

Lecture 1: Course Overview

This lecture introduces the course ...

- What is a database, a DBMS, a *relational* database?
- What are we going to learn in this course?
- What activities will part of the course?
- How will you be assessed?

What is a database?

“A database is a collection of related data.”
[Elmasri and Navathe §1.1]

What is a database?

“A database is a collection of related data.”
[Elmasri and Navathe §1.1]

It records a lot of facts about some subject, such as

- customers, orders and product stock
- students, courses and results
- blog or discussion forum entries

What's a DBMS?

A database management system (DBMS) is a computer application for creating, maintaining and using databases.

What's a DBMS?

A database management system (DBMS) is a computer application for creating, maintaining and using databases.

A DBMS has several advantages over “raw” computer files

- data is *abstract*, independent of computer and programming concerns

[Elmasri and Navathe §1.3]

What's a DBMS?

A database management system (DBMS) is a computer application for creating, maintaining and using databases.

A DBMS has several advantages over “raw” computer files

- data is *abstract*, independent of computer and programming concerns
- different users can have customised *views* of the data

[Elmasri and Navathe §1.3]

What's a DBMS?

A database management system (DBMS) is a computer application for creating, maintaining and using databases.

A DBMS has several advantages over “raw” computer files

- data is *abstract*, independent of computer and programming concerns
- different users can have customised *views* of the data
- supports many simultaneous users, *transaction* processing

[Elmasri and Navathe §1.3]

What's a DBMS?

A database management system (DBMS) is a computer application for creating, maintaining and using databases.

A DBMS has several advantages over “raw” computer files

- data is *abstract*, independent of computer and programming concerns
- different users can have customised *views* of the data
- supports many simultaneous users, *transaction* processing
- database is *self-describing*

[Elmasri and Navathe §1.3]

What is a *relational* database?

[Elmasri and Navathe, Chapter 5]

- In a relational database, the data is organised as relations.

What is a *relational* database?

[Elmasri and Navathe, Chapter 5]

- In a relational database, the data is organised as relations.
- *relation* is a mathematical idea, but it's pretty much like a table, with rows and named columns.

What is a *relational* database?

[Elmasri and Navathe, Chapter 5]

- In a relational database, the data is organised as relations.
- *relation* is a mathematical idea, but it's pretty much like a table, with rows and named columns.
- The relational model is the most common *data model*.

What is a *relational* database?

[Elmasri and Navathe, Chapter 5]

- In a relational database, the data is organised as relations.
- *relation* is a mathematical idea, but it's pretty much like a table, with rows and named columns.
- The relational model is the most common *data model*.
- It is simple, yet mathematically solid.

What is a *relational* database?

[Elmasri and Navathe, Chapter 5]

- In a relational database, the data is organised as relations.
- *relation* is a mathematical idea, but it's pretty much like a table, with rows and named columns.
- The relational model is the most common *data model*.
- It is simple, yet mathematically solid.
- Ideas from maths and logic give us

What is a *relational* database?

[Elmasri and Navathe, Chapter 5]

- In a relational database, the data is organised as relations.
- *relation* is a mathematical idea, but it's pretty much like a table, with rows and named columns.
- The relational model is the most common *data model*.
- It is simple, yet mathematically solid.
- Ideas from maths and logic give us
 - a powerful but easy language: SQL (Structured Query Language)

What is a *relational* database?

[Elmasri and Navathe, Chapter 5]

- In a relational database, the data is organised as relations.
- *relation* is a mathematical idea, but it's pretty much like a table, with rows and named columns.
- The relational model is the most common *data model*.
- It is simple, yet mathematically solid.
- Ideas from maths and logic give us
 - a powerful but easy language: SQL (Structured Query Language)
 - design and evaluation techniques (normalisation)

What is a *relational* database?

[Elmasri and Navathe, Chapter 5]

- In a relational database, the data is organised as relations.
- *relation* is a mathematical idea, but it's pretty much like a table, with rows and named columns.
- The relational model is the most common *data model*.
- It is simple, yet mathematically solid.
- Ideas from maths and logic give us
 - a powerful but easy language: SQL (Structured Query Language)
 - design and evaluation techniques (normalisation)
 - theory for optimising query processing (relational algebra)

Overloaded Word Alert!

Be careful to distinguish between the various technical and ordinary usages of the words “relation” “relationship” “relational” “related” etc.

- the normal, non-technical sense, eg “a database is a collection of related data”

Overloaded Word Alert!

Be careful to distinguish between the various technical and ordinary usages of the words “relation” “relationship” “relational” “related” etc.

- the normal, non-technical sense, eg “a database is a collection of related data”
- the mathematical sense of relation: a subset of a cartesian product

Overloaded Word Alert!

Be careful to distinguish between the various technical and ordinary usages of the words “relation” “relationship” “relational” “related” etc.

- the normal, non-technical sense, eg “a database is a collection of related data”
- the mathematical sense of relation: a subset of a cartesian product
- relationships as a modelling concept in the entity-relationship technique

What are we going to learn in this course?

the *central* topics of the course are

- the relational data model

What are we going to learn in this course?

the *central* topics of the course are

- the relational data model
 - a bit of basic maths called “relational algebra” (don't panic!)

What are we going to learn in this course?

the *central* topics of the course are

- the relational data model
 - a bit of basic maths called “relational algebra” (don’t panic!)
 - various ways in which a database can be “right”:
intergrity, constraints, normalisation

What are we going to learn in this course?

the *central* topics of the course are

- the relational data model
 - a bit of basic maths called “relational algebra” (don’t panic!)
 - various ways in which a database can be “right”:
integrity, constraints, normalisation
- SQL
the standard language for creating maintaining and using relational databases

What are we going to learn in this course?

the *central* topics of the course are

- the relational data model
 - a bit of basic maths called “relational algebra” (don’t panic!)
 - various ways in which a database can be “right”:
integrity, constraints, normalisation
- SQL
the standard language for creating maintaining and using relational databases

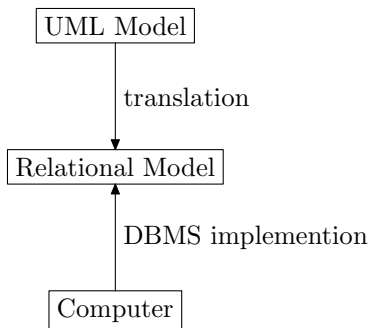
What are we going to learn in this course?

the *central* topics of the course are

- the relational data model
 - a bit of basic maths called “relational algebra” (don’t panic!)
 - various ways in which a database can be “right”:
integrity, constraints, normalisation
- SQL
the standard language for creating maintaining and using relational databases

but wait, there’s more!

Three Conceptual Levels



The relational model is a convenient half-way point between our everyday view of the world and its representation in computer hardware. We will also learn how to connect this half-way point with the real world and with computing systems.

Other Topics

So, we will also learn about

modelling finding an appropriate database structure by modelling the subject with UML

Other Topics

So, we will also learn about

modelling finding an appropriate database structure by modelling the subject with UML

normal forms and functional dependencies, for assessing design quality, and “bottom up” design

Other Topics

So, we will also learn about

modelling finding an appropriate database structure by modelling the subject with UML

normal forms and functional dependencies, for assessing design quality, and “bottom up” design

implementation a bit about what goes on “under the hood” of the DBMS: file access methods, query and transaction processing

What activities will be part of the course?

lectures 30 hours of listening to me talk!
printable slides will be on the web-site

What activities will be part of the course?

- lectures 30 hours of listening to me talk!
printable slides will be on the web-site
- reading and exercises

What activities will be part of the course?

lectures 30 hours of listening to me talk!
printable slides will be on the web-site

reading and exercises

tutorial/labs 5 * 2 hour practical hands-on sessions
each fortnight starting next week

What activities will be part of the course?

lectures 30 hours of listening to me talk!
printable slides will be on the web-site

reading and exercises

tutorial/labs 5 * 2 hour practical hands-on sessions
each fortnight starting next week

assignments 2 assignments

What activities will be part of the course?

lectures 30 hours of listening to me talk!
printable slides will be on the web-site

reading and exercises

tutorial/labs 5 * 2 hour practical hands-on sessions
each fortnight starting next week

assignments 2 assignments

exams mid-semester and final. Ironically, the exams make
a huge contribution to your learning!

Course Resources

textbook *Fundamentals of Database Systems*, 5th edition,
Ramez Elmasri and Shamkant B. Navathe,
Addison-Wesley, 2007.

Course Resources

textbook *Fundamentals of Database Systems*, 5th edition,
Ramez Elmasri and Shamkant B. Navathe,
Addison-Wesley, 2007.

forum on streams

Course Resources

- textbook** *Fundamentals of Database Systems*, 5th edition, Ramez Elmasri and Shamkant B. Navathe, Addison-Wesley, 2007.
- forum** on streams
- website** The course website
<http://cs.anu.edu.au/students/comp2400>.
Lecture notes, handouts assignments etc will be made available there.

Course Resources

- textbook** *Fundamentals of Database Systems*, 5th edition, Ramez Elmasri and Shamkant B. Navathe, Addison-Wesley, 2007.
- forum** on streams
- website** The course website
<http://cs.anu.edu.au/students/comp2400>.
Lecture notes, handouts assignments etc will be made available there.
- lecturer** You can contact me by email greg.okeefe@anu.edu.au *but use the forum for questions about course content*

How will you be assessed?

Assessment for the course will be as follows

10% tutorial participation and exercises

20% 2 assignments, 10% each

70% exams, best of

- 20% mid-semester + 50% final
- 70% final

To do this week

- Get a copy of the text: *Fundamentals of Database Systems* by Elmasri and Navathe, and read the first two chapters.

To do this week

- Get a copy of the text: *Fundamentals of Database Systems* by Elmasri and Navathe, and read the first two chapters.
- Find a suitable tutorial time on streams and join a group

To do this week

- Get a copy of the text: *Fundamentals of Database Systems* by Elmasri and Navathe, and read the first two chapters.
- Find a suitable tutorial time on streams and join a group
- Come along again on Thursday to start learning about the relational model and SQL!