

CS2102 Database Systems

Lecture 0 – Course Admin

Lectures

- **Lecturers**

- Christian Von Der WETH
 - Email `chris@comp.nus.edu.sg`
 - Office COM2-02-60
- Adi Yoga Sidi PRABAWA
 - Email `adi-yoga@comp.nus.edu.sg`
 - Office COM2-02-55
- HUANG Zhiyong
 - Email `huangzy@comp.nus.edu.sg`
 - Office AS6-05-11

- **Synchronous Online Lectures:** Tuesday, 12:00 – 14:00

- Recordings will be made available at LumiNUS → Multimedia

Tutorials

- **Synchronous Online Tutorial**

- Duration **1 Hour**
- Starts **Week 3**

- ***Not all tutorial questions will be discussed in class***

- **Tutorial Participation** (2 marks)

- Each student will be pre-assigned to prepare the answer for a question
- The answer files **must** be uploaded to the appropriate LumiNUS file folder (*for the tutorial group*) by **Monday, 09:00**
 - Each file must be named using the format **T<xx>_Q<yy>_<name>.pdf** where
 - xx
 - yy
 - name
 - For each discussed question, one of the assigned students will be randomly picked to lead discussion
 - Other students are expected to participate as well
 - Ask questions
 - Suggest alternative answers
 - etc

Prerequisites

- **Advanced Programming**

- CS1020(E)
- CS2020
- CS2030
- CS2040(C)

- **Discrete Mathematics**

- CS1232
- MA1100

Learning Objectives

The aim of this module is to introduce the fundamental concepts and techniques necessary for the understanding and **practice of design and implementation of database applications** and of the **management of data** with **relational database management systems**.

The module covers practical and theoretical aspects of design with entity-relationship model, theory of functional dependencies and normalization by decomposition in third and Boyce-Codd normal forms.

The module covers practical and theoretical aspects of programming with SQL data definition and manipulation sublanguages as well as relational algebra/calculus

- **Theory**

- Entity-relationship model
- Relational algebra
- Functional dependencies
- Normal forms

- **Practice**

- SQL (*sequel*)
 - Data definition language
 - Queries
 - Stored procedures
 - Triggers

Learning Objectives

- How to *design* an entity-relationship data model to capture the data requirements for an application and translate the conceptual data model to a relational database schema
- How to *refine* a relational database schema using schema decompositions to avoid anomalies
- How to use SQL to *define* relational schemas and write SQL *queries* on the schemas
- How to *reason* about the correctness of relational queries based on the concepts of formal query languages (relational algebra/calculus)
- How to *apply* knowledge of relational database systems to develop database applications

- **Design**

- Entity-relationship model
- SQL data definition language

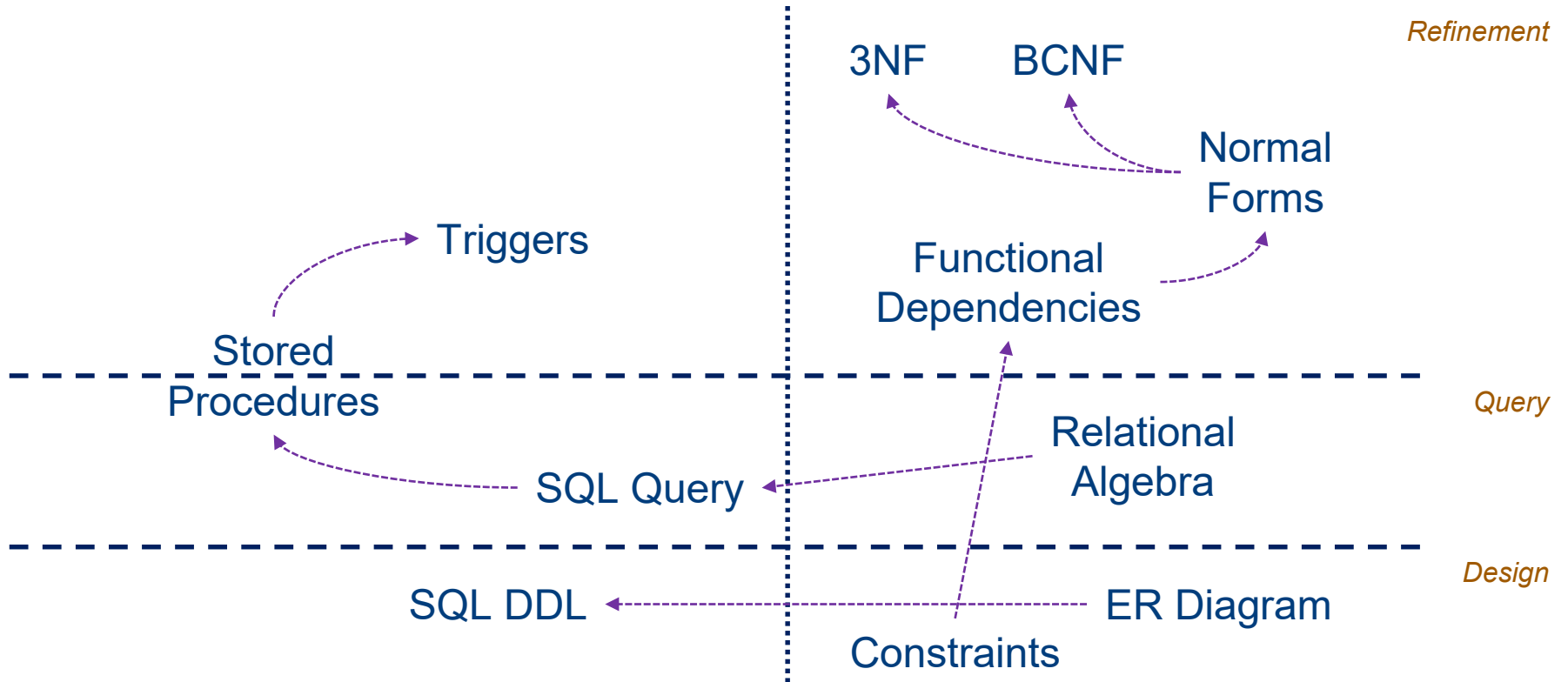
- **Query**

- Relational algebra
- SQL queries
 - Stored procedures
 - Triggers

- **Refinement**

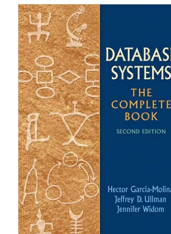
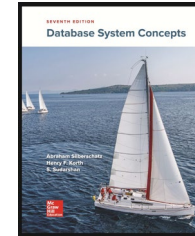
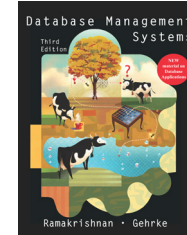
- Functional dependencies
- Normal forms

Roadmap



Reference Books

- Raghu Ramakrishnan & Johannes Gehrke
Database Management Systems
McGraw-Hill, Third Edition, 2003
(QA76.9 Dbm.Ra 2003)
<http://pages.cs.wisc.edu/~dbbook/>
- Avi Silberschatz, Hank Korth & S. Sudarshan
Database System Concepts
McGraw-Hill, Seventh Edition, 2019
(QA76.9 D3 Sil 2020)
<https://www.db-book.com/db7/index.html>
- Hector Garcia-Molina, Jeffrey Ullman & Jennifer Widom
Database System: The Complete Book
Prentice Hall, Second Edition, 2009
(QA76.9 Dbm.Gar 2009)
<http://infolab.stanford.edu/%7Eullman/dscb.html>



Workload and Assessment

- **Workload**

- Number of credits **4MC**
- Workload per week **10 hours**
 - 2 lecture hours
 - 1 tutorial hour
 - 4 assignment & project hours
 - 3 preparatory work hours

- **Tentative Module Assessment:**

Assessment Component	%
Tutorial Participation	2
Assignments	10
Group Project	28
Midterm Test	20
Final Test	40

Assignments

- **Individual Assignments**

- Total marks **10 Marks**

- **Tentative Assignment Deadlines**

#	Assessment Component	Due Date
1	SQL	Week 8
2	Schema Refinement	Week 12

Projects

- **Objective:** Develop a database application
 - **Software** PostgreSQL Database Server
 - **Languages** SQL and PL/pgSQL
- **Groups:** Teams of 4 members each
 - Team members could belong to different tutorial groups
 - If you are on ATAP, please declare to your other team members
 - If you are unable to form a team or belong to a team with fewer than 4 members, you may be assigned or *reassigned* to a random team
 - Registration will be opened by end of Week 2
 - Teams to be formed by end of Week 3

Projects

- **Tentative Project Deadlines:**

Task	Due Date
Project Team Registration	Week 3
Submission of ER Data Model Design	Week 6
Submission of Project Report and Code	Week 12
Project Evaluation	Week 13

Assessments

- **Midterm Test:** Week 8, Oct 5 (Tuesday), 12:00 – 14:00
- **Final Test:** Week 15, Nov 24 (Wednesday), 09:00 – 11:00
- Both test will be conducted online using **Exemplify**
 - Closed-book tests
 - Allowed to have one double-sided A4-sized cheatsheet
 - You may run postgresql
- ❖ Might change to face-to-face tests if conditions improved

Softwares

- **Exemplify**

- <https://wiki.nus.edu.sg/display/DA/Student>
- ***Will be used for both midterm and final tests***
- **System requirements:**
 - <https://wiki.nus.edu.sg/display/DA/Device+Minimum+System+Requirements>
- **CIT Briefing Slides:**
 - <https://wiki.nus.edu.sg/display/DA/CIT+Student+Briefing+Slides>
- **Installation:**
 - <https://wiki.nus.edu.sg/display/DA/Download+and+Install+Exemplify>
- **Service Help Desk:** <http://cit.nus.edu.sg/da-help/>
- **Emergency Email:** citbox25@nus.edu.sg

Softwares

- **PostgreSQL**

- <https://www.postgresql.org/>
- **Installation Guides:**
 - https://wiki.postgresql.org/wiki/Detailed_installation_guides



Course Policies

- **Students' Responsibilities**

- Attending lectures and tutorials
- Checking LumiNUS and/or emails for course-related announcements/updates
- For clarifications on lecture materials:
 - The best way is to post your questions on ***LumiNUS Forum***
 - Questions emailed to the lecturer may be posted to LumiNUS and answered there
 - You may also email the lecturer to arrange for consultation sessions

Course Policies

- **Zero-Tolerance for Plagiarism**

- Students will be reported to University for disciplinary action for plagiarism/cheating offence
 - Offenders will receive F grade for the module
- Resources:
 - <https://www.nus.edu.sg/celc/programmes/plagiarism.html>
 - <https://www.comp.nus.edu.sg/cug/plagiarism/>

Course Policies

- <https://www.comp.nus.edu.sg/cug/plagiarism/>

*All students share the responsibility for upholding the academic standards and reputation of the University. Academic honesty is a prerequisite condition in the pursuit and acquisition of knowledge. Academic dishonesty is any misrepresentation with the intent to deceive or failure to acknowledge the source or falsification of information or inaccuracy of statements or cheating at examinations/tests or inappropriate use of resources. There are many forms of academic dishonesty and plagiarism is one of them. **Plagiarism is generally defined as the practice of taking someone else's work or ideas and passing them off as one's own** (The New Oxford Dictionary of English). The University does not condone plagiarism.*

Course Policies

- <https://www.nus.edu.sg/registrar/administrative-policies-procedures/acceptance-record.html>
*Students should adopt this rule - **You have the obligation to make clear to the assessor which is your own work, and which is the work of others.** Otherwise, your assessor is entitled to assume that everything being presented for assessment is being presented as entirely your own work. This is a minimum standard.*

*A student may not knowingly intend to plagiarise, but that should not be used as an excuse for plagiarism. **Students should seek clarification from their instructors or supervisors if they are unsure whether or not they are plagiarising the work of another person.***