

SchoolNova



IT102

Normalization

Database Tables and Normalization

- If the database tables are treated as though they were files in a file system, the RDBMS never has a chance to demonstrate its superior data-handling capabilities.
- It is possible to create poor table structures even in a good database design. How do we produce good table structures?
- **Normalization** is a process for assigning attributes to entities. It reduces data redundancies and helps eliminate the data anomalies that result from these redundancies.
- Normalization does not fully eliminate redundancies, it produces the controlled redundancy that lets us link database tables.
- Normalization works through a series of stages called normal forms. Most common are the first three normal forms:
 - ◆ 1NF
 - ◆ 2NF
 - ◆ 3NF
- Although normalization is very important for database design, you should not assume that the highest level of normalization is always the most desirable.
- Generally, the higher the normal form, the more joins are required to produce a specified output and the more slowly the database system responds to end user demands.
- A successful design must also consider end user demand for fast performance.

1NF

- Review the STUDENT table. Note that it contains repeating groups; every student can have a group of several data entries, leading to data redundancy.
- Data redundancies yield the following anomalies:
 - ◆ Update anomalies: changing the 'Management 101' course name requires updates many alterations.
 - ◆ Insertion anomalies: creating a new Course or Major requires assigning at least one student to it.
 - ◆ Deletion anomalies: if the 'Management 101' course is discontinued, multiple updates or deletions need to be made.

Student Table

Student Number	Student Name	Student Major	Course	Course Grade
86432	Stephanie Moore	Information Systems	ISYS 315	A
			Accounting 324	B
			Management 401	C-
86789	Bob Archer	Management	Management 101	B+
			History 201	A-
98653	Haley Mills	Music	Music 371	C+
			Math 212	B-
			Management 101	B

- 1NF means that
 - ◆ All the key attributes are defined (primary key);
 - ◆ The table must not contain repeating groups. In other words, each row/column intersection can contain only one value, not a set of values.
 - ◆ All attributes are dependent on the primary key.
- If repeating groups exist, they must be eliminated by making sure that each row defines a single entity.

Primary Key

- Primary key – a data field or a combination of data fields that makes each record in the database unique.
- Concatenated primary key – a primary key that is comprised of two or more data fields. Sometimes a record (row) in a database table can not be uniquely identified by using a single field.
- Review the redesigned STUDENTS table. What kind of primary key is defined?
- Review the CREATE TABLE statement, what is the primary key of the ATTENDANCE table?

```
CREATE TABLE IF NOT EXISTS `ATTENDANCE` (  
  `A_S_ID` INT NOT NULL,  
  `A_C_ID` INT NOT NULL,  
  `A_DATE` DATE NOT NULL,  
  `HOMEWORK` TINYINT(1) NOT NULL DEFAULT 0,  
  `CREATE_DATE` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  INDEX `fk_ATTENDANCE_STUDENTS1_idx` (`A_S_ID` ASC),  
  PRIMARY KEY (`A_S_ID`, `A_C_ID`, `A_DATE`),  
  INDEX `fk_ATTENDANCE_CLASSES1_idx` (`A_C_ID` ASC),  
  CONSTRAINT `fk_ATTENDANCE_STUDENTS1`  
    FOREIGN KEY (`A_S_ID`)  
    REFERENCES `STUDENTS` (`S_ID`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_ATTENDANCE_CLASSES1`  
    FOREIGN KEY (`A_C_ID`)  
    REFERENCES `CLASSES` (`C_ID`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;
```

Student Table

Student Number (PK)	Course (PK)	Student Name	Student Major	Course Grade
86432	ISYS 315	Stephanie Moore	Info Systems	A
86432	Accounting 324	Stephanie Moore	Info Systems	B
86432	Management 401	Stephanie Moore	Info Systems	C-
86789	Management 101	Bob Archer	Management	B+
86789	History 201	Bob Archer	Management	A-
98653	Music 371	Haley Mills	Music	C+
98653	Math 212	Haley Mills	Music	B-
98653	Management 101	Haley Mills	Music	B

Exercise

- Normalize the BOOKS table to 1NF;
- Write the CREATE TABLE statement(s) based on your normalized design;
- Write INSERT statements to populate the table with data.
- Write a SELECT statement that selects only authors and orders them by name.

BOOK_TITLE	AUTHORS	NUM_OF_PAGES	PRICE
Effective Java Programming Language Guide	Joshua Bloch	252	\$44.99
Design Patterns: Elements of Reusable Object-Oriented Software	Erich Gamma Richard Helm Ralph Johnson John Vlissides	395	\$54.99
Refactoring: Improving the Design of Existing Code	Martin Fowler Kent Beck John Brant William Opdyke Don Roberts	464	\$59.99

Homework

- StonyBrook Hospital has hired you to create a database for managing patients billing.
- Review the patient's bill sample.
- Create a database design (on paper) that includes tables, relationships, columns, primary keys and foreign keys.
- Optional: create the tables in MySQL based on your design.
- **Required:** Bring your database design (on paper) to the next class.

Patient bill						
Patient #: 12345			Date: 7/20/08			
Patient Name: Mary Baker			Date admitted: 7/14/08			
Patient Address: 300 Oak Street			Discharge date: 7/17/08			
City-State-Zip: Boulder, CO 80638						
Cost Center	Cost Name	Date Charged	Item Code	Desc	Charge	Bal Due
100	Room & Board	7/14/08	2000	Semi-prv room	200.00	
		7/14/08	2005	Television	5.00	
		7/15/08	2000	Semi-prv room	200.00	
		7/16/08	2000	semi-prv room	200.00	
				Subtotal		605.00
110	Laboratory	7/14/08	1580	Glucose	25.00	
		7/15/08	1585	Culture	20.00	
				Subtotal		45.00
125	Radiology	7/15/08	3010	X-ray chest	30.00	
				Subtotal		30.00
				Balance due		680.00