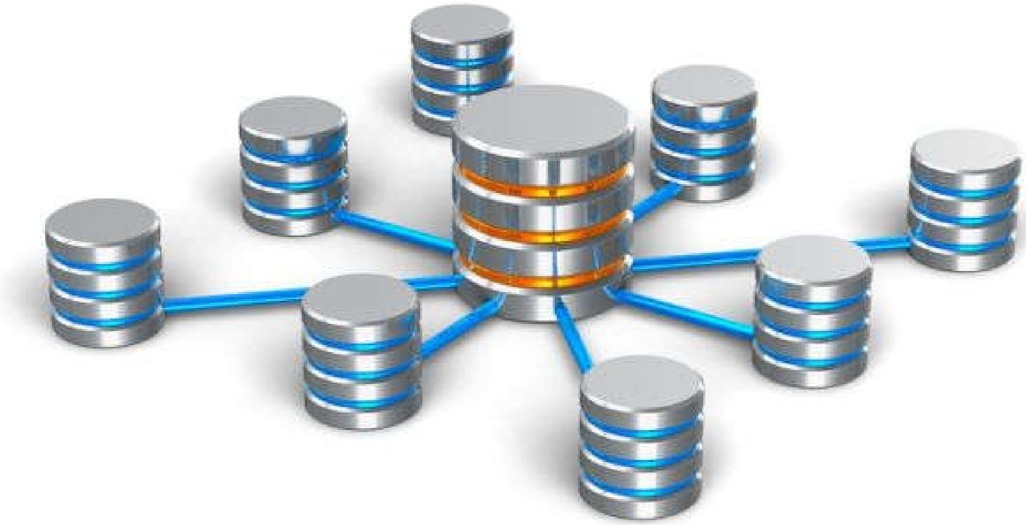
# Chapter 6: Basic SQL

Database Systems CS203

[redacted]

[redacted]



By: Anam Qureshi

# Outline

- SQL Data Definition and Data Types
- Specifying Constraints in SQL
- Basic Retrieval Queries in SQL
- INSERT, DELETE, and UPDATE Statements in SQL
- Additional Features of SQL

# Basic SQL

- SQL language
  - Considered one of the major reasons for the commercial success of relational databases
- SQL
  - The origin of SQL is relational predicate calculus called tuple calculus (see Ch.8) which was proposed initially as the language SQUARE.
  - SQL Actually comes from the word “SEQUEL” which was the original term used in the paper: “SEQUEL TO SQUARE” by Chamberlin and Boyce. IBM could not copyright that term, so they abbreviated to SQL and copyrighted the term SQL.
  - Now popularly  known as “Structured Query language”.
  - SQL is an informal  or practical rendering of the relational data model with syntax

# SQL Data Definition, Data Types, Standards

- Terminology:
  - **Table**, **row**, and **column** used for relational model terms relation, tuple, and attribute
- `CREATE` statement
  - Main SQL command for data definition
- The language has features for : Data definition, Data Manipulation, Transaction control (Transact-SQL, Ch. 20), Indexing (Ch.17), Security specification (Grant and Revoke- see Ch.30), Active databases (Ch.26), Multi-media (Ch.26), Distributed databases (Ch.23) etc.

# SQL Standards

- SQL has gone through many standards: starting with SQL-86 or SQL 1.A. SQL-92 is referred to as SQL-2.
- Later standards (from SQL-1999) are divided into **core** specification and specialized **extensions**. The extensions are implemented for different applications – such as data mining, data warehousing, multimedia etc.
- SQL-2006 added XML features (Ch. 13); In 2008 they added Object-oriented features (Ch. 12).
- SQL-3 is the current standard which started with SQL-1999. It is not fully implemented in any RDBMS.

# Schema and Catalog Concepts in SQL

- We cover the basic standard SQL syntax – there are variations in existing RDBMS systems
- **SQL schema**
  - Identified by a **schema name**
  - Includes an **authorization identifier** and **descriptors** for each element
- **Schema elements** include
  - Tables, constraints, views, domains, and other constructs
- Each statement in SQL ends with a **semicolon**

# Schema and Catalog Concepts in SQL (cont'd.)

- `CREATE SCHEMA` statement
  - `CREATE SCHEMA COMPANY AUTHORIZATION 'Jsmith';`
- **Catalog**
  - Named collection of schemas in an SQL environment
- SQL also has the concept of a cluster of catalogs.

# The CREATE TABLE Command in SQL

- Specifying a new relation
  - Provide name of table
  - Specify attributes, their types and initial constraints
- Can optionally specify schema:
  - `CREATE TABLE COMPANY.EMPLOYEE ...`

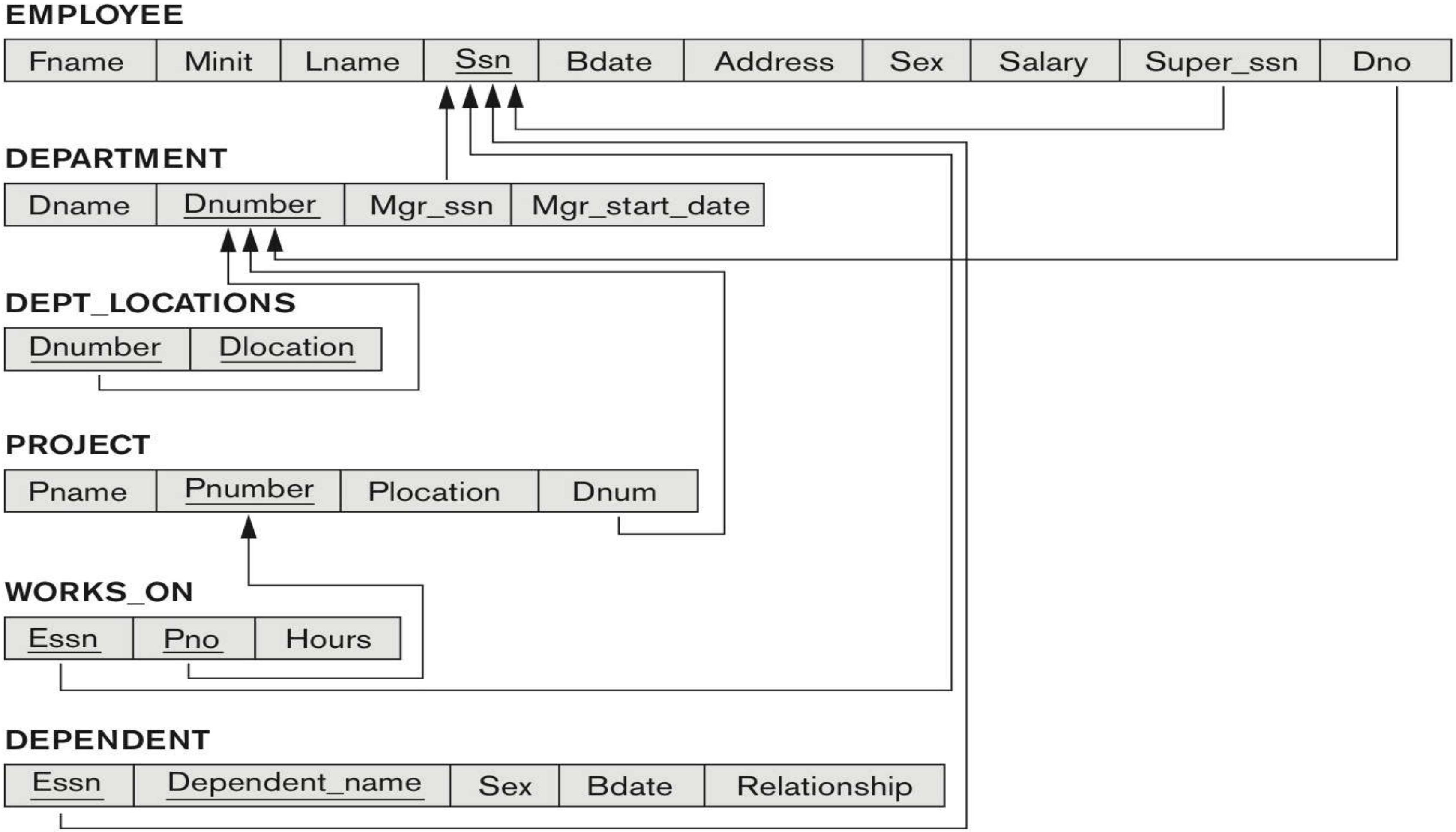
    or

  - `CREATE TABLE EMPLOYEE ...`

# The CREATE TABLE Command in SQL (cont'd.)

- **Base tables (base relations)**
  - Relation and its tuples are actually created and stored as a file by the DBMS
- **Virtual relations (views)**
  - Created through the `CREATE VIEW` statement. Do not correspond to any physical file.

# COMPANY relational database schema (Fig. 5.7)



**EMPLOYEE**

| Fname | Minit | Lname | Ssn | Bdate | Address | Sex | Salary | Super_ssn | Dno |
|---|---|---|---|---|---|---|---|---|---|

**DEPARTMENT**

| Dname | Dnumber | Mgr_ssn | Mgr_start_date |
|---|---|---|---|

**DEPT_LOCATIONS**

| Dnumber | Dlocation |
|---|---|

**PROJECT**

| Pname | Pnumber | Plocation | Dnum |
|---|---|---|---|

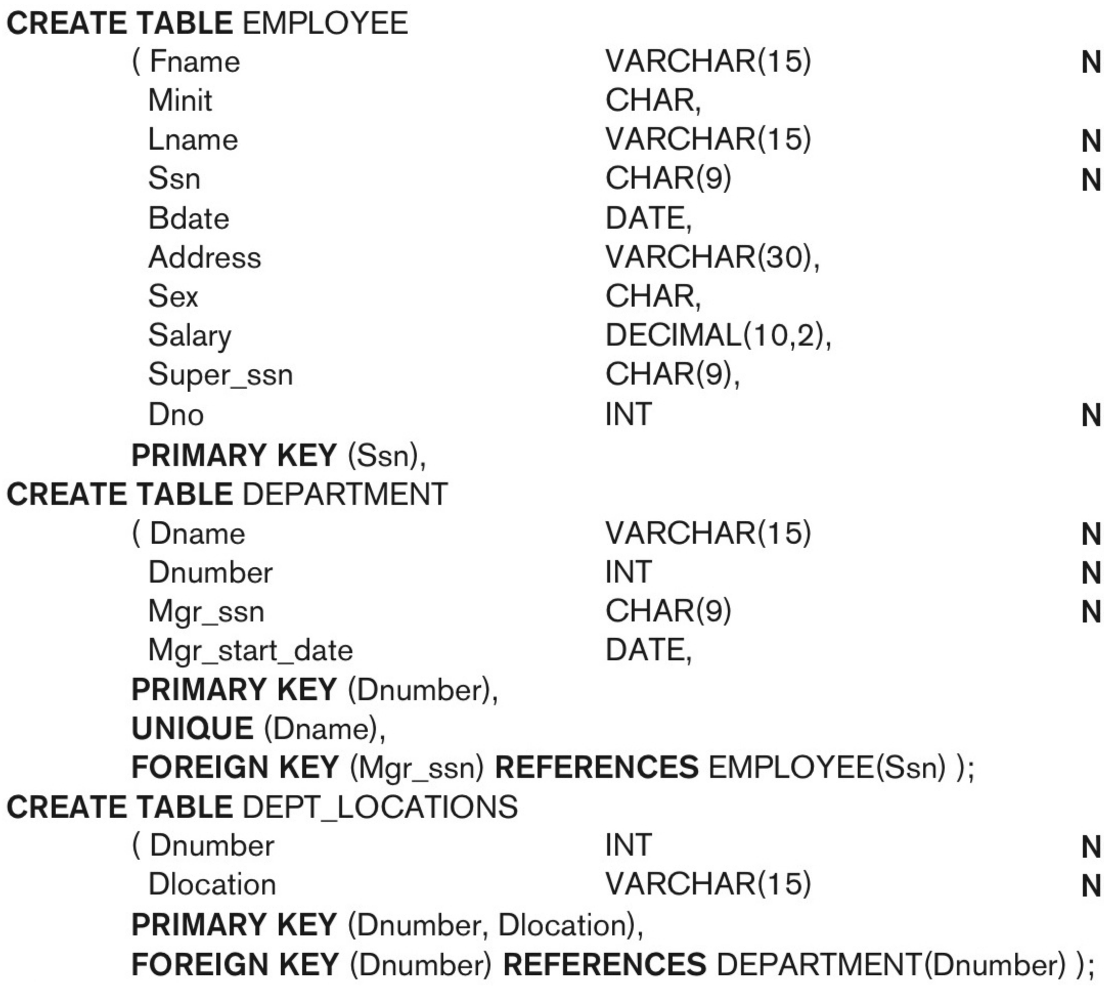
**WORKS_ON**

| Essn | Pno | Hours |
|---|---|---|

**DEPENDENT**

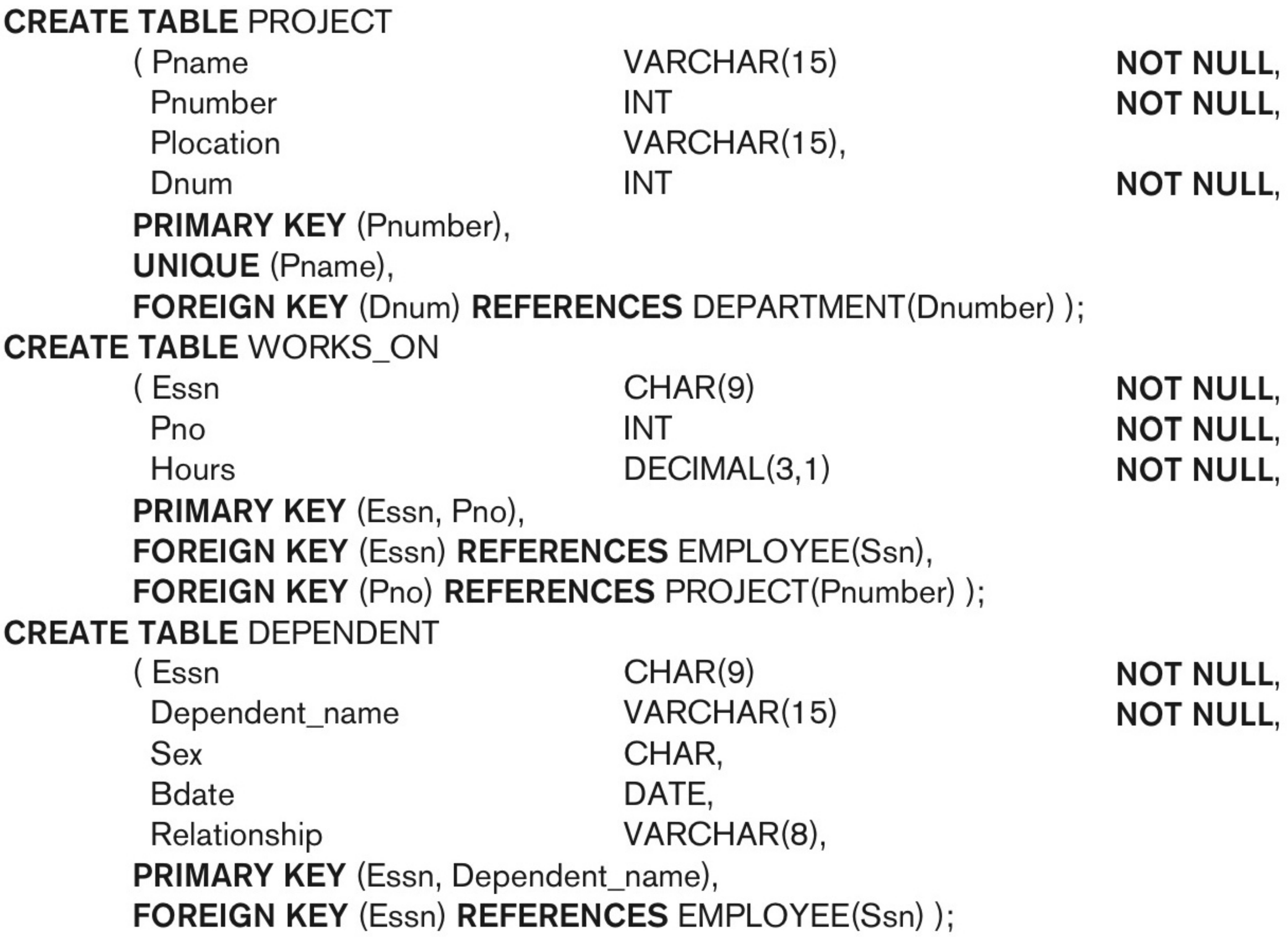
| Essn | Dependent_name | Sex | Bdate | Relationship |
|---|---|---|---|---|

# SQL CREATE TABLE data definition statements for defining the COMPANY schema from Figure 5.7 (Fig. 6.1)



```
CREATE TABLE EMPLOYEE
        ( Fname                 VARCHAR(15)           NOT NULL,
          Minit                 CHAR,
          Lname                 VARCHAR(15)           NOT NULL,
          Ssn                   CHAR(9)               NOT NULL,
          Bdate                 DATE,
          Address               VARCHAR(30),
          Sex                   CHAR,
          Salary                DECIMAL(10,2),
          Super_ssn             CHAR(9),
          Dno                   INT                   NOT NULL,
        PRIMARY KEY (Ssn),
CREATE TABLE DEPARTMENT
        ( Dname                 VARCHAR(15)           NOT NULL,
          Dnumber               INT                   NOT NULL,
          Mgr_ssn               CHAR(9)               NOT NULL,
          Mgr_start_date        DATE,
        PRIMARY KEY (Dnumber),
        UNIQUE (Dname),
        FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn) );
CREATE TABLE DEPT_LOCATIONS
        ( Dnumber               INT                   NOT NULL,
          Dlocation             VARCHAR(15)           NOT NULL,
        PRIMARY KEY (Dnumber, Dlocation),
        FOREIGN KEY (Dnumber) REFERENCES DEPARTMENT(Dnumber) );
```

# SQL CREATE TABLE data definition statements for defining the COMPANY schema from Figure 5.7 (Fig. 6.1)-continued

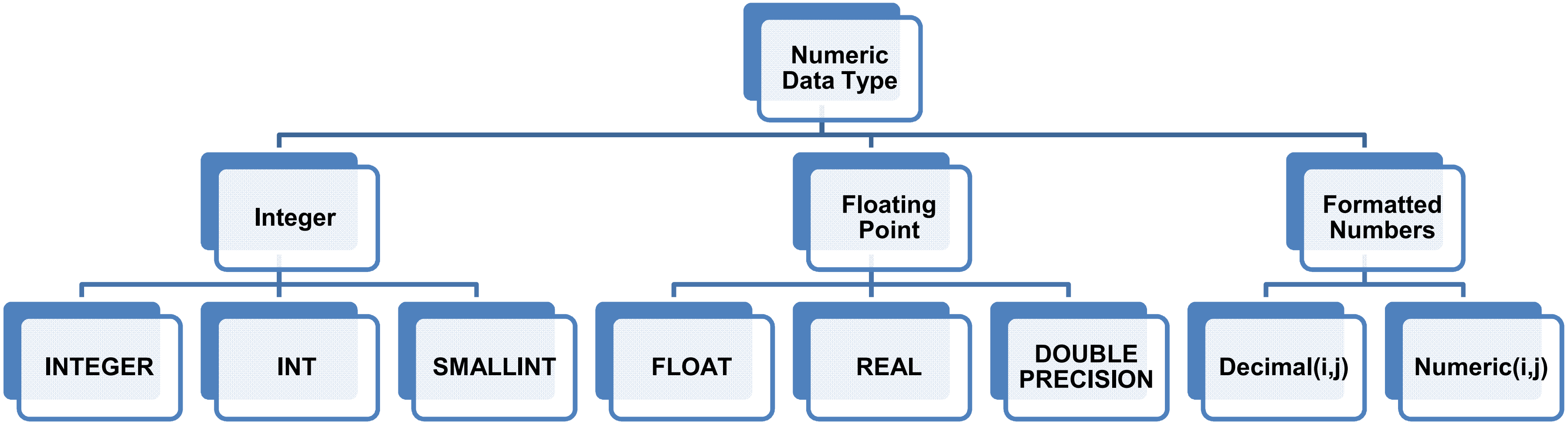


```
CREATE TABLE PROJECT
        ( Pname                        VARCHAR(15)                  NOT NULL,
          Pnumber                      INT                          NOT NULL,
          Plocation                    VARCHAR(15),
          Dnum                         INT                          NOT NULL,
        PRIMARY KEY (Pnumber),
        UNIQUE (Pname),
        FOREIGN KEY (Dnum) REFERENCES DEPARTMENT(Dnumber) );
CREATE TABLE WORKS_ON
        ( Essn                         CHAR(9)                      NOT NULL,
          Pno                          INT                          NOT NULL,
          Hours                        DECIMAL(3,1)                 NOT NULL,
        PRIMARY KEY (Essn, Pno),
        FOREIGN KEY (Essn) REFERENCES EMPLOYEE(Ssn),
        FOREIGN KEY (Pno) REFERENCES PROJECT(Pnumber) );
CREATE TABLE DEPENDENT
        ( Essn                         CHAR(9)                      NOT NULL,
          Dependent_name               VARCHAR(15)                  NOT NULL,
          Sex                          CHAR,
          Bdate                        DATE,
          Relationship                 VARCHAR(8),
        PRIMARY KEY (Essn, Dependent_name),
        FOREIGN KEY (Essn) REFERENCES EMPLOYEE(Ssn) );
```

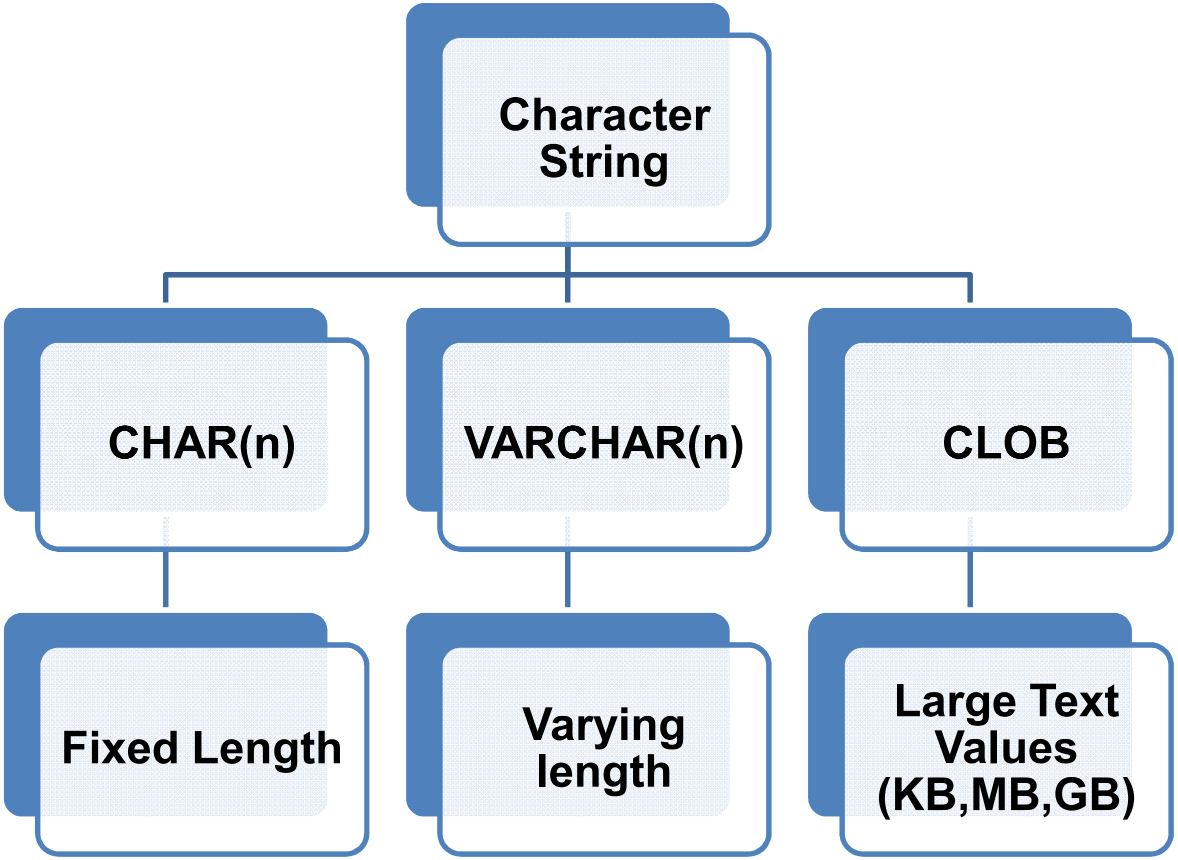
# The CREATE TABLE Command in SQL (cont'd.)

- Some foreign keys may cause errors
  - Specified either via:
    - Circular references
    - Or because they refer to a table that has not yet been created
- DBA's have ways to stop referential integrity enforcement to get around this problem.

# Attribute Data Types and Domains in SQL



# Attribute Data Types and Domains in SQL



Character String

CHAR(n)

VARCHAR(n)

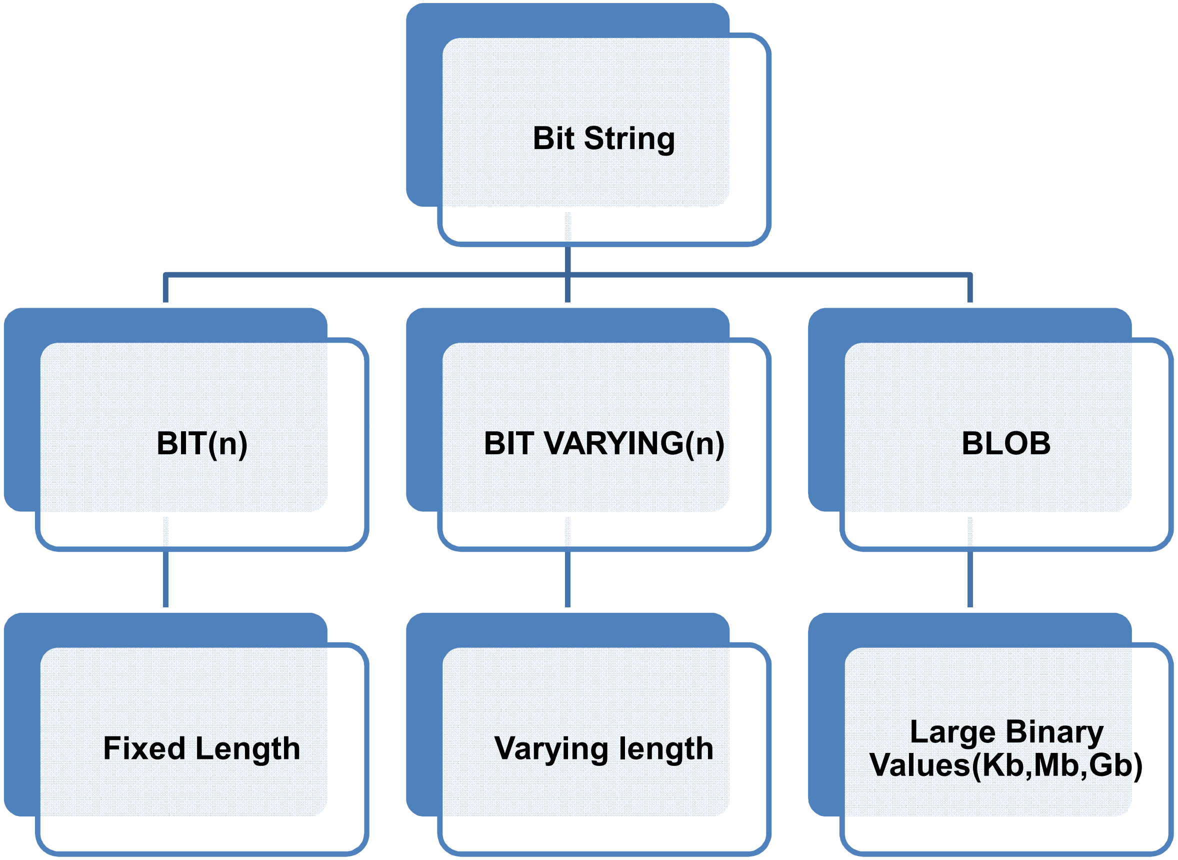
CLOB

Fixed Length

Varying length

Large Text Values (KB,MB,GB)

# Attribute Data Types and Domains in SQL



Bit String

BIT(n)

BIT VARYING(n)

BLOB

Fixed Length

Varying length

Large Binary Values(Kb,Mb,Gb)

# Attribute Data Types and Domains in SQL

- **Boolean Data Type**
    - True
    - False
    - NULL
- **Time**
    - HH:MM:SS
- **Date**
    - YYYY-MM-DD
    - Multiple mapping functions available in RDBMSs to change date formats
- **Timestamp**
    - TIMESTAMP ‘2014-09-27 09:12:47.648302’
- **Interval**
    - Specifies a relative value that can be used to increment or decrement an absolute value of a date, time, or timestamp
- **DATE, TIME, Timestamp, INTERVAL** data types can be **cast** or converted to string formats for comparison

# Attribute Data Types and Domains in SQL

- **Domain**
  - Name used with the attribute specification
  - Makes it easier to change the data type for a domain that is used by numerous attributes
  - Improves schema readability
  - Example:
    - `CREATE DOMAIN SSN_TYPE AS CHAR(9);`
- **TYPE**
  - User Defined Types (UDTs) are supported for object-oriented applications. (See Ch.12) Uses the command:

    `CREATE TYPE`