COMM1822

Term 2 2022

Introduction to Databases for Business Analytics

Week 10 Course Rev

Lecturer-in-Charge: Kam-Fung (Henry) Cheung
Email: kf.cheung@unsw.edu.au
Tutors: Theresa Tran
Liam Li Chen
Kathy Xu
PASS Leader: Srilekha Chandrashekara Kolaki

Copyright

- There are some file-sharing websites that specialise in buying and selling academic work to and from university students.

- If you upload your original wor other student downloads and presents it as their own eit **ight be found guilty of collusion — even years after graduatio**

- These file-sharing websites may also accept purchase of course materials, **such as copies of lecture slides and tutorial handouts. By law, the copyright on course materials, developed by UNSW staff in the course of their employment, belongs to UNSW. It constitutes copyright infringement, if not academic misconduct, to trade these materials.**

Acknowledgement of Country

UNSW Business School acknowledges the Bidjigal (Kensington campus) and Gadigal (City campus) the traditional custodians of the lands where each campus is located.

We acknowledge all Aboriginal and Torres St Islander Elders, past and present and thei communities who have shared and practiced their teachings over thousands of years including business practices.

We recognise Aboriginal and Torres Strait Islander people's ongoing leadership and contributions, including to business, education and industry.

UNSW Business School. (2022, May 7). *Acknowledgement of Country* [online video]. Retrieved from https://vimeo.com/369229957/d995d8087f

Agenda

- ❑ Housekeeping
- ❑ MyExperience
- ❑ Exam Advice
- ❑ Course Review
- ❑ Thank you and Q&A

Within-Group Peer Review

- Email the LiC (form on Moodle)

myExperience

Tell us about your experience. Shape the future of education at UNSW.

Visit Moodle to complete the myExperience survey or accessing myexperience.unsw.edu.au directly.

UNSW SYDNEY

Exam Advice

On different slides

Course Review

UNSW SYDNEY

Topics and Key Concepts

- Business Rules
- Conceptual Model
- Relation Model
- Normalization
- RDBMS/SQL
- Big Data

Database Design: Overview

The processes that we follow when designing a database for an organization:

1. We gather **business re** nization.
2. We develop a **concept** lling technique.
3. We convert the conceptual model (ER **elational model**, a set of relations.
4. We normalize the relation model (relations) to remove any anomalies and convert to **internal model**. (e.g., Oracle)
5. We physically implement this internal model in a **database** by creating a table for each normalized relations.

Conceptual Model

What is a conceptual model?

- A conceptual data model is a **representation of organizational data**.
- **ER modelling** is common modelling technique (use Chen's notation for exam).
- The result of ERM is an **ER model**: a detailed, logical representation of the data for an organization or for a business

What is shown in an ER model?

- An ER model is normally expressed as an **ER dia** a graphical representation of an ER model.
- The ER model is expressed in terms of **entities** in the business environment.
- The ER model also shows the **relationships** (associations) among those entities.
- The ER model also shows **attributes** of both the entities and their relationships.

Note: In the exam, solve all ER questions without supertype/subtype structures (unless explicitly asked for)!

Relational Model

What is a relation a model?

- A **relational model** represents data in relations.
- A **relation** can be thought of (and implemented as) a two-dimensional **table**.
- The name of a relation and [illegible] lation is called the **relational schema**.

What is shown in a relation model?

- A relational model shows **relations**, the name and structure of a two-dimensional table.
- A relation model shows **attributes**, the names of the columns of relations.
- A relation model shows **tuples**, the rows of relations.

Normalization

- **Normalization** is a **process for converting** complex data structures into simple data structures. This can be accomplished in stages.

- **What is the outcome**

  - **First normal form (1NF):** A moved, so there is a single value at the intersection of each row and column of the
  - **Second normal form (2NF):** Non-key attribute hole key for identification. (No partial dependencies exist.)
  - **Third normal form (3NF):** Non-key attributes do not depend on other non-key data elements (which is called transitive dependencies).
  - **Boyce-Codd Normal Form (BCNF):** If no non-key attribute determines part of the PK.
  - We usually normalize to **3NF**, which is an industry standard.

Relational Languages – Theory

What is the “theory” behind relational databases?

- **Relational algebra** and **relational calculus** are defined by Codd (1971) as the basis for relatio quages are not very user-friendly languages.

- **Relational algebra** operations can b in three main categories:
  - **Union, Intersection & Difference:** set operations.
  - **Selection & Projection:** choose/remove parts of a relation.
  - **Cartesian Product & Join:** operations combine the tuples of two relations.

Relational Languages – Practice

What is the practice of relational databases?

- The **RDBMS** – for example Oracle – provides data access via a query language.
- The RDBMS's query language – mo[illegible]y **SQL** – contains three components:
  - **Data Definition Language (DDL)** is used to specify the database schema or modify an existing one.
  - **Data Manipulation Language (DML)** is used to manipulate the data.
  - **Data Control Language (DCL)** is used for controlling the data.

SQL DML

ISO SQL standard uses the terms tables, columns, and rows.

- **SELECT** clause tell which attributes of the tuples matching the condition are produces as part o
- **FROM** clause gives th
- **WHERE** clause is a condition which t satisfy in order to match the query.


```
SLECT <columns>
FROM <tables>
[WHERE <conditions>]
```

Database Development

Database Development

- Information systems development overview
- Software development lifecycle (**SDLC**)
- Database developmen
- **Interaction** between S

Database Administrator (DBA)

- DBA vs. Data Administrator (DA)
- DBA tasks
- DBA ethics

DBLC & SDLC

Database Administration (DBA)

When a new DBMS is introduced to an organization, three important aspects have to be addressed.

- **Technological**
- **Managerial**
- **Cultural**

The person responsible for the contr tabase is called the **Database Administrator (DBA)**.

Database Administration (DBA)

- The **role** and **position** of DBA may **vary** in companies.
- Some of the larger corporations make a distinction between **DA (data administrator)** and **D** **trator)**.
- **DA:** A **high-level** funct or the **overall management of data resources** in tion, including maintaining corporate-wide definition ards.
- **DBA:** A **technical** function that is responsible for **physical database design** and for dealing with technical issues such as security enforcement, database performance and backup and recovery.

Big Data

What is Big Data?

- Buzz Word!
- Cannot fit into a USB flash drive
- A large and complex dataset
- Social media
- IoT streaming of data
- Capturing of Media

3Vs (Volume, Velocity and Variety) a

Big Data is classified into three types

- Structured
- Unstructured
- Semi-Structured

Views

A view contains no data of its own.

- A view is a logical table based on a table or another view

A view is stored as a S e data dictionary.

- The tables on which a ase tables.

The two main purposes of using a vie

- Reduce the complexity of some queries; and
- Restrict users' access to sensitive data

Exercise

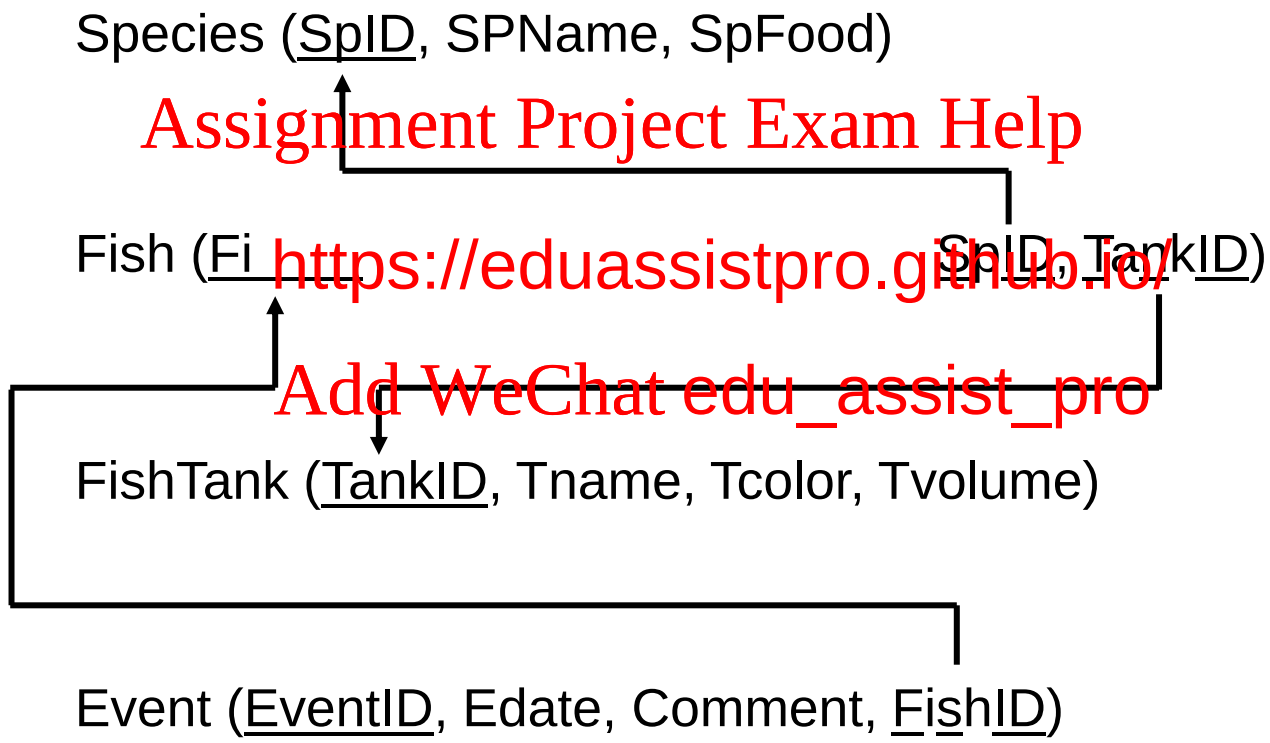
Exercise 1 – Aquariums

Each fish belongs to a species. Each fish also has a name, and you also want to register their colour and weight. You would like to store the specific food recommendation for each species. Naturally, the database should be able to tell you which fish is in which tank. Also, the names, volumes, and colou red. Finally, there are events involving your fish (birth, fights, and comment.

- ❑ Create the ER model.
- ❑ Create the relational model.

Solution to Exercise 1 - ER

Solution to Exercise 1 – Relational Model



Exercise 2 – SQL

Create SQLs for the following questions:

1. List all details about the fish stored in the fish table, sorted by fish name in ascending order.
2. What is the average vol
3. You have called your Cl ity events involving Sharky have occurred between 1 January 2012 2013? List the name of the fish and the number of events!
4. How can you find about Sharky's are of which type species? List the species name and food.
5. What colours are the fish tanks? Do not include duplicates.

Solution to Exercise 2 – SQL

1. List all details about the fish stored in the fish table, sorted by fish name in ascending order.


```
SELECT      *
FROM        FISH
ORDER BY    Fname;
```


2. What is the average volume of a fish tank?

```
SELECT      AVG (Tvolume) “Average Volume”
FROM        FISH_TANK;
```

Solution to Exercise 2 – SQL

3. You have called your Clownfish “Sharky”. How many events involving Sharky have occurred between 1 January 2012 and 1 April 2013? List the name of the fish and the number of events.


```
SELECT      Fname, COU                    vents”
FROM        FISH JOIN EVENT USING (F
WHERE       Fname = ‘Sharky’
AND         Edate BETWEEN DATE ‘2012-01-01’ AND ‘2013-04-01’
GROUP BY    Fname;
```

Solution to Exercise 2 – SQL

4. How can you find about Sharky's are of which type species? List the species name and food.


```
SELECT    Fnam
FROM      FISH
WHERE     Fname = 'Sharky';
```


5. What colours are the fish tanks? Do not include duplicates.

```
SELECT    DISTINCT (TColor)
FROM      FISHTANK
```

Species (SpID, SPName, SpFood)
Fish (FishID, Fname, Color, Weight, SpID, TankID)
FishTank (TankID, Tname, Tcolor, Tvolume)
Event (EventID, Edate, Comment, FishID)

Exercise 3 – Normalization

Scenario: Using the INVOICE table structure below:

a. Normalize the table to 1NF, and draw the dependency diagram. Assume that the table does not contain repeating groups and that an invoice-number references more than one product.

b. Normalize the table to its 2NF, and indicate functional dependencies.

c. Normalize the table to its 3NF, an

Solution to Exercise 3

Solution to Exercise 3

Exercise 4 – Functional Dependency

Prove or disprove the following property using **Armstrong's Axioms Primary Rules** only.

If $V \rightarrow W$, V $\{\}$ $\{Y, K\} \rightarrow \{T, W\}$.

If the property is held, you have to clearly state the rules used. If a property is NOT held, disprove it by giving a counter example (a set of sample instance).

Solution to Exercise 4

If $V \rightarrow W$, $V \subseteq Y$ and $\{Y, K\} \rightarrow T$, then $\{Y, K\} \rightarrow \{T, W\}$.

The property holds. The proof is as follows:

$(1)\{Y, K\} \rightarrow T$ (Given)

$(2)\{Y, K, V\} \rightarrow \{T, V\}$ (Augmentatio

$(3)V \rightarrow W$ (Given)

$(4)\{T, V\} \rightarrow \{T, W\}$ (Augmentation of T and (3))

$(5)\{Y, K, V\} \rightarrow \{T, W\}$ (Transitivity of (2) and (4))

$(6)V \subseteq Y$ (Given)

$(7)\{Y, K, V\} = \{Y, K\}$ (Trivial with (6))

$(8)\{Y, K\} \rightarrow \{T, W\}$ (Trivial with (5) and (7))

$\subseteq Y$, i.e., V is part of Y.
Thus, $\{Y, V\} = Y$.

Armstrong's Axioms Primary Rules

i. Inclusion (Reflexive) rule: If $Y \subseteq X$, then $X \rightarrow Y$.
ii. Augmentation rule: If $X \rightarrow Y$, then $\{W, X\} \rightarrow \{W, Y\}$.
iii. Transitivity rule: If $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$.

Questions

Source: metlifepetinsurance.com