

ISYS1055/1057 Database Concepts S1/2019 Assignment 2

This is an individual assignment. Plagiarism in oral, written or visual presentations is the presentation of the work, idea or creation of another person, without appropriate referencing, as though it's one's own. Plagiarism is not acceptable and may result in charges of academic misconduct which carries a range of penalties. It is also a disciplinary offence for students to allow their work to be plagiarised by another student.

Submission

Final submission is due at 23:59 27 May 2019 Monday in Week 12. Submit via the assignment submission system on Canvas ONE pdf file named after your student number. The penalty for late submission is 3 marks per day or part day. After five days, assignments get 0 marks.

- **Your submission is successful only when you see your submitted file in your Canvas account. No confirmation email is sent to you.**
- Never leave submission to the last minute -- you may have difficulty uploading files.
- You can submit multiple times – a new submission will override any earlier submissions.
But if your final submission is after the due time, late penalty applies.

Assessment

There are five questions of 30 points in total (10+4+4+7+5=30). This assignment is worth 30% of the overall assessment for ISYS1057 and ISYS1055. The assessment components and weights for the course are:

Assignment 1	Assignment 2	Exam
20%	30%	50%

There is not a hurdle for any assessment component.

Question 1. SQL (10 points).

*In addition to the lecture notes, you should also study by yourself the SQL*Plus tutorial on Canvas (the Oracle section) and other resources for syntax and useful functions.*

The relational schema for the Academics database is as follows:

```
DEPARTMENT(deptnum, descrip, instname, deptname, state, postcode)
ACADEMIC(acnum, deptnum*, famname, givenname, initials, title)
PAPER(panum, title)
AUTHOR(panum*, acnum*)
FIELD(fieldnum, id, title)
INTEREST(fieldnum*, acnum*, descrip)
```

Some notes on the Academics database:

- An academic department belongs to one institution (instname) and often has many academics. An academic only works for one department.
- Research papers (PAPER) are often authored by several academics, and of course an academic often writes several papers (AUTHOR).
- A research field (FIELD) often attracts many academics and an academic can have interest in several research fields (INTEREST).

Primary keys are underlined and foreign keys are marked with *. You should download the SQL script for defining and populating the database *academics.sql* from Canvas (the Oracle section) and run *academics.sql* in your Oracle account to build the database.

Write ONE SQL query for each of questions 1.1) through to 1.10). Your query must run in Oracle SQL Developer. Each query is worth 1 point.

Notes for marking:

- Each question is worth 1 point.
- For questions with “*You must ...*” or “*You must not ...*” requirement, queries failing to meet the requirement receive maximum 0.5 mark. For example, question 1.2) has “*You must use a subquery*”. A query not using the subquery operator receives maximum 0.5 mark.
- Do not include the output of the query or the script used to create the tables.
- Your query should not output duplicates but use DISTINCT only if necessary.
- Queries are marked in terms of both correctness and efficiency. Unnecessary joins will incur deduction.

- 1.1) For each academic, give the acnum, givenname, famname and the total number of papers s/he has written. Note that if an academic has not written any paper, his/her total should be zero. You can use or not use JOIN operators.
- 1.2) List departments where at least one academic does not have any research interest. List the deptnum, deptname and instname of these departments. You must use a subquery.
- 1.3) List the fieldnum, title and the total number of interested academics (under the heading "NO. ACADEMICS INTERESTED") for each research field that some academics are interested in. The list should be in increasing order of fieldnum. Note: research fields that no academics are interested in are excluded.
- 1.4) Find research fields that have at least ten interested academics. Give the fieldnum, title and the number of interested academics for these research fields.
- 1.5) Return the acnum of academic(s) who wrote the largest number of papers. You must NOT use MAX. An SQL query that lists all academics in decreasing order of their total number of papers is incorrect.
- 1.6) Give the total number of academics who have not written any papers. You must use the NOT IN operator.
- 1.7) Find research fields whose title contains the string 'data' and has at least one interested academic from the department with deptnum 100. List the fieldnum and title of these research fields. You must use the EXISTS operator. Ensure your query is case-insensitive.
- 1.8) List papers (panum) that have authors from the same department. Do NOT use any JOIN operator. Do NOT use any subqueries.
- 1.9) The SQL query below is meant to list research fields (fieldnum) where **no** academics from 'RMIT CS' (deptnum=126) have an interest. But it is incorrect. Give the correct SQL query.

```

select fieldnum
from interest, academic
where interest.acnum=academic.acnum and deptnum != 126;

```
- 1.10) Consider the SQL query given below. Give the English explanation for the output of a) the subquery, and b) the whole SQL query. Literal explanation will receive zero marks.

```

select S.acnum
from interest S
where upper(S.descrip) like '%LOGIC%'
and exists (select fieldnum
            from interest
            where acnum = S.acnum

```

and upper(descrip) not like '%LOGIC%');

Question 2. The Relational model (4 points).

Consider the Customer Order database of the ABC company, which keeps data for customers and their orders. A database of only one relation is designed, as shown below:

ABC(customerNo, customerName, phone, streetAddr, city, state, postCode, orderNo, productNo, quantity, unitPrice, salesRep)

- A customer has a unique customer number (customerNo) and is also described by name, phone number, street address (streeAddr), city, state, and postCode.
- An order is identified by a unique order number (orderNo).
- A product has a unique product number (productNo) and has a unit price (unitPrice).
- A customer can place several orders and an order is made by only one customer.
- An order can include several products and each has some quantity, and a product can be on several orders.
- An order has one sales representative (salesRep) and one sales representative can be responsible for several orders.

Answer questions:

- 2.1) (1 point) Give all likely FDs.
- 2.2) (1 point) Give the candidate keys for the ABC relation. Explain your answer.
- 2.3) (1 point) Give {CustomerNo}⁺ and {orderNo, salesRep}⁺ based on the FDs for Question 2.1).
- 2.4) (1 point) Is the relation ABC in BCNF or 3NF? Explain your answer.

Question 3. Normalisation (4 points).

A shop keeps data about customers, salesmen, products and orders. Consider the attributes below and their associated FDs, where emplID is the employee ID for salesmen.

custNo \rightarrow custName, address, credit-limit, discount
productNo \rightarrow price, desc
orderNo \rightarrow emplID, custNo, discount
orderNo, productNo \rightarrow quantity, emplID
custNo, orderNo \rightarrow discount
orderNo \rightarrow address

Consider the below Transaction relation;

Transaction (custNo, emplID, productNo, orderNo, quantity, discount)

Answer questions.

- 3.1) (1 point) Give the minimal basis for the given FDs.
- 3.2) (1 point) The Transaction relation is not in BCNF or 3NF. Give the reason using the FDs on attributes.
- 3.3) (2 points) Follow the 3NF decomposition algorithm to decompose the Transaction relation into relations in BCNF or 3NF. Give;
 - The relations after decomposition.
 - Specify the primary key for each relation.
 - Specify any foreign keys for each relation.

Question 4. ER model (7 points).

Due to your experience designing the database for the O Athletic Games, you are asked to design the ER diagram for a bigger database for the 2020 Summer Olympic games in Tokyo (<https://tokyo2020.org/en/>). Requirements are as follows.

- The Tokyo 2020 Olympic Games will feature 33 sports, as shown on this page: <https://tokyo2020.org/en/games/sport/olympic/>. Some sports organise events into disciplines (for ease of scheduling events). For example, the Hockey sport has two events “Men 12-team tournament” and “Women 12-team tournament”. But the Aquatics sport has disciplines such as Swimming and Diving, and Swimming has events like “Men 100m freestyle” and “Women 100m butterfly”.
- Each athlete competing at the Olympics has a unique athlete number and has a first name, last name, and other personal details. Each athlete must represent a country.
- Each Olympic venue has name and address and description of facilities. Each venue is designed for some sports or disciplines. For example, on this page: <https://tokyo2020.org/en/games/venue/olympic/>, the Olympic Stadium is designed for the purpose of opening and closing ceremonies, athletics and football events.
- Data should be kept for event schedules. Events are scheduled to venues with date and time according to its design purpose, as on this page: <https://tokyo2020.org/en/games/schedule/olympic/>. An event has many participants and they first compete in groups numbered sequentially and then some are selected to participate in the final competition event for the medals.
- Event results for all athletes, including medals, rank and result recorded should be kept in the database. As a fictitious example, for Women’s 50 metres freestyle swimming event, Cate Campbell won the Gold medal in first place with a recorded time of 24.42 seconds; Amanda Smith is placed fourth with a recorded time of 25.50 seconds.

You shall design the database such that it can be used to

- Search for Olympians and their results.
- Search the event schedule for event details.
- List sports and their events.

You are encouraged to use the “Olympian Search Page” of the Australian Olympic Committee as a motivating example for the usage of your database: <http://olympics.com.au/olympians>. The difference is that your database is only for the 2020 Tokyo Olympics and does not keep historical data for previous games.

According to the requirements, give the ER diagram for the database using the UML class symbols (as used in the lecture notes), making assumptions where necessary. You must

represent entities, relationships and their attributes, and all applicable constraints in your diagram. Explain any constraints that can not be expressed in the diagram.

Question 5. ER to relational schema mapping (5 points).

Consider the below Metro Train database ER model using the UML Class symbols.

- The Employee-Driver-TicketInspector-StationMaster subclass hierarchy is partial and disjoint.
- A train line can have several runs numbered sequentially and it can be express.
- Some stations are premium stations with facilities such as toilets.

For ease of reading the ER diagram can also be downloaded at <https://drive.google.com/open?id=1zIEiXCipBEJ7sdt3tZIB-8TtHTdE2iin>. Map the diagram to a relational database schema. Indicate the primary key (underline) and any foreign keys (asterisk) in each relation.

