

Final Report for DB Term Project



담 당 : 백두권 교수님

학 과 : 컴퓨터학과

2000160166	선 태 봉
2000160184	이 진 무
2000160197	하 동 욱

■ 주제

다이어트를 필요로 하는 사람들을 위한 체계적인 섭취 칼로리와 소비 칼로리 관리

■ 팀 구성원

컴퓨터학과 2000160166 선 태 봉

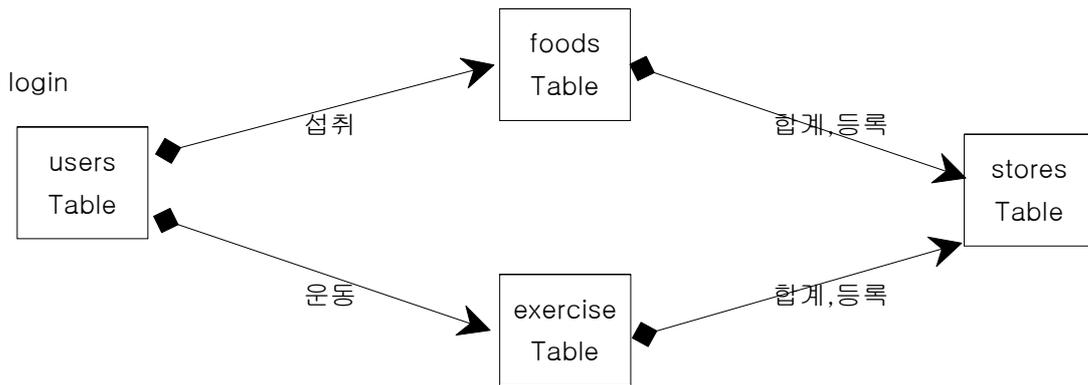
컴퓨터학과 2000160184 이 진 무

컴퓨터학과 2000160197 하 동 욱

■ 팀명

광란의 다이어트 (crazy diet)

■ Entity Relation Diagram



■ Attributes

1. users table

-uid(user id), passwd(user password), name(user name), sex, birthyear, height, weight

2. foods table

-name(음식의 이름), kcal(기준단위당 칼로리), ate(해당음식의 섭취량), type(음식의 종류)

3. exercises table

-name(운동의 이름), kcal(기준시간당 칼로리), done(얼마나 운동을 했나)

4. stores table

-date(날짜), intakes(총 섭취 칼로리), exers(총 소모 칼로리), height(키), weight(몸무게), memo(간단한 메모)

■ Relational Database

1.login

uid	passwd	name	birthyear	sex	weight	height
gm0707	1234	이진무	1981	m	80	180

2.Foods

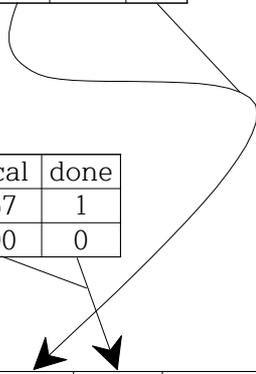
name	kcal	type	ate
김치볶음밥	450	1	1
김치찌개	157	2	0

3.Exercises

name	kcal	done
산보	67	1
걷기	90	0

4.Stores

date	intakes	exers	height	weight	memo
2002-06-03	450	67	180	80	



■ Development Environment

-Server : klug.korea.ac.kr (IP 163.152.44.26)

OS- Linux RedHat 7.2, CPU- Pentium3 800, SQL- MySQL 3.2.3,
Apache Web Server, DB controlling- phpMyAdmin 2.2.6

-Clients: 247, 246호 pc's

OS- Microsoft Windows 2000 Professional, CPU- Pentium3,4,
Web Editor- Macromedia Dreamweaver4, Terminal- Putty,
Graphic- Adobe Photoshop6.0
Monitor Resolution 1024*768

■ Team원별 분담 내용

	선 태 봉	이 진 무	하 동 욱
역할	ERD design database 관련사항 RDB 설계 재료 입력 form design 음식 입력 form design login form design Searching form design PHP 관련사항 Interface design	중간 presentation database 관련작업 MySQL database 구축 PHP, MySQL연동 관련 phpMyAdmin 관련 새로운 item 입력 form design Documentation	RDB 설계 관련 프로그램 벤치마킹 일정, 업무분담계획 하루 칼로리 섭취, 소비입력 form design login form design Analysis form design PHP 관련사항 Interface design

■ Architecture

-원래의 계획은 Microsoft Visual C++과 Microsoft Access를 이용하여서 Windows Application program을 만들려는 계획이었으나, 이 project는 성격상 web을 이용하는 것이 타당하다고 생각되어서 바꾸게 되었다.

-전형적인 조합인 Apache + PHP + MySQL을 이용하였다.

-사이트주소 : <http://klug.korea.ac.kr/~gm0707>

@web page의 기본 구성

```

index.php - this.php + main.php + intakemenu.php
              |
              + consume.php
              |
              + new.php
              |
              + modify.php
              |
              + search.php
              |
              + analysis.php
              + register.php
    
```

-사용자가 등록을 하면 기본적으로 등록되어 있는 음식과 운동의 종류를 가진 데이터베이스가 생성된다.

this.php => secret.php (로그인)

(사용SQL)

```
UPDATE ".$uid."_foods set ate = 0
```

```
UPDATE ".$uid."_exers set done = 0
```

```
SELECT exers, intakes FROM ".$uid."_stores WHERE date = '$today'
```

this.php => register.php (새로 가입)

(사용SQL)

user table에 새로운 사용자의 정보를 입력한다.

```
INSERT INTO users (uid, passwd, name, sex, birthyear ) values(".$uid.",
"$.passwd.", "$.name.", "$.sex.", "$.birthyear.")
```

새로운 사용자의 foods table을 만들고 기존의 음식 정보들을 삽입한다.

```
CREATE TABLE `crazy_diet`.`.$uid."_foods` (`name` varchar(20) NOT NULL default
",`kcal` int(11) NOT NULL default '0',`type` int(11) NOT NULL default '0',`ate` int(11)
NOT NULL default '0',PRIMARY KEY (`name`)) TYPE=MyISAM
INSERT INTO `crazy_diet`.`.$uid."_foods` SELECT * FROM `crazy_diet`.`foods`
```

새로운 사용자의 exercises table을 만들고 기존의 운동 정보들을 삽입한다.

```
CREATE TABLE `crazy_diet`.`.$uid."_exers` (`name` varchar(20) NOT NULL default
",`kcal` int(11) NOT NULL default '0',`done` int(11) NOT NULL default '0',PRIMARY
KEY (`name`)) TYPE=MyISAM
INSERT INTO `crazy_diet`.`.$uid."_exers` SELECT * FROM `crazy_diet`.`exercises`
```

새로운 사용자의 stores table을 만든다.

```
CREATE TABLE `crazy_diet`.`.$uid."_stores` (`date` varchar(20) NOT NULL default
",`intakes` text NOT NULL,`exers` text NOT NULL,`height` int(3) NOT NULL default
'0',`weight` int(3) NOT NULL default '0',`memo` text NOT NULL, PRIMARY KEY
(`date`)) TYPE=MyISAM
```

-처음 사이트에 접속을 하면 로그인화면이 나오고 거기서 받은 아이디를 가지고 사용자의 데이터베이스로 접속을 시킨다.

-사용자는 Intake, Consume창에서 자신이 먹은 음식과 그 양과 한 운동의 종류와 시간을 입력할 수 있다.

음식의 종류를 선택하면 해당 음식들이 database에서 나온다. 그리고 사용자가 음식의 종류를 선택하고 섭취량을 입력하면 database의 ate field에 저장한다.

intakemenu.php => intake.php => update_input.php

(사용SQL)

```
SELECT name, kcal, ate FROM ".$uid."_foods WHERE ate != 0
SELECT name FROM ".$uid."_foods WHERE type=$foods
UPDATE ".$uid."_foods SET ate = $amount WHERE name = '$food'
```

사용자가 지우고 싶은 섭취 음식의 양을 초기화시켜줄 수 있다.

intakemenu.php => kill_intake.php

(사용SQL)

```
UPDATE ".$uid."_foods SET ate = 0 WHERE name = '$target'
```

-마찬가지로 운동의 종류와 시간을 입력하면 exercises의 네 번째 field인 done field에 저장된다.

consume.php => update_exer.php

(사용SQL)

```
SELECT name FROM ".$uid."_exers
```

```
UPDATE ".$uid."_exers SET done = $minutes WHERE name = '$exercise'
```

여기에서도 지우고 싶은 운동시간을 지울 수 있다.

consume.php => kill_exer.php

(사용SQL)

```
UPDATE ".$uid."_exers SET done = 0 WHERE name = '$target'
```

-New, Modify창에서는 사용자가 새로운 tuple을 자신의 database에 추가할 수 있다.

new.php => insert_food.php, insert_exer.php

(사용SQL)

```
INSERT INTO ".$uid."_foods (name, kcal, type) values('".$name."', '".$kcal."', '".$type."')
```

```
INSERT INTO ".$uid."_exers (name, kcal) values('".$name2."', '".$kcal2."')
```

-기존의 내용을 수정할 수 있다.

음식

modify.php => modify_foods.php => modify_foods2.php

(사용SQL)

```
SELECT name FROM ".$uid."_foods WHERE type=$foods
```

```
UPDATE ".$uid."_foods SET kcal = $kcal WHERE name = '$food'
```

운동

modify.php => modify_exer.php

(사용SQL)

```
UPDATE ".$uid."_exers SET kcal = $kcal WHERE name = '$exercise'
```

-자신의 database에 어떤 내용이 들어있는지 검색할 수 있다. 검색 결과에서 자신이 먹은 음식이나 한 운동을 update할 수 있다.

search.php=>search_food.php, search_exer.php => update_foods.php, update_exer.php

(사용SQL)

```
SELECT name FROM ".$uid."_foods WHERE name LIKE '%".$search.%'
```

```
SELECT name FROM ".$uid."_exers WHERE name LIKE '%".$search.%'
```

-Analysis창에서는 Intake, Consume창에서 입력받아서 저장되어 있는 database를 바탕으로 하루 balance를 보여준다. Foods table에서 모든 tuple에서 칼로리와 양을 곱한 것을 모두 더하여 총 섭취 칼로리를 구한다. 그리고 Exercises table에서도 마찬가지로 작업을 한다. 사용자의 성별, 생년, 키, 몸무게를 바탕으로 체질을 계산해 준다.

analysis.php

(사용 SQL)

```
SELECT name, kcal, ate FROM ".$uid."_foods WHERE ate != 0
```

```
SELECT name, kcal, done FROM ".$uid."_exers WHERE done !=0
```

```
SELECT sex FROM users WHERE uid='$uid'
```

```
SELECT height, weight FROM ".$uid."_stores
```

```
SELECT birthyear FROM users WHERE uid='$uid'
```

-매일 자신의 정보를 database에 저장한다. 일기장 기능이라고 볼 수 있다.

■ User Interface

1.Login

-사용자의 아이디와 암호를 입력받고 확인이 되면 main화면으로 들어가게 된다.

-새로 사용자가 등록이 되면 기존의 foods와 exercises table로부터 그 사용자의 table이 새로 생성된다.

2.Foods

-5개의 종류로 분류되어있는 음식들 중에 사용자가 섭취한 음식과 양을 입력하면

-아래에 사용자가 선택한 결과가 테이블로 나타난다.

-삭제하고 싶은 것이 있으면 삭제도 가능하다.

3.Exercise

-사용자가 한 운동과 시간을 입력하면

-아래에 선택한 결과가 테이블로 나타난다.

-마찬가지로 삭제가 가능하다.

4.Add Data

-기존에 주어지지 않은 음식이나 운동을 사용자가 추가할 수 있는 창이다.

-음식을 입력할 때에는 종류를 선택하도록 하였다.

5.Update Data

-기존에 주어진 음식, 운동의 데이터 중에서 수정하고 싶은 것을 수정하는 창이다.

6.Search

-어떤 종류의 음식, 운동이 있는지 모를 경우, 검색을 할 수 있다.

-검색결과에 먹은 양이나 운동시간을 추가해서 사용자의 데이터베이스에 넣을 수 있다.

7.Analysis

-앞의 창에서 선택한 것들을 바탕으로

-최종적으로 사용자가 얼마만큼의 열량을 소비하고 섭취했는지를 종합적으로 보여준다.

-사용자의 몸무게와 키를 보고 기초 대사량을 측정해주고 Broca에 의한 비만도, BMI에 의한 비만판정을 통해 사용자의 상태를 파악해준다.

-자신의 섭취, 운동 정보를 날짜별로 저장할 수 있고 메모를 할 수 있어서 일기장 기능도 가능하다.

-수시로 자신의 인적사항(키, 몸무게)을 update시킬 수 있다.

8.Logout / Home

-logout을 하면 this.php로 이동하면서 새로운 사용자를 기다린다.

-home에서는 사용자에게 간단한 사이트 이용 방법을 설명한다.