SQL - DATA DEFINITION

Aims:

At the end of this group of two lectures, you should be able to use SQL to define and populate and manipulate your database

Reading: Elmasri & Navathe, Chapters 6

COSC265, 2019

Slide 4-1

OVERVIEW

1. Introduction to SQL
   - History of the language
   - SQL statements
   - Oracle: SQL in context
   - Elements of SQL
2. Data definition in SQL
   - CREATE/ALTER/DROP SCHEMA
   - CREATE TABLE statement

COSC265, 2019

Slide 4-2

HISTORY OF SQL

- Ancestor: SEQUEL (Structured English Query Language), a language developed for System R at the IBM San Jose Research Lab in early 1970
- SQL: a DDL, VDL and DML language in one
- Standards:
  - SQL1 (ANSI 1986, adopted by ISO in 1987)
  - Update SQL89 (Integrity Enhancement Feature, IEF)
  - SQL2 (SQL-92)
  - SQL:1999 (SQL3)
  - SQL:2003 (introduces XML-related specifications)
  - SQL:2006 (XML and XQuery)
  - SQL:2008
  - SQL:2011
  - SQL:2016

COSC265, 2019

Slide 4-3

SQL

- DDL: Manipulate database objects
  (tables, constraints, domains, procedures, ...)
- VDL (views)
- DML: Manipulate data
  (select, insert, update, and delete data)
- Transaction control
- Integrity control (DDL)
- Embedded and dynamic SQL
- Authorization

COSC265, 2019

Slide 4-4

ORACLE: SQL IN CONTEXT

- In Oracle, SQL statements may be:
  - entered directly from SQL*Plus
  - entered through GUI interfaces (SQL Developer)
  - embedded within programs
    (Embedded SQL, PL/SQL)
  - used in application development tools

COSC265, 2019

Slide 4-5

ELEMENTS OF SQL STATEMENTS

- Terminology: tables, columns, rows

- Important distinction between SQL and the formal relational model: SQL allows a table (relation) to have two or more tuples that are identical in all their attribute values

- Hence, an SQL relation (table) is a *multi-set* (sometimes called a *bag*) of tuples; it *is not* a set of tuples

- SQL relations can be constrained to be sets by specifying PRIMARY KEY or UNIQUE attributes, or by using the DISTINCT option in a query

COSC265, 2019

Slide 4-6

NAMING CONVENTIONS

- Database names - up to 8 characters
- Other names: sequences of no more than 30 alphanumeric characters, starting with an alphabetic character and excluding Oracle reserved words and keywords
- Table names must not begin with "sys_"
  (reserved for use by Oracle)
- Case insensitive (except in string constants)

COSC265, 2019

Slide
4-7

DATA TYPES

- Even when two DBMSs have the data type with the same name, they might be different. Check documentation!
- Numeric (*integer, int, smallint, numeric, float, real, double precision*, ...)
  - Oracle: NUMBER(*p,s*)
  - Floating numbers, with precision (max 38) and scale (max 127)
  - NUMBER(17, 12): column length is 17, max. 12 decimal digits (the integer portion of a number is the difference between the two numbers)
- Character types
  (*char, character, varchar, nchar*, ...)

COSC265, 2019

Slide
4-8

DATA TYPES (CONT)

- Abstract types: *date, time, money, object_key, table_key*
  - Default Oracle format for Date is 'dd-mmm-yy'
  - Other formats can be specified: e.g. 'dd/mm/yyyy'
  - Valid date range from January 1, 4712 BC to 31.12.9999
- Binary data (*bit, binary, blob*, ...)
- Boolean (*true, false, unknown*)
- TIME(i): hh:mm:ss:ii...i
  additional digits specifying fractions of a second
- TIMESTAMP: date + time
- INTERVAL: DAY/TIME intervals or YEAR/MONTH

COSC265, 2019

Slide
4-9

OPERATORS

- Arithmetic (+, -, *, /)
- String concatenation (||)
- Comparisons (=, <>, !=, <, <=, >, >=)
- Logical (and, or, not)
- Set (union, intersection, set difference)

COSC265, 2019

Slide
4-10

FUNCTIONS

- Scalar functions
  - Type conversion: to_char, to_date, to_number, …
  - Numeric: abs, ceil, floor, exp, log, mod, power, round, sqrt, tan, …
  - String: substring, upper, lower, translate, convert, instr, length, lpad, …
  - Date/time functions: Sysdate, add_months, months_between, next_day, new_time, …
  - nvl, decode, case
- Aggregate (set) functions
  - sum, count, max, min, avg

COSC265, 2019

Slide
4-11

- **Expression**
  - An attribute
  - A combination of attributes, operators and functions

- **Predicates**
  - Like
  - Between
  - In
  - Any/all
  - Exists
  - Is Null

COSC265, 2019

Slide
4-12

DATA DEFINITION IN SQL

○ Statements that allow the user to define, alter or drop various database objects
○ Schema:
- CREATE SCHEMA
- ALTER SCHEMA
- DROP SCHEMA

COSC265, 2019

Slide 4-13

CREATE SCHEMA

○ Introduced in SQL2

○ *CREATE SCHEMA schema-name AUTHORIZATION auth*
[object-definition
{object-definition}]

○ *auth* – the user who owns the schema

COSC265, 2019

Slide 4-14

CREATE TABLE STATEMENT

○ Specifies a new base relation by giving it a name, and specifying each of its attributes and their types

○ NOT NULL option for mandatory attributes

○ Key attributes can be specified via the PRIMARY KEY and UNIQUE phrases

○ Foreign key can be specified via FOREIGN KEY and REFERENCES

○ Other constraints: CHECK

COSC265, 2019

Slide 4-15

CREATE TABLE (CONT)

○ CREATE TABLE [schema.]table-name
(column-name column-spec
{, column-name column-spec})
[,[constraint constraint-name] table-constraint
{,[constraint constraint-name] table-constraint}
[with_clause]

○ CREATE TABLE [schema.]table-name
[(column-name column-spec
{, column-name column-spec})]
AS subselect [with_clause]

COSC265, 2019

Slide 4-16

CHECKLIST FOR CREATING TABLES

○ What are the attributes to be stored?
○ What are the data types for the attributes?
○ Which columns make the primary key?
○ Are there any foreign keys?
○ Which columns do not allow NULLs?
○ Which columns do not allow duplicates?
○ Are there default values for certain columns?
○ Are there any limitation on the values allowed in attributes?

COSC265, 2019

Slide 4-17

CREATING THE MOVIE DATABASE

○ CREATE TABLE DIRECTOR
○ (DNUMBER /* Unique number */ INTEGER NOT NULL
constraint dir_pk PRIMARY KEY,
LNAME /* Last name */ VARCHAR(16) NOT NULL,
FNAME /* First name */ VARCHAR(15),
BORN /* Year of birth */ INTEGER
constraint dir_born check
(BORN between 1880 and 1990),
DIED /* Year of death */ INTEGER
constraint dir_died check (DIED>1930),
constraint corr_years check (born <= died));

○ Note the comments
- Multiline /* ... */
- Single line --

COSC265, 2019

Slide 4-18

CREATING THE MOVIE DATABASE (CONT)

o CREATE TABLE MOVIE
(MNUMBER /* Unique number */ integer not null
PRIMARY KEY,
TITLE /* Title */ VARCHAR(50) not null,
TYPE /* Type of the movie */ VARCHAR(15) not null,
AANOM /* Number of Academy Awards nominations*/
integer,
AAWON /* Number of AA won */ integer,
YEAR /* Year when the movie was made */ integer,
CRITICS /* Critics rating */ varchar(2),
DIRECTOR /* Director's number */ integer
        REFERENCES DIRECTOR);

COSC265, 2019

Slide
4-19

CREATING THE MOVIE DATABASE (CONT)

o CREATE TABLE STAR
(SNUMBER /* Unique number */ integer not null,
LNAME /* Last name */ VARCHAR(15) not null,
FNAME /* First name */ VARCHAR(15),
BORN /* Year of birth */ integer
        constraint check_born check (BORN between
                1880 and 2000),
DIED /* Year of death */ integer constraint
        check_died check (DIED>1930),
CITY /* City of birth */ varchar(15),
        constraint corr_syears check (born <= died),
PRIMARY KEY (SNUMBER));

COSC265, 2019

Slide
4-20

CREATING THE MOVIE DATABASE (CONT)

o CREATE TABLE CUSTOMER
(LNAME /* Last name */ varchar(15) not null,
FNAME /* First name */ varchar(15) not null,
CNUMBER /* Unique number */ integer not null,
ADDRESS /* Customer's address */ varchar(40),
RENTALS /* The number of DVDs rented */
        integer check (rentals>=0),
BONUS /* 1/10 of RENTALS */ integer,
JDATE /* Date of joining the club */ date,
PRIMARY KEY (CNUMBER));

COSC265, 2019

Slide
4-21

CREATING THE MOVIE DATABASE (CONT)

o CREATE TABLE DVD
(CODE /* Unique number */ integer not null,
MOVIE /* Movie number */ integer not null,
PDATE /* Purchase date */ DATE,
TIMES /* Times rented */ integer default 0,
CUSTOMER /* Customer renting the DVD */ integer,
HIREDATE /* Date of hire */ date,
PRIMARY KEY (CODE),
FOREIGN KEY (CUSTOMER) REFERENCES
CUSTOMER(CNUMBER),
FOREIGN KEY (MOVIE) REFERENCES MOVIE,
constraint check_times check (TIMES >= 0));

COSC265, 2019

Slide
4-22

CREATING THE MOVIE DATABASE (CONT)

o CREATE TABLE STARS
(MOVIE /* Movie number */ integer not null
        REFERENCES MOVIE(MNUMBER),
STAR /* Star number */ integer not null
        REFERENCES STAR(SNUMBER),
ROLE varchar(20) not null,
PRIMARY KEY (MOVIE,STAR,ROLE));

COSC265, 2019

Slide
4-23

CREATING DOMAINS

o CREATE DOMAIN domain-name type
    {CHECK check-statement};
o Example:
CREATE DOMAIN LNAME VARCHAR(20)
CHECK (VALUE IS NOT NULL);

Not available in Oracle!

COSC265, 2019

Slide
4-24

DROP TABLE

- Used to remove a relation (base table) *and its definition*

- The relation can no longer be used in queries, updates, or any other commands since its description no longer exists

- **DROP TABLE DEPENDENT;**

COSC265, 2019

Slide
4-25

ALTER TABLE

- Used to add, delete or modify attributes or constraints to one of the base relations

- Adding an attribute
  ALTER TABLE EMPLOYEE
  ADD JOB VARCHAR(12);

- The new attribute will have NULLs in all the tuples of the relation right after the command is executed; hence, the NOT NULL constraint is *not allowed* for such an attribute

- The database users must still enter a value for the new attribute JOB for each EMPLOYEE tuple. This can be done using the UPDATE command.

COSC265, 2019

Slide
4-26

ALTER TABLE (CONT)

- ALTER TABLE [schema.]table-name
  [ADD | MODIFY] [constraint-name column-name]
  spec |
  DROP CONSTRAINT constraint-name
  [RESTRICT | CASCADE]

COSC265, 2019

Slide
4-27

REFERENTIAL INTEGRITY OPTIONS

- We can specify RESTRICT, CASCADE, SET NULL or SET DEFAULT on referential integrity constraints (foreign keys)

  create table DEPT
   (DNAME varchar(10) not null,
    DNUMBER integer not null,
    MGRSSN char(9),
    MGRSTARTDATE char(9),
    primary key (DNUMBER),
    unique (DNAME),
    foreign key (MGRSSN) references EMP
  on delete SET DEFAULT
  on update CASCADE   );

COSC265, 2019

Slide
4-28

REFERENTIAL INTEGRITY OPTIONS

    CREATE TABLE   EMP
      (ENAME varchar(30) not null,
          ESSN char(9),
          BDATE date,
          DNO integer default 1,
          SUPERSSN char(9),
          PRIMARY KEY (ESSN),
          FOREIGN KEY (DNO) references DEPT
      ON DELETE SET DEFAULT ON UPDATE CASCADE,
          FOREIGN KEY (SUPERSSN) references EMP
      ON DELETE SET NULL ON UPDATE CASCADE);

COSC265, 2019

Slide
4-29

DROP SCHEMA

- DROP SCHEMA schema-name [CASCADE | RESTRICT]

- CASCADE: all defined database objects are deleted at the same time

COSC265, 2019

Slide
4-30