LECTURE 1:
INTRODUCTION

COMP2004J: Databases and Information Systems

Dr. Ruihai Dong (ruihai.dong@ucd.ie)
UCD School of Computer Science
Beijing-Dublin International College

Module Timetable

- Lectures:
  - Tuesdays 18:00-19:35 in Room 301, Teaching Building 3
- Labs:
  - Wednesday 18:00-19:35 in Room 216, Teaching Building 4
  - Week 3 – Week 15
- Link: https://csmoodle.ucd.ie/moodle/course/view.php?id=649
- Enrolment Key : P2004J18

Module Format

- Lectures will run from week 1 to week 15 (inclusive).
- Labs will run from week 3 to week 15 (inclusive).

- If you have any questions about the module, please ask me or a teaching assistant (TA).

- After a lecture, during labs or by email:
  - My email address is ruihai.dong@ucd.ie
  - TA email address is linyi.yang@ucdconnect.ie

- Assessment is split between
  - 70% Final exam at end of term
  - 30% Practical work in Labs during semester

Practical Work and Assignments

- Weekly worksheets (**10%)** are **Individual** (not group)

- This means that you must submit **your own work** only.
  If you submit somebody else's work and pretend that you wrote it, this is called **plagiarism.**

- Major Assignment (**20%**) is **Group work** (group assignment).
- 3 members each team.

- Plagiarism is a **very serious** academic offence.

Plagiarism & UCD Computer Science

- **Plagiarism is a serious academic offence**
  - [Student Code, section 6.2] or [UCD Registry Plagiarism Policy] or [CS Plagiarism policy and procedures]
- Our staff and demonstrators are **proactive** in looking for possible plagiarism in all submitted work
- Suspected plagiarism is reported to the CS Plagiarism subcommittee for investigation
  - Usually includes an interview with student(s) involved
  - 1st offence: **usually** 0 or NG in the affected components
  - 2nd offence: referred to the **University disciplinary committee**
- Student who enables plagiarism is equally responsible

http://www.ucd.ie/students/guide/academicregs.html

http://libguides.ucd.ie/academicintegrity

Asking for help

- If you find things difficult, help is available.
  - There is a lecturer and many TAs in every lab.
  - You can ask a question after class.
  - You can email a TA with a question outside class.
  - You can email me with a question outside class.
  - You can get help from your classmates.
    - Getting help to understand something is not the same as copying a solution!

Module Content

- Reading List
- Fundamentals of Database Systems (6th Edition)
  - ISBN-10: 0136086209 | ISBN-13: 978-0136086208
- Database Systems: A Practical Approach to Design, Implementation, and Management (6th Edition)
  - ISBN-10: 0132943263 | ISBN-13: 978-0132943260
- All materials will be covered in lecture notes
- Practical work will using MySQL Relational DataBase Management System (RDBMS)
  - Free, cross platform, open source database management system

Topics

- Introduction to Databases
- Database Models
- Relational Database Model
- Structured Query Language (1974, 1981)
- Programmatic DB Use – Build you own first Information System
- Database Design (Entity-Relationship Model)
- Database Normalisation
- Object Relational Mapping

INTRODUCTION TO DATABASES AND INFORMATION SYSTEMS

Why Study Databases?

- A huge amount of information being stored.
- The College, Medical records, Employers, Companies, Government Agencies, etc.
- Managing that data is a **really big** task
- Data Base Management Systems (DBMS)
- Storing is easy, **managing** is the issue

What is a Database?

- Initial Definition: A database is a collection of related data
- This is a collection of related data with an implicit meaning and hence is a database
- For example, consider the names, telephone numbers, and addresses of the people you know

What is a Database?

- You may have recorded this data in an indexed address book (The contacts in your phone)

More Specific Properties of a Database

- A database represents some aspect of the real world
  - Changes to the real world are reflected in the database

David	1,000$
Ruihai	300$

More Specific Properties of a Database

- A database is a logically connected collection of data with some meaning
  - A random assortment of data cannot correctly be called a database

Place	County	Phone code	Approx. population
Basingstoke	Hampshire	01256	82913
Brighton	East Sussex	01273	155919
Carlisle	Cumbria	01228	103700
Huddersfield	Yorkshire	01484	146234
Luton	Bedfordshire	01582	203800
Nottingham	Nottinghamshire	0115	292400
Rhyll	Clwyd	01745	24889
Woking	Surrey	01483	62796

Zhang San 010-2895331
Ha Ha :D
Math 85%
An example of [illegible] database

- A database is designed, built, and populated with data for a specific purpose
  - It has an intended group of users and some applications in which these users are interested

Database Management Systems

- A database management system (DBMS) is a collection of programs that enables users to create and maintain a database

- The DBMS is a general-purpose software system that allows the processes of **defining**, **constructing**, **manipulating**, and **sharing** databases among various users and applications

DBMS Models

- Nowadays almost all databases are Relational
- This is a specific model and this module we will concentrate on this model
- Other models that have been used are
  - Hierarchical
  - Network
- Other models have been suggested in recent years
  - Object-Oriented

Managing Data

- Businesses have always maintained data
- For centuries this was done on paper

> “The palest ink is better than the best memory.” (Chinese proverb)
> 好记忆不如烂笔头

- Huge files that had to be manually searched in order to find information

Managing Data

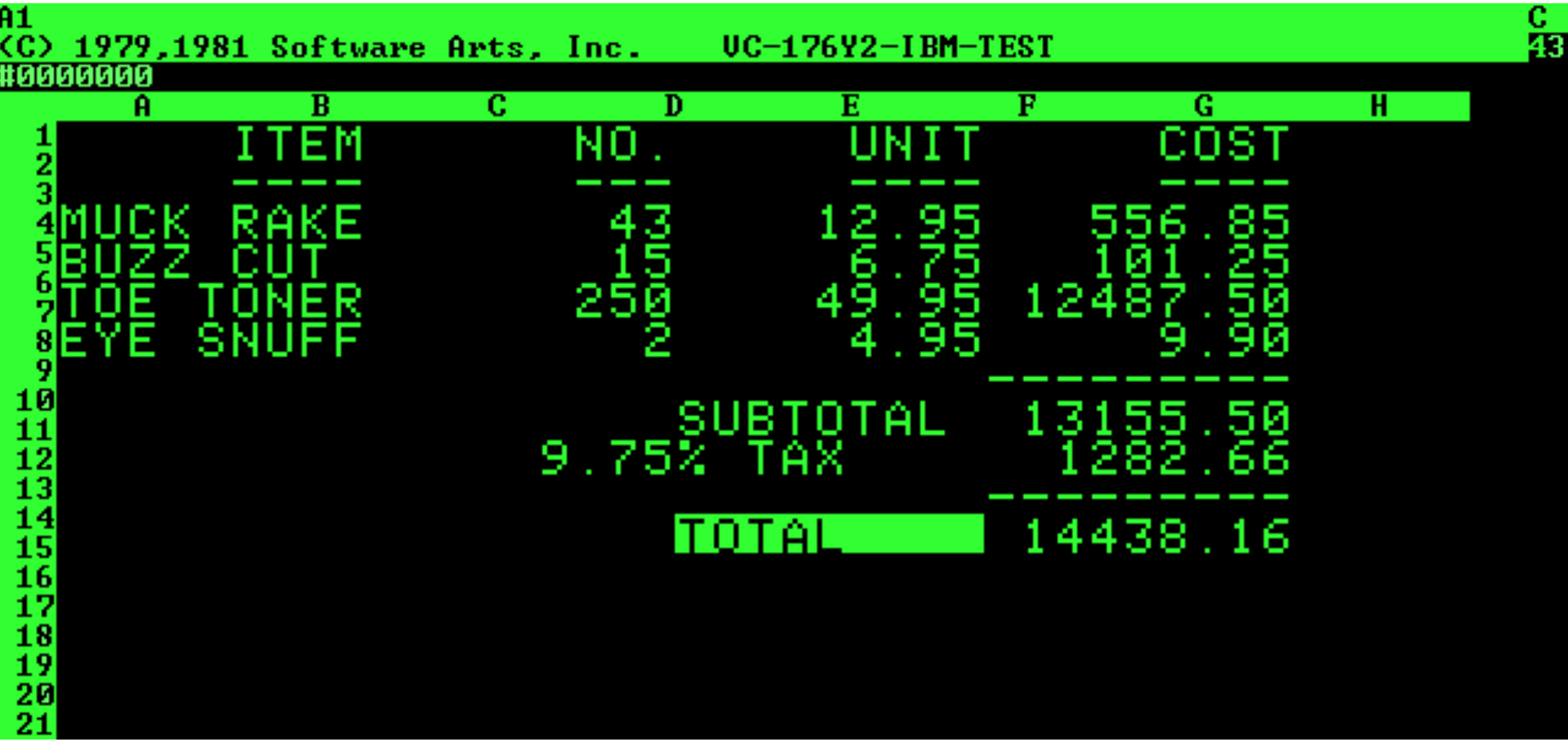
- The advent of computers allowed electronic storage
- First attempts stored electronic versions of documents
- The first major problem was how to store financial information

Spreadsheets

Spreadsheets

- Spreadsheets have been used by accountants for centuries.
- Very early electronic versions of spreadsheets were developed for mainframe computers in the 1960s.
- Modern computerised spreadsheets began with Daniel Bricklen and Bob Frankston.
- They developed VisiCalc in 1978, which became the basis for all electronic spreadsheets since.

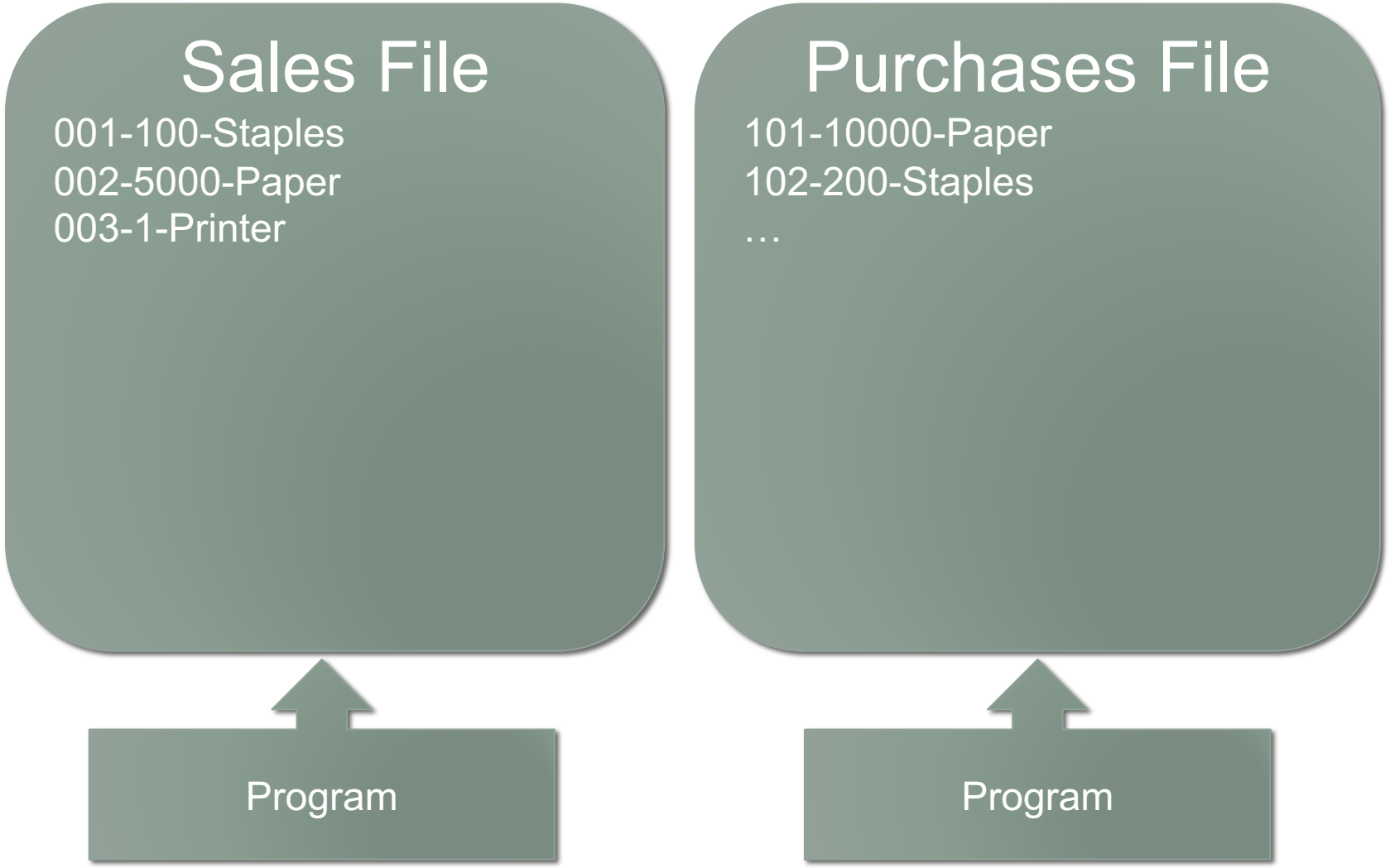
VisiCalc


A1
C
(C) 1979,1981 Software Arts, Inc.
VC-176Y2-IBM-TEST
43
#00000000
A B C D E F G H
1 ITEM NO. UNIT COST
2 ---- --- ---- ----
3
4 MUCK RAKE 43 12.95 556.85
5 BUZZ CUT 15 6.75 101.25
6 TOE TONER 250 49.95 12487.50
7 EYE SNUFF 2 4.95 9.90
8 ----------
9 SUBTOTAL 13155.50
10 9.75% TAX 1282.66
11 ----------
12 TOTAL 14438.16
13
14
15
16
17
18
19
20
21

Storing Data

- Storing data as simply electronic copies of existing documents is not sufficient.
- How do we search for information?
  - How many BSc II students do we have?
  - Is Ruihai Dong in the list?
  - What modules is he registered for?
- This type of query requires that we change how things are done.
- People then began to look at how to model data for electronic use.

File Based Database



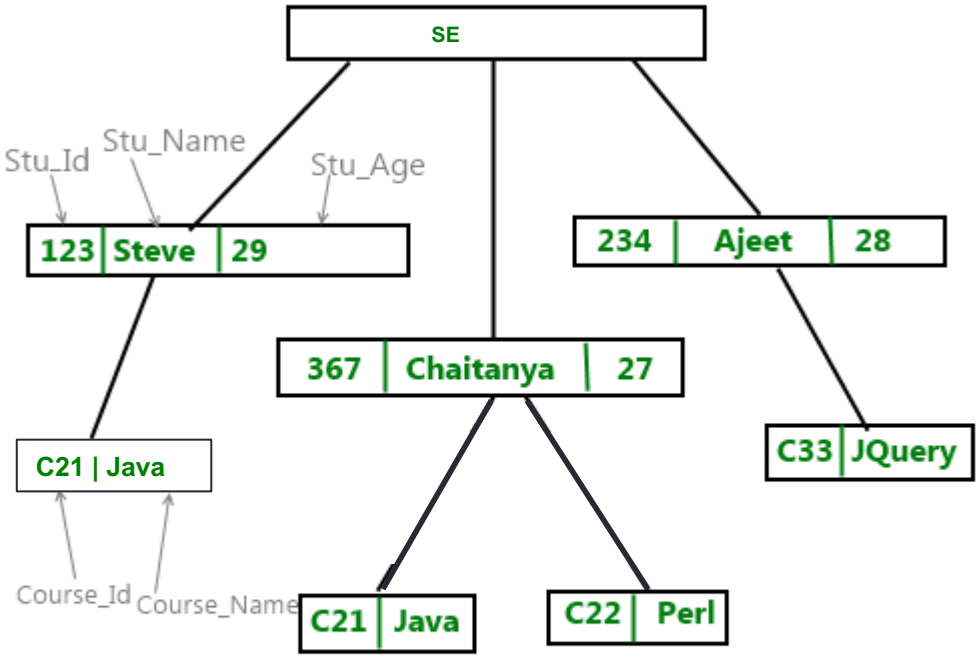
File Based Database Systems

- Advantages
  - More lightweight than a DBMS
  - Might be good enough for small data, for example a personal address book
- Disadvantages
  - No query language
  - No scalability
  - Hard to update schema and modify data
  - Recovery?

Hierarchical Database Model

- The oldest DBMS model is the hierarchical model (Information Management System from IBM)
- Developed in the 1960s to overcome the problems with file processing
- The model was not standardised
- It was based on a tree structure consisting of nodes, branches and roots
- It allowed for 1 to many relationships
- The best known implementation is IMS by IBM

Hierarchical Based Database



The layout is highly efficient for operations that “drill down,” so queries such as “how many courses does Chaitanya choose?” are easy to answer.

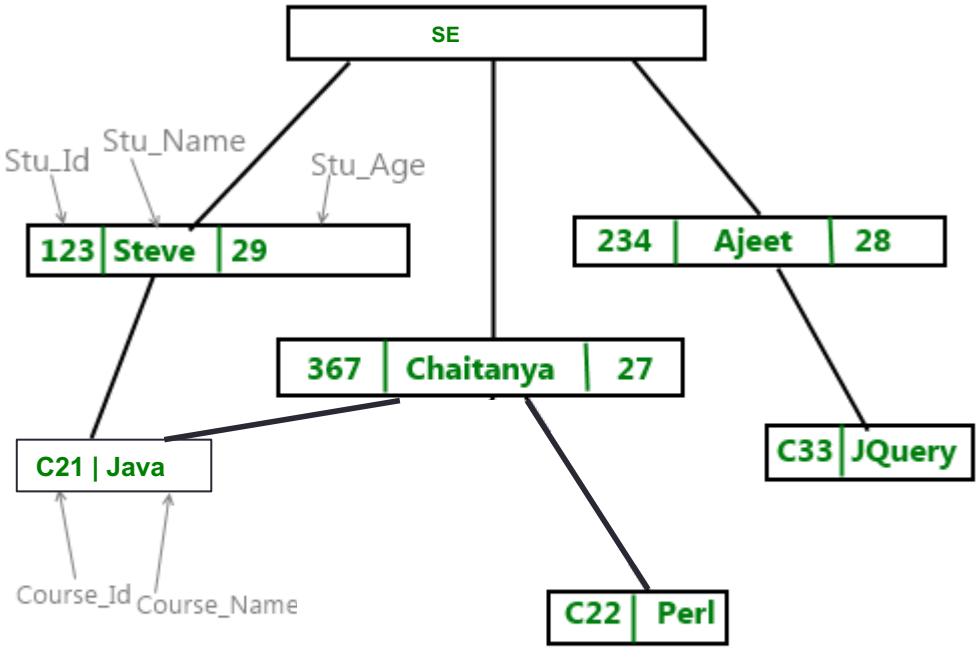
However, non-hierarchical queries are difficult to express and very costly to evaluate: queries such as “how many students do study Java?”

The hierarchical model is also prone to redundancy.

Network Database System

- The network model was developed by committee
- The Database Task Group (DBTG) of CODASYL (COnference on DAta SYstems Languages) developed the model so that it could be standardised
- CODASYL are the people who developed the COBOL programming language and much of the language of the network model looks like COBOL
- The model allows the implementation of many-to-many (M:M) relationships, which ultimately get translated into one-to-many (1:M).

Network Database System



The network model generalizes the hierarchical model to represent relationships as directed graphs rather than trees. Doing so improves efficiency significantly: an application can easily add connections between records in order to accelerate important queries.

This model also has some problems, the main problem being that you need to be a database expert to use this database successfully. It is very difficult for the general public to use.

Relational Database Model

- Formulated by Edgar Codd of IBM in 1970.
- Commercial RDBMS in 80s.
- “Codd's 12 Rules” (actually 13) that all RDBMSs should follow.
- Most widely used Model at present
  - Access, Oracle, MySQL, MariaDB, MS SQL Server, DB2, Sybase ASE, PostgresSQL etc.

Relational Concepts

- Data is represented as a collection of **relations**

- Each relation is **table** of values
- Each table consists of **rows** and **columns**

Student

StudentNo	FirstName	LastName	YearofEntry	Major
1312345	Lina	Xu	2013	Finance
1318999	Wan	Wan	2013	Software Engineering
1218985	Ning	Cao	2012	Internet of Things

Relational Concepts

- Each **row** represents an **entity** or **record**
- Rows are unordered

StudentNo	FirstName	LastName	YearofEntry	Major
1312345	Lina	Xu	2013	Finance
1318999	Wan	Wan	2013	Software Engineering
1218985	Ning	Cao	2012	Internet of Things

Relational Concepts

- No duplicate rows are allowed

- Each relation has a **primary key**, the value of which uniquely identifies the **record/entity**

StudentNo	FirstName	LastName	YearofEntry	Major
1312345	Lina	Xu	2013	Finance
1318999	Wan	Wan	2013	Software Engineering
1218985	Ning	Cao	2012	Internet of Things

1218985	**Ning**	**Ca**	**2012**	**Internet of Things**
1218986	**Ning**	**Cao**	**2012**	**Internet of Things**

Relational Concepts

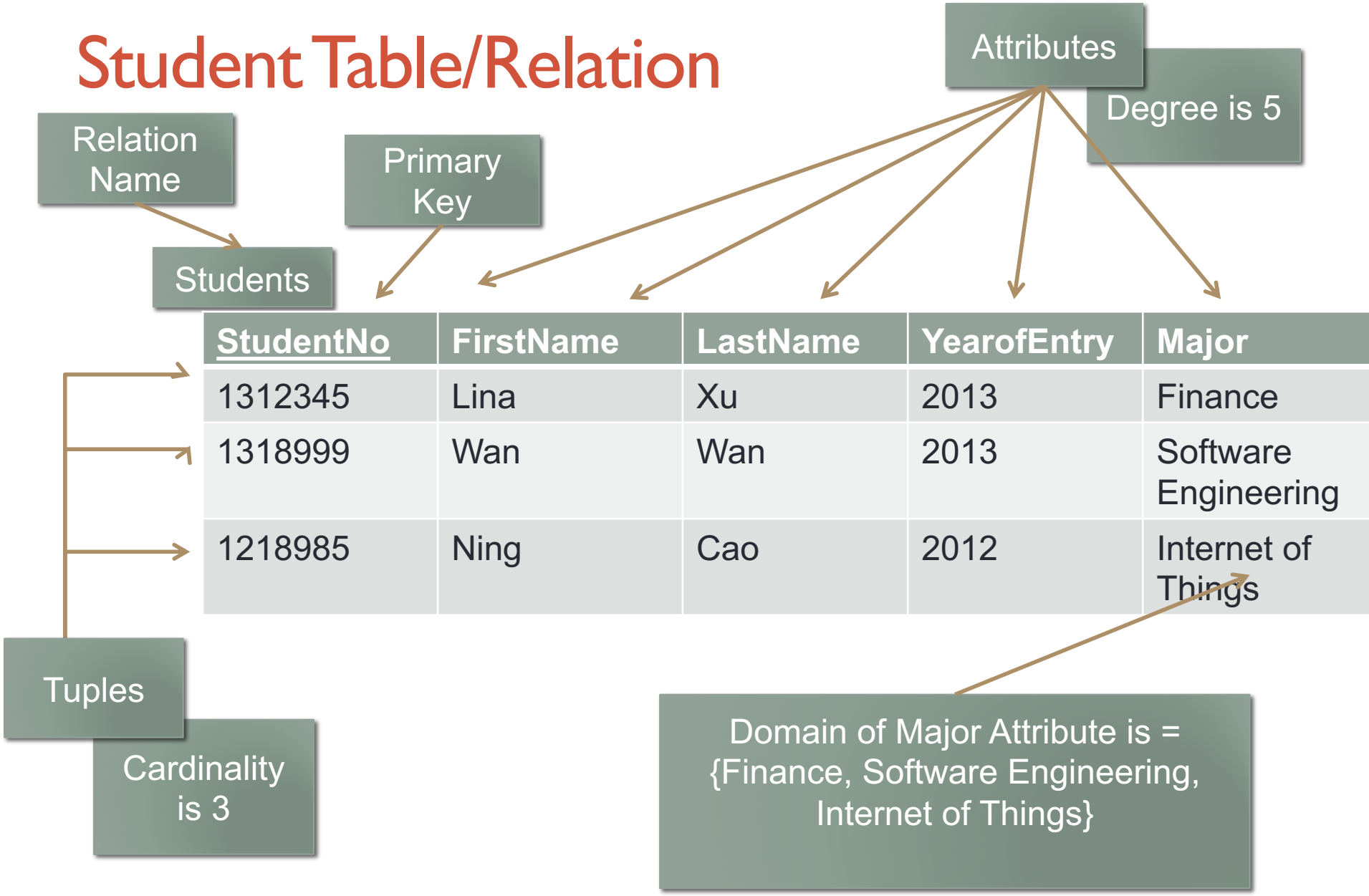
- Each column represents an **attribute**
- Table name and column name are used to help interpret the values

StudentNo	StudentName	Major	YearofEntry
1312345	Xu Lina	Finance	2013
1318999	Jie Wan	Software Engineering	2013
1218985	Cao Ning	Internet of Things	2012

Database Terminology

- **Relation** is a mathematical term for a **table**
- **Row** is called a **Tuple**
- **Column** is called an **Attribute**
- **Domain** is used to describe the types of values that can appear in a column
- **Degree** is the number of attributes
- **Cardinality** – the number of tuples/rows in a relation
- **Atomic Value** – precisely one value at each row intersection
- **Null Value** – Missing, not known or irrelevant data (not the same as zero or blank)

Student Table/Relation



Advantages of Database Approach

- Data can be shared
- Redundancy can be reduced
- Integrity can be maintained
- Security can be enforced
- Conflicting requirements can be balanced
- Standards can be enforced