MySQL

MySQL

- MySQL is a relational database management system (RDBMS)
- We use the MySQL database included in MAMP (Do NOT use MAMP Pro)

Structured Query Language (SQL)

- SQL is a database language for:
  - creating database and table structures
  - performing data manipulation and administration
  - querying the database to extract useful information

- It is a nonprocedural language
  - The user specifies what must be done, but not how
  - Where the data is *actually* stored isn't important

    …only the specified relations and relationships matter

Structured Query Language (SQL)

- All relational DBMS software thus supports SQL
  - Oracle, MySQL, SQL Server, DB2, MS Access
  - Many database vendors have developed extensions
    - Basic, simple vocabulary of < 100 "words"
    - Different "dialects" with minor differences

Categories of SQL commands

- Data Definition Language (DDL)
  - Commands that define a database, including creating, altering, and dropping tables and stored procedures, and establishing constraints
    - CREATE TABLE, set PRIMARY KEY
- Data Manipulation Language (DML)
  - Commands that are used to manipulate data and extract information
    - SELECT, UPDATE, INSERT, DELETE

Data types (cont.)

- MySQL data types (**http://dev.mysql.com/doc/refman/5.1/en/data-types.html**):
  - Primary numeric types:
    - TINYINT -128 to 127 (or 0 to 255) = 1 byte
    - SMALLINT -32768 to 32767 (or 0 to …) = 2 bytes
    - MEDIUMINT -8.39×10^6 to 8.39×10^6 = 3 bytes
    - INT -2.15×10^9 to 2.15×10^9 = 4 bytes
    - BIGINT -9.22×10^{18} to 9.22×10^{18} = 8 bytes
    - DECIMAL(*M, D*) *M* total digits / *D* digits after decimal

MySQL data types (cont.)

- Primary date and time types:
  - DATE 'YYYY-MM-DD' format
    range: '1000-01-01' to '9999-12-31'
  - DATETIME 'YYYY-MM-DD HH:MM:SS' format
    range: '… 00:00:00' to '… 23:59:59'

Invalid dates and times are converted to zero values: '0000-00-00'

Some built-in functions:

NOW( ), CURDATE( ), DATEDIFF( ), INTERVAL
DATE( ), TIME( ), DAY( ), YEAR( ), MONTH( ), etc.

MySQL data types (cont.)

- Primary string types:
  - CHAR(*n*) - always allocates *n* bytes of storage
  - VARCHAR(*n*) - only allocates used space (plus 1 byte)

  Values that exceed *n* characters in length are truncated

  - BLOB - used for large binary strings of data
  - TEXT - used for large character strings of data
  - ENUM - string object with a value chosen from a list of permitted values that is specified at table creation time

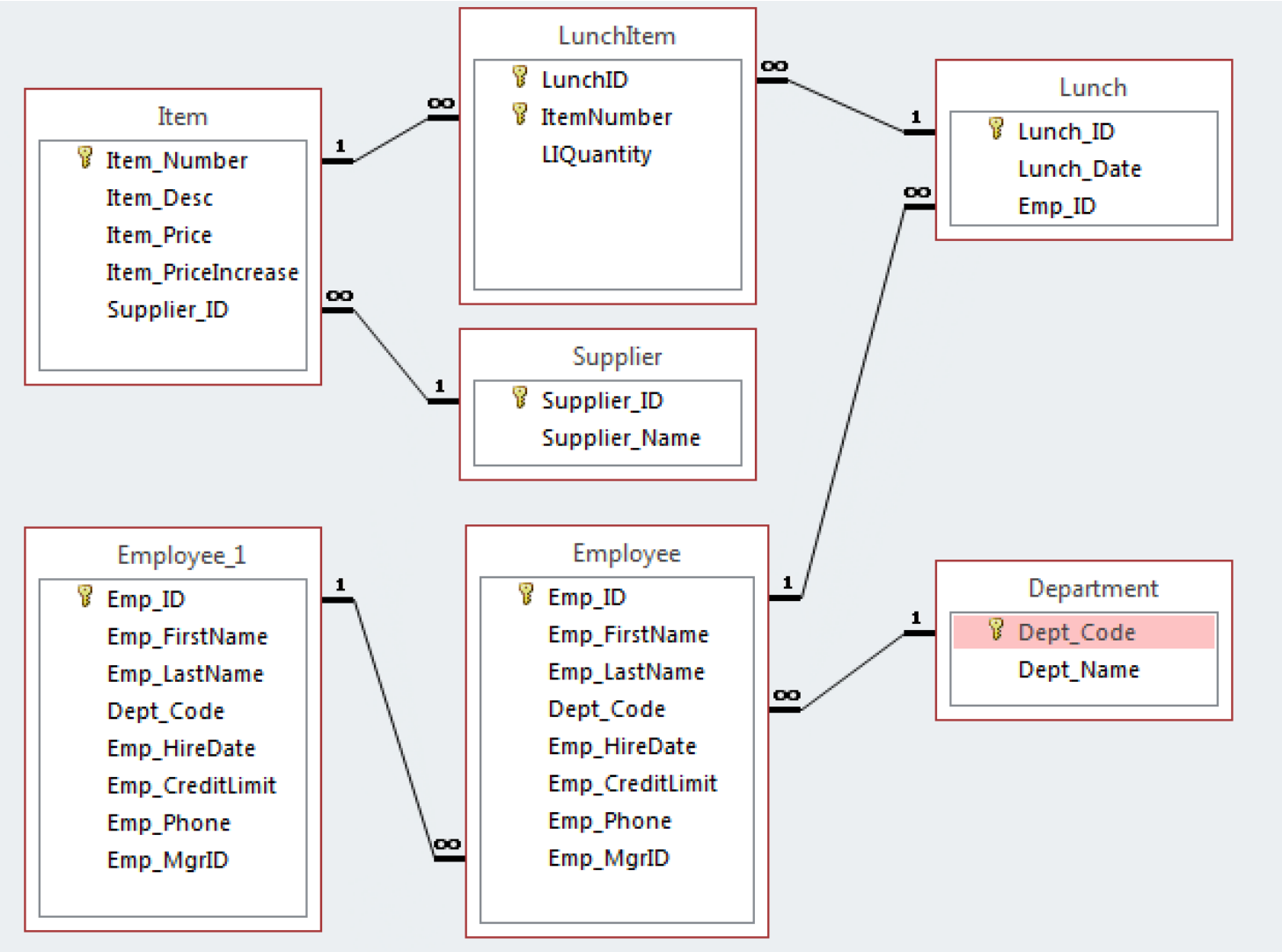
Sample Database

- **Posted on Canvas:**
  - **lunchDB.sql** (MySQL 5.0)
  - We will be running examples of queries on this database as we cover the material in class, to help illustrate how the different SQL statements work.

Sample Database

- 6 Tables:

(Employee_1 is a replica of the Employee table for showing the self-referencing foreign key relationship)



Demo 1: Create and Import a database

- Create a database from scratch
- Import a .sql database dump to MySQL
- Browse the database
- Create a new user and modify user privileges
- Assign auto-incremental attributes

Data Manipulation Language (DML)

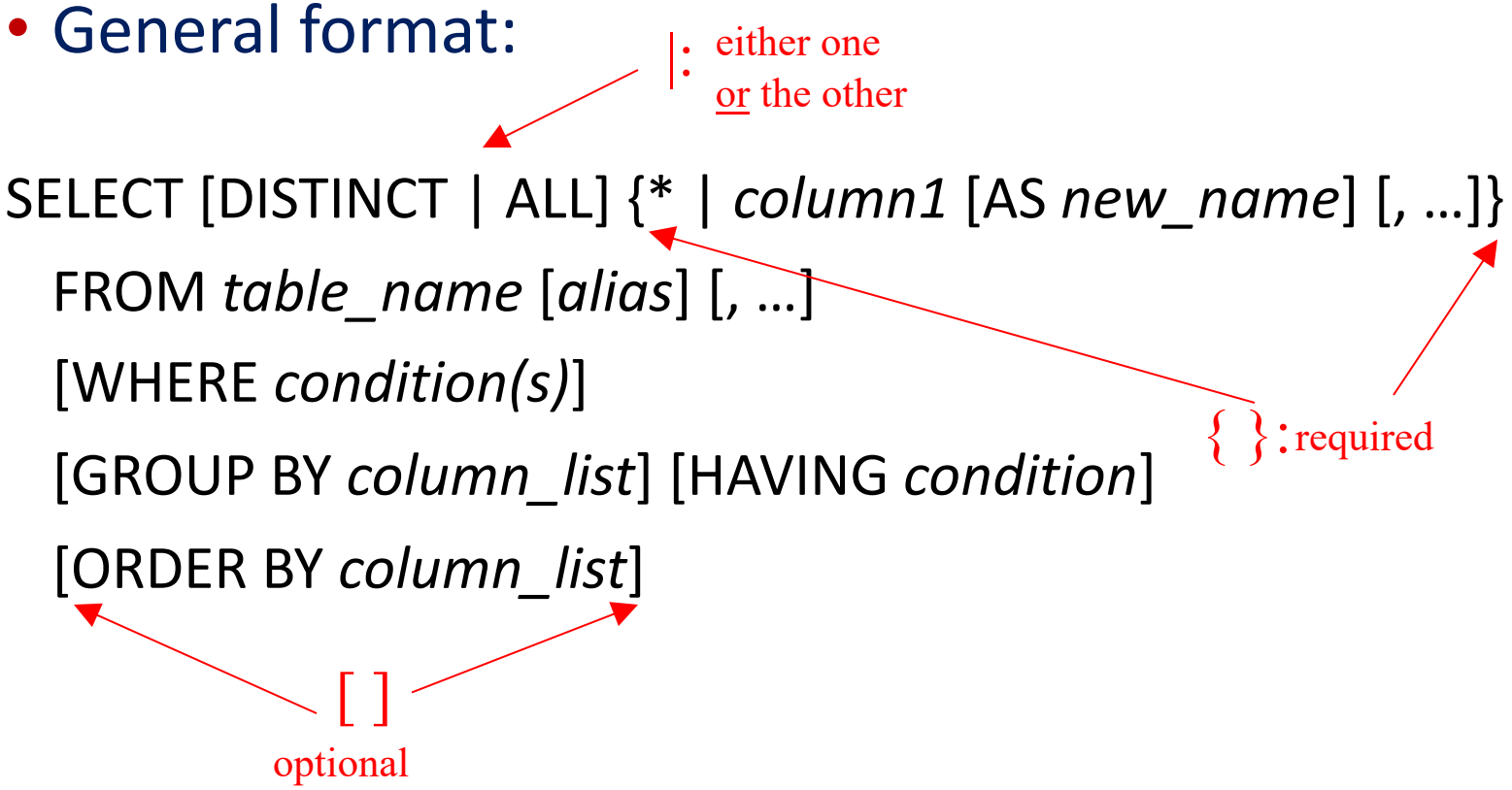
- Primary commands:
  - Data retrieval ("standard" queries):
    - SELECT
  - Action queries:
    - INSERT
    - UPDATE
    - DELETE

SQL Statements in General

- SQL statements are case-insensitive
- Enclose non-numeric values (string, date/time) in single quotation marks;
- Valid data/time format: 'yyyy-mm-dd' or 'yyyy-mm-dd hh:mm:ss'

SELECT statement

- General format:



SELECT Statements Review

1. List the full details of every employee;
2. List the first and last name of every employee with a credit limit less than $25;
3. List the first and last name of all employees who were hired in 1997;
4. List the ID of every employee whose last name is four letters long and ends in 'e'; (Hint: % matches 0 or more characters; _ matches exactly one character)
5. List all employees who are in the department of 'Sal' or 'Mkt';
6. List all employees who do not have a manager;

SELECT Statements Review

7. List all employee last names and dept codes from departments whose codes begin with 'S'. Sort the list in descending order of departmental code and ascending order of employee last name;
8. List all employee last names together with twice their credit limit, where this second column is named "double credit";
9. List all unique managers' IDs;
10. Count the number of employees hired after 1996;
11. Count the number of managers;
12. For each dept_code, list the number of employees as count and the total credit limit as total. Sort the list by the total credit limit in ascending order;
13. List each employee's first name, last name then 'works in the' Dept_Code and 'dept'. Name the Column "Employee Assignment";

INSERT statement

- General format:

```
INSERT INTO tablename(columnName1, columnName2,…)
   VALUES(value1, value2,…)
```

UPDATE statement

- General format:

UPDATE tablename

SET columnName1=value1, columnName2=value2,…

WHERE criteria

DELETE statement

- General format:

DELETE FROM tablename

WHERE criteria

INSERT/UPDATE/DELETE Statements Review

1. Add a new lunch item (item_number: 11, item_desc: House Special Salad, item_price: 4.5, item_priceincrease: 0, supplier_ID: Vsp);
2. Increase the credit limit by 20% of all employees whose current credit limit is less than $20;
3. Delete the employee records for those who work for the 'Shp' department.