cis20.2
design and implementation of software applications 2
spring 2010
lecture # II.2

today's topics:

- relational databases
- SQL

relational databases

- a *relational* database consists of multiple *tables*
- each table is defined as having a number of *fields*
- data is *stored* in a table so that a single entry in a table, called a *record*, provides one data element for each field
- a table can be thought of as a spreadsheet, where the *fields* are *columns* in the spreadsheet, and the *records* are *rows*
- records can have "unique" fields, which are called *keys*
- if a record does not have a value for a particular field, then a *NULL* value is entered
- "relational" databases consist of multiple tables that *relate* to each other by having one column (field) in common

SQL

- SQL = Structured Query Language
- **MySQL** is a free database management system (DBMS) that implements SQL `http://www.mysql.com`
- basic *data definition* commands:
  - `CREATE` — to create a table
  - `DESCRIBE` — to describe a table's definition
  - `DROP` — to delete a table
- basic *data manipulation* commands:
  - `INSERT` — to put data into a table
  - `SELECT` — to see what is in a table
  - `UPDATE` — to edit data that is already in a table
  - `DELETE` — to remove data from a table

example: create, describe, insert, select

- **tblUser**

userID	lastname	firstname
1	sklar	elizabeth
2	mouse	mickey
3	mouse	minnie
4	potter	harry

- The userID uniquely identifies a single person in the **tblUser** table.
- **tblBday**

bdayID	month	day
1	12	11
2	10	9
3	8	7

- The bdayID uniquely identifies a single birth date in the **tblBday** table.
- These are connected using a "relation" called **rltUserBday**.

- **rltUserBday**

userID	bdayID
1	1
2	2
3	3
4	1

- The **rltUserBday** table is used to *join* the **tblUser** table to the **tblBday** table in order to look up a person's birthday.
- Note that users with userID=1 and userID=4 have the same birthday!
- Here is the "join" command in mysql:

```
SELECT *
FROM   tblUser, tblBday, rltUserBday
WHERE  tblUser.userID=rltUserBday.userID
AND    tblBday.bdayID=rltUserBday.bdayID;
```


- Here are the commands to generate the example.
  Note the convention of putting MySQL keywords in all CAPS.
- First, create the user table:

```
mysql> CREATE TABLE tblUser (
userID INT(11) NOT NULL PRIMARY KEY AUTO_INCREMENT,
lastname TEXT,
firstname TEXT);
```

  Note the keywords:
  - "NOT NULL" means that the field can never be null (empty)
  - "PRIMARY KEY" means that the field must be unique
  - "AUTO_INCREMENT" means that MySQL will generate a unique key automatically

- Second, create the birthday table:

```
mysql> CREATE TABLE tblBday (
bdayID INT(11) NOT NULL PRIMARY KEY AUTO_INCREMENT,
month INT,
day INT);
```


- Third, create the relation:

```
mysql> CREATE TABLE rltUserBday (
userID INT(11) NOT NULL PRIMARY KEY,
bdayID INT(11) NOT NULL);
```

- Now look at your tables:

```
mysql> SHOW TABLES;
```

  to get a list of all the tables in your database

```
mysql> DESCRIBE tblUser;
```

  to look at the definition of the user table

- Next, put data into your user table using the "INSERT" command:

```
mysql> INSERT INTO tblUser (lastname, firstname)
VALUES ('sklar', 'elizabeth');

mysql> INSERT INTO tblUser (lastname, firstname)
VALUES ('mouse', 'mickey');

mysql> INSERT INTO tblUser (lastname, firstname)
VALUES ('mouse', 'minnie');

mysql> INSERT INTO tblUser (lastname, firstname)
VALUES ('potter', 'harry');
```

  and look at your data:

```
mysql> SELECT * FROM tblUser;
```

- Then, put data into your bday table:

```
mysql> INSERT INTO tblBday (month, day)
VALUES (12, 11);

mysql> INSERT INTO tblBday (month, day)
VALUES (10, 9);

mysql> INSERT INTO tblBday (month, day)
VALUES (8, 7);
```

and look at your data:

```
mysql> SELECT * FROM tblBday;
```


- Then, populate the user-bday relation. You can do this manually:

```
mysql> INSERT INTO rltUserBday (userID,bdayID) VALUES (1,1);
```

which requires that you know what the values of `userID` and `bdayID` are

- You can also do this with a query that looks up the ID values and inserts them automatically into the relation:

```
INSERT INTO rltUserBday (userID,bdayID)
SELECT userID,bdayID
FROM tblUser,tblBday
WHERE lastname='sklar'
AND firstname='elizabeth'
AND month=1
AND day=1;
```


updating table values

- use the `UPDATE` command to change the values in a table

```
UPDATE tblUser
SET lastname='mantle'
WHERE firstname='mickey';
```


deleting entries from tables

- use the `DELETE` command to remove an entry (row) from a table

```
mysql> DELETE
FROM tblUser
WHERE lastname='mouse';
```

- WARNING: be careful when deleting! I always run a `SELECT` command first, to make sure that I am deleting the row(s) that I wanted:

```
mysql> SELECT *
FROM tblUser
WHERE lastname='mouse';

mysql> DELETE
FROM tblUser
WHERE lastname='mouse';
```

deleting tables

- you can delete an entire table using the DROP command

```
mysql> DROP TABLE tblUser;
```