Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

Высшего профессионального обучения

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

Пояснительная записка к курсовому проекту

по дисциплине «Базы данных»

на тему

«База данных «Издание»

Консультант по нормоконтролю Выполнил:

аспирант каф. КСУП Студент гр.515-2 \_\_\_\_\_\_\_И. В. Ячный \_\_\_\_\_\_А.А. Видершпан

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2016 г. «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2016 г.

Проверил:

Ст. преподаватель каф. КСУП

\_\_\_\_\_\_\_\_\_ Е. Н. Рыбалка

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2016 г.

Томск 2016

**Реферат**

Курсовой проект 50 с., 17 табл., 33 рис., 6 источников.

Пояснительная записка содержит концептуальную модель процесса «Издание» и описание структуры базы данных, предназначенной для его информационной поддержки. Концептуальное моделирование выполнено с использованием методологии IDEF1X.

Оглавление

[1 Описание предметной области 6](#_Toc451976762)

[1.1 Постановка задачи 7](#_Toc451976763)

[1.2 Перечень вводимой информации 7](#_Toc451976765)

[1.3 Концептуальная модель данных 8](#_Toc451976766)

[2 Глоссарий модели 9](#_Toc451976768)

[3 Типы полей 10](#_Toc451976769)

[4 Первичные, внешние и альтернативные ключи 16](#_Toc451976770)

[4.1 Ключи таблицы Avtor 16](#_Toc451976771)

[4.2 Ключи таблицы Book 16](#_Toc451976772)

[4.3 Ключи таблицы Authorship 17](#_Toc451976773)

[4.4 Ключи таблицы Edition\_department 17](#_Toc451976774)

[4.5 Ключи таблицы Stock\_department 18](#_Toc451976775)

[4.6 Ключи таблицы Sale\_department 18](#_Toc451976776)

[4.7 Ключи таблицы Marketing\_department 18](#_Toc451976777)

[4.8 Ключи таблицы Translation\_department 19](#_Toc451976778)

[4.9 Ключи таблицы Group\_departments 19](#_Toc451976779)

[4.10 Ключи таблицы Publish\_house 20](#_Toc451976780)

[5 Определение правил ограничения ввода 21](#_Toc451976781)

[5.1 Ограничения таблицы Avtor 21](#_Toc451976782)

[5.2 Ограничения таблицы Book 21](#_Toc451976783)

[5.3 Ограничения таблицы Sale\_department 22](#_Toc451976784)

[5.4 Ограничения таблицы Marketing\_department 22](#_Toc451976785)

[5.5 Ограничения таблицы Edition\_department 23](#_Toc451976786)

[5.6 Ограничения таблицы Publish\_house 24](#_Toc451976787)

[6 Организация ввода данных 25](#_Toc451976788)

[6.1 Данные таблицы Avtor 26](#_Toc451976789)

[6.2 Данные таблицы Book 27](#_Toc451976790)

[6.3 Данные таблицы Authorship 28](#_Toc451976791)

[6.4 Данные таблицы Edition\_department 29](#_Toc451976792)

[6.5 Данные таблицы Stock\_department 30](#_Toc451976793)

[6.6 Данные таблицы Sale\_department 31](#_Toc451976794)

[6.7 Данные таблицы Marketing\_department 32](#_Toc451976795)

[6.8 Данные таблицы Translation\_department 33](#_Toc451976796)

[6.9 Данные таблицы Group\_departments 34](#_Toc451976797)

[6.10 Данные таблицы Publish\_house 35](#_Toc451976798)

[7 Запросы 36](#_Toc451976799)

[Заключение 49](#_Toc451976800)

[Список использованных источников 50](#_Toc451976801)

**Введение**

Курсовой проект выполнен с целью практического освоения основных приемов и правил методологии информационного моделирования IDEF1X и структурированного языка запросов SQL. В качестве предметной области разрабатываемой базы данных (БД) выбрана система «Издание».

Предложенный в настоящей курсовой работе проект направлен на достижение указанных целей.

Концептуальная модель данных представлена в виде IDEF1X-диаграммы данных, показывающая сущности предметной области и выявляющая обусловленные правила издательств. Диаграмма сопровождается глоссарием, содержащим формальные определения имен всех сущностей и хранимых элементов данных.

Практическое значение: разработанная БД позволяет автоматизировать контроль над изданием или контроль любого издательства.

Актуальность курсовой работы объясняется тем, что в настоящее время, специалистам в IT-сфере необходимо эффективно распределять и систематизировать большой поток информации. Для этого специалист должен уметь разрабатывать и внедрять распределенные системы обработки данных.

База данных разработана в программной среде MYSQL, с использованием языка программирования SQL. Результатом курсовой работы должна стать готовая БД, удовлетворяющая всем требованиям технического задания.

# 1 Описание предметной области

В данном курсовом проекте исследуется работа издательства. Издательство –  предприятие, которое работает в области литературы, искусства, музыки или науки, и продукция которой может воспроизводиться и распространяться.

Автор пишет книгу, отдает ее в издание, она проходит множества этапов по редактированию, в ходе чего ее издают.

Современное издательство является крупным многопрофильным, специализированным предприятием. Сущность издательской деятельности — распространение информации и создание прибавочной стоимости объектов авторского права. Издательство, руководствуясь своим опытом, приобретает исключительное право у автора (писателя, художника) на издаваемое произведение, и организует его воспроизведение (изготовление) и распространение.

## 1.1 Постановка задачи

## Задачей курсового проекта является построение IDEF1x-диаграммы, показывающей все сущности и связи между ними. Освоение программной среды MySQL. Результатом работы курсового проекта должна быть готовая БД, удовлетворяющая всем требованиям технического задания.

## 1.2 Перечень вводимой информации

* данные об авторе (ФИО автора, ученая степень);
* данные об авторстве (номер автора, номер книги);
* данные о книге (название книги, ISBN, количество страниц, количество глав, дата печати);
* данные об отделе склада (номер книги, количество);
* данные об отделе продаж (номер книги, цена, количество продаж);
* данные об отделе перевода (номер книги, язык оригинала, язык перевода);
* данные об отделе редакции (номер книги, формат, тип переплета, тип бумаги);
* данные об отделе маркетинга (номер книги, репутация, бюджет);
* данные о группе отделов (номер отдела редакции, номер отдела перевода, номер отдела склада, номер отдела продаж, номер отдела маркетинга);
* данные об издательстве (название издательства, город, ИНН, масштаб, территориальный признак, природа информации, номер группы отдела, дата основания).

## 1.3 Концептуальная модель данных

## Концептуальная модель представляет собой описание основных сущностей (таблиц) и связей между ними без учета принятой модели БД и синтаксиса целевой БД.

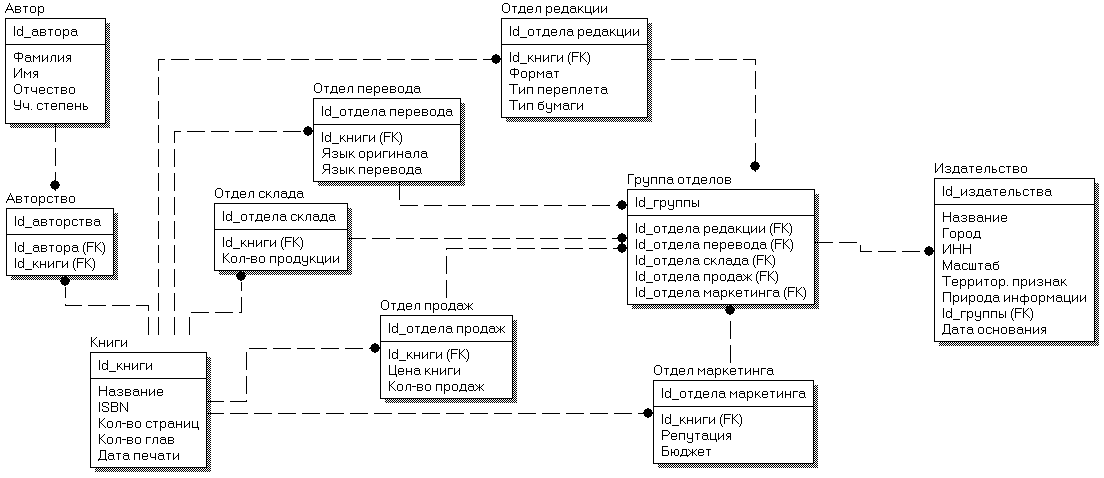
На рисунке 1.3.1 приведен блок физической информационной модели «Издание». Для ее построения использовалась программа Erwin 4.0

Рисунок 1.3.1 – концептуальная модель «Издание»

# 2 Глоссарий модели

Описание всех реализованных сущностей приведено в таблице 2.1.

Таблица 2.1 – Сущности

|  |  |
| --- | --- |
| *Имя* | *Определение* |
| Автор | Человек пишущий книги. |
| Авторство | Принадлежность произведения автору. |
| Книги | Вид печатной продукции, написанный автором. |
| Отдел продаж | Отдел, отслеживающий цену каждой книги и количество их продаж. |
| Отдел склада | Отдел, где хранятся все изданные книги. |
| Отдел маркетинга | Отдел, отслеживающий репутацию книг, а так же бюджет каждой книги. |
| Отдел перевода | Отдел, занимающийся переводом книг с одного языка на другой. |
| Отдел редакции | Отдел, занимающийся редактированием и печатью книг. |
| Группа отделов | Все отделы, которые участвовали в издание книги. |
| Издательство | Предприятие, занимающиеся изданием книг. |

# 3 Типы полей

Поля таблицы предназначены для хранения в них данных. Это могут быть числа, текстовая информация, даты, графические файлы и т. д. Для определения типа данных, размещаемых в поле, используются тип поля, его ширина и количество знаков после запятой.

Для заданной логической модели «Издание» были созданы таблицы со следующим описанием полей.

Описание полей для таблицы Avtor представлено в таблице 3.1:

Таблица 3.1 – Описание полей таблицы Avtor

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_avtor | Int Autoinc | 11 | Личный номер автора |
| S\_name | Varchar | 30 | Фамилия автора |
| Name | Varchar | 30 | Имя автора |
| T\_name | Varchar | 30 | Отчество автора |
| Academ\_degree | Varchar | 30 | Ученая степень |

Описание полей для таблицы Authorship представлено в таблице 3.2:

Таблица 3.2 - Описание полей таблицы Authorship

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_avtor | Int | 11 | Личный номер автора |
| Id\_book | Int | 11 | Личный номер книги |

Описание полей для таблицы Book представлено в таблице 3.3:

Таблица 3.3 - Описание полей таблицы Book

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_book | Int Autoinc | 11 | Личный номер книги |
| Name\_book | Varchar |  | Название книги |
| ISBN | Bigint | 13 | Международный стандартный книжный номер |
| Number\_pages | Smallint | 4 | Количество страниц |
| Number\_chapters | Tinyint | 3 | Количество глав |
| Print\_date | Date | 8 | Дата пописи |

Описание полей для таблицы Edition\_department представлено в таблице 3.4:

Таблица 3.4 - Описание полей таблицы Edition\_department

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_edition | Int Autoinc | 11 | Личный номер редакции |
| Id\_book | Int | 11 | Личный номер книги |
| Format | Varchar | 20 | Формат печати |
| Type\_paper | Varchar | 30 | Тип бумаги |
| Type\_binding | Varchar | 30 | Тип переплета |

Описание полей для таблицы Stock\_department представлено в таблице 3.5:

Таблица 3.5 - Описание полей таблицы Stock\_department

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_stock | Int Autoinc | 11 | Личный номер склада |
| Id\_book | Int | 11 | Личный номер книги |
| Number\_products | Int | 11 | Количество книг |

Описание полей для таблицы Sale\_department представлено в таблице 3.6:

Таблица 3.6 - Описание полей таблицы Sale\_department

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_sale | Int Autoinc | 11 | Личный номер отдела продаж |
| Id\_book | Int | 11 | Личный номер книги |
| Price | Smallint | 4 | Цена |
| Number\_sales | Mediumint | 9 | Количество продаж |

Описание полей для таблицы Marketing\_department представлено в таблице 3.7:

Таблица 3.7 - Описание полей таблицы Marketing\_department

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_marketing | Int Autoinc | 11 | Личный номер книги |
| Id\_book | Int | 11 | Личный номер книги |
| Budget | Mediumint | 9 | Бюджет |
| Reputation | Float | 5,1 | Репутация |

Описание полей для таблицы Translation\_department представлено в таблице 3.8:

Таблица 3.8 - Описание полей таблицы Translation\_department

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_translation | Int Autoinc | 11 | Личный номер книги |
| Id\_book | Int | 11 | Личный номер книги |
| Original\_language | Varchar | 40 | Язык оригинала |
| Translated\_language | Varchar | 40 | Язык перевода |

Описание полей для таблицы Group\_departments представлено в таблице 3.9:

Таблица 3.9 - Описание полей таблицы Group\_departments

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_group | Int | 11 | Личный номер группы отделов |
| Id\_edition | Int | 11 | Личный номер редакции |
| Id\_stock | Int | 11 | Личный номер склада |
| Id\_sale | Int | 11 | Личный номер отдела продаж |
| Id\_marketing | Int | 11 | Личный номер книги |
| Id\_translation | Int | 11 | Личный номер книги |

Описание полей для таблицы Publish\_house представлено в таблице 3.10:

Таблица 3.10 - Описание полей таблицы Publish\_house

|  |  |  |  |
| --- | --- | --- | --- |
| *Имя* | *Тип данных* | *Длина поля* | *Определение* |
| Id\_izd | Int Autoinc | 11 | Личный номер издательства |
| Name\_ph | Varchar | 30 | Название |
| City | Varchar | 30 | Город |
| Scale | Varchar | 30 | Масштаб |
| Geo\_indication | Varchar | 30 | Территориальный признак |
| Nature\_info | Varchar | 40 | Природа информации |
| INN | Bigint | 11 | Идентификационный номер налогоплательщика |
| Id\_group | Int | 11 | Личный номер группы отделов |
| Founding\_date | Date | 4 | Дата основания |

# 4 Первичные, внешние и альтернативные ключи

При определении первичных и внешних ключей, также как и при создании таблиц, необходимо учитывать связи типа «предок-потомок» между таблицами. Так как внешний ключ – это атрибут первичного ключа таблицы предка, то сначала надо определить все первичные ключи предков, затем внешние и первичные ключи потомков, и так далее по иерархии. Ключи любого типа определяются при помощи индексов различного типа.

Для создания первичных, внешних и альтернативных ключей была использована электронная документация "MyQSL" [[1](#_Список_использованных_источников)].

В соответствии с правилом определения типа ключа, для таблиц БД «Издание» были определены следующие первичные и внешние ключи:

* 1. **Ключи таблицы Avtor**

Таблица Avtor содержит 1 ключ (рисунок 4.1.1):

* первичный ключ id\_avtor, представляющий собой уникальный номер автора.



Рисунок 4.1.1 – Ключ таблицы Avtor

## 4.2 Ключи таблицы Book

Таблица Book содержит 2 ключа (рисунок 4.2.1):

* первичный ключ id\_book, представляющий собой уникальный номер книги;
* уникальный ключ ISBN, представляет собой Международный стандартный книжный номер.

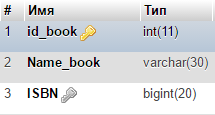


Рисунок 4.2.1 – Ключи таблицы Book

## 4.3 Ключи таблицы Authorship

Таблица Authorship содержит 3 ключа (рисунок 4.3.1):

* первичный ключ id\_authorship, представляющий собой уникальный номер авторства;
* внешний ключ id\_book, создает связь между таблицами;
* внешний ключ id\_avtor, создает связь между таблицами.

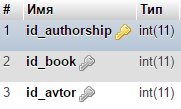


Рисунок 4.3.1 – Ключи таблицы Authorship

**4.4 Ключи таблицы Edition\_department**

Таблица Edition\_department содержит 2 ключа (рисунок 4.4.1):

* первичный ключ id\_edition, представляющий собой уникальный номер отдела редакции;
* внешний ключ id\_book, создает связь между таблицами.

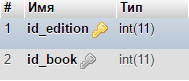


Рисунок 4.4.1 – Ключи таблицы equipment

* 1. **Ключи таблицы** **Stock\_department**

Таблица Stock\_department содержит 2 ключа (рисунок 4.5.1):

* первичный ключ id\_stock, представляющий собой уникальный номер отдела склада;
* внешний ключ id\_book, создает связь между таблицами.

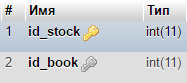


Рисунок 4.5.1 – Ключи таблицы Stock\_department

**4.6 Ключи таблицы Sale\_department**

Таблица Sale\_department содержит 2 ключа (рисунок 4.6.1):

* первичный ключ id\_sale, представляющий собой уникальный номер отдела продаж;
* внешний ключи id\_book, создает связь между таблицами.

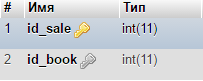


Рисунок 4.6.1 – Ключи таблицы Sale\_department

**4.7 Ключи таблицы Marketing\_department**

Таблица Marketing\_department содержит 3 ключа (рисунок 4.7.1):

* первичный ключ id\_marketing, представляющий собой уникальный номер отдела маркетинга;
* внешний ключ id\_book, создает связь между таблицами.

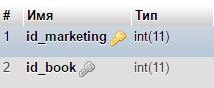


Рисунок 4.7.1 – Ключи таблицы Marketing\_department

## 4.8 Ключи таблицы Translation\_department

Таблица Translation\_department содержит 2 ключа (рисунок 4.8.1):

* первичный ключ id\_translation, представляющий собой уникальный номер отдела перевода;
* внешний ключ id\_book, создает связь между таблицами.

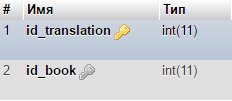


Рисунок 4.8.1 – Ключи таблицы Translation\_department

## 4.9 Ключи таблицы Group\_departments

Таблица Group\_departments содержит 6 ключей (рисунок 4.9.1):

* первичный ключ id\_group, представляющий собой уникальный номер группы отделов;
* внешний ключ id\_edition, создает связь между таблицами;
* внешний ключ id\_stock, создает связь между таблицами;
* внешний ключ id\_sale, создает связь между таблицами;
* внешний ключ id\_marketing, создает связь между таблицами;
* внешний ключ id\_translation, создает связь между таблицами.



Рисунок 4.9.1 – Ключи таблицы Group\_departments

**4.10 Ключи таблицы Publish\_house**

Таблица Publish\_house содержит 2 ключа (рисунок 4.10.1):

* первичный ключ id\_izd, представляющий собой уникальный номер издательства;
* внешний ключ id\_group, создает связь между таблицами.

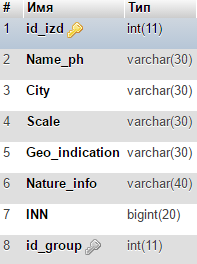


Рисунок 4.10.1 – Ключи таблицы Publish\_house

**5** **Определение правил ограничения ввода**

**Задание:** в соответствии с описанием модели ПО, установить правила, ограничивающие ввод значений. Проверить работу установленных правил.

Для создания ограничений была использована книга "MyQSL" [1].

## 5.1 Ограничения таблицы Avtor

Для ограничения полей таблицы Avtor было выбрано следующие поле: Academ\_degree. Его ограничение представлено в таблице 5.1.1.

Таблица 5.1.1 - Ограничения поля таблицы Avtor

|  |  |  |
| --- | --- | --- |
| *Имя поля* | *Ограничение поля* | *Пояснение* |
| Academ\_degree | new.Academ\_degree NOT IN ('Бакалавр','Магистр','Доктор философии','Кандидат наук','Доктор наук') | Ученая степень может быть только 5 видов: бакалавр, магистр, Доктор философии, кандидат наук, Доктор наук |

## 5.2 Ограничения таблицы Book

Для ограничения полей таблицы Book были выбраны следующие поля: ISBN, Number\_pages. Их ограничения представлены в таблице 5.2.1.

Таблица 5.2.1 - Ограничения полей таблицы Book

|  |  |  |
| --- | --- | --- |
| *Имя поля* | *Ограничение поля* | *Пояснение* |
| ISBN | CHAR\_LENGTH(CAST(new.ISBN AS CHAR)) <> 13 | Данное поле должно содержать ровно 13 цифр |
| Number\_pages | new.Number\_pages >2500 | Количество страниц не может быть больше 2500 штук |

## 5.3 Ограничения таблицы Sale\_department

Для ограничения полей таблицы Sale\_department было выбрано следующие поле: Price. Его ограничение представлено в таблице 5.3.1.

Таблица 5.3.1 - Ограничения поля таблицы Sale\_department

|  |  |  |
| --- | --- | --- |
| *Имя поля* | *Ограничение поля* | *Пояснение* |
| Price | new.Price >20000 | Цена книги не может превышать 20000 |

## 5.4 Ограничения таблицы Marketing\_department

Для ограничения полей таблицы Marketing\_department было выбрано следующие поле: Budget. Его ограничение представлено в таблице 5.4.1.

Таблица 5.4.1 - Ограничения поля таблицы Marketing\_department

|  |  |  |
| --- | --- | --- |
| *Имя поля* | *Ограничение поля* | *Пояснение* |
| Budget | new.Budget >6000000 | Бюджет не может превышать 6000000 |

## 5.5 Ограничения таблицы Edition\_department

Для ограничения полей таблицы Edition\_department было выбраны следующие поля: Format, Type\_paper, Type\_binding. Их ограничения представлены в таблице 5.5.1.

Таблица 5.5.1 - Ограничения полей таблицы Edition\_department

|  |  |  |
| --- | --- | --- |
| *Имя поля* | *Ограничение поля* | *Пояснение* |
| Format | new.Format NOT IN ('Сверхкрупный','Крупный','Средний','Малый','Сверхмалый') | Формат может быть только 5 видов:сверхкрупный, крупный, средний, малый, сверхмалый |
| Type | new.Type\_paper NOT IN ('Офсетная','Мелованная','Дизайнерская') | Тип бумаги может быть только 3 видов: офсетная, мелованная, дизайнерская |
| Type\_binding | new.Type\_binding NOT IN ('Мягкий переплет','Твердый переплет','Брошюровка скобой') | Тип переплета может быть только 3 видов: мягкий переплет, твердый переплет, брошюровка скобой |

## 5.6 Ограничения таблицы Publish\_house

Для ограничения полей таблицы Publish\_house были выбраны следующие поля: Scale, Geo\_indication, Nature\_info, INN. Их ограничения представлены в таблице 5.6.1.

Таблица 5.6.1 - Ограничения полей таблицы Publish\_house

|  |  |  |
| --- | --- | --- |
| *Имя поля* | *Ограничение поля* | *Пояснение* |
| Scale | new.Scale NOT IN ('Крупный','Средний','Малый') | Масштаб может быть только 3 видов: крупный, средний, малый |
| Geo\_indication | new.Geo\_indication NOT IN ('Местный','Национальный','Транснациональный') | Территор. признак может быть только 3 видов: местный, национальный, транснациональный |
| Nature\_info | new.Nature\_info NOT IN ('Текстовые книги','Электронные издания','Картографическая продукция') | Природа информации может быть только 3 видов: текстовые книги, электронные издания, картографическая продукция |
| INN | CHAR\_LENGTH(CAST(new.INN AS CHAR)) <> 10 | Поле ИНН не может содержать больше 10 цифр |

# 6 Организация ввода данных

**Задание:** Необходимо ввести данные в таблицы БД для будущей демонстрации работы SQL запросов:

* данные не должны носить фривольный характер;
* конкретные значения атрибутов должны быть максимально приближены к реальности;
* обязательно соблюдать ограничения ссылочной целостности. Это означает, что атрибуты, отмеченные в подчиненной таблице спецификатором FK (внешний ключ), не могут принимать значения, которые не существуют среди значений соответствующего атрибута родительской таблицы. На диаграмме линия связи «предок-потомок» отмечается точкой со стороны подчиненной таблицы;
* подчиненная таблица должна содержать несколько записей, соответствующих одной записи родительской таблицы.
* для соблюдения правил ссылочной целостности рекомендуется такая же последовательность заполнения таблиц от предков к потомкам.

Данные заполняем ориентируясь на источник [[2](#_Список_использованных_источников)], [[3](#_Список_использованных_источников)], [[4](#_Список_использованных_источников)], [[5](#_Список_использованных_источников)].

## 6.1 Данные таблицы Avtor

Введенные данные, удовлетворяющие ТЗ для таблицы Avtor на рисунке 6.1.1.



Рисунок 6.1.1 - Введенные данные в таблицу Avtor

## 6.2 Данные таблицы Book

Введенные данные, удовлетворяющие ТЗ для таблицы Book на рисунке 6.2.1.



Рисунок 6.2.1 - Введенные данные в таблицу Book

## 6.3 Данные таблицы Authorship

Введенные данные, удовлетворяющие ТЗ для таблицы Authorship на рисунке 6.3.1.



Рисунок 6.3.1 - Введенные данные в таблицу Authorship

## 

## 6.4 Данные таблицы Edition\_department

Введенные данные, удовлетворяющие ТЗ для таблицы Edition\_department на рисунке 6.4.1.



Рисунок 6.4.1 - Введенные данные в таблицу Edition\_department

## 6.5 Данные таблицы Stock\_department

Введенные данные, удовлетворяющие ТЗ для таблицы Stock\_department на рисунке 6.5.1.

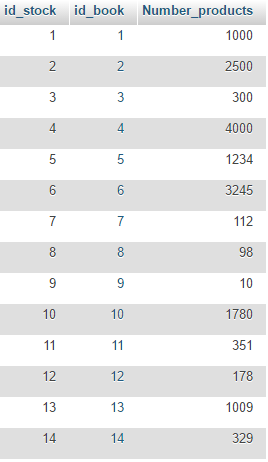


Рисунок 6.5.1 - Введенные данные в таблицу Stock\_department

## 6.6 Данные таблицы Sale\_department

Введенные данные, удовлетворяющие ТЗ для таблицы Sale\_department на рисунке 6.6.1.

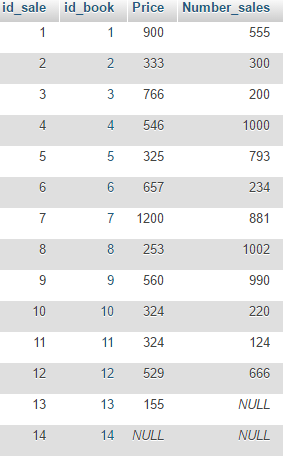


Рисунок 6.6.1 - Введенные данные в таблицу Sale\_department

## 6.7 Данные таблицы Marketing\_department

Введенные данные, удовлетворяющие ТЗ для таблицы Marketing\_department на рисунке 6.7.1.

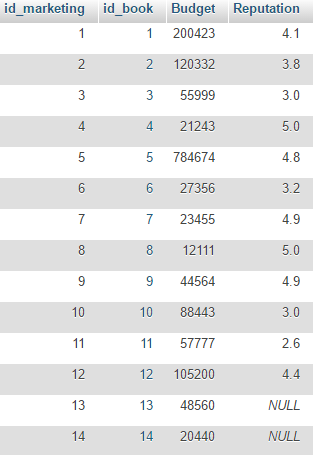


Рисунок 6.7.1 - Введенные данные в таблицу Marketing\_department

## 6.8 Данные таблицы Translation\_department

Введенные данные, удовлетворяющие ТЗ для таблицы Translation\_department на рисунке 6.8.1.



Рисунок 6.8.1 - Введенные данные в таблицу Translation\_department

## 6.9 Данные таблицы Group\_departments

Введенные данные, удовлетворяющие ТЗ для таблицы Group\_departments на рисунке 6.9.1.

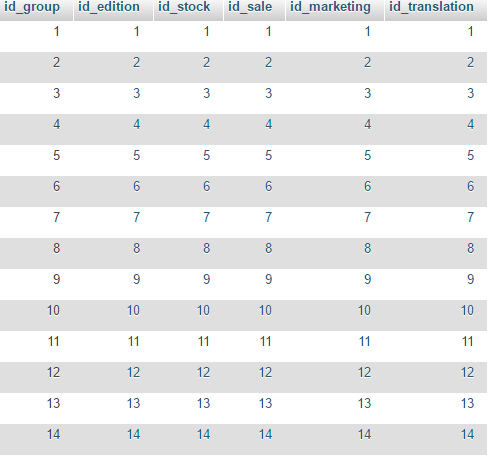


Рисунок 6.9.1 - Введенные данные в таблицу Group\_departments

## 6.10 Данные таблицы Publish\_house

Введенные данные, удовлетворяющие ТЗ для таблицы Publish\_house на рисунке 6.10.1

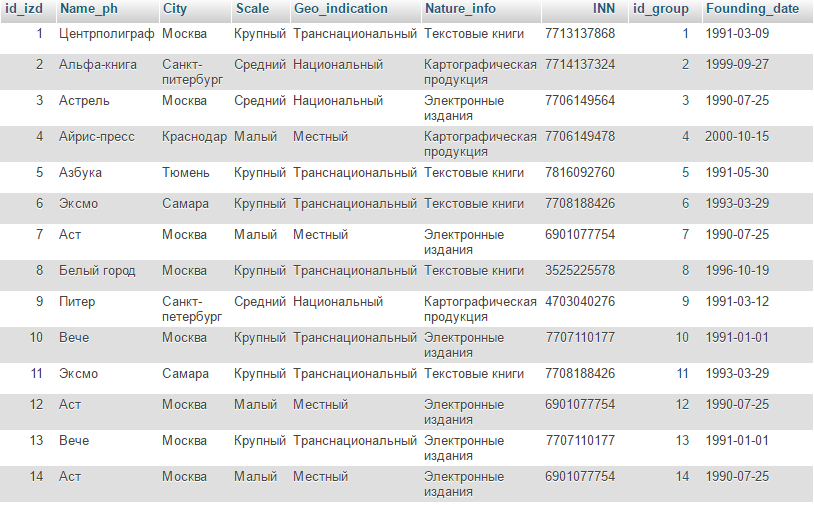
****

Рисунок 6.10.1 - Введенные данные в таблицу Publish\_house

# 7 Запросы

При необходимости просмотреть, добавить, изменить или удалить данные из базы данных, удобно использовать SQL запросы. Запросы можно использовать для фильтрации данных, выполнения расчетов на основе данных и отображения сводных данных. Кроме того, запросы позволяют автоматизировать выполнение многих задач управления данными и просматривать изменения в данных перед их использованием.

SQL запрос, представляет собой обращение к данным для получения информации и выполнения действий с данными. Запрос можно использовать для получения ответа на поставленный вопрос, выполнения расчетов, объединения данных из разных таблиц или же для добавления, изменения или удаления данных в таблице. Запросы, используемые для извлечения данных из таблицы или выполнения расчетов, называются запросами на выборку. Запросы, используемые для добавления, изменения или удаления данных, называются запросами на изменение.

Для выполнения запросов была использован сайт «Политерм» [[6](#_Список_использованных_источников)].

Осуществляем первый запрос: найти все книги, у которых количество страниц варьируется от 100 до 450 страниц. Результаты запроса показаны на рисунке 7.1.

SELECT \*

# выбираем все записи

FROM Book B

# из таблицы Book

WHERE B.Number\_pages BETWEEN 100 AND 450;

# которые удовлетворяют условию BETWEEN 100 AND 450

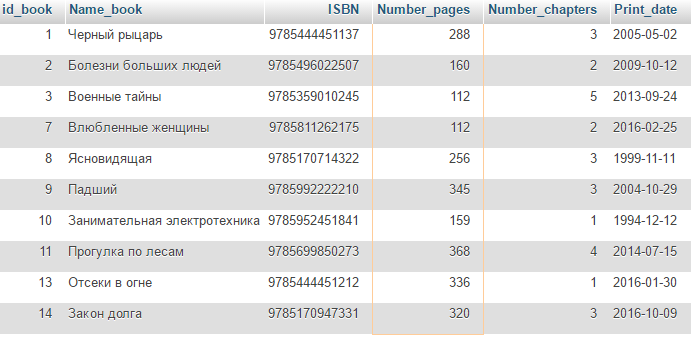


Рисунок 7.1 - Результат первого запроса

Осуществляем второй запрос: найти сумму бюджета всех книг, у которых оценка варьируется от 3 до 5. Результат показан на рисунке 7.2.

SELECT SUM(Budget)

# используем агрегирующею функцию Sum, которая сложит весь

# бюджет книг

FROM Marketing\_department M

# из таблицы Marketing\_department

WHERE M.Reputation BETWEEN 3 AND 5;

# используем условие, где поле M.Reputation удовлетворяет

# условию BETWEEN 3 AND 5



Рисунок 7.2 – Результат второго запроса

Осуществляем третий запрос: найдем все языки оригинала и сгруппируем их. Результат запроса показан на рисунке 7.3.

SELECT Original\_language, COUNT(\*)

# выберем поле Original\_language, и агрегирующею функцию

# COUNT(\*), которая будет считать количество записей

FROM Translation\_department

# из таблицы Translation\_department

GROUP BY Original\_language;

# и сгрупперуем результат запроса по полю Original\_language,

# c помощью конструкции GROUP BY

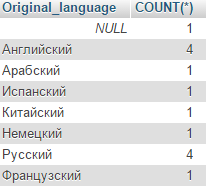


Рисунок 7.3 - Результат третьего запроса

Осуществляем четвертый запрос: найдем всех авторов, которые не написали ни дну книгу. Результат выполнения SQL запроса представлен на рисунке 7.4.

SELECT A.id\_avtor

# выбираем поле id\_avtor

FROM Avtor A

# из таблицы Avtor

LEFT JOIN Authorship ON A.id\_avtor = Authorship.id\_avtor

# используем оператор левого присоединения LEFT JOIN,

# который присоединит 2 таблицы по ключам

WHERE Authorship.id\_book IS NULL;

# так же используем условие,

# где поле Authorship.id\_book равно NULL



Рисунок 7.4 – Результат четвёртого запроса

Осуществляем пятый запрос: выведем всех авторов, которые написали книгу и присоединим таблицу с книгами, которые они написали. Результат запроса показан на рисунке 7.5.

SELECT \*

# выберем все записи

FROM Avtor

# из таблицы Avtor

INNER JOIN Authorship ON Avtor.id\_avtor = Authorship.id\_avtor

# присоединим таблицу Authorship,

# используя оператор INNER JOIN

INNER JOIN Book ON Authorship.id\_book = Book.id\_book

# присоединим таблицу Book

WHERE Authorship.id\_book is NOT NULL;

# используем условие, где поле Authorship.id\_book равно NULL

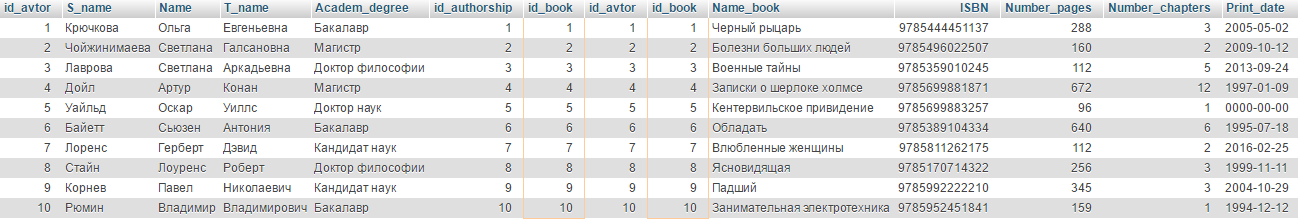


Рисунок 7.5 – Результат пятого запроса

Осуществляем шестой запрос: найдем всех авторов, которые написали больше чем одну книгу. Результат запроса представлен на рисунке 7.6.

SELECT S\_name, Name, T\_name, COUNT(\*)

# выберем поля S\_name, Name, T\_name

# и агрегирующею функцию COUNT(\*)

FROM Avtor A

# из таблицы Avtor

INNER JOIN Authorship Au ON A.id\_avtor = Au.id\_avtor

# присоединим к таблице Avtor таблицу Authorship

GROUP BY A.id\_avtor HAVING COUNT(\*) > 1;

# сгруппируем поле A.id\_avtor и используем условие отбора

# при помощи конструкции HAVING



Рисунок 7.6 - Результат шестого запроса

Осуществляем седьмой запрос: найдем все книги, которые прошли через все отделы издательства. Результат запроса представлен на рисунке 7.7.

SELECT Name\_book, ISBN

# выберем поля Name\_book, ISBN

FROM Book B

# из таблицы Book

INNER JOIN Edition\_department E ON B.id\_book = E.id\_book

# присоединим к таблице Book таблицу Edition\_department

INNER JOIN Stock\_department ST ON B.id\_book = ST.id\_book

# присоединим к таблице Book таблицу Stock\_department

INNER JOIN Sale\_department S ON B.id\_book = S.id\_book

# присоединим к таблице Book таблицу Sale\_department

INNER JOIN Marketing\_department M ON B.id\_book = M.id\_book

# присоединим к таблице Book таблицу Marketing\_department

INNER JOIN Translation\_department T ON B.id\_book = T.id\_book

# присоединим к таблице Book таблицу Translation\_department

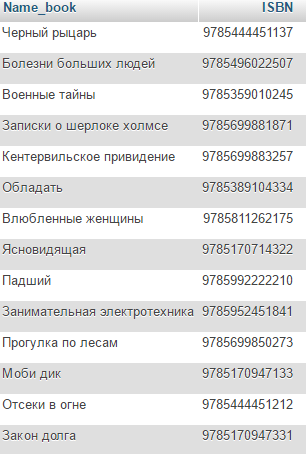


Рисунок 7.7 - Результат седьмого запроса

Осуществляем восьмой запрос, состоящий из двух: в первом запросе найдем все книги, у которых количество страниц варьируется от 100 до 370 штук; во втором запросе найдем все книги, у которых количество глав варьируется от 1 до 4. Результаты двух запросов объединим. Результат запроса представлен на рисунке 7.8

SELECT Name\_book,Number\_pages

# выберем поля Name\_book,Number\_pages

FROM Book B

# из таблицы Book

WHERE B.Number\_pages BETWEEN 100 AND 370

# используем условие, где поле B.Number\_pages удовлетворяет

# условию BETWEEN 100 AND 370

Union

# объединяем 2 запроса при помощи конструкции Union

SELECT Name\_book,Number\_chapters

# выбираем поля Name\_book,Number\_chapters

FROM Book B

# из таблицы Book

WHERE B.Number\_chapters BETWEEN 1 AND 4;

# используем условие, где поле B.Number\_chapters

# удовлетворяет условию BETWEEN 1 AND 4



Рисунок 7.8 - Результат восьмого запроса

Осуществляем девятый запрос: найдем язык оригинала и количество книг написанных на нем, сгруппируем и отсортируем по убыванию. Результат запроса представлен на рисунке 7.9

SELECT Original\_language, COUNT(\*)

# выберем поле Original\_language

# и агрегирующею функцию COUNT(\*)

FROM Translation\_department

# из таблицы Translation\_department

GROUP BY Original\_language ORDER BY Original\_language DESC;

# группируем по полю Original\_language, при помощи GROUP BY

# и сортируем по полю Original\_language, при помощи ORDER BY

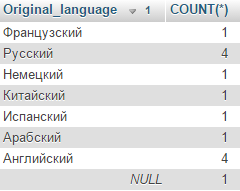


Рисунок 7.9 - Результат девятого запроса

Осуществляем десятый запрос: найдем всю информацию о книге, чей личный номер равен 6 . Результат запроса представлен на рисунке 7.10

SELECT S\_name,T\_name,Name\_book,ISBN,Number\_pages,Number\_chapters,Print\_date,Format,Type\_paper,Type\_binding,Price,Number\_sales,Budget,Reputation

#выберем поля S\_name,T\_name,Name\_book,ISBN,Number\_pages,

# Number\_chapters,Print\_date,Format,Type\_paper,Type\_binding,

# Price,Number\_sales,Budget,Reputation

FROM Book B

# из таблицы Book

INNER JOIN Authorship AU ON B.id\_book = AU.id\_book

# присоединим к таблице Book таблицу Authorship

INNER JOIN Avtor A ON AU.id\_avtor = A.id\_avtor

# присоединим к таблице Authorship таблицу Avtor

INNER JOIN Edition\_department E ON B.id\_book = E.id\_book

# присоединим к таблице Book таблицу Edition\_department

INNER JOIN Sale\_department S ON B.id\_book = S.id\_book

# присоединим к таблице Book таблицу Sale\_department

INNER JOIN Marketing\_department M ON B.id\_book = M.id\_book

# присоединим к таблице Book таблицу Marketing\_department

WHERE B.id\_book = 6;

# используем условие, где поле B.id\_book,

# удовлетворяет условию B.id\_book = 6



Рисунок 7.10 - Результат десятого запроса

Осуществляем одиннадцатый запрос: найдем издательства, у которых территориальный признак равен «Транснациональный», а название начинается на «А» и книги которые эти издательства издали. Результат запроса представлен на рисунке 7.11

SELECT Name\_book,Name\_ph,City,Scale,Geo\_indication,Nature\_info,INN

# выберем поля Name\_book,Name\_ph,City,Scale,

# Geo\_indication,Nature\_info,INN

FROM Publish\_house P

# из таблицы Publish\_house

INNER JOIN Group\_departments G ON P.id\_group = G.id\_group

# присоединим к таблице Publish\_house

# таблицу Group\_departments

INNER JOIN Edition\_department E ON G.id\_edition = E.id\_edition

# присоединим к таблице Group\_departments

# таблицу Edition\_department

INNER JOIN Book B ON E.id\_book = B.id\_book

# присоединим к таблице Edition\_department таблицу Book

WHERE P.Geo\_indication = 'Транснациональный' AND P.Name\_ph LIKE 'а%';

# используем условие, где поле P.Geo\_indication,

# удовлетворяет условию

# P.Geo\_indication = 'Транснациональный'

# и удовлетворяет условию

# P.Name\_ph LIKE 'а%'



Рисунок 7.11 - Результат одиннадцатого запроса

Осуществляем двенадцатый запрос: найдем все данные о книги, номер который найдем из подзапроса. Результат запроса представлен на рисунке 7.12

SELECT S\_name,T\_name,Name\_book,ISBN,Number\_pages,Number\_chapters,

Print\_date,Format,Type\_paper,Type\_binding,Price,

Number\_sales,Budget,Reputation

# выберем поля S\_name,T\_name,Name\_book,ISBN,Number\_pages,

# Print\_date,Format,Type\_paper,Type\_binding,Price,

# Number\_sales,Budget,Reputation

FROM Book B

# из таблицы Book

LEFT JOIN Authorship AU ON B.id\_book = AU.id\_book

# присоединяем к таблице Book таблицу Authorship

LEFT JOIN Avtor A ON AU.id\_avtor = A.id\_avtor

# присоединяем к таблице Authorship таблицу Avtor

LEFT JOIN Edition\_department E ON B.id\_book = E.id\_book

# присоединяем к таблице Book таблицу Edition\_department

LEFT JOIN Sale\_department S ON B.id\_book = S.id\_book

# присоединяем к таблице Book таблицу Sale\_department

LEFT JOIN Marketing\_department M ON B.id\_book = M.id\_book

# присоединяем к таблице Book таблицу Marketing\_department

WHERE B.id\_book =

# используем условие, которое является подзапросом

(SELECT COUNT(\*)

# выберем все поля

FROM Book B

# из таблицы Book

WHERE B.Number\_pages BETWEEN 50 AND 800);

# используем условие, где поле B.Number\_pages,

# удовлетворяет условию BETWEEN 50 AND 800



Рисунок 7.12 - Результат двенадцатого запроса

# Заключение

В результате выполнения курсового проекта была построена концептуальная модель процесса «Издание» и описана структура базы данных.

Были построены таблицы с детальным описанием всех сущностей, атрибутов и доменов, созданы ключи, ограничения, последовательности, запросы. По данной предметной области и ее модели возможна реализация этой модели в СУБД.

Также в ходе данной курсовой работы была изучена программная среда phpMyAdmin.

Были изучены и применены принципы работы SQL запросов.

# Список использованных источников

1. <http://www.mysql.ru/docs/>
2. <http://www.list-org.com/>
3. <http://www.gnpbu.ru/sprav/pub/list/>
4. <http://knigi.tomsk.ru/>
5. <https://www.livelib.ru/>
6. https://www.politerm.com/