marks)

(c) Combine the above into a single transaction (not a single statement), making it permanent before exiting. (10 marks)

**ALTER TABLE *tablename***
**{[ADD | MODIFY] *column-definition*}**

**CREATE TABLE *tablename***
**(*{column-definition}***
**[PRIMARY KEY (*{column-name},*)]**
**{[FOREIGN KEY (*{column-name}*) REFERENCES *tablename*]})**

**COMMIT**

**DELETE FROM *tablename* WHERE *condition***

**DROP [TABLE *tablename*|DROP VIEW *viewname*]**

**INSERT INTO *tablename* [*{column-name,}*] VALUES (*data-value-list*)**

**ROLLBACK WORK**

**SELECT *column-list* FROM *tablename***
**[JOIN] *tablename2* [USING (*attribute-list*)|ON *condition*]**
**[WHERE *condition*]**
**[ORDER BY *column-list*]**
**[GROUP BY *column-name*]**
**[HAVING *condition*]**
***Conditions : =,>,<,>=,<=,<>,* BETWEEN .. AND.., IN *(list),* IS NULL, LIKE**
***Logical operators:* AND, OR, NOT**
***Set operations:* UNION, MINUS, INTERSECT**

**SELECT SYSDATE FROM DUAL;**

**SELECT column-list FROM**
**<tablelist> WHERE <column-list> IN | ALL |ANY |EXISTS (Select statement)**

**UPDATE *tablename***
**[SET *column-name*= <*data-value*>] [WHERE *condition*]**

***Column-definition = column-name* [CHAR [(*n*)] | VARCHAR(*n*) | NUMBER [ *n,p*] | DATE | DATETIME] {[NOT NULL | UNIQUE | PRIMARY KEY]}**