

# 饮食助手系统-概念设计

## 一、 实体

### 1.1 食物实体

#### 1. 描述

食物实体中存放了食物名称及其相关的营养素数据，我们这里的食物特指未经过烹饪的食物，包括各类水果、蔬菜、肉类、豆制品、主食以及市面上可以买到的各种零食、即食物品等。

#### 2. 属性

- 食物编号  
唯一地标识每个食物，为食物实体的主键，要求非空
- 食物名称  
每种食物的名称
- 热量  
食物所含的热量，单位为千卡，要求为非负数
- 蛋白质含量  
食物所含的蛋白质含量，单位为克，要求为非负数
- 总脂肪含量  
食物所含的脂肪含量，包括饱和脂肪和反式脂肪，单位为克，要求为非负数
- 碳水化合物含量  
食物所含的碳水化合物含量，包括膳食纤维和糖类，单位为克，要求为非负数
- 钠含量  
食物所含的钠元素含量，单位为毫克，要求为非负数
- 胆固醇含量  
食物所含的胆固醇含量，单位为毫克，要求为非负数
- 钙含量  
食物所含的钙元素含量，单位为毫克，要求为非负数
- 铁含量  
食物所含的铁元素含量，单位为毫克，要求为非负数
- 维生素 C 含量  
食物所含的维生素 C 含量，单位为毫克，要求为非负数

#### 3. 图示

为了清晰地显示实体中的各个属性，图片中实体名、属性名的命名采用首字母大

写，单词间使用空格分隔的格式，实际构建数据库时全部小写，单词间使用下划线分隔。之后的所有图示均同理。

Food	
PK	<u>Food ID</u>
	Food Name Calories Protein Total Fat Total Carbonhydrated Sodium Cholestoral Calcium Iron Vitamin C

## 1.2 菜肴实体

### 1. 描述

菜肴实体中存放了菜肴名称及其相关的营养素数据，我们这里的菜肴指使用各种食材烹饪所得到的食物，菜肴营养素含量并不是简单地等于其所有食材的营养素之和，而是以实际烹饪后的营养素为准。

### 2. 属性

- 菜肴编号  
唯一地标识每种菜肴，为菜肴实体的主键，要求非空
- 菜肴名称  
每种菜肴的名称
- 热量  
以下属性均与食物重复，不再赘述
- 蛋白质含量
- 总脂肪含量
- 碳水化合物含量
- 钠含量
- 胆固醇含量
- 钙含量
- 铁含量
- 维生素 C 含量

### 3. 图示

Dish	
PK	<u>Dish ID</u>
	Dish Name Calories Protein Total Fat Total Carbonhydrated Sodium Cholestoral Calcium Iron Vitamin C

### 1.3 一餐实体

#### 1. 描述

一餐实体中存放了名称及其相关的营养素数据，我们这里的一餐即为表面意思，每一餐可以包括多种食物以及菜肴，一餐的营养素即为其中所有食物及菜肴的营养素之和

#### 2. 属性

- 编号  
唯一地标识每一餐，为实体的主键，要求非空
- 名称  
给这一餐取的名称
- 热量  
以下属性均与食物重复，不再赘述
- 蛋白质含量
- 总脂肪含量
- 碳水化合物含量
- 钠含量
- 胆固醇含量
- 钙含量
- 铁含量
- 维生素 C 含量

#### 3. 图示

Meal	
<b>PK</b>	<b><u>Meal ID</u></b>
	Meal Name Calories Protein Total Fat Total Carbonhydrated Sodium Cholestoral Calcium Iron Vitamin C

#### 1.4 饮食计划实体

##### 1. 描述

饮食计划实体记录了用户一天内所有的饮食情况以及营养素的摄入情况。在系统自动推荐饮食计划或者用户记录自己的饮食计划是时会用到。

##### 2. 属性

- 饮食计划编号  
唯一地标识饮食计划，为实体的主键，要求非空
- 饮食计划名称
- 总卡路里  
这一日摄入食物的总卡路里
- 总蛋白质摄入量
- 总脂肪摄入量
- 总碳水摄入量

##### 3. 图示

Meal Plan	
<b>PK</b>	<b><u>Meal Plan ID</u></b>
	Plan Name Calories Protein Total Fat Total Carbonhydrated

## 1.5 用户实体

### 1. 描述

用户实体中存放了用户账号名称及其相关的数据，描述了一个用户的基本信息。

### 2. 属性

- 用户账号  
用户账号，唯一地标识每个用户，用户注册后不可更改，为用户实体的主键，要求非空
- 用户名称  
用户对外显示的网络昵称，要求非空
- 用户头像  
用户对外显示的头像
- 用户性别
- 用户生日

### 3. 图示

User	
PK	<u>Account Number</u>
	User Name Avatar Sex Birthday

## 1.6 身体数据实体

### 1. 描述

身体实体中存放了用户的身体参数的相关数据，包括身高、体重、体脂率、肌肉量等，用于用户身体数据的追踪记录。

### 2. 属性

- 数据编号  
唯一地标识每组统计数据，为用户身体数据实体的主键，要求非空
- 日期  
用于标注用户的每日身体数据记录的日期
- 年龄
- 身高
- 体重
- 体脂率

- 肌肉量
- 臂围
- 腰围
- 胸围
- 肩宽
- 臀围

### 3. 图示

User Body Statistic	
PK	<u>Statistic ID</u>
	Date Age Height Weight Body Fat Rate Lean Muscle Mass Arm Circumference Waistline Chest Circumference Shoulder Breadth Hip Circumference

## 1.7 目标实体

### 1. 描述

目标实体中存放了用户当前身体健康方面的目标，目标可以设置为增肌、减脂、维持体重等。用户可以设置多个目标，并为每个目标定义一个名字。

### 2. 属性

- 目标编号  
唯一地标识每个目标，为用户目标实体的主键，要求非空
- 目标名称  
由用户制定
- 目标类型  
包括减脂、增肌、维持体重等
- 开始日期  
用户目标的开始日期
- 结束日期  
用户目标的结束日期

- 目标体重  
用户计划达到的体重
- 目标体脂率  
用户计划达到的体脂率
- 目标肌肉量  
用户计划达到的肌肉量

### 3. 图示

User Target	
PK	<u>Target ID</u>
	Target Name Target Start Date End Date Target Weight Target Body Fat Rate Target Lean Muscle Mass

## 1.8 日常摄入实体

### 1. 描述

日常摄入实体记录了用户一天内各营养素的摄入情况，在系统智能推荐每日营养素摄入量、用户记录每日营养素摄入量时会用到。

### 2. 属性

- 摄入记录编号  
唯一地标识每个摄入记录，为日常摄入实体的主键，要求非空
- 日期  
该摄入记录的日期
- 摄入卡路里总量
- 摄入蛋白质总量
- 摄入脂肪总量
- 摄入碳水化合物总量
- 饮水总量

### 3. 图示

Daily Intake	
<b>PK</b>	<b><u>Intake Record ID</u></b>
	Date Calories Protein Fat Carbonhydrated Water

## 1.9 文章实体

### 1. 描述

文章实体记录了用户发表的文章，用户可以发表自己在某一方面（如减脂）的经验与推荐等。

### 2. 属性

- 文章编号  
唯一地标识每篇文章，为主键，要求非空
- 标题  
用户为文章所起的标题
- 日期  
用户发表该篇文章的日期
- 文本  
文章内容
- 点赞数量

### 3. 图示

Article	
<b>PK</b>	<b><u>Article ID</u></b>
	Date Title Text Like Number



## 1.10 评论实体

### 1. 描述

评论体记录了用户对某篇文章发表的评论。

### 2. 属性

- 评论编号  
唯一地标识每个评论，为主键，要求非空
- 文本  
评论内容

### 3. 图示

Comment	
PK	<u>Comment ID</u>
	Text

## 二、 实体之间的联系

### 2.1 食物与一餐

#### 1. 描述

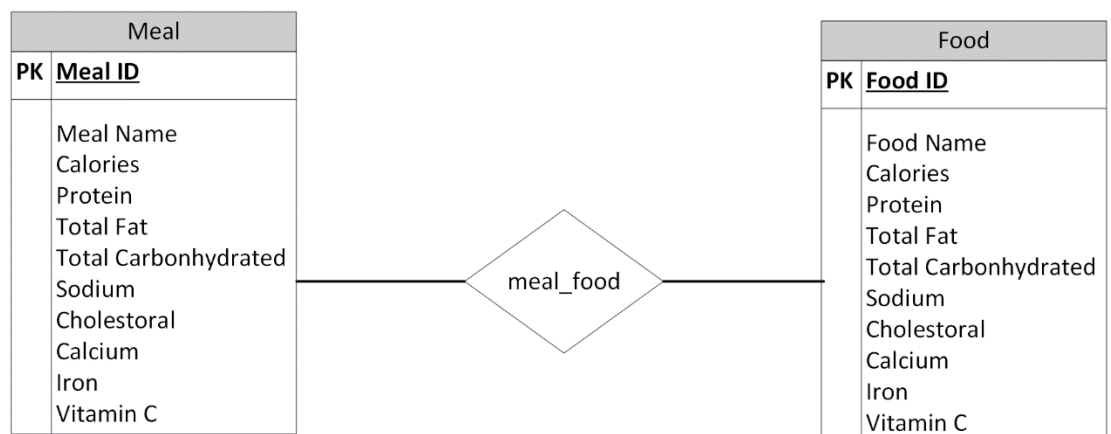
每一餐包含多种食物，食物也包含在不同的餐中，二者为多对多的关系，每一餐中的食物需要标注份量。

#### 2. 关系模式

下划线表示主码

meal\_food(meal\_id, food\_id, serving\_size)

#### 3. 图示



## 2.2 菜肴与一餐

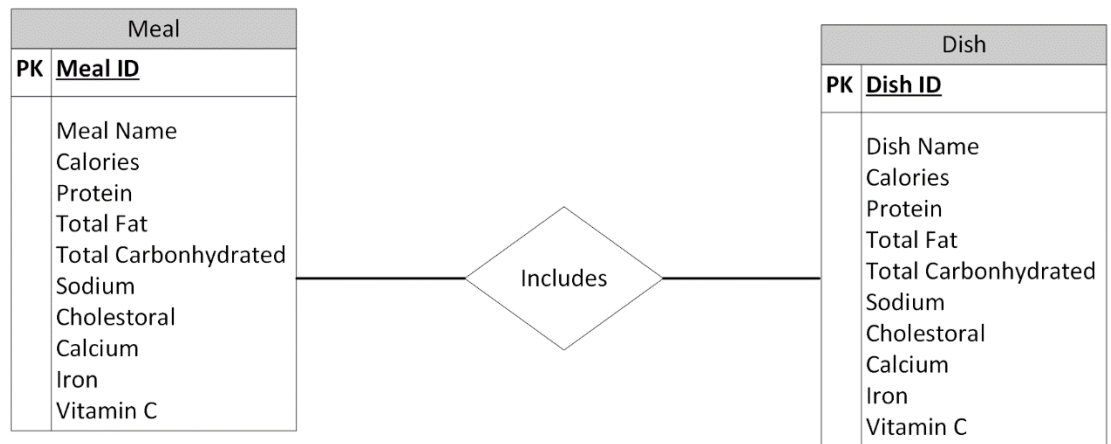
### a) 描述

每一餐包含多种菜肴，菜肴也包含在不同的餐中，二者为多对多的关系，每一餐中的菜肴需要标注份量。

### b) 关系模式

meal\_dish(meal\_id, dish\_id, serving\_size)

### c) 图示



## 2.3 饮食计划与一餐

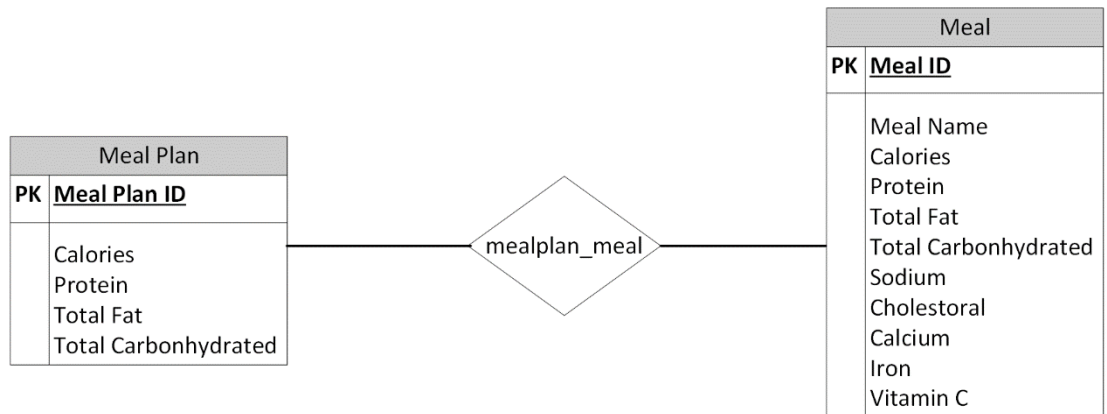
### 1. 描述

饮食计划规划了一天中每一餐应该吃哪些食物。每一餐也可出现在不同的饮食计划中。所以饮食计划实体集相对一餐实体集为多对多的关系。在两者的联系中应该考虑每一餐的时间，即应该作为早餐还是午餐等。

### 2. 关系模式

mealplan\_meal(meal\_plan\_id, meal\_id, time)

### 3. 图示



## 2.4 用户与目标（制定目标）

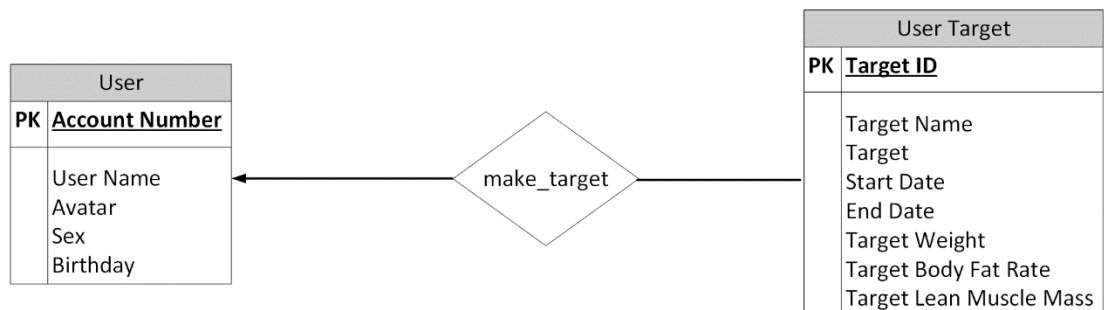
### 1. 描述

每个用户可以指定多个目标，如减脂、增肌等。制定出的每个计划只对应一个用户，所以用户实体集相对用户目标实体集为一对多的关系。

### 2. 关系模式

`make_target(target_id, account_num,)`

### 3. 图示



## 2.5 用户与身体数据（记录身体数据）

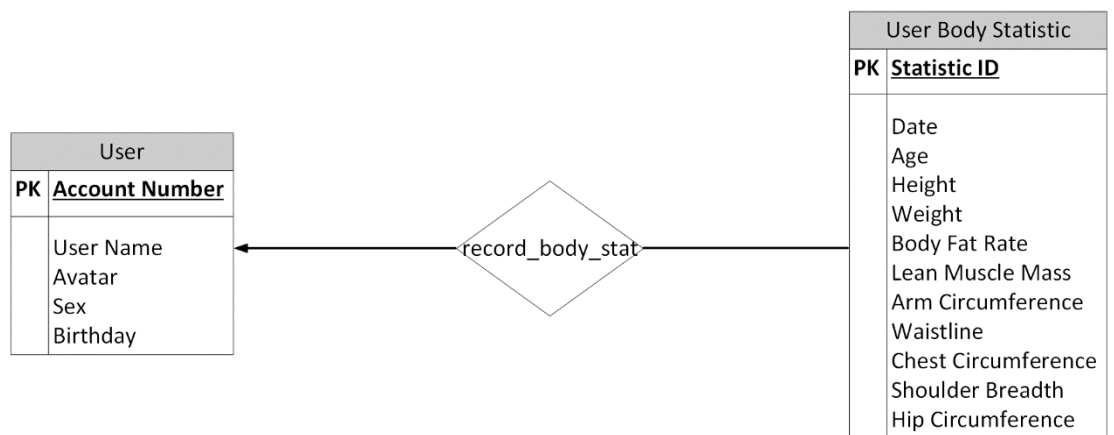
### 1. 描述

此处的身体数据为用户特定日期的身体数据，因此每个用户可以有多个身体数据记录，而每条身体数据记录仅对应一个用户实体，所以用户实体集相对身体数据实体集为一对多的关系。

### 2. 关系模式

`record_body_stat(statistic_id, account_num,)`

### 3. 图示



## 2.6 用户与饮食计划及日常摄入（制定饮食计划与记录日常营养素摄入）

### 1. 描述

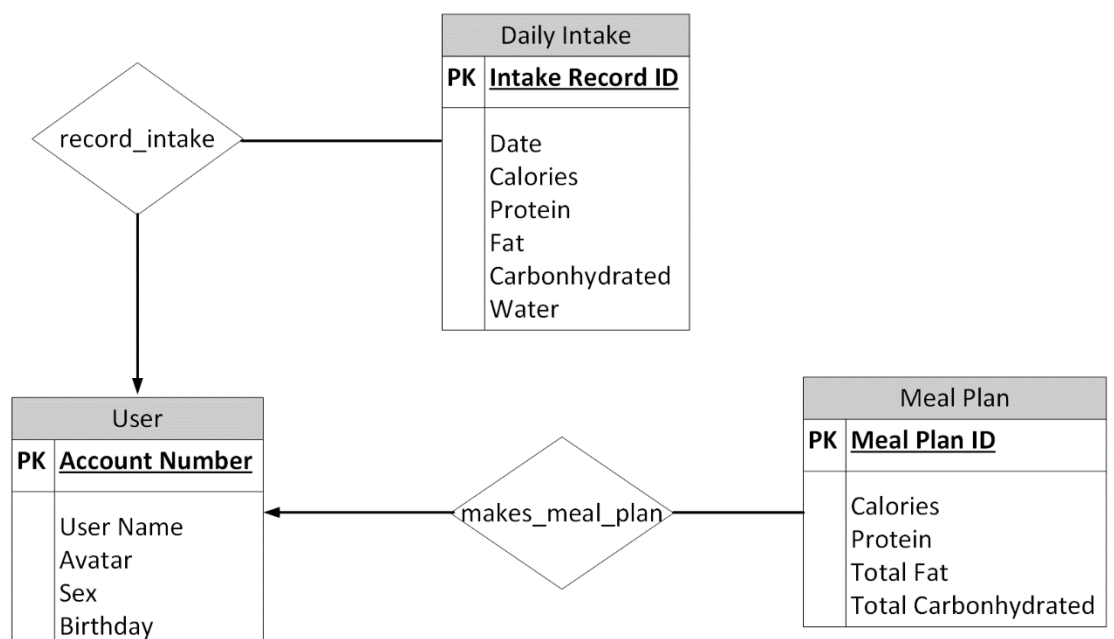
每个用户在每一天的饮食计划不同，因此可以有多个饮食计划，而每个饮食计划仅对应一个用户。此外，用户也可以设定自己每日营养素的摄入量目标，或者是记录今日营养素的摄入量。因此用户实体集相对饮食计划实体集为一对多的关系，用户实体集相对日常摄入量为多对一的关系。

### 2. 关系模式

`record_intake(intake_record_id, account_num)`

`makes_meal_plan(meal_plan_id, account_num, date)`

### 3. 图示



## 2.7 用户、文章及评论

### 1. 描述

每个用户可以将自己在健康方面的经验写成文章，发布在社区中，而其它用户可以对该文章发表评论。一个用户可以发布多篇文章，一篇文章仅对应一个用户；用户与评论、文章与评论也是同理。

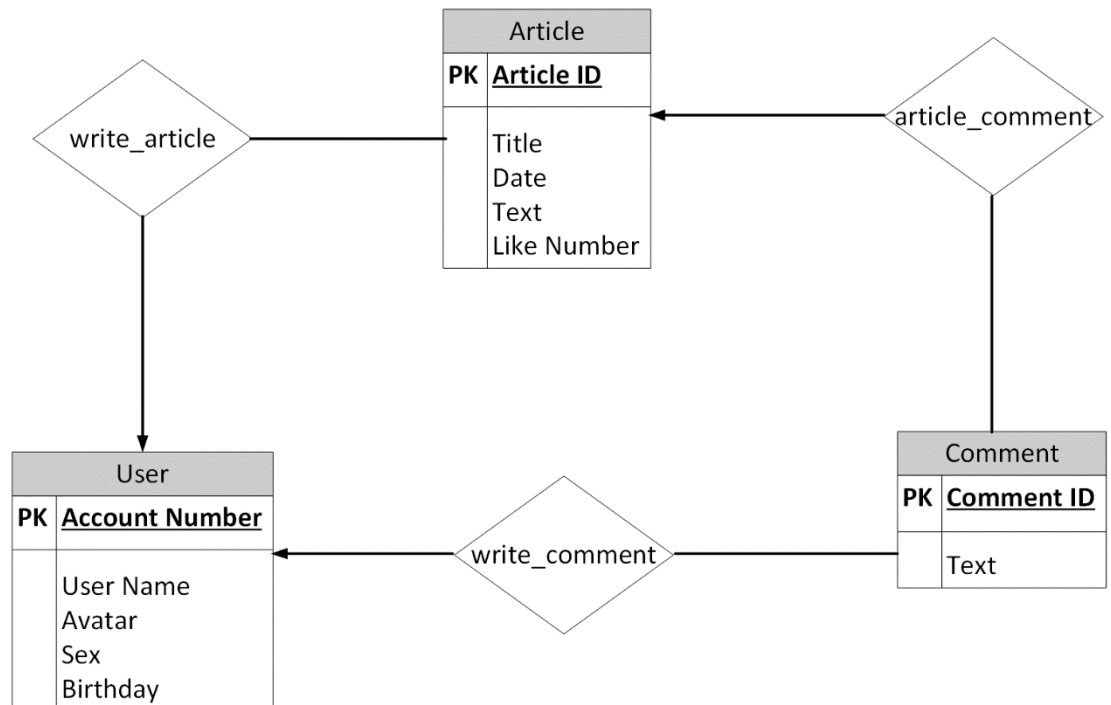
### 2. 关系模式

write\_article(article\_id, account\_num)

write\_comment(comment\_id, account\_num)

article\_comment(comment\_id, article\_id)

### 3. 图示



## 三、 全局 E-R 图

