

# SQL FOR DB2

Chapter 8

Physical Database Design

# CHAPTER OBJECTIVES

- ① Explain what physical database design is
- ① Explain the steps for physical database design



# PHYSICAL DATABASE DESIGN

- ⊙ Physical database design provides the bridge between logical data modeling and implementation
- ⊙ Logical data modeling analyzes an organization's information requirements to produce a data model that specifies entity types, attributes of entities, relationships among entities, and integrity rules
- ⊙ These elements are represented as tables and views, columns, foreign keys, domains, and other design objects based on the relational model
- ⊙ Physical database design focuses on the platform specific details



# STEPS IN DATABASE DESIGN

- ◎ Step 1: Create an SQL table for each entity
- ◎ Step 2: Specify primary, candidate, and foreign keys
- ◎ Step 3: Specify view implementation
- ◎ Step 4: Specify security implementation
- ◎ Step 5: Specify additional indexes

# STEP 1: CREATE AN SQL TABLE FOR EACH ENTITY

- ⦿ Each entity is represented in the data model as a table containing columns that represent the entity's attributes.
- ⦿ For an SQL table, the following design information should be specified:
  - ⦿ Full table name, system table name
  - ⦿ The name of entity in the related data model
  - ⦿ Descriptive text for the table
  - ⦿ Schema that will contain the table
  - ⦿ User profile that will own the table

# COLUMN DESIGN INFORMATION

- ⦿ For each column, the following design information should be specified:
  - ⦿ Name of column
  - ⦿ Table name that contains the column
  - ⦿ Descriptive text for column
  - ⦿ Column data type, length, decimal positions, allowable values
  - ⦿ Whether null is allowed
  - ⦿ Default value or value used for special cases
  - ⦿ Default column headings, display and print formats

# STEP 2: SPECIFY PRIMARY, CANDIDATE, FOREIGN KEYS

- ⊙ A primary key is one or more columns that uniquely identify rows.
- ⊙ Candidate keys are other column(s) that are also unique for each row
- ⊙ A foreign key is one or more column(s) in a dependent table that references the primary key in a parent table
- ⊙ Each of these types of keys requires some implementation technique to enforce the corresponding integrity constraints
- ⊙ These techniques could be:
  - ⊙ Constraints in the SQL create table statements
  - ⊙ Trigger programs
  - ⊙ Stored procedures

# STEP 3: SPECIFY VIEW IMPLEMENTATION

- ⦿ Users may need to see only selected rows or columns from data that is stored in tables
- ⦿ Users may also need to see data from several tables
- ⦿ Provide the full description for each view, this might take the form of the necessary definitions for an SQL Create View statement
- ⦿ For an SQL view the following design information should be specified:
  - ⦿ View name, system name
  - ⦿ Descriptive text
  - ⦿ Schema that will contain the view
  - ⦿ User profile that will own the view

# STEP 4: SPECIFY SECURITY IMPLEMENTATION

- ⊙ Specify public and private authority for table and program objects
- ⊙ SQL allows column-level security for UPDATE statements
- ⊙ SQL views can implement access that is restricted to selected rows or a subset of columns
- ⊙ For more context-sensitive control of database access, stored procedures can be used
- ⊙ Implement an audit trail

# STEP 5: SPECIFY ADDITIONAL INDEXES

- ⊙ Indexes is one of the most common options for improving performance
- ⊙ An index usually provides an efficient way to access a specific row by key value or to process some or all of a table in a particular sequence
- ⊙ The two most common reasons to consider additional indexes are for retrieving groups of rows in a particular sequence and as an efficient way to select a subset of rows
- ⊙ Avoid an index on a frequently updated column

# CHAPTER SUMMARY

- ⦿ Physical database design is the application development state that produces a specification for the subsequent implementation
- ⦿ Database design used the logical data model that is produced in an earlier stage of the development process
- ⦿ The major steps in the database design are:
  - ⦿ Step 1: Create an SQL table for each entity
  - ⦿ Step 2: Specify primary, candidate, and foreign keys
  - ⦿ Step 3: Specify view implementation
  - ⦿ Step 4: Specify security implementation
  - ⦿ Step 5: Specify additional indexes