Web System Development with Ruby on Rails

Day 6(25/Oct/2011)

Database Language: SQL

Today's Theme

Learn What is Database

- Learn database language: SQL

- Add one table to the project

Is this too easy, and boring?

If you think that this lecture is so easy to understand that you guess what to do before the lecturer explains…

Then, please kick off another WEB application project of your own design with your originality. It may help you raise a new question, how to write program to realize certain design of your own. (This will be added to your score.)

Please do not waste your time, raise new questions and make the most of this class.

What is database?

- Collection of data composed specially for computer to read and write easily.

- What is 'Data'
  - Information managed by electronic and/or magnetic signals; Pictures, programs, music, and literatures are all data when they are transformed in the figure of electronically accessible form.

Ain't “File” enough?

- What is “File”
  - Collection of data on the computer, which is packed so that it could be accessed as one unit.

- What differs between Database and File?
  - Name differs! Of course! Is that all?
  - When files gather, they turn to become database? Is that so?

Problems of using files

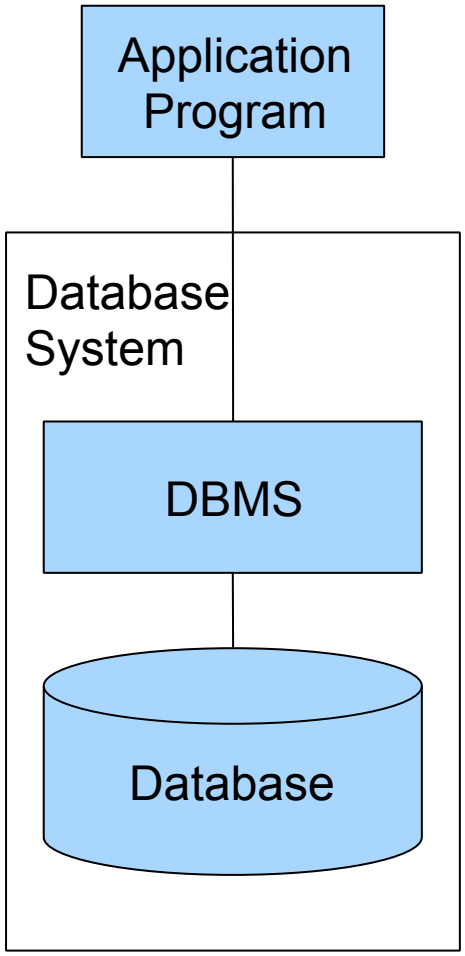
- If data is modified, all the associated files should be updated.
  - Complex, if information is linked together.
  - If some are left un-updated, contents would become different from file to file. It will cause the lack of “integrity”
- Waste of storage
  - Independent files require independent management area. They should be packed.

Solving Problems of separate files

- Fundamental Solution
  - Sharing data, recorded in one place
- Dogma:
  - One fact in one place.
  - One fact in Only one place.

DBMS

- DBMS stands for "Database Management System"
- DBMS and DB are put together to make "Database System"



SQL is a DB Manipulation Language

- SQL
  - Structured Query Language
  - SQL is used to access and manipulate database for other application programs and/or users.
  - SQL is a kind of DML(Data Manipulation Language)

- SQLite3 is one of installation of SQL
  - Oracle, SQL Server, DB2 are famous
  - PostgreSQL and MySQL are “freeware”

Modeling Data

- What does it mean "Data modeling?"
  - Describe data structure
    - Relations between properties
  - Describe Constraint for Conformance
    - Conformance – Having no contradiction between components
  - Describe how to manipulate data
    - Separate input fields and calculated fields
- Define Schema by data modeling

What is Schema

- Schema
  - Data structure obtained by describing application data model
  - Relations between components of data
- Three Layer Schema model:
  - External Schema, Conceptual Schema, and Internal Schema

OR Mapping

- Object Relational Mapping
  - In Object-Oriented Language, Class structure can be directly linked to database
  - Available on the platform such as Rails
  - But, sometimes, Database Structure does not match with object oriented Program's Class.

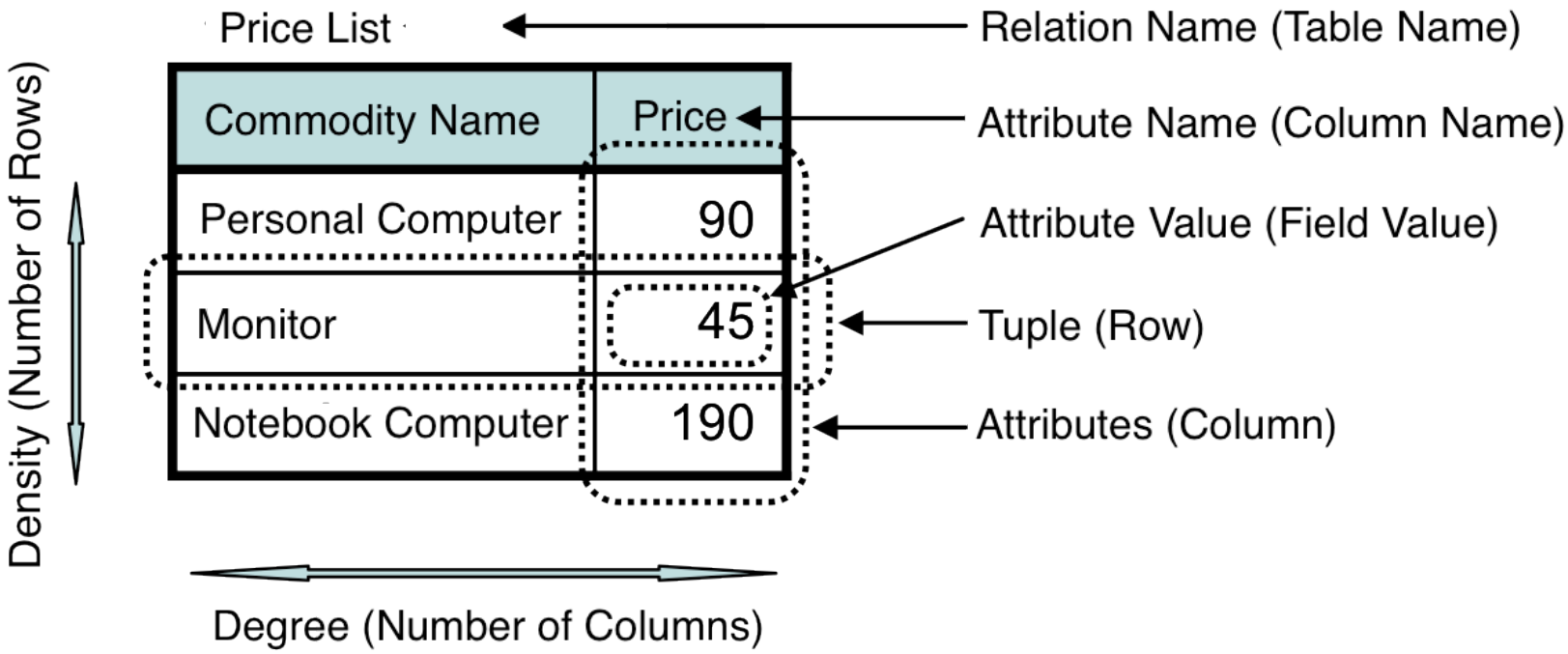
  OR Mapping is to solve this problem.
- In "Ruby on Rails", Model = Class
  - Class – Grammar
  - Model – Structure
    - Designing Schema has directly become Ruby Class.
    - Here we start with simple structure.

DB and Tables

- Information is stored in Tables
- Tables are called as “Relations”, in RDB; Relational Database
  - Tables have Columns(Attributes) [vertical]
  - Records are stored in Tapples(lines) [horizontal]
    - Instances/ records

Relation(Table) Structure

- There are Definitions in “Relational DB”, and popular names in SQL, enclosed with ().



Primary Key

- Simply think as ID (Identifier)
  - Such as “Membership Code”, “Student Number”
- It should be UNIQUE
  - Unique – means “only one, and identical”
  - No other data has the same primary key value
  - No other student has the same “Student Number” with you!

Start and Stop using MySQL

- Type the following command to run MySQL
  - Mysql -u root
- -u option followed by user name
- When password is set for the user to use MySQL, -p option should be added
  - Mysql -u root -p
  - If you forget the password for “root”, it is just a tragedy!
    So we do not set password for the lecture environment, but for the real service environment, password should be set which you NEVER forget!
- When you stop using mysql, simply type “exit” on the mysql prompt;
  - Mysql >> exit

We do not use MySQL this year.

Three Project Sub-versions in sqlite3

There are three project sub-versions.

- Development --- Logic Development
- Test --- Test Run
- Product --- Commercial Version

There are three different database files.

- development.sqlite3
- test.sqlite3
- product.sqlite3

Starting and Stop using sqlite3

Move to (Project root)/db, i.e. memopad/db directory, then type

sqlite3 development.sqlite3

To see the command list of sqlite3, type

.help

To exit from sqlite3, type

.exit

```
[root@cisnote memopad]# pwd
/home/rails3work2/memopad
[root@cisnote memopad]# ls
app     config.ru  doc      Gemfile.lock  log     Rakefile     script  tmp
config  db         Gemfile  lib           public  README.rdoc  test    vendor
[root@cisnote memopad]# cd db
[root@cisnote db]# ls
development.sqlite3  migrate  schema.rb  seeds.rb
[root@cisnote db]# sqlite3 development.sqlite3
SQLite version 3.3.6
Enter ".help" for instructions
sqlite> .exit
[root@cisnote db]#
```

Sqlite3 Command (1/2)

```
SQLite version 3.7.11 2012-03-20 11:35:50
Enter ".help" for instructions
Enter SQL statements terminated with a ";"
sqlite> .help
.backup ?DB? FILE      Backup DB (default "main") to FILE
.bail ON|OFF           Stop after hitting an error.  Default OFF
.databases             List names and files of attached databases
.dump ?TABLE? ...      Dump the database in an SQL text format
                       If TABLE specified, only dump tables matching
                       LIKE pattern TABLE.
.echo ON|OFF           Turn command echo on or off
.exit                  Exit this program
.explain ?ON|OFF?      Turn output mode suitable for EXPLAIN on or off.
                       With no args, it turns EXPLAIN on.
.header(s) ON|OFF      Turn display of headers on or off
.help                  Show this message
.import FILE TABLE     Import data from FILE into TABLE
.indices ?TABLE?       Show names of all indices
                       If TABLE specified, only show indices for tables
                       matching LIKE pattern TABLE.
.load FILE ?ENTRY?     Load an extension library
.log FILE|off          Turn logging on or off.  FILE can be stderr/stdout
.mode MODE ?TABLE?     Set output mode where MODE is one of:
                       csv      Comma-separated values
                       column   Left-aligned columns.  (See .width)
                       html     HTML <table> code
```

Sqlite3 Command (2/2)

```
                    insert   SQL insert statements for TABLE
                    line     One value per line
                    list     Values delimited by .separator string
                    tabs     Tab-separated values
                    tcl      TCL list elements
.nullvalue STRING      Print STRING in place of NULL values
.output FILENAME       Send output to FILENAME
.output stdout         Send output to the screen
.prompt MAIN CONTINUE  Replace the standard prompts
.quit                  Exit this program
.read FILENAME         Execute SQL in FILENAME
.restore ?DB? FILE     Restore content of DB (default "main") from FILE
.schema ?TABLE?        Show the CREATE statements
                    If TABLE specified, only show tables matching
                    LIKE pattern TABLE.
.separator STRING      Change separator used by output mode and .import
.show                  Show the current values for various settings
.stats ON|OFF          Turn stats on or off
.tables ?TABLE?        List names of tables
                    If TABLE specified, only list tables matching
                    LIKE pattern TABLE.
.timeout MS            Try opening locked tables for MS milliseconds
.vfsname ?AUX?         Print the name of the VFS stack
.width NUM1 NUM2 ...   Set column widths for "column" mode
.timer ON|OFF          Turn the CPU timer measurement on or off
sqlite>
```

“Database” in MySQL

When we use MySQL, once we login to mysql as root user, we can access to all ‘Databases’ on that computer. On the other hand, when we use SQLite3, we start using sqlite3 by opening one database file.

So, in using MySQL, we create/remove database in MySQL data storage, and we connect or use database.

In SQLite3, we had already chosen one database when we open one sqlite3 file.

Create and Remove Database in MySQL

- Creating Database
  - Mysql >> create database [DB_name];
  - ex: create database friendDB;
    - Here, the database “friendDB” is a large container for all “fiends'” tables.
- Remove Database
  - MySQL >> drop database [DB_name];
  - ex: drop database friendDB;
  - Ultimately, be careful, this command simply erase everything!
- Show the list of databases
  - mysql >> show databases;

Declare using database in MySQL

- mysql >> use [DB_name];
- ex： use friendDB;

- The Command, “connect [DB_name]” too has the same result.

Another way to use SQLite easily

Use SQLite Database Browser.

Go to

http://sqlitebrowser.sourceforge.net/index.html

for more information.

But, for the learning purpose, we dare learn typing SQL commands.

Create Table (1/2)

- sql >> create table TableName(
  ColName [Data Type] [Attributes],
  );
  Repeat
  - ColumnName [Data Type] [Attribute]

  the times of number of columns
- Data Type : int, decimal, text, varchar, datetime, timestamp, blob
- Column Attributes :
  primary key, not null

Creating Table(2/2)

Design a table which contains only member ID and member name.

```
create table MemberT(
MemberID  char(6) primary key,
MemberName  char(16) not null );
```

Check the result of above command for MySQL.

```
show columns from MemberT;   (on mySQL)
.schema MemberT    (on SQLite3)
```

```
+------------+----------+------+-----+---------+-------+
| Field      | Type     | Null | Key | Default | Extra |
+------------+----------+------+-----+---------+-------+
| MemberID   | char(6)  | NO   | PRI | NULL    |       |
| MemberName | char(16) | NO   |     | NULL    |       |
+------------+----------+------+-----+---------+-------+
```

See if tables exist

- See the list of the table (in the selected database)
  - sqlite >> .tables
  - Mysql >> show tables;
- If you find the table already exists, then you may need to check the columns of the table.
  - mysql >> show columns from TableName;

Storage Classes and Datatypes

Each value stored in an SQLite database (or manipulated by the database engine) has one of the following storage classes:

NULL. The value is a NULL value.

INTEGER. The value is a signed integer, stored in 1, 2, 3, 4, 6, or 8 bytes depending on the magnitude of the value.

REAL. The value is a floating point value, stored as an 8-byte IEEE floating point number.

TEXT. The value is a text string, stored using the database encoding (UTF-8, UTF-16BE or UTF-16LE).

BLOB. The value is a blob of data, stored exactly as it was input.

http://www.sqlite.org/datatype3.html

rails/ mysql / ruby

- In ruby on rails, SQL types for the migration, and the ruby Class (types) are listed in the next page.

- MySQL may need the maximal digits for such types as int and varchar.
  - In ruby language, Class Description start with Capital charcters. (same with Java)

Supported Datatypes

:binary

:boolean

:date

:datetime

:decimal

:float

:integer

:primary_key

:string

:text

:time

:timestamp

These will be mapped onto an appropriate underlying database type.

See the table contents

- Use Select Statement
  - If you can write SQL “select” statements to obtain data with complex conditions, it means that you have intermediate skill of database.
- If we only want to show all fields from all records of a table, then type;
  - sql >> select * from (TableName);
  - ex.: select * from friends;
- *(asterisk) is a wild card to see all columns. We usually list the column names we want to see.

Why not trying the following site?
http://www.1keydata.com/sql/sql.html

Write data into table (1/2)

- sql >> insert into tableName

  values (column1Value, column2Value ･･･);

  We list the column values in the order we first created the table, up to the number of columns we described in the table declaration.

- We can also write insert statement as the following sample. In first (), column names are listed.
- **INSERT INTO Apparel_Store (name, phone) VALUES ('Shimamura', '+81-493-72-XXXX');**

Write data into table (2/2)

To insert the member data into the member table, created in the saveral pages before;

```
insert into MemberT values (
      'A001', 'Aoyama' );
insert into MemberT values (
      'B002', 'Konaka' );
```

To see the insersion result, type;

```
select * from MemberT;
```

```
+------------+-------------+
| MemberID   | MemberName  |
+------------+-------------+
| A001       | Aoyama      |
| B002       | Konaka      |
+------------+-------------+
```

```
+------------+----------+------+-----+---------+-------+
| Field      | Type     | Null | Key | Default | Extra |
+------------+----------+------+-----+---------+-------+
| MemberID   | char(6)  | NO   | PRI | NULL    |       |
| MemberName | char(16) | NO   |     | NULL    |       |
+------------+----------+------+-----+---------+-------+
```

Update Data

- Sql >> update "table_name”
  set ”column1" = [value1], “column2” = [value2]..
  where {condition} ;
- Modify the rows that satisfy the {condition} clause so that the field columns in the set clause to have the values.
- **Important Notice:** If you forget writing where clause, update command replaces the column value of all records in the table. (And we cannot restore!)

Before Object Oriented Database

We embedded the scripts to control the database in SQL in the WEB programs.

There were no automated process to access database, and we had to write SQL statements for every steps that we access the database to store, fetch, search, and delete information.

In ruby too, the database access phases are almost the same, but we do not have to write SQL at all. (I want you to read SQL.)

Today's Theme

- Add one table: Category
  - Memo has their contents in memos table.
  - Category of memos are stored in categories.
- We give One to Many relationship from categories to memos next week.

Register Categories

- Think the categories of your memos.
  - Internationalize (Localize?) the WEB page labels and link names, into your mother tongue.
  - Those ‘improvements’ of your WEB application may be considered as the points to be added to your score.

- Memos have categories; ‘Idea’, ‘To Do’, ‘Phone’, ‘Dinner’, ‘Party’, and such.

Generate Category table

Type the following command;

```
rails g scaffold category name:string
```

Then, we create the database

```
rake db:migrate
```

Please refer to the page 23 & 24, in the slides of Day 2.

No replacing scaffolds.css.scss

```
[root@cisnote memopad]# pwd
/home/rails3work2/memopad
[root@cisnote memopad]# rails g scaffold category name:string
               active_record
      create     db/migrate/20121023061740_create_categories.rb
      create     app/models/category.rb
                 test_unit
      create       test/unit/category_test.rb
      create       test/fixtures/categories.yml
               resource_route
       route     resources :categories
               scaffold_controller
      create     app/controllers/categories_controller.rb
                 erb
      create       app/views/categories
      create       app/views/categories/index.html.erb
      create       app/views/categories/edit.html.erb
      create       app/views/categories/show.html.erb
      create       app/views/categories/new.html.erb
      create       app/views/categories/_form.html.erb
                 test_unit
                  coffee
       create       app/assets/javascripts/categories.js.coffee
                  scss
       create       app/assets/stylesheets/categories.css.scss
                 scss
    conflict     app/assets/stylesheets/scaffolds.css.scss
  Overwrite /home/rails3work2/memopad/app/assets/stylesheets/scaffolds.css.scss?
 (enter "h" for help) [Ynaqdh] n
        skip     app/assets/stylesheets/scaffolds.css.scss
[root@cisnote memopad]#
```

Test run and register data

Complete the migration, then next, run the server

rails s

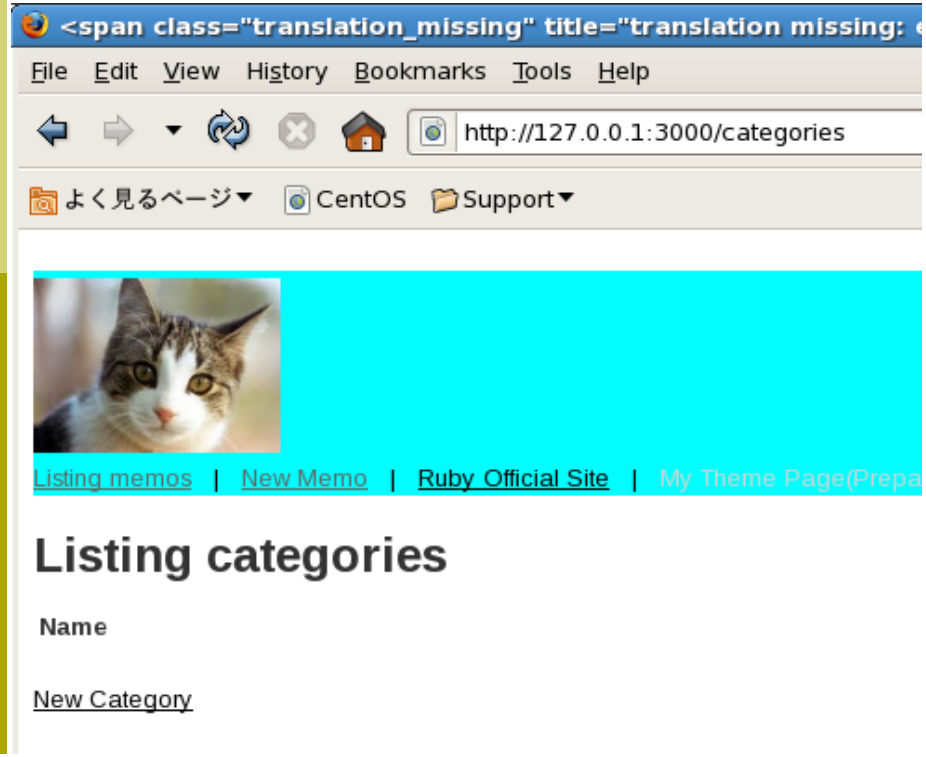
And add some data into categories

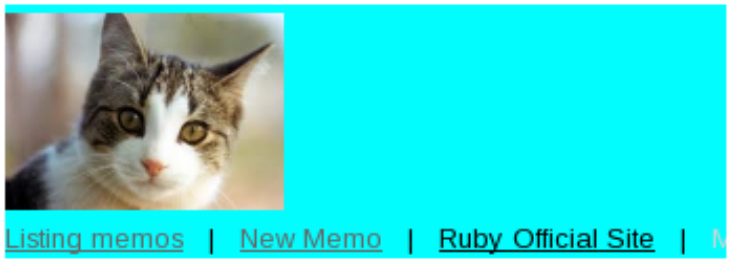
```
[root@cisnote memopad]# rake db:migrate
==  CreateCategories: migrating ================================================
-- create_table(:categories)
   -> 0.0254s
==  CreateCategories: migrated (0.0256s) =======================================

[root@cisnote memopad]#
```

Categories program screen

Like memos listing screen, we have whole set of screens to add, edit, show and remove the categories.





Check where database is created.

- In the sqlite3 database, the database categories should be generated.
  - Move to memopad/db, and find development.sqlite3 file.
  - This is the database file for memopad project.

- In the GNOME terminal, type

  sqlite3 development.sqlite3

Type SQL command in sqlite3

- When you typed SQL command, do not forget to add ;(semicolon) at the last of the sentence.
- Type

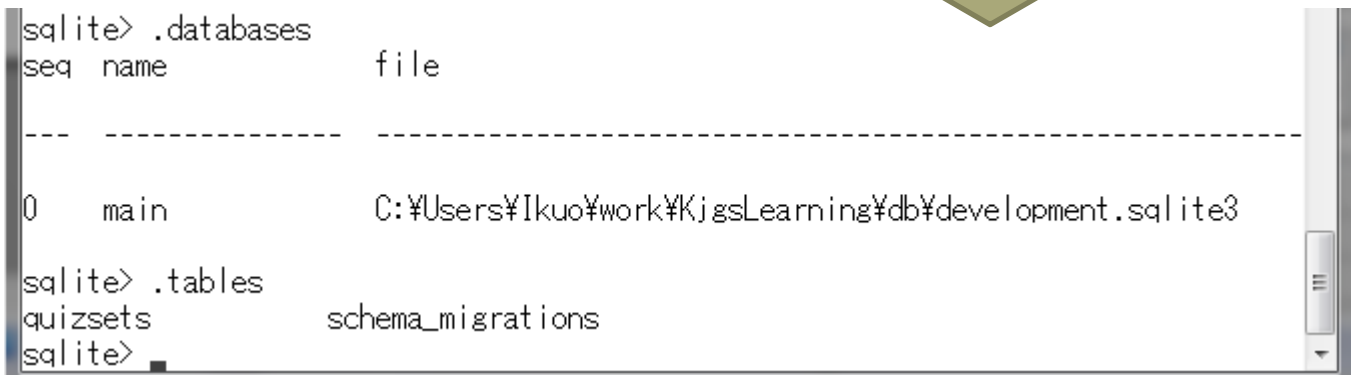
.tables

to see memos and categories tables there. Type

select * from categories;

To see the contents of categories;

Screen shots の更新



Check the schema of categories

- Type

  .schema categories

to see the schema of categories.

- When we generated the table, we did not specified the field ID, but it is generated.
- It is important to set relationship.
  - Now, if we add the field ‘category_id in the other table, it will become the relation to the table ‘categories.’ (singular_id)

Sqlite3 commands

```
[root@cisnote memopad]# cd db
[root@cisnote db]# sqlite3 development.sqlite3
SQLite version 3.3.6
Enter ".help" for instructions
sqlite> .tables
categories          memos               schema_migrations
sqlite> select * from categories
   ...> ;
1|To Do|2012-10-23 06:35:26.155296|2012-10-23 06:35:26.155296
2|Idea|2012-10-23 06:35:39.686580|2012-10-23 06:35:39.686580
3|Phone message|2012-10-23 06:36:07.151289|2012-10-23 06:36:07.151289
4|Dinner|2012-10-23 06:36:18.886586|2012-10-23 06:36:18.886586
sqlite> .schema categories
CREATE TABLE "categories" ("id" INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, "nam
e" varchar(255), "created_at" datetime NOT NULL, "updated_at" datetime NOT NULL)
;
sqlite>
```

Singular form and Plural form

When we typed table generation command, we specified memo in singular form.

The generated table's name was plural form, memos.

In ruby on rails environment, plural forms have significant grammatical meaning, so please be careful if the word is in singular form or plural form.

Irb (interactive ruby)

Run irb, and then type the following commands.

We can convert the form of nouns.

```
1: require “rubygems”
2: require “active_support/inflector”
3: puts “ox”.pluralize
4: puts “data”.singularize
```

Continued to the next week…

We have generated a new table, ‘categories.’

We give relationship to memos and categories next week.

And our original plan to ask report submission this week is postponed to the next week. So, no report theme today.

In the report of next week, I ask you the screenshots of your project. If you generate your ORIGINAL project, you are most welcome to do so.

If you were absent for the day…

- Generate categories table, and add some data.
- Report the screenshots of categories listing screen, and the screenshots of sqlite3 commands and responses in the terminal.