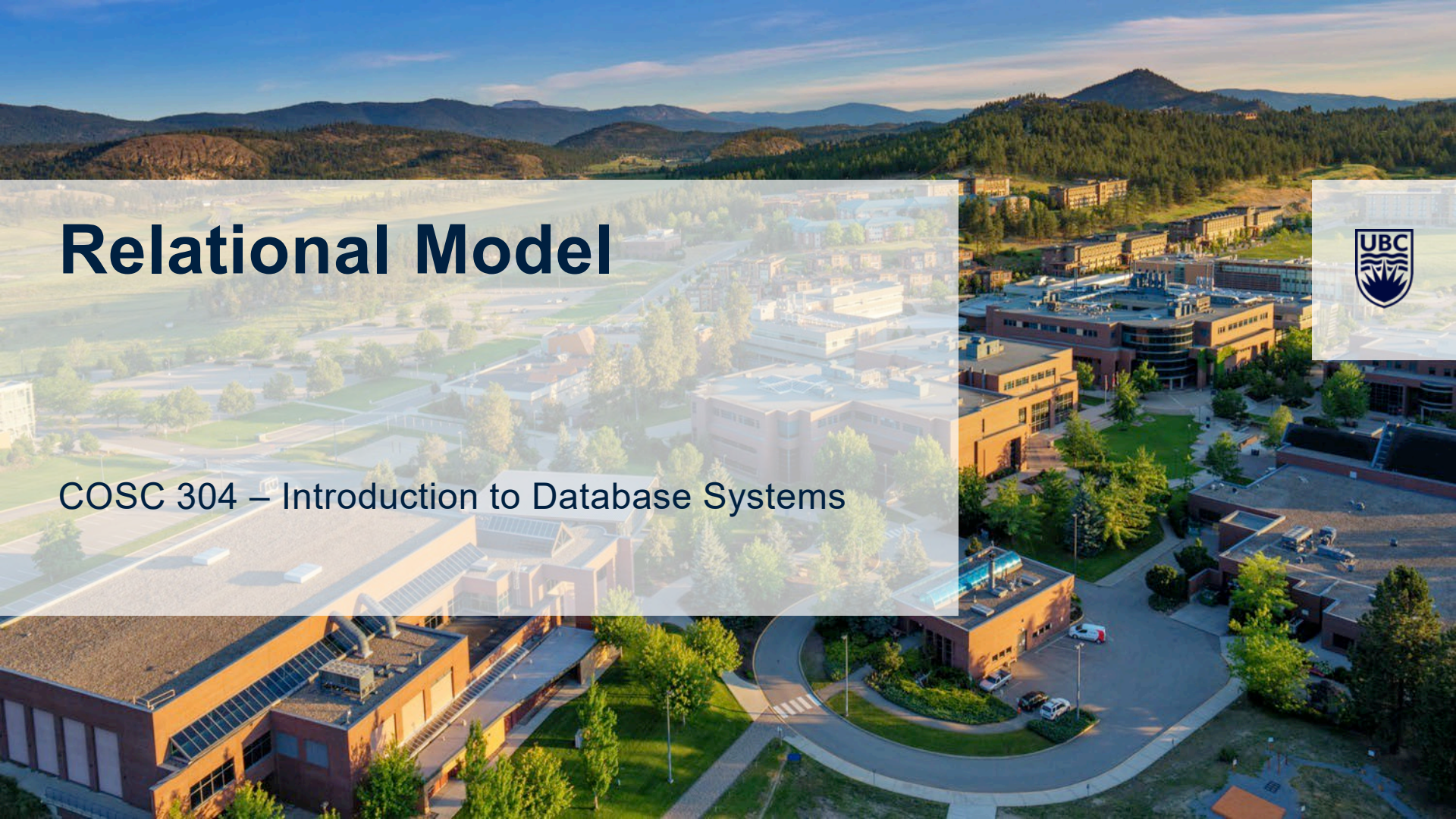


Relational Model

COSC 304 – Introduction to Database Systems

Relational Model History

The relational model was proposed by E. F. Codd in 1970. Commercial implementations appeared in the late 1970s and early 1980s.

One of the first relational database systems, System R, developed at IBM led to several important breakthroughs:

- the first version of SQL
- various commercial products such as Oracle and DB2
- extensive research on concurrency control, transaction management, and query processing and optimization

Currently, the relational model is the foundation of the majority of commercial database systems.

The Relational Model: Terminology

The ***relational model*** organizes data into tables called relations.

A ***relation*** is a table with columns and rows.

An ***attribute*** is a named column of a relation.

A ***tuple*** is a row of a relation.

A ***domain*** is a set of allowable values for one or more attributes.

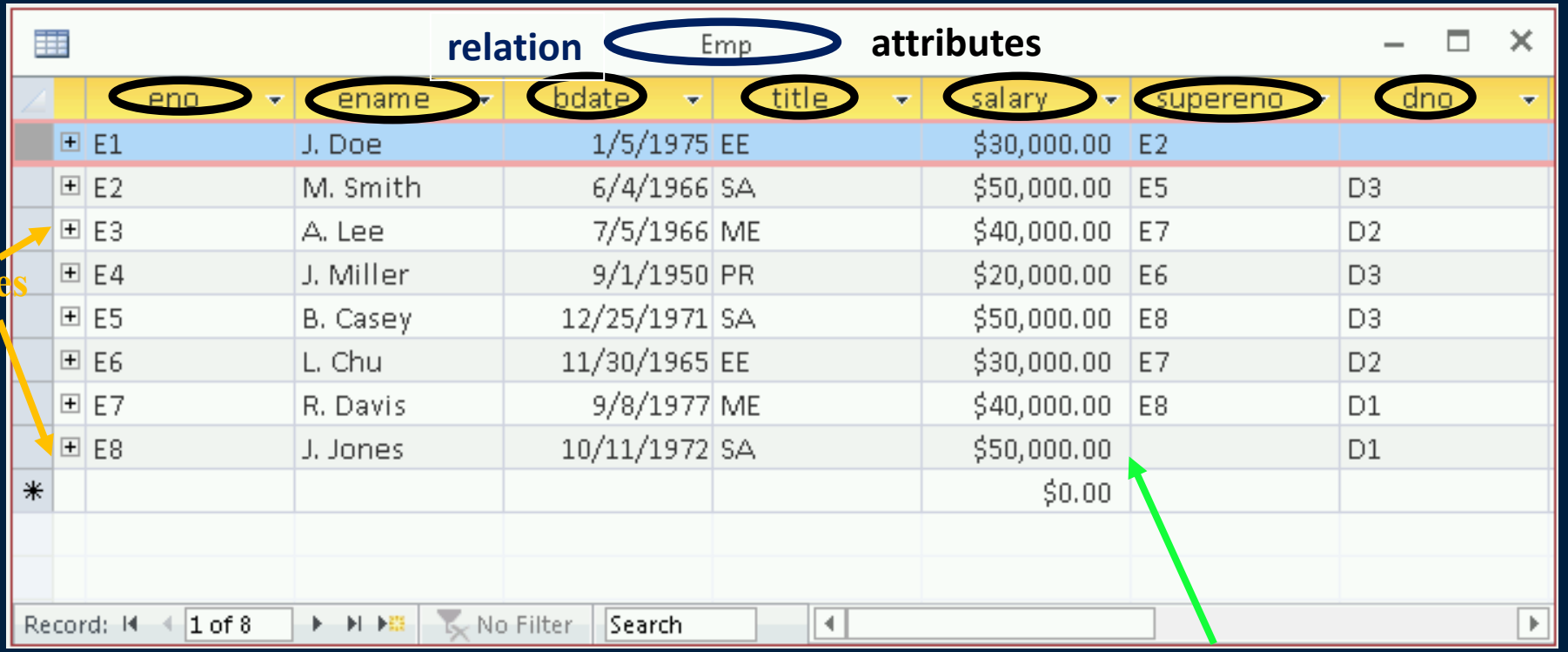
The ***degree*** of a relation is the number of attributes it contains.

The ***cardinality*** of a relation is the number of tuples it contains.

The ***intension*** is the structure of the relation including its domains.

The ***extension*** is the set of tuples currently in the relation. All data in the DB

Relation Example



eno	ename	bdate	title	salary	supereno	dno
E1	J. Doe	1/5/1975	EE	$30,000.00	E2	
E2	M. Smith	6/4/1966	SA	$50,000.00	E5	D3
E3	A. Lee	7/5/1966	ME	$40,000.00	E7	D2
E4	J. Miller	9/1/1950	PR	$20,000.00	E6	D3
E5	B. Casey	12/25/1971	SA	$50,000.00	E8	D3
E6	L. Chu	11/30/1965	EE	$30,000.00	E7	D2
E7	R. Davis	9/8/1977	ME	$40,000.00	E8	D1
E8	J. Jones	10/11/1972	SA	$50,000.00		D1
*				$0.00		

Degree =7

Cardinality = 8

Domain of salary is *currency*

Relational Model Formal Definition

The relational model is formally defined in terms of sets and set operations.

A ***relation schema*** ***R*** (A_1, A_2, …, A_n) has each attribute A_i with a name and a domain $dom(A_i)$.

A ***relation instance*** denoted $r(R)$ is a set of n-tuples $<d_1, d_2, ..., d_n>$ where each d_i is an element of $dom(A_i)$ or is **null**.

- The relation instance is the *extension* of the relation.
- A value of **null** represents a missing or unknown value.

Example: Product (id, name, supplierId, categoryId, price)

- ***R*** = Product (relation name)
- Set ***A*** = {id, name, supplierId, categoryId, price}
- dom(price) is set of all possible positive currency values
- dom(name) is set of all possible strings that represent people's names

Relation Practice Questions

WorksOn

eno	pno	resp	hours
E1	P1	Manager	12
E2	P1	Analyst	24
E2	P2	Analyst	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36

Record: 1 of 9 No Filter Search

1) What is the name of the relation?
2) What is the cardinality of the relation?
3) What is the degree of the relation?
4) What is the domain of `resp`? What is the domain of `hours`?
5) What is larger the size of the intension or extension?
6) Is a relation's cardinality always bigger than its degree?

Database Definition Matching Question

Question: Given the three definitions, select the ordering that contains their related definitions.

1) relation

2) tuple

3) attribute

A) column, row, table

B) row, column, table

C) table, row, column

D) table, column, row

Cardinality and Degree Question

Question: A database table has 5 rows and 10 columns. Select ***one*** true statement.

A) The table's degree is 50.

B) The table's cardinality is 5.

C) The table's degree is 5.

D) The table's cardinality is 10.

Relation Properties

1) No two relations have the same name.
2) Each attribute of a relation has a distinct name.
3) Each tuple is distinct. There are no duplicate tuples. Only in RM not SQL
4) The order of attributes is not important.
5) The order of tuples has no significance.

Relational Keys

Keys are used to uniquely identify a tuple in a relation.

- Note that keys apply to the schema not to the data. That is, looking at the current data cannot tell you for sure if the set of attributes is a key.

A ***superkey*** is a set of attributes that uniquely identifies a tuple in a relation.

A (***candidate***) ***key*** is a *minimal* set of attributes that uniquely identifies a tuple in a relation.

- There may be more than 1 candidate key for a relation with different # of attributes.

A ***primary key*** is the candidate key designated as the distinguishing key of a relation.

A ***foreign key*** is a set of attributes in one relation referring to the primary key of a relation. 引用完整性

- Foreign keys enforce referential integrity. Note: A FK may refer to its own relation.

Keys and Superkeys Question

Question: **True or false:** A key is always a superkey.

A) true

B) false

Keys and Superkeys Question (2)

Question: **True or false:** It is possible to have more than one key for a table and the keys may have different numbers of attributes.

A) true

B) false

Keys and Superkeys Question (3)

Question: **True or false:** It is possible to always determine if a field is a key by looking at the data in the table.

A) true

B) false

Example Relational Data Questions

Keys are underlined

Emp Relation

eno	ename	title	salary
E1	J. Doe	EE	30000
E2	M. Smith	SA	50000
E3	A. Lee	ME	40000
E4	J. Miller	PR	20000
E5	B. Casey	SA	50000
E6	L. Chu	EE	30000
E7	R. Davis	ME	40000
E8	J. Jones	SA	50000

WorksOn Relation

eno	pno	resp	hours
E1	P1	Manager	12
E2	P1	Analyst	24
E2	P2	Analyst	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36
E7	P5	Engineer	23
E8	P3	Manager	40

Proj Relation

pno	pname	budget
P1	Instruments	150000
P2	DB Develop	135000
P3	CAD/CAM	250000
P4	Maintenance	310000
P5	CAD/CAM	500000

Questions:

1) Is *ename* a key for *emp*?
2) Is *eno* (by itself) a key for *WorksOn*?
3) List all the superkeys for *WorksOn*.

Keys Practice Question

Consider a relation storing driver information including:

- SSN, name, license plate number and state (unique together)

Driver Relation

SSN	name	LicNum	LicState
123-45-6789	S. Smith	123-456	IA
111-11-1111	A. Lee	123-456	NY
222-22-2222	J. Miller	555-111	MT
333-33-3333	B. Casey	678-123	OH
444-44-4444	A. Adler	456-345	IA

Questions:

1) List the candidate keys for the relation.
2) Pick a primary key for the relation.
3) Is *name* a candidate key for *Driver*?
4) List all the superkeys for *Driver*.

Assumptions:

1) A person has only one license plate.
2) A license plate uniquely identifies a person.

Relational Integrity

Integrity rules are used to insure the data is accurate.

Constraints are rules or restrictions that apply to the database and limit the data values it may store.

Types of constraints:

- ***Domain constraint*** - Every value for an attribute must be an element of the attribute's domain or be `null`.
  - `null` represents a value that is currently unknown or not applicable.
  - `null` is not the same as zero or an empty string.
- ***Entity integrity constraint*** - No attribute of a primary key can be null.
- ***Referential integrity constraint*** - If a foreign key exists in a relation, then the foreign key value must match a primary key value of a tuple in the referenced relation or be null.

Foreign Keys Example

Emp Relation

eno	ename	title	salary
E1	J. Doe	EE	30000
E2	M. Smith	SA	50000
E3	A. Lee	ME	40000
E4	J. Miller	PR	20000
E5	B. Casey	SA	50000
E6	L. Chu	EE	30000
E7	R. Davis	ME	40000
E8	J. Jones	SA	50000

WorksOn.eno is FK to Emp.eno

WorksOn.pno is FK to Proj.pno

WorksOn Relation

eno	pno	resp	hours
E1	P1	Manager	12
E2	P1	Analyst	24
E2	P2	Analyst	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36
E7	P5	Engineer	23
E8	P3	Manager	40

Proj Relation

pno	pname	budget
P1	Instruments	150000
P2	DB Develop	135000
P3	CAD/CAM	250000
P4	Maintenance	310000
P5	CAD/CAM	500000

Foreign Keys Example (2)

Proj Relation

pno	pname	budget	dno
P1	Instruments	150000	D1
P2	DB Develop	135000	D2
P3	CAD/CAM	250000	D3
P4	Maintenance	310000	null
P5	CAD/CAM	500000	D1

Proj.dno is FK to Dept.dno

Department Relation

dno	dname
D1	Management
D2	Consulting
D3	Accounting
D4	Development

Integrity Constraints Question

Question: What constraint says that a primary key field cannot be null?

A) domain constraint

B) referential integrity constraint

C) entity integrity constraint

Entity Integrity Constraint Question

Question: A primary key has three fields. Only one field is `null`. Is the entity integrity constraint violated?

A) Yes

B) No

Referential Integrity Constraint Question

Question: A foreign key has a `null` value in the table that contains the foreign key fields. Is the referential integrity constraint violated?

A) Yes

B) No

Integrity Questions

Emp Relation

eno	ename	title	salary
E1	J. Doe	EE	AS
E2	null	SA	50000
E3	A. Lee	null	40000
E4	J. Miller	PR	20000
E5	B. Casey	SA	50000
null	L. Chu	EE	30000
E7	R. Davis	ME	null
E8	J. Jones	SA	50000

Proj Relation

pno	pname	budget
P1	Instruments	150000
P2	DB Develop	135000
P3	CAD/CAM	250000
P4	Maintenance	310000
P5	null	null

WorksOn Relation

eno	pno	resp	hours
E1	P0	null	12
E2	P1	Analyst	null
null	P2	Analyst	6
E3	P3	Consultant	10
E9	P4	Engineer	48
E4	P2	Programmer	18
E5	null	Manager	24
E6	P4	Manager	48
E7	P6	Engineer	36
E7	P4	Engineer	23
null	null	Manager	40

Question: How many rows have violations of integrity constraints? Note: salary, budget, hours are number fields.

A) 8 B) 9 C) 10 D) 11 E) 12

Conclusion

The ***relational model*** represents data as relations which are sets of tuples. Each relational schema is a set of attributes with domains.

- A relation is a table with columns and rows. An attribute is a named column. A tuple is a row.
- Degree is the number of attributes, and cardinality is the number of tuples.

Keys are used to uniquely identify tuples in relations.

- Superkey is any set of attributes that identifies a tuple in a relational. A candidate key is a minimal set of attributes that identifies a tuple.

The relational model has ***constraints*** to guarantee data integrity including: domain, entity integrity and referential integrity constraints.

Objectives

- Define: relation, attribute, tuple, domain, degree, cardinality, intension, extension, relation schema, relation instance, null
- List the properties of relations.
- Define: superkey, key, candidate key, primary key, foreign key
- Define: integrity, constraints, domain constraint, entity integrity constraint, referential integrity constraint
- Given a relation be able to:
  - identify its cardinality, degree, domains, keys, and superkeys
  - determine if constraints are being violated


UBC
THE UNIVERSITY OF BRITISH COLUMBIA