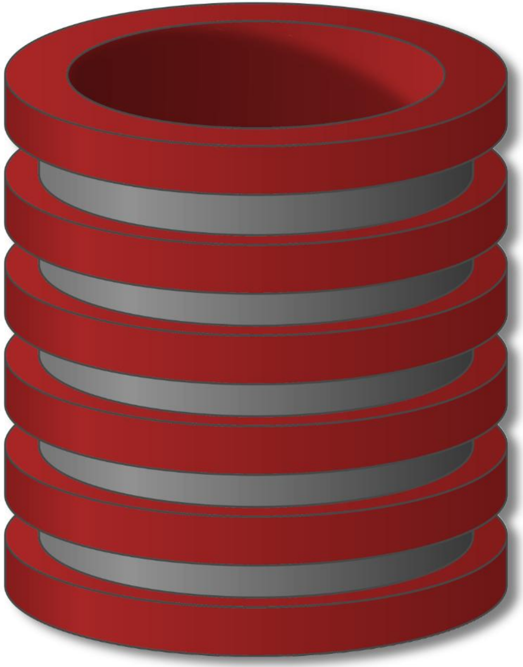

Relational Databases

The Relational Model

Jennifer Widom

The Relational Model

- Used by all major commercial database systems
- Very simple model
- Query with high-level languages: simple yet expressive
- Efficient implementations

The Relational Model

Schema = structural description of relations in database

Instance = actual contents at given point in time

Student

ID	name	GPA	photo
123	Amy	3.9	☺
234	Bob	3.4	NULL
345	Craig	NULL	😐
	⋮		

College

name	state	enr
Stanford	CA	15,000
Berkeley	CA	36,000
MIT	MA	10,000
	⋮	

The Relational Model

Database = set of named **relations** (or **tables**)

Each relation has a set of named **attributes** (or **columns**)

Each **tuple** (or **row**) has a value for each attribute

Each attribute has a **type** (or **domain**)

Student

ID	name	GPA	photo
123	Amy	3.9	☺
234	Bob	3.4	NULL
345	Craig	NULL	😐
	⋮		

College

name	state	enr
Stanford	CA	15,000
Berkeley	CA	36,000
MIT	MA	10,000
	⋮	

The Relational Model

Schema – structural description of relations in database

Instance – actual contents at given point in time

Student

ID	name	GPA	photo
123	Amy	3.9	☺
234	Bob	3.4	NULL
345	Craig	NULL	😐
	⋮		

College

name	state	enr
Stanford	CA	15,000
Berkeley	CA	36,000
MIT	MA	10,000
	⋮	

The Relational Model

NULL – special value for “unknown” or “undefined”

Student

ID	name	GPA	photo
123	Amy	3.9	☺
234	Bob	3.4	NULL
345	Craig	NULL	😐
	⋮		

College

name	state	enr
Stanford	CA	15,000
Berkeley	CA	36,000
MIT	MA	10,000
	⋮	

Jennifer Widom

The Relational Model

Key – attribute whose value is unique in each tuple

Or set of attributes whose combined values are unique

Student

ID	name	GPA	photo
123	Amy	3.9	☺
234	Bob	3.4	NULL
345	Craig	NULL	😐
	⋮		

College

name	state	enr
Stanford	CA	15,000
Berkeley	CA	36,000
MIT	MA	10,000
	⋮	

Jennifer Widom

Creating relations (tables) in SQL

```
Create Table Student(ID, name, GPA, photo)

Create Table College
  (name string, state char(2), enrollment integer)
```

The Relational Model

- Used by all major commercial database systems
- Very simple model
- Query with high-level languages: simple yet expressive
- Efficient implementations