## **Лабораторная работа №1 Проектирование базы данных с использованием ER-технологии**

**Теоретические сведения**

Для заданной предметной области должен быть определен состав реляционных таблиц и логические связи между таблицами. Для каждого атрибута должны быть заданы тип и размер данных, ограничения целостности. Для каждой таблицы – первичный ключ, потенциальные ключи и внешние ключи.

Разработка логической модели методом «сущность-связь»(ER-методом) предусматривает выполнение следующих шагов: построение ER-диаграммы, включающей все сущности и связи, важные с точки зрения интересов предметной области;

* 1. анализ связей и определение их характеристик – степени связи, мощности и класса принадлежности;
  2. построение набора предварительных отношений с указанием предполагаемого первичного ключа для каждого отношения;
  3. подготовка списка всех неключевых атрибутов и назначение каждого из этих атрибутов одному из предварительных отношений;
  4. проверка нахождения всех полученных отношений в нормальной форме Бойса-Кодда;
  5. построение модели данных.

**Практическая работа**

При выполнении лабораторной работы необходимо:

* для своего варианта, соответствующего определенной предметной области, построить логическую модель данных в соответствии со стандартом IDEF1X;
* построить физическую модель;
* составить отчет по лабораторной работе.

**Пример выполнения работы**

Особенности диалекта SQL в СУБД MySQL рассмотрим на примере учебной базы данных *book* Интернет-магазина, торгующего компьютерной литературой. В базе данных должна поддерживаться следующая информация:

* тематические каталоги, по которым сгруппированы книги;
* предлагаемые книги (название, автор, год издания, цена, имеющееся на складе количество);
* зарегистрированные покупатели (имя, отчество, фамилия, телефон, адрес электронной почты, статус – авторизованный, неавторизованный, заблокированный, активный с хорошей кредитной историей);
* покупки, совершенные в магазине (время совершения покупки, число приобретенных экземпляров книги).

Логическая модель данных предметной области в стандарте IDEF1X представлена на рис. 1. Выделены сущности *КАТАЛОГ, КНИГА, КЛИЕНТ, ЗАКАЗ*, между которыми установлены неидентифицирующие связи мощностью oдин-ко-многим, определенные спецификой предметной области.

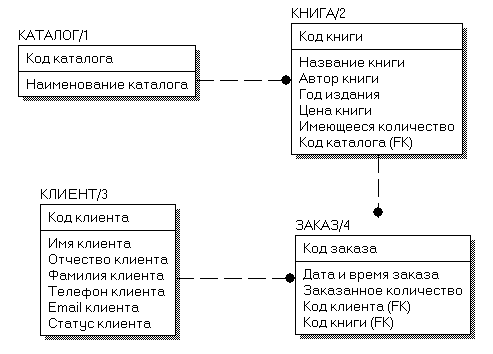


Рис. 1. Логическая модель данных предметной области

Физическая модель данных предметной области в стандарте IDEF1X для целевой СУБД MySQL представлена на рис. 2.

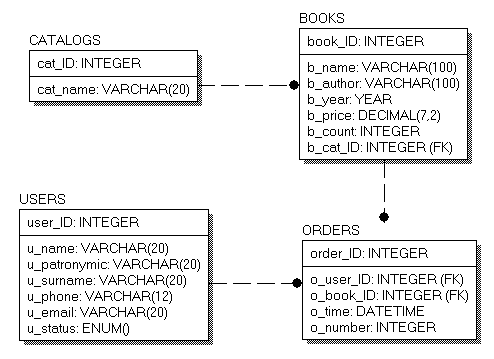


Рис. 2. Физическая модель предметной области

База данных *book* состоит из четырех таблиц:

* *catalogs* – список торговых каталогов;
* *books* – список предлагаемых книг;
* *users* – список зарегистрированных пользователей магазина;
* *orders* – список заказов (осуществленных сделок).

Таблица *catalogs* состоит из двух полей:

* *cat\_ID* – уникальный код каталога;
* *cat\_name* – имя каталога.

Оба поля должны быть снабжены атрибутом *not null*, поскольку неопределенное значение для них недо­пустимо.

Таблица *books* состоит из семи полей:

* *book\_ID* – уникальный код книги;
* *b\_name* – название книги;
* *b\_author* – автор книги;
* *b\_year* – год издания;
* *b\_price* – цена книги;
* *b\_count* – количество книг на складе;
* *b\_cat\_ID* – код каталога из таблицы *catalogs*.

Цена книги *b\_price* и количество экземпляров на складе *b\_count* могут иметь атрибут *null*. На момент доставки час­то неизвестны количество товара и его цена, но отразить факт наличия товара в прайс-листе необходимо.

Поле *b\_cat\_ID* устанавливает связь между таблицами *catalogs* и *books*. Это поле должно быть объявлено как внешний ключ (FK)с правилом каскадного удаления и обновления. Обновление таблицы *catalogs* вызовет автоматическое обновление таблицы *books*. Удаление каталога в таблице *catalogs* приведет к автоматическому удалению всех записей в таблице *books*, соответствующих каталогу.

Таблица *users* состоит из семи полей:

* *user\_ID* – уникальный код покупателя;
* *u\_name* – имя покупателя;
* *u\_patronymic* – отчество покупателя;
* *u\_surname* – фамилия покупателя;
* *u\_phone* – телефон покупателя (если имеется);
* *u\_email* – e-mail покупателя (если имеется);
* *u\_status* – статус покупателя.

Статус покупателя представлен полем типа *enum*, которое может принимать одно из четырех значений:

* *active* – авторизованный покупатель, который может осуществлять покупки через Интернет;
* *passive* – неавторизованный покупатель (значение по умолчанию), который осуществил процедуру регистрации, но не подтвердил ее и пока не может осуществлять покупки через Интернет, однако ему доступны каталоги для просмотра;
* *lock –* заблокированный покупатель, не может осуществлять покупки и просматривать каталоги магазина;
* *gold* – активный покупатель с хорошей кредитной историей, которому предоставляется скидка при следующих покупках в магазине.

Поля *u\_phone* и *u\_email* могут быть снабжены атрибутом *null*. Остальные поля должны получить атрибут *not null*.

Таблица *orders* вклю­чает пять полей:

* *order\_ID* – уникальный номер сделки;
* *o\_user\_ID* – номер пользователя из таблицы *users*;
* *o\_book\_ID* – номер товарной позиции из таблицы *books*;
* *o\_time* – время совершения сделки;
* *o\_number* – число приобретенных товаров.

Поля таблицы *orders* должны быть снабжены атрибутом *not null*, т. к. при совершении покупки вся информация должна быть занесена в таблицу.

В таблице *orders* устанавливается связь с таблицами *users* (за счет поля *o\_user\_id*) и *books* (за счет поля *o\_book\_id*). Эти поля объявлены как внешние ключи (FK)с правилом каскадного удаления и обновления. Обновление таблиц *users* и *books* приведет к автоматическому обновлению таблицы *orders*. Удаление любого пользователя в таблице *users* приведет к автоматическому удалению всех записей в таблице *orders*, соответствующих этому пользователю.

**Содержание отчета:**

1. **Логическая и физическая модель предметной области (Visio, ERwin, MySQL Workbench).**
2. **Таблица сущностей.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Название сущности | Количество | Изменение количества сущностей | Идентификатор | Ограничение доступа | Тип связи |
|  |  |  |  |  |  |

Примечание.

Название сущности – название выделенной в предметной области сущности.

Количество – предлагаемое количество экземпляров сущности.

Изменение количества сущностей – процент изменения сущностей в единицу времени (в таблице за единицу времени взят 1 год).

Идентификатор – имя файла данных в базе данных, в котором будут храниться экземпляры данной сущности.

Ограничение доступа – список пользователей, которым разрешен доступ к данной сущности.

Тип связи – тип связи для каждой сущности, с которой связана данная сущность.

1. **Таблицы атрибутов.**

Таблица атрибутов сущности

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя атрибута | Формат | Область допустимых значений | Ограничение доступа | Роль атрибута |
|  |  |  |  |  |

1. **Список возможных запросов.**