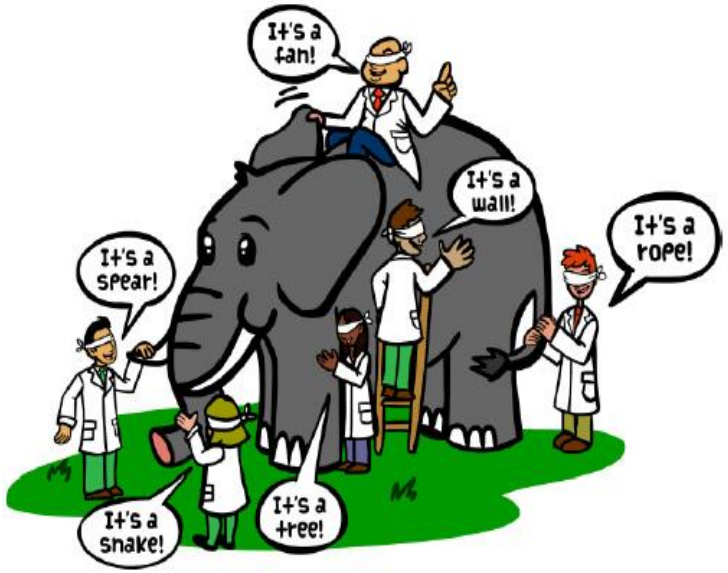
CPSC 5021: Database Systems

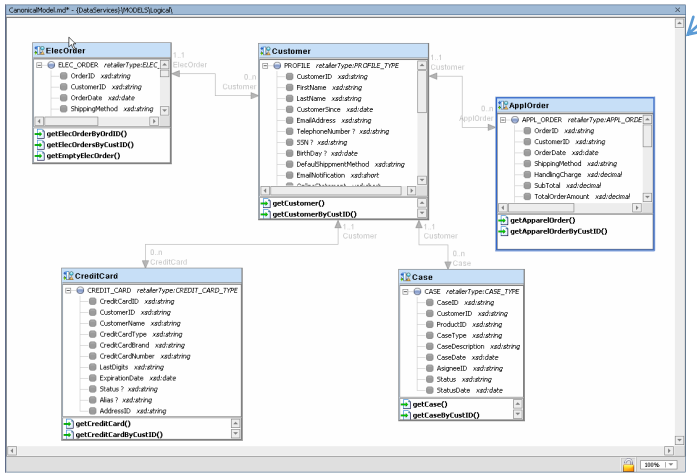
Data Modeling

Lin Li

Data Model

- Why do we need data model?





data model

programmers

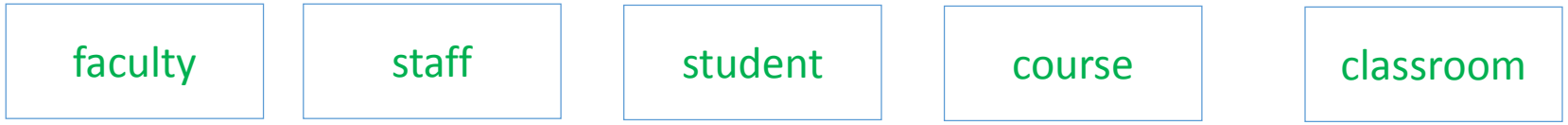
users

designers

Designers, programmers, and users see data in different ways.

Data Model Basic Building Blocks

- Entity: Some unit of data that can be classified.



- Attribute: a characteristic of an entity.

- Relationship: describes an association among entities

Data Model Basic Building Blocks

- Relationship

- One-to-many (1:M) relationship (e.g. building vs. classroom)
- Many-to-many (M:N) relationship (e.g. student vs. course)
- One-to-one (1:1) relationship (e.g. *chair* prof. vs. department)

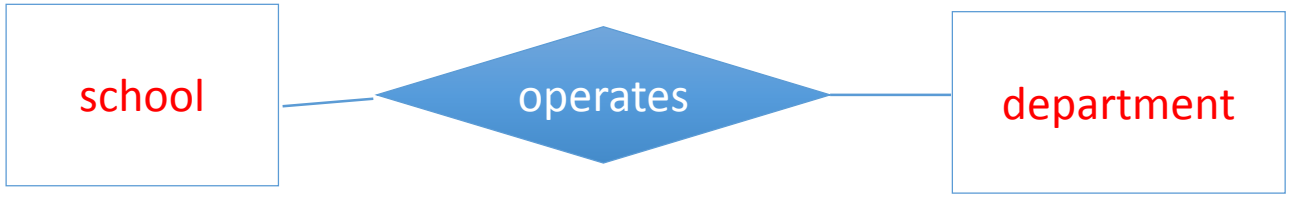
- Constraint: a restriction placed on the data.

e.g. (1) A professor can teach at most 4 classes

(2) A student's age must be a valid positive integer

How to Do Data Modeling?

- Example: Tiny college is divided into several schools: business, arts and sciences, education, and applied sciences. Each school comprises several departments. For example, the school of business has an accounting department, a marketing department, etc.



1:M

Data Modeling Exercises

- Each department may offer courses.
- A department may offer several classes of the same course.
- Each department should have one or more professors assigned to it. One and only one of those professors chairs the department.
- Each professor may teach up to 4 classes.
- A student may enroll in up to 6 classes, and each class may have up to 35 students.

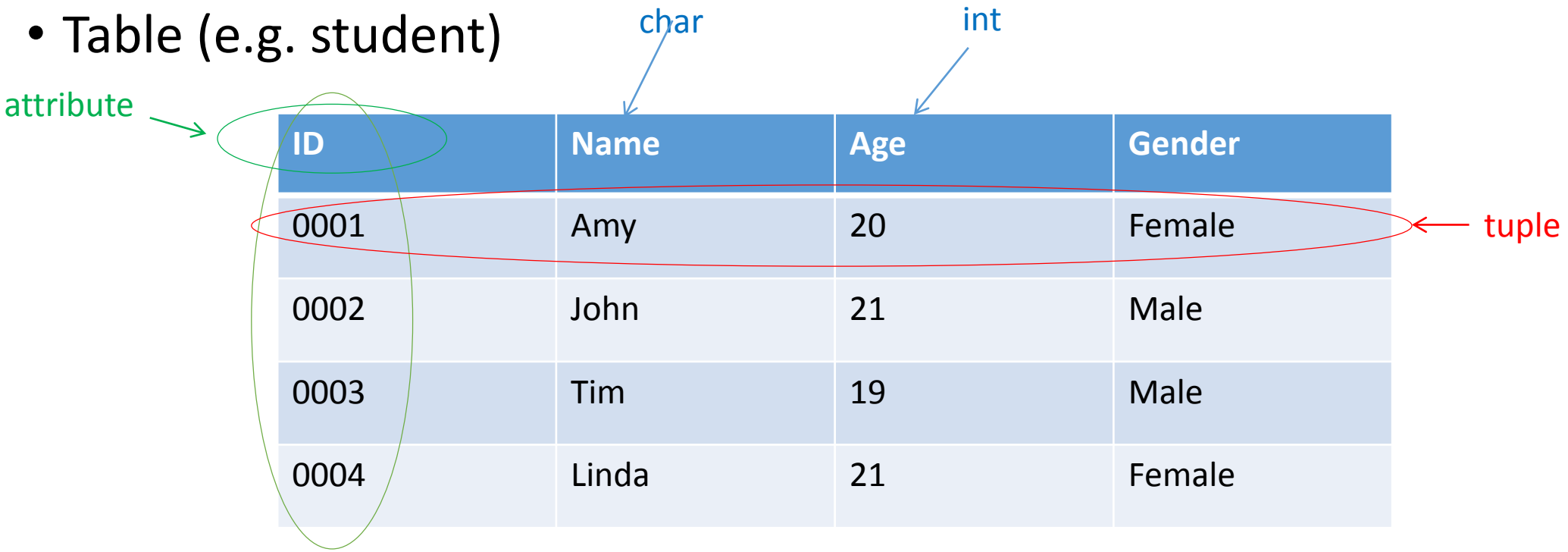
The Evolution of Data Models

TABLE 2.1 Evolution of Major Data Models

GENERATION	TIME	DATA MODEL	EXAMPLES	COMMENTS
First	1960s–1970s	File system	VMS/VSAM	Used mainly on IBM mainframe systems Managed records, not relationships
Second	1970s	Hierarchical and network	IMS, ADABAS, IDS-II	Early database systems Navigational access
Third	Mid-1970s	Relational	DB2 Oracle MS SQL Server MySQL	Conceptual simplicity Entity relationship (ER) modeling and support for relational data modeling
Fourth	Mid-1980s	Object-oriented Object/ relational (O/R)	Versant Objectivity/DB DB2 UDB Oracle 11g	Object/relational supports object data types Star Schema support for data warehousing Web databases become common
Fifth	Mid-1990s	XML Hybrid DBMS	dbXML Tamino DB2 UDB Oracle 11g MS SQL Server	Unstructured data support O/R model supports XML documents Hybrid DBMS adds object front end to relational databases Support large databases (terabyte size)
Emerging Models: NoSQL	Late 2000s to present	Key-value store Column store	SimpleDB (Amazon) BigTable (Google) Cassandra (Apache)	Distributed, highly scalable High performance, fault tolerant Very large storage (petabytes) Suited for sparse data Proprietary API

The Relational Model

- Relational database management system (RDBMS): Users usually see it as a collection of tables.
- Table (e.g. student)



ID	Name	Age	Gender
0001	Amy	20	Female
0002	John	21	Male
0003	Tim	19	Male
0004	Linda	21	Female

Table

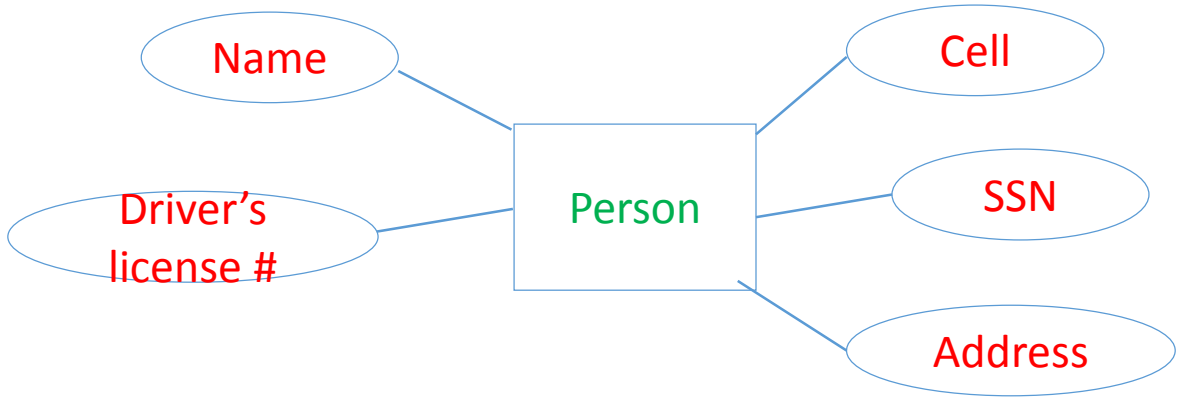
Animals

Name	Type	Birth_Date	Consumption	Avg amount of milk per day
Sara	Cow	01/02/2008	2000	2
Lisa	Cow	03/12/2008	1500	1.5
John	Cow	04/05/2009	1850	1.75
Jimmy	Pig	09/25/2010	800	
Mia	Dog	08/23/2010	300	

Null

Key

- Key: One or more attributes that uniquely identify each row.

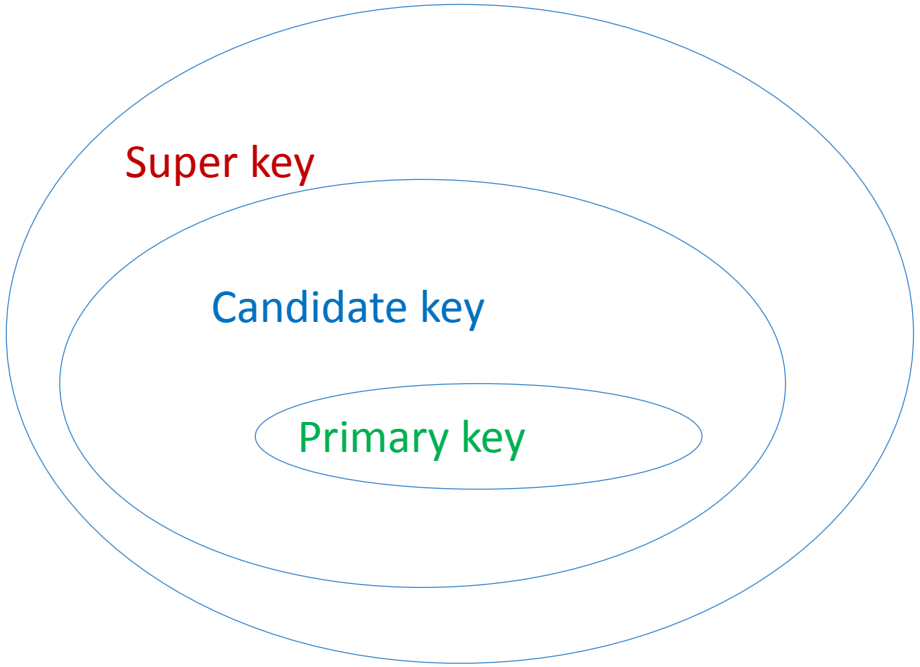


- Super Key: Any key that uniquely identifies each row.
- Candidate Key: A super key without unnecessary attributes.

Is "SSN" a candidate key?
- Yes
Is "SSN, name" a candidate key?
- No

Key

- Primary key: A candidate key selected by the DB designer to uniquely identify a row.



Super key:
(1)SSN
(2)Driver's license # + Address + Name;
(3)Driver's license # + Address
……
Candidate key:
(1)SSN
(2) Driver's license # + Address

Primary key:
e.g. SSN

Primary Key

Product

Name	Amount
Cow Milk	10
Chicken Egg	200
Chicken	50
Apple	300
Orange	600

Primary key: Name

(1) Must be unique
(2) Cannot have null values

The Relational Model

FIGURE 2.1 Linking relational tables

Table name: AGENT (first six attributes)

Database name: Ch02_InsureCo

AGENT_CODE	AGENT_LNAME	AGENT_FNAME	AGENT_INITIAL	AGENT_AREACODE	AGENT_PHONE
501	Alby	Alex	B	713	228-1249
502	Hahn	Leah	F	615	882-1244
503	Okon	John	T	615	123-5589

Link through AGENT_CODE

Table name: CUSTOMER

CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_AREACODE	CUS_PHONE	CUS_INSURE_TYPE	CUS_INSURE_AMT	CUS_RENEW_DATE	AGENT_CODE
10010	Ramas	Alfred	A	615	844-2573	T1	100.00	05-Apr-2012	502
10011	Dunne	Leona	K	713	894-1238	T1	250.00	16-Jun-2012	501
10012	Smith	Kathy	W	615	894-2285	S2	150.00	29-Jan-2013	502
10013	Olowski	Paul	F	615	894-2180	S1	300.00	14-Oct-2012	502
10014	Orlando	Myron		615	222-1672	T1	100.00	28-Dec-2013	501
10015	O'Brian	Amy	B	713	442-3381	T2	850.00	22-Sep-2012	503
10016	Brown	James	G	615	297-1228	S1	120.00	25-Mar-2013	502
10017	Williams	George		615	290-2556	S1	250.00	17-Jul-2012	503
10018	Farriss	Anne	G	713	382-7185	T2	100.00	03-Dec-2012	501
10019	Smith	Olette	K	615	297-3809	S2	500.00	14-Mar-2013	503

SOURCE: Course Technology/Cengage Learning

Foreign Key

- Foreign Key: An attribute whose values match primary key values in the related table or be null.

Product

Name
Cow Milk
Chicken Egg
Chicken
Apple
Orange

Sales

Date	Name	Amount
10/21/12	Apple	500
10/21/12	~~Banana~~	380
10/21/12	Cow milk	3
10/21/12	Orange	6

- **Exercise:** Identify the primary key and the foreign key

EMPLOYEE

EMP_CODE	EMP_LNAME	JOB_CODE
14	Rudell	2
15	McDade	1
16	Ruellardo	1
17	Smith	3
20	Smith	2

BENEFIT

EMP_CODE	PLAN_CODE
15	2
15	3
16	1
17	1
17	3
17	4
20	3

JOB

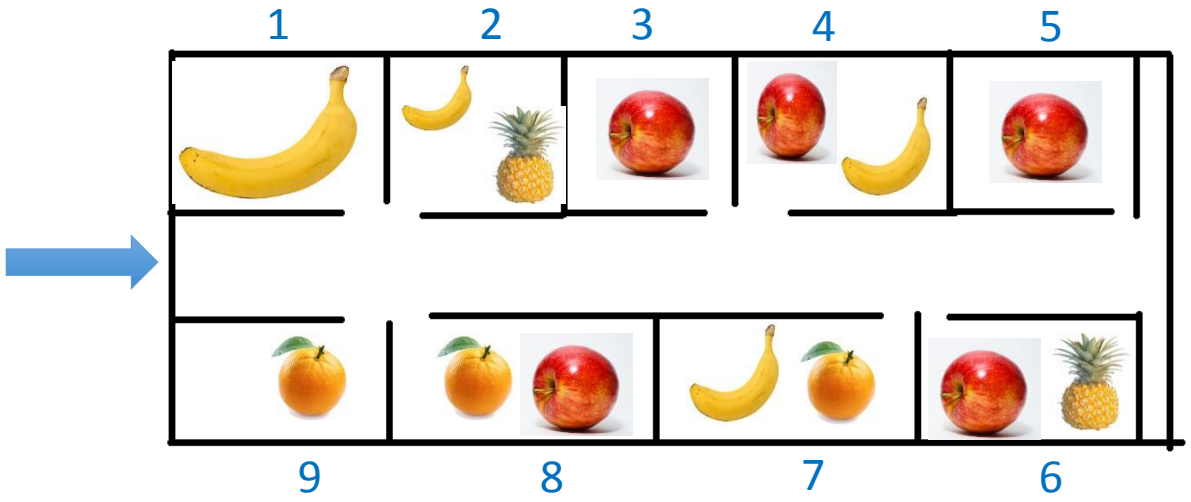
JOB_CODE	JOB_DESCRIPTION
1	Clerical
2	Technical
3	Managerial

PLAN

PLAN_CODE	PLAN_DESCRIPTION
1	Term life
2	Stock purchase
3	Long-term disability
4	Dental

Index

school of fruits



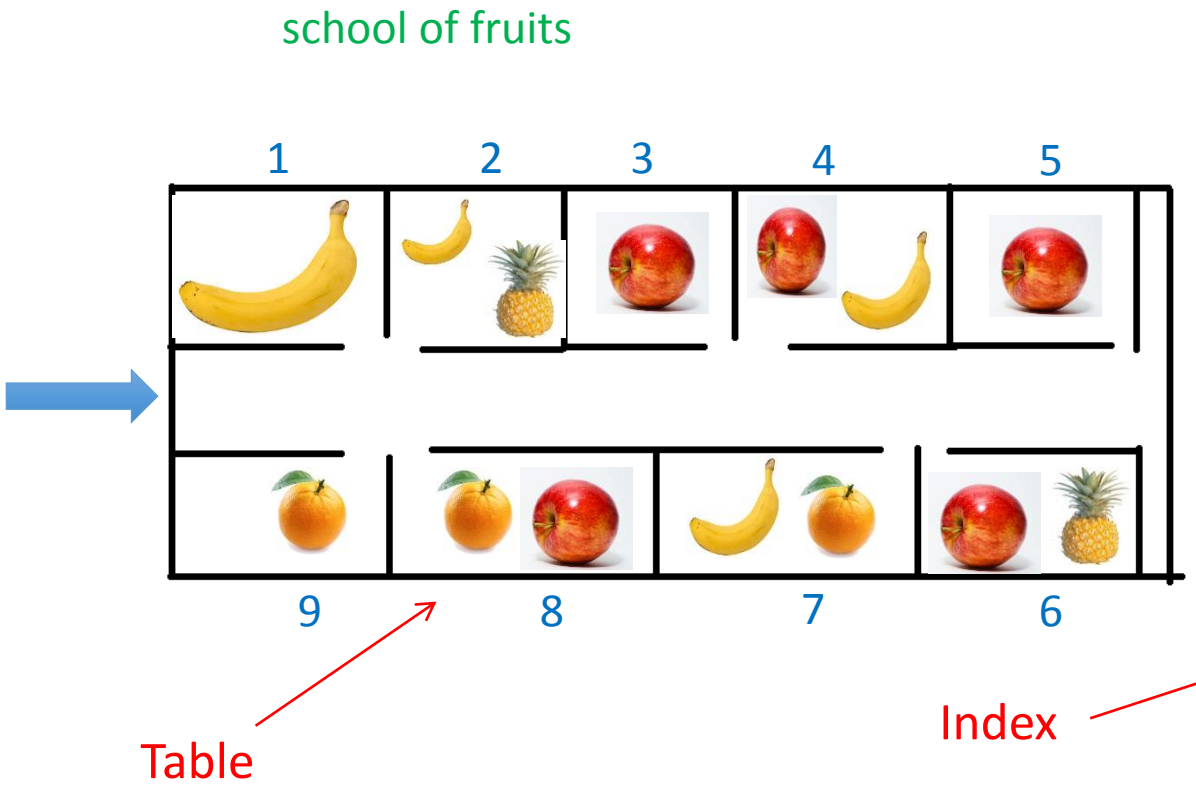
In which classroom there is at least one banana?

Find out the room where a pineapple is there.

Efficiency!

Index

- An index is a data structure defined on columns in a database table to significantly speed up data retrieval operations.



Index

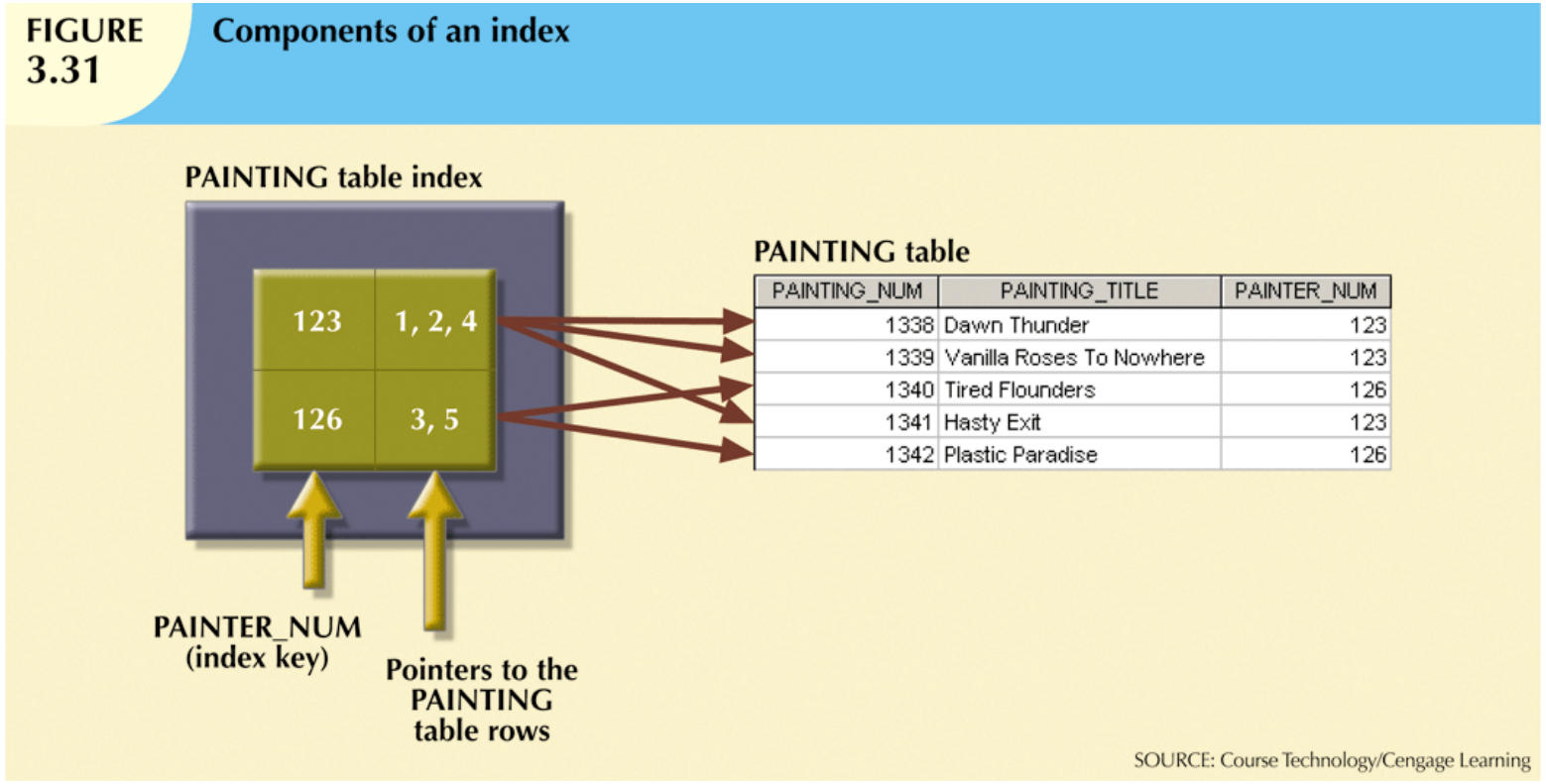


FIGURE 3.31
Components of an index
PAINTING table index
123
1, 2, 4
126
3, 5
PAINTER_NUM (index key)
Pointers to the PAINTING table rows
PAINTING table
PAINTING_NUM
PAINTING_TITLE
PAINTER_NUM
1338 Dawn Thunder 123
1339 Vanilla Roses To Nowhere 123
1340 Tired Flounders 126
1341 Hasty Exit 123
1342 Plastic Paradise 126
SOURCE: Course Technology/Cengage Learning

Relational Schema

- A textual representation of the database tables where each table is listed by its name followed by the list of its attributes in parentheses.
- Example: The relational schema for tables on slides 15 would be:
  - ✓ EMPLOYEE(EMP_CODE, EMP_LNAME, JOB_CODE)
  - ✓ BENEFIT(EMP_CODE, PLAN_CODE)
  - ✓ JOB(JOB_CODE, JOB_DESCRIPTION)
  - ✓ PLAN(PLAN_CODE, PLAN_DESCRIPTION)

Relationships within the Relational Database

- 1:M relationship – the relational database norm!

FIGURE 3.18 The implemented 1:M relationship between PAINTER and PAINTING

Table name: PAINTER
Primary key: PAINTER_NUM
Foreign key: none

Database name: Ch03_Museum

PAINTER_NUM	PAINTER_LNAME	PAINTER_FNAME	PAINTER_INITIAL
123	Ross	Georgette	P
126	Itero	Julio	G

Table name: PAINTING
Primary key: PAINTING_NUM
Foreign key: PAINTER_NUM

PAINTING_NUM	PAINTING_TITLE	PAINTER_NUM
1338	Dawn Thunder	123
1339	Vanilla Roses To Nowhere	123
1340	Tired Flounders	126
1341	Hasty Exit	123
1342	Plastic Paradise	126

SOURCE: Course Technology/Cengage Learning

Relationships within the Relational Database

FIGURE 3.22 The implemented 1:1 relationship between PROFESSOR and DEPARTMENT

Table name: PROFESSOR
Primary key: EMP_NUM
Foreign key: DEPT_CODE

Database name: Ch03_TinyCollege

EMP_NUM	DEPT_CODE	PROF_OFFICE	PROF_EXTENSION	PROF_HIGH_DEGREE
103	HIST	DRE 156	6783	Ph.D.
104	ENG	DRE 102	5561	MA
105	ACCT	KLR 229D	8665	Ph.D.
106	MKT/MGT	KLR 126	3899	Ph.D.
110	BIOL	AAK 160	3412	Ph.D.
114	ACCT	KLR 211	4436	Ph.D.
155	MATH	AAK 201	4440	Ph.D.
160	ENG	DRE 102	2248	Ph.D.
162	CIS	KLR 203E	2359	Ph.D.
191	MKT/MGT	KLR 409B	4016	DBA
195	PSYCH	AAK 297	3550	Ph.D.
209	CIS	KLR 333	3421	Ph.D.
228	CIS	KLR 300	3000	Ph.D.
297	MATH	AAK 194	1145	Ph.D.
299	ECON/FIN	KLR 284	2851	Ph.D.
301	ACCT	KLR 244	4683	Ph.D.
335	ENG	DRE 208	2000	Ph.D.
342	SOC	BBG 208	5514	Ph.D.
387	BIOL	AAK 230	8665	Ph.D.
401	HIST	DRE 156	6783	MA
425	ECON/FIN	KLR 284	2851	MBA
435	ART	BBG 185	2278	Ph.D.

1:1 relationship - Very rare!

The 1:M DEPARTMENT employs PROFESSOR relationship is implemented through the placement of the DEPT_CODE foreign key in the PROFESSOR table.

The 1:1 PROFESSOR chairs DEPARTMENT relationship is implemented through the placement of the EMP_NUM foreign key in the DEPARTMENT table.

Table name: DEPARTMENT
Primary key: DEPT_CODE
Foreign key: EMP_NUM

DEPT_CODE	DEPT_NAME	SCHOOL_CODE	EMP_NUM	DEPT_ADDRESS	DEPT_EXTENSION
ACCT	Accounting	BUS	114	KLR 211, Box 52	3119
ART	Fine Arts	A&SCI	435	BBG 185, Box 128	2278
BIOL	Biology	A&SCI	387	AAK 230, Box 415	4117
CIS	Computer Info. Systems	BUS	209	KLR 333, Box 56	3245
ECON/FIN	Economics/Finance	BUS	299	KLR 284, Box 63	3126
ENG	English	A&SCI	160	DRE 102, Box 223	1004
HIST	History	A&SCI	103	DRE 156, Box 284	1867
MATH	Mathematics	A&SCI	297	AAK 194, Box 422	4234
MKT/MGT	Marketing/Management	BUS	106	KLR 126, Box 55	3342
PSYCH	Psychology	A&SCI	195	AAK 297, Box 438	4110
SOC	Sociology	A&SCI	342	BBG 208, Box 132	2008

SOURCE: Course Technology/Cengage Learning

Relationships within the Relational Database

- M:N relationship → break it into 1:M relationships

FIGURE 3.25 Converting the M:N relationship into two 1:M relationships

Table name: STUDENT
Primary key: STU_NUM
Foreign key: none

Database name: Ch03_CollegeTry2

STU_NUM	STU_LNAME
321452	Bowser
324257	Smithson

Table name: ENROLL
Primary key: CLASS_CODE + STU_NUM
Foreign key: CLASS_CODE, STU_NUM

CLASS_CODE	STU_NUM	ENROLL_GRADE
10014	321452	C
10014	324257	B
10018	321452	A
10018	324257	B
10021	321452	C
10021	324257	C

→ the linking table

Table name: CLASS
Primary key: CLASS_CODE
Foreign key: CRS_CODE

CLASS_CODE	CRS_CODE	CLASS_SECTION	CLASS_TIME	CLASS_ROOM	PROF_NUM
10014	ACCT-211	3	TTh 2:30-3:45 p.m.	BUS252	342
10018	CIS-220	2	MWF 9:00-9:50 a.m.	KLR211	114
10021	QM-261	1	MWF 8:00-8:50 a.m.	KLR200	114

SOURCE: Course Technology/Cengage Learning