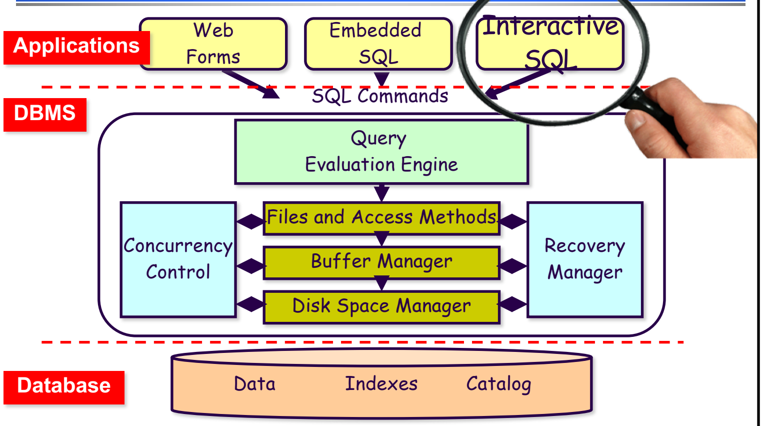
Structured Query Language
SQL - DDL

- SQL Overview
- SQL Datatypes
- DDL statements

Database Management System (DBMS)



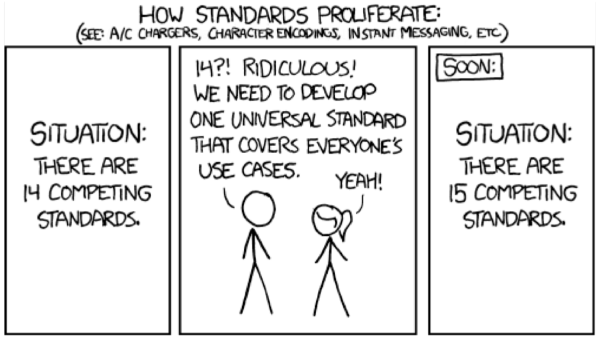
Relational Model - History

- Before: records, pointers, sets, etc.
  - Hierarchical Data Model (IBM IMS,1966-68)
  - Network Data Model (CODASYL DBTG, 1969)
- Introduced by E.F. Codd in 1970
- Revolutionary!
- First systems: 1977-8
  - System R; Ingres
- Turing award in 1981

SQL

- SQL is the query language for the **System R** developed at IBM San Jose [Astraham, Gray, Linsday, Selinger,…]
- SQL is the de-facto standard on most RDBMS
- Most successful standardization effort
  - SQL (ANSI 1986)
  - SQL1 (ANSI 1989)
  - SQL2 or SQL92 (ANSI 1992)
  - SQL3 (ANSI 1999/2000/2003) -- Core and Packages
  - SQL 2008
  - SQL 2013

A word about Standards


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

http://xkcd.com/927/

Database Languages

- **Data Definition Language (*DDL*):**
  - Define schemas
  - Define **Integrity Constraints**
    - Example: unique *SIDs*
  - More…
- **Data Manipulation Language (*DML*):**
  - To ask questions = ***Query***
    - Example: Which students have GPA > 3.75?
  - To insert, delete and update data

Basic SQL-DDL COMMANDS

- For database schemas:

  CREATE SCHEMA, DROP SCHEMA
- For tables:

  CREATE TABLE, DROP TABLE, ALTER TABLE
- For domains:

  CREATE DOMAIN, DROP DOMAIN [SQL99]
- For views:

  CREATE VIEW, DROP VIEW
- For integrity constraints

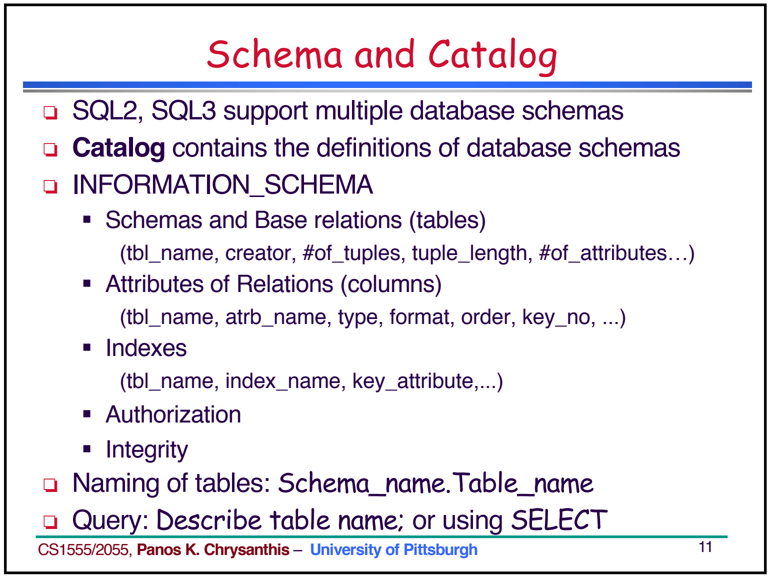
  CREATE IC, DROP IC

  For Indexes [defunct in SQL2]

Database Schema

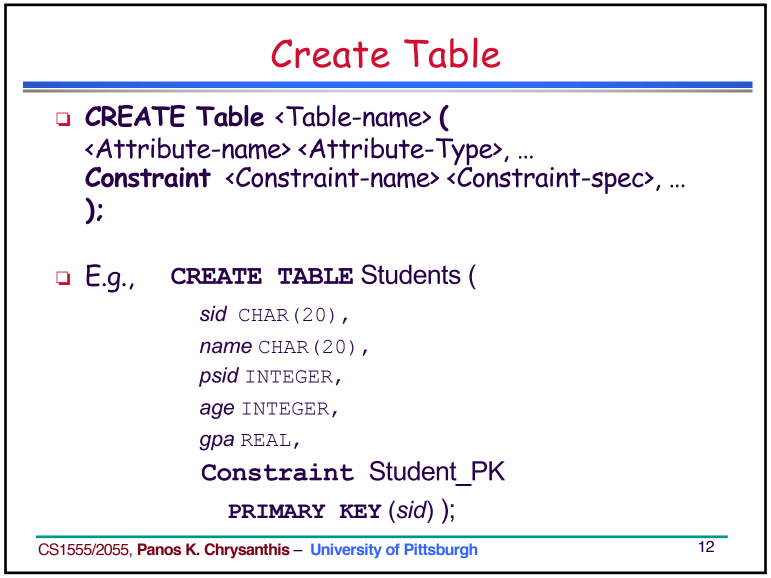
- **CREATE SCHEMA** <database-name>
  AUTHORIZATION <user-identifier>;
- E.g. **CREATE SCHEMA** micro_db

  AUTHORIZATION panos;
- **DROP SCHEMA** <db-name> [**RESTRICT** | **CASCADE**];
  - *Restrict*: removes the schema if the db has no data
  - *Cascade*: removes everything, data and definitions
- E.g., **DROP SCHEMA** micro_db **RESTRICT**;



Schema and Catalog

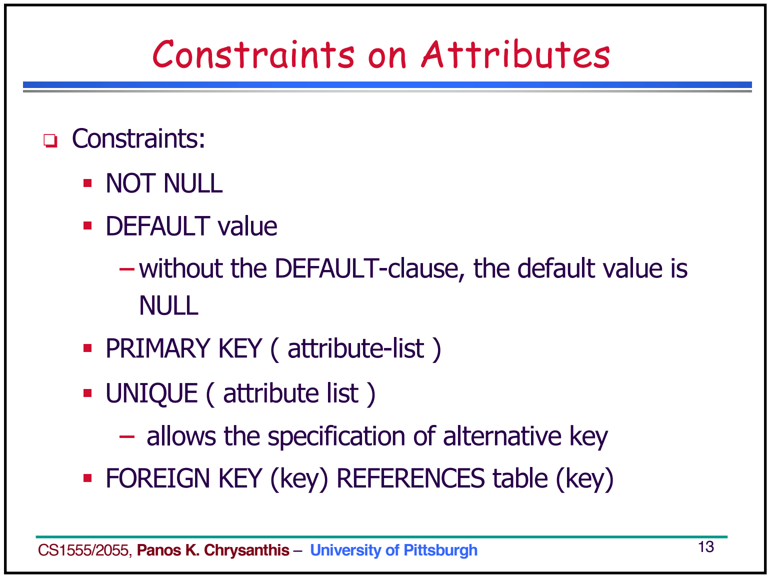
- SQL2, SQL3 support multiple database schemas
- **Catalog** contains the definitions of database schemas
- INFORMATION_SCHEMA
  - Schemas and Base relations (tables)
    (tbl_name, creator, #of_tuples, tuple_length, #of_attributes…)
  - Attributes of Relations (columns)
    (tbl_name, atrb_name, type, format, order, key_no, ...)
  - Indexes
    (tbl_name, index_name, key_attribute,...)
  - Authorization
  - Integrity
- Naming of tables: Schema_name.Table_name
- Query: Describe table name; or using SELECT



Create Table

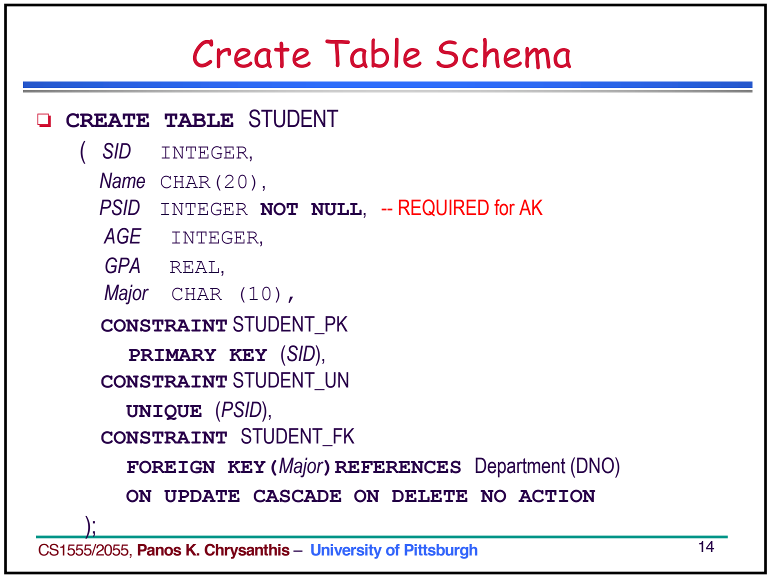
- **CREATE Table** <Table-name> **(**
  <Attribute-name> <Attribute-Type>, …
  Constraint <Constraint-name> <Constraint-spec>, …
  );
- E.g.,

```
CREATE TABLE Students (
    sid CHAR(20),
    name CHAR(20),
    psid INTEGER,
    age INTEGER,
    gpa REAL,
    Constraint Student_PK
      PRIMARY KEY (sid) );
```



Constraints on Attributes

- Constraints:
  - NOT NULL
  - DEFAULT value
    – without the DEFAULT-clause, the default value is NULL
  - PRIMARY KEY (attribute-list)
  - UNIQUE (attribute list)
    – allows the specification of alternative key
  - FOREIGN KEY (key) REFERENCES table (key)



Create Table Schema

```
CREATE TABLE STUDENT
 ( SID    INTEGER,
   Name   CHAR(20),
   PSID   INTEGER NOT NULL,  -- REQUIRED for AK
   AGE    INTEGER,
   GPA    REAL,
   Major  CHAR (10),
   CONSTRAINT STUDENT_PK
     PRIMARY KEY (SID),
   CONSTRAINT STUDENT_UN
     UNIQUE (PSID),
   CONSTRAINT STUDENT_FK
     FOREIGN KEY (Major) REFERENCES Department (DNO)
     ON UPDATE CASCADE ON DELETE NO ACTION
 );
```