THE UNIVERSITY OF SYDNEY

School of Information Technologies
Bryn Jeffries

COMP9120/INFO2x20: Database Systems **Semester 1, 2017**

Assignment 2: Logical Database Design

1 Introduction

This assignment takes you through the process of translating an ER diagram into a relational model and implementing this with DDL. The objectives are to gain practical experience in logical database design and to experience group-based data design work. This is a group assignment for teams of 2 to 3 members. You should inform your tutor of any changes to your group membership by the Week 6 tutorial.

Please also keep an eye on the discussion forum and further announcements in Ed STEM. There will undoubtedly be issues and further clarification required for this assignment, and you are expected to read through this document early on and ask any questions for have on Ed.

2 Design Brief: Relational Database Schema for Olympic Games

Your task is to create a relational database schema translated from an entity-relationship diagram. The ER diagram is supplied at the end of this document, based upon an example solution to the Olympics scenario started in Assignment 1 (INFO2120 and COMP9120 students should use the diagram on Page 6, INFO2820 student should use the one on Page 7). Note that there is an additional Staff entity type, inheriting from Member, to represent people who perform office tasks such as booking journeys for other members.

In additional to the model given, the following details apply:

- All Member IDs are created as 10-digit numeric codes;
- All Vehicle codes are created as 8-digit alphanumeric codes;
- The `medal` attribute on the *participates* relationship can be 'gold', 'silver' or 'bronze', or left as NULL if no medal was received.

2.1 Expectations

A credit-level solution should show sound knowledge of logical database implementation by including:

- Tables and attributes with suitable data types to capture all information in the model;
- Appropriate key constraints (PRIMARY KEY, UNIQUE) for all tables;
- Correct foreign key constraints ON DELETE clauses where suitable;
- Appropriate additional integrity constraints expressed by means of, e.g., NOT NULL, DEFAULT or CHECK clauses;
- INSERT statements to populate each relation with at least one record, to demonstrate a database instance consistent with the ER model.

For Distinction/High Distinction level submissions, proficiency can be demonstrated through appropriate use of the following features:

- examples of DML statements to demonstrate functionality of integrity constraints (put in a separate file to your main DML statements)
- an *assertion* definition (commented out in your DDL) to constrain the `nbooked` attribute of `Journey` to be kept consistent with the number of bookings make for that journey, and a *trigger* definition giving equivalent functionality;
- a *view* to present all details of an athlete (including those inherited from Member), plus columns reporting the number of gold, silver and bronze medals received by that athlete.
- advanced domain constraints, e.g., using regular expressions.

2.2 Advanced component for INFO2820 Students

The ER diagram provided on Page 7 includes an implementation of the hierarchical locations, and team-based events. These extra components are shown in red, and must also be implemented in the relational database.

You should map the two `participates` relationships for team and individual events so that all results are maintained in a common table. This should be supported by a commented *assertion* and equivalent *trigger* to enforce the constraint that only team-based results should be recorded for team events, and individual athletes' results for individual events.

3 Submission Details

A final submission of the whole assignment must be made via elearning by 11:59pm Tuesday of Week 8. Regular progress updates will be expected in tutorials. **All group members are expected to attend tutorials to explain their attempts.** Please consult sit.info2120@sydney.edu.au (undergraduates) or sit.comp9120@sydney.edu.au (postgraduates) if you are unable to attend your tutorial so that alternative arrangements can be made.

You should prepare for the Week 6 tutorial by bringing a draft RM diagram, showing all relations, primary keys and foreign keys, to discuss with your tutor.

We intend to provide a code submission portal to give immediate feedback on parts of your DML and DDL. Monitor Ed STEM for further details.

3.1 Submission Items

Please ensure that your submission is a single file or ZIP archive that contains the following items:

- A (PDF, PNG or JPEG format) relational model diagram showing the mapping of the ER model to relations, clearly indicating all primary key and foreign key constraints;
- (Text format, `.sql` extension) DDL statements to create all the tables, along with appropriate constraints, for your relational model; and
- (Text format, `.sql` extension) DML statements to populate each of your tables with example data.

3.2 Plagiarism

By uploading your submission to eLearning your group implicitly agrees to abide by the University policies regarding academic honesty, and in particular that all the work is original and not plagiarised from the work of others. If you believe that part of your submission is not the work of your group members you must bring this to the attention of your tutor or lecturer immediately. See the policy slides released in Week 1 for further details. It is a Faculty Policy that all written assignments use TurnitIn to check for plagiarism within submissions.

In assessing a piece of submitted work, the School of IT may reproduce it entirely, may provide a copy to another member of faculty, and/or communicate a copy of this assignment to a plagiarism checking service or in-house computer program. A copy of the assignment may be maintained by the service or the School of IT for the purpose of future plagiarism checking.

3.3 Late submissions

An example solution to the assignment will be presented in the week of the final submission deadline, so please keep to the deadline. Late final submissions will be penalised 20% per day late.

4 Guidelines

4.1 RM Diagram

- Your diagram should be clearly laid out and easy to interpret, and follow the conventions introduced in the lecture material.
- Your RM diagram can spread across multiple pages, but please try to keep them neatly organised and target an A4 page size.
- Your relations can stack attributes horizontally (as in the lecture slides) or vertically (as in the Grok tutorials), but be consistent.
- You relations should capture all details and be **consistent with the provided ER diagram**.
- Take care to show **primary keys** for all relations by underlining their attributes.
- Take care to show all **foreign keys** by dashed-underlining those attributes and joining them with arrows to their referenced attributes.
- You can capture additional **constraints** (additional `UNIQUE` keys, important `NOT NULL` attributes and domains) as text annotations in your diagram.

4.2 Database DDL

- You should include all DDL statements necessary to fully instantiate a working database, **consistent with your RM diagram**.
- Your file should run without errors in PostgreSQL 9.5 (undergraduate) or Oracle 12c (postgraduate), and you may be required to demonstrate this to your tutor.
- You can annotate your statements using '--' at the start of lines of comment.
- Take care to include appropriate choices of data-types.

- You should clearly capture all `PRIMARY KEY`, `UNIQUE` **and** `FOREIGN KEY` **constraints**.
- It is recommended that you give meaningful names to all your constraints using the `CONSTRAINT` clause.
- You may also wish to refine your schema with `CHECK` constraints to enforce domains constraint, in which case consult your tutor.
- You should group your statements for ease of reading (e.g., by keeping all table constraints within the relevant `CREATE TABLE` statement rather than declaring them externally, if possible).

4.3 Example DML

- Your sample data file should use `INSERT` statements to populate every relation with at least one row of data.
- Your file should run without errors in PostgreSQL 9.5 (undergraduate) or Oracle 12c (postgraduate), and you may be required to demonstrate this to your tutor.

5 Marking

This assignment is worth 10% of your final grade for the unit of study. Your group's draft and final submissions will be marked according to the attached rubric.

5.1 Feedback

You will have several opportunities to present your progress to your tutor and receive feedback. You are expected to make use of this feedback in your final submission. Your group's final submission will receive minimal summative feedback, explaining your final mark with respect to the rubric.

5.2 Group member participation

If members of your group do not contribute sufficiently you should alert your tutor as soon as possible. The tutor has the discretion to scale the group's mark for each member as follows:

Level of contribution	**Proportion of final grade received**
No participation.	0%
Full understanding of the submitted work.	50%
Minor contributor to the group's submission.	75%
Major contributor to the group's submission.	100%

All group members should expect to be asked questions about their group's progress during tutorials, and should be able to explain any part of the work, not just their own contribution. The participation mark will be dependent upon how satisfactory these explanations are.

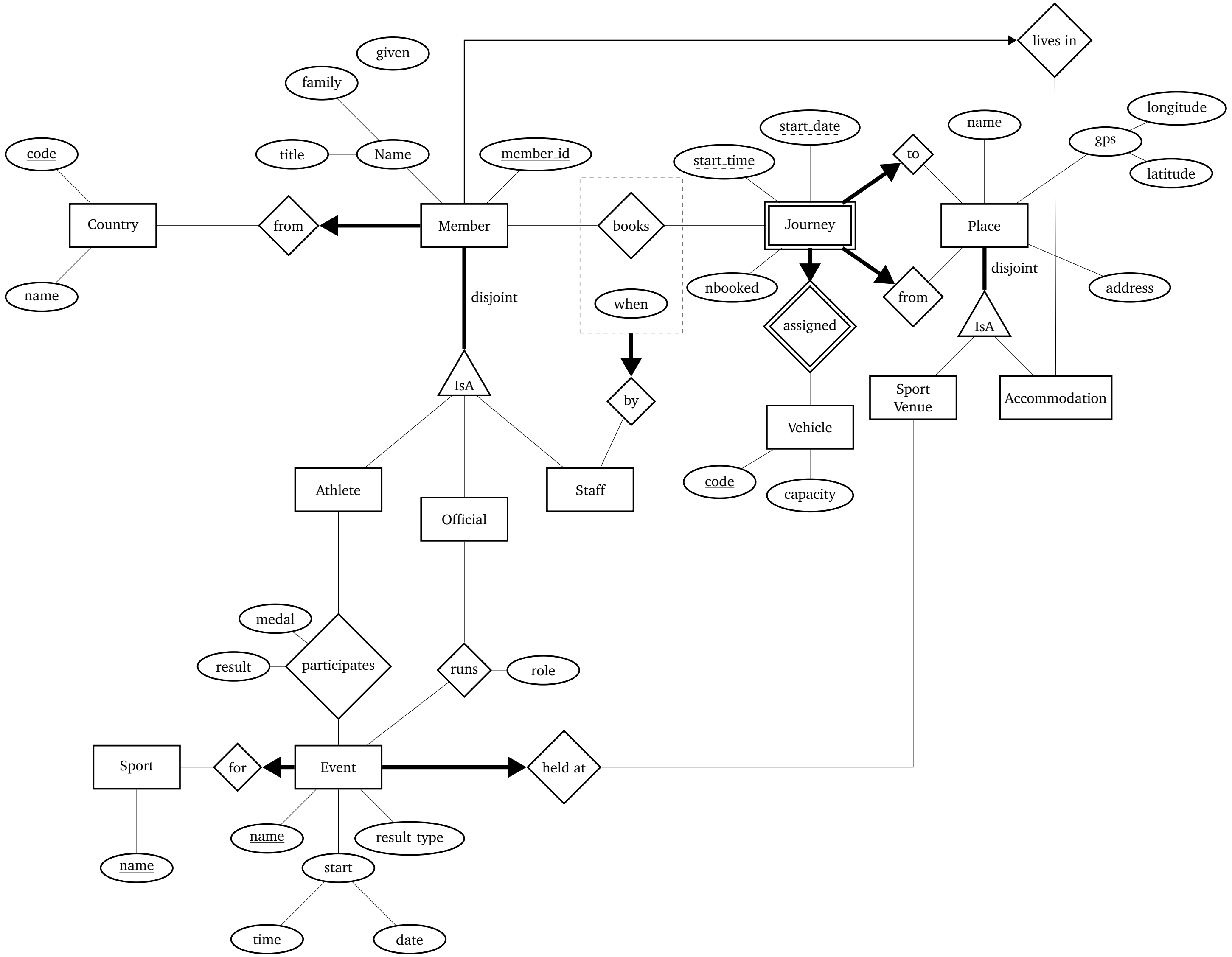
Note that a submission that cannot be adequately explained by multiple members of a group will be treated suspiciously, and may be referred for investigation of academic dishonesty.

5.3 Rubric

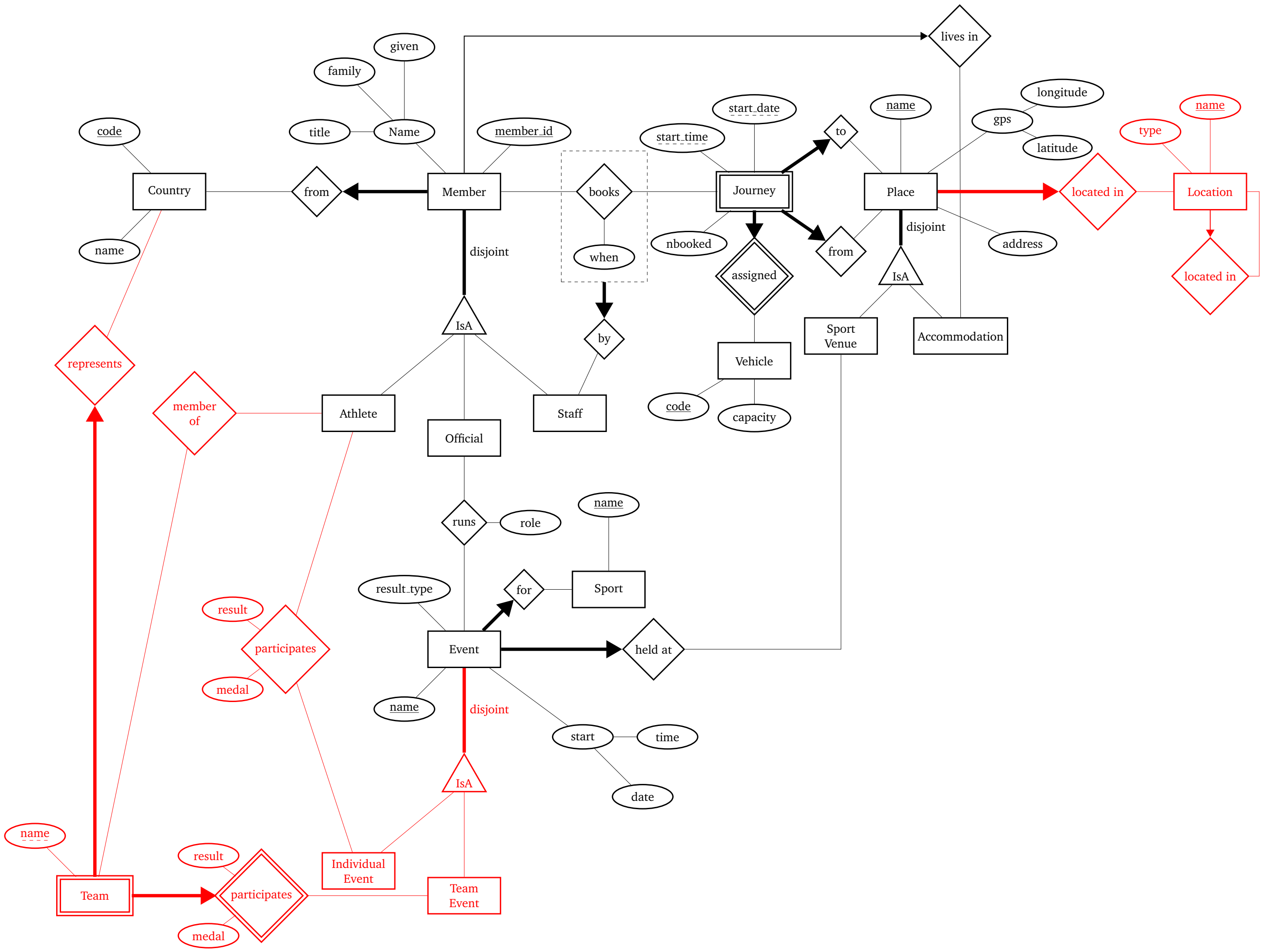
Your submissions will be marked according to the following rubric, with a maximum possible score of 10 points.

	Novice (0 pts)	**Competent (1 pt)**	**Proficient (2 pts)**
Relational Model	Less than competent model of the given scenario	Most entities and relationships of core model are correctly mapped	Complete mapping of ER model with foreign key dependencies correctly shown.
DDL Implementation	Less than competent model of the given scenario	Not all major entities and relationships of core model are correctly captured	Tables completely match RM diagram, and views correctly implemented.
Key Constraints	No key constraints captured	Most primary and foreign keys correctly defined	all necessary primary keys and foreign keys given including some useful ON DELETE and ON UPDATE clauses, where applicable.
Semantic constraints	No constraints other than key constraints captured	Some semantic integrity constraints such as CHECK, DEFAULT or NOT NULL were defined, but either incorrectly or incomplete	All necessary semantic integrity constraints for the model were given, including assertion and trigger definitions.
Example Data	No example data given or yielded multiple errors	All main tables populated with a consistent set of data	Database fully populated with a consistent set of data, with good examples of each significant constraint.

A Olympics ER Model (INFO2120/COMP9120)



B Olympics ER Model (INFO2820)


lives in
given
family
title
Name
member_id
code
Country
name
from
Member
books
when
start_date
start_time
Journey
nbooked
to
from
assigned
name
Place
longitude
gps
latitude
address
disjoint
IsA
Sport Venue
Accommodation
located in
type
name
Location
located in
disjoint
IsA
by
represents
member of
Athlete
Official
Staff
Vehicle
code
capacity
runs
role
name
Sport
for
result_type
Event
held at
result
participates
medal
name
disjoint
IsA
start
time
date
name
Team
result
participates
medal
Individual Event
Team Event