SQL

Brief Intro

Orlando Karam

okaram@spsu.edu

SOUTHERN POLYTECHNIC STATE UNIVERSITY

Basics

- Structured Query Language
- Standard language for relational databases
- Every RDBMS implements, with slight variations, including
  - –Special data types
  - –Stored procedures
  - –Kinds of indexes, file organization ...

Basics

- Data Definition Language (DDL)
  - CREATE / DROP / ALTER
    - Table
    - View
    - Index, Sequence, ...
- Data Manipulation Language (DML)
  - INSERT
  - UPDATE
  - DELETE
  - SELECT
  - Transactions
- Data Control Language (DCL)

Conventions

- -- comments until end of line
- /* can also use C-style comments */
- SQL is case insensitive (except for data)
- But we usually type reserved words in ALL CAPS
- Use single quotes for 'character constants'
  - –programming quotes, don't let your word processor auto-correct them :) 'wrong'

CREATE TABLE

```
CREATE TABLE table_name (
  field type constraints,
  field2 type2 ,
  CONSTRAINT name ...,
  ...
);
```

```
CREATE TABLE Book (
    ISBN  CHAR(9)      PRIMARY KEY,
    Title VARCHAR(20) UNIQUE NOT NULL,
    Pages Integer
);
```

```
CREATE TABLE Student (
   Id CHAR(3) PRIMARY KEY,
   Name VARCHAR(20) NOT NULL,
   Age INT DEFAULT 20 CHECK(Age>0 AND AGE<100),
   Gender CHAR NOT NULL,
   Deg_code CHAR(2) NOT NULL REFERENCES Degree(code),
   Major CHAR(3),
   credits INTEGER
);
```

Common Datatypes

- CHAR(n)
  - –fixed length strings, padded with spaces at end
- VARCHAR(n)
  - –variable length strings, but no longer than n
- NUMERIC(prec,dec)
  - –fixed precision numbers (not floats)
  - –precision is **total** number of digits
  - –dec is how many after the decimal point
  - –NUMERIC(3,2) max value is 9.99
- DATE, TIMESTAMP
  - –Represent dates, or specific points in time

Common constraints

- PRIMARY KEY
- NOT NULL
- UNIQUE
- REFERENCES (foreign key)
  - after REFERENCES put name of table, then field in parenthesis
  - StudentId REFERENCES Student(Id)
- CHECK
  - Allows for predicates after
  - CHECK(age>20)

INSERT

- INSERT INTO table (fields) VALUES (values)
- Character constants have single quotes ‘a’

```
INSERT INTO Student (Id,Name,Major,Age)
  VALUES (1,'Orlando Karam','CS',21);
```

SELECT (Retrieving Data)

- SELECT fields FROM table WHERE conditions
- Can use fields or expressions (a+3), * for all fields
- Conditions use normal operators (=,>) and are combined with AND, OR, NOT

```
SELECT *
FROM Student
```

```
SELECT Id,Name
FROM Student
WHERE Major='CS'
```

More Examples

```
SELECT Id,Name,Age+5
FROM Student
WHERE Major='CS' AND Gender='F'
```

```
SELECT Id,Name,Age+5
FROM Student
WHERE Age>=20 OR Age <=10
```

DELETE

- DELETE FROM table WHERE conditions
- If no conditions, all data
- Does NOT delete the meta-data, use DROP TABLE for that

```
DELETE
FROM Student
WHERE Id=1
```

UPDATE (change data)

- UPDATE table SET field=value WHERE conditions

```
UPDATE Student
SET Name='Alfredo
Karam', Age=25
WHERE Id=1
```