СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 6](#_Toc515496598)

[1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ 7](#_Toc515496599)

[1.1 Анализ предметной области и краткая постановка задачи. 7](#_Toc515496600)

[1.2 Определение основных бизнес-функций высоконагруженной информационной системы 8](#_Toc515496601)

[1.3 Определение основных функций интерфейса клиентской части информационной системы 8](#_Toc515496602)

[2 РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ ВЫСОКОНАГРУЖЕННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ 9](#_Toc515496603)

[2.1 Логическое и физическое моделирование данных 9](#_Toc515496604)

[2.2 Разработка поддержки целостности данных 10](#_Toc515496605)

[2.3 Реализация бизнес-функций информационной системы на стороне сервера MySQL 18](#_Toc515496606)

[ВЫВОДЫ 20](#_Toc515496607)

[ПЕРЕЧЕНЬ ССЫЛОК 22](#_Toc515496608)

ВВЕДЕНИЕ

Автоматизация бизнес процессов является актуальной в любой сфере деятельности человека. Почувствовав, что использование автоматизированных средств во много раз облегчает работу и при этом только улучшает её качество, достаточно трудно отказаться от приобретения и внедрения компьютера.

В настоящее время все большее внимание уделяется техническим задачам поддержки работы высоконагруженных систем хранения данных. Это связано, в первую очередь, с ростом количества пользователей сети Интернет и числа глобальных единых сервисов в виде поисковых систем, социальных сетей и сайтов, поддерживающих электронную коммерцию. Рост числа пользователей меняет сценарии использования информационных систем и нагрузку доступа к базам данных. Отмеченные факторы имеют стойкую тенденцию к росту, что усиливает интерес к области построения высоконагруженных и распределенных систем. В связи с этим задачи масштабирования серверных решений по нагрузке и оптимизации доступа к информации, хранящейся в базах данных, является актуальной.

Целью курсового проектирования является приобретение студентами практических навыков создания высоконагруженных систем хранения данных в виде информационных систем, разработки баз данных, масштабирования, создания серверной части информационной системы и ее оптимизация по критерию минимизации времени доступа к информации. Данная программа обладает всеми необходимыми компонентами для выполнения любых задач связанных с поиском и обработкой информации, удобным графическим интерфейсом.

Использование Интернета в торговой деятельности также достаточно перспективное направление, так как позволяет за счет экономии средств (нет необходимости в торговых помещениях, количество персонала минимально необходимое и т.д.) существенно повысить уровень доходов. К тому же подобный вид торговли весьма удобен и для клиентов.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ
   1. Анализ предметной области и краткая постановка задачи.

Объектом исследования является высоконагруженная информационная система «Библиотека». Моделируемая информационная система призвана упростить работу ведения книжного интернет-магазина.

Пользователь имеет возможность заказать книгу в онлайн-библиотеке, заполнив форму заказа после выбора нужного продукта. Данная форма доступна только авторизированным пользователям - клиентам.

Данная информационная система должна иметь подключение к сети общего пользования - интернет. Все компоненты ИС «Библиотека» будут находиться на одном сервере.

Всего имеется 2 типа пользователей, у каждого из которых свои функции:

* незарегистрированный пользователь;
* зарегистрированный пользователь;

Информационный ресурс должен обеспечивать возможность выполнения функций авторизации пользователя и управления продукцией магазина.

Незарегистрированные пользователи должны иметь следующие возможности:

* регистрация в системе;
* просмотр домашней страницы;
* просмотр информации обо всех книгах;
* просмотр информации обо всех писательствах;
* просмотр информации обо всех издательствах.

Зарегистрированные пользователи имеют аналогичные возможности, а также:

* функцию создания и возврата заказа;
* просмотр информации о своих заказах;
* редактировать персональную информацию.
  1. Определение основных бизнес-функций высоконагруженной информационной системы

Основные бизнес-функции высоконагруженной информационной системы «Библиотека»:

* Регистрация пользователя в системе;
* Поиск информации о книге, писателе, издательстве;
* Заказ книги.
  1. Определение основных функций интерфейса клиентской части информационной системы

Форма поиска – текстовое поле, в которое пользователь вводит текст для поиска, кнопка «Поиск»

Форма регистрации – текстовые поля для ввода имени, логина, номера телефона, пароля и кнопка «Регистрация»

Форма заказа книги – кнопка «Заказать»

Форма просмотра заказанных книг – кнопка «Вернуть заказ»

Клиентская часть информационной системы «Библиотека», которая базируется на бизнес логике в соответствии с серверной частью должна выполнять функции, которые будут предоставлять пользователю необходимую информацию и давать возможность взаимодействовать с системой:

* функция вывода результатов по тексту поиска;
* функция вывода детальной информации о книге, писателе, издательстве;
* функция оформления заказа;
* функция вывода списка книг, взятых в библиотеке;
* функция просмотра информации о пользователе.

1. РАЗРАБОТКА СЕРВЕРНОЙ ЧАСТИ ВЫСОКОНАГРУЖЕННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ
   1. Логическое и физическое моделирование данных

Основой для информационной системы является проектирование БД. Именно БД позволяет эксплуатировать ИС, выполнять ее текущее обслуживание, модифицировать и обрабатывать информацию о работе предприятия, принимать информационные потоки.

Для данной предметной области была разработана модель БД в виде ER-диаграммы (рисунок 3.1), в которой присутствуют следующие сущности:

* book – предназначена для хранения названия товара (title), цены (price), жанра(genre), описания (description). publisher\_ref – внешний ключи, id – первичный ключ.
* writer – содержит имя авторов книг (full\_name), дату рождения (birth\_date). id – первичный ключ.
* participation – содержит writer\_ref, book\_ref – составной ключ, внешние ключи.
* publisher – содержит информацию о издательстве: название издательства (title), город (city). id – первичный ключ.
* order – предназначена для хранения информации о заказе. Содержит такие атрибуты как: дата заказа (order\_date), дату возврата (return\_date). А также первичный ключ – id, внешние ключи – book\_ref, user\_ref.
* application\_user – сущность, предназначена для хранения логина (user\_name), электронной почты (email) и пароля (password\_hash) пользователей. id – первичный ключ.

Для каждой сущности были определены связи с другими сущностями.

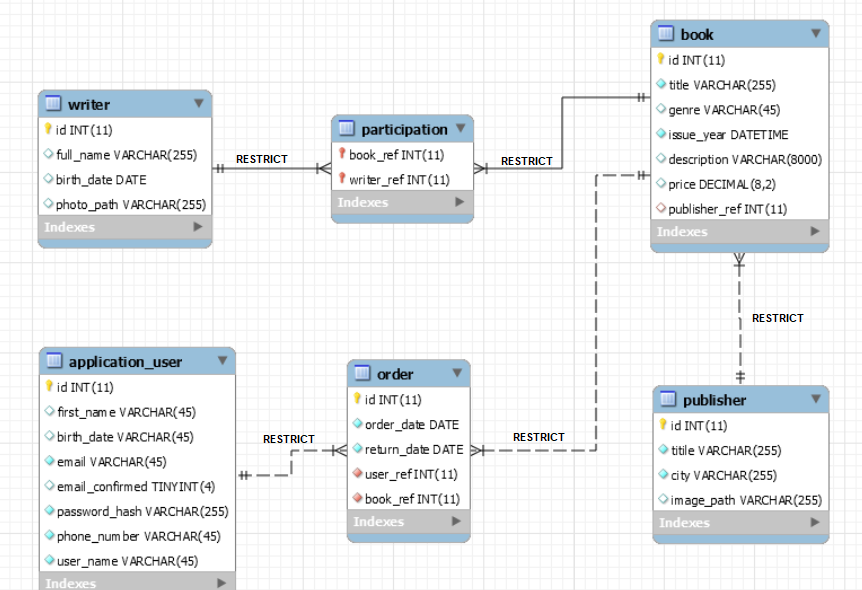


Рисунок 3.1 – EER-диаграмма для базы данных с типом таблиц – InnoDb

* 1. Разработка поддержки целостности данных

Логическая модель данных была преобразована в физическую путем выбора базы данных и задания правил целостности. Они необходимы для описания действий сущностей относительно друг друга, то есть, какие изменения происходят в сущности, если связанная с ней другая сущность была изменена или удалена.

После построения схемы физической модели необходимо задать правила целостности данных, то есть характеристики для каждого атрибута каждой сущности, чтобы обеспечить соответствие информации её внутренней логике, структуре и всем заданным правилам. Целостность БД не гарантирует достоверности содержащейся в ней информации, но обеспечивает, по крайней мере, правдоподобность этой информации, отвергая заведомо невероятные, невозможные значения.

Правила целостности для данной предметной области для всех связей являются по умолчанию “RESTRICT” и представлены на ниже:

Сущность book

`id` int(11) NOT NULL,

`title` varchar(255) NOT NULL,

`genre` varchar(45) DEFAULT NULL,

`issue\_year` datetime NOT NULL,

`description` varchar(8000) DEFAULT NULL,

`price` decimal(8,2) DEFAULT NULL,

`publisher\_ref` int(11) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id\_UNIQUE` (`id`),

KEY `publisher\_fk\_idx` (`publisher\_ref`),

CONSTRAINT `book\_publisher\_fk` FOREIGN KEY (`publisher\_ref`) REFERENCES `publisher` (`id`)

Сущность writer

`id` int(11) NOT NULL,

`full\_name` varchar(255) DEFAULT NULL,

`birth\_date` date DEFAULT NULL,

`photo\_path` varchar(255) DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id\_UNIQUE` (`id`)

Сущность participant

`book\_ref` int(11) NOT NULL,

`writer\_ref` int(11) NOT NULL,

PRIMARY KEY (`book\_ref`,`writer\_ref`),

KEY `participation\_writer\_fk` (`writer\_ref`),

CONSTRAINT `participation\_book\_fk` FOREIGN KEY (`book\_ref`) REFERENCES `book` (`id`),

CONSTRAINT `participation\_writer\_fk` FOREIGN KEY (`writer\_ref`) REFERENCES `writer` (`id`)

Сущность publisher

`id` int(11) NOT NULL,

`titile` varchar(255) NOT NULL,

`city` varchar(255) NOT NULL,

`image\_path` varchar(255) DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id\_UNIQUE` (`id`)

Сущность order

`id` int(11) NOT NULL,

`order\_date` date NOT NULL,

`return\_date` date NOT NULL,

`user\_ref` int(11) NOT NULL,

`book\_ref` int(11) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id\_UNIQUE` (`id`),

KEY `user\_fk\_idx` (`user\_ref`),

KEY `book\_fk\_idx` (`book\_ref`),

CONSTRAINT `order\_book\_fk` FOREIGN KEY (`book\_ref`) REFERENCES `book` (`id`),

CONSTRAINT `order\_user\_fk` FOREIGN KEY (`user\_ref`) REFERENCES `application\_user` (`id`)

Сущность application\_user

`id` int(11) NOT NULL,

`first\_name` varchar(45) DEFAULT NULL,

`birth\_date` varchar(45) DEFAULT NULL,

`email` varchar(45) NOT NULL,

`email\_confirmed` tinyint(4) DEFAULT NULL,

`password\_hash` varchar(255) NOT NULL,

`phone\_number` varchar(45) NOT NULL,

`user\_name` varchar(45) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id\_UNIQUE` (`id`),

UNIQUE KEY `user\_name\_UNIQUE` (`user\_name`),

UNIQUE KEY `email\_UNIQUE` (`email`)

Также была разработана база данных с типом таблиц – MyISAM (Рисунок 3.2)

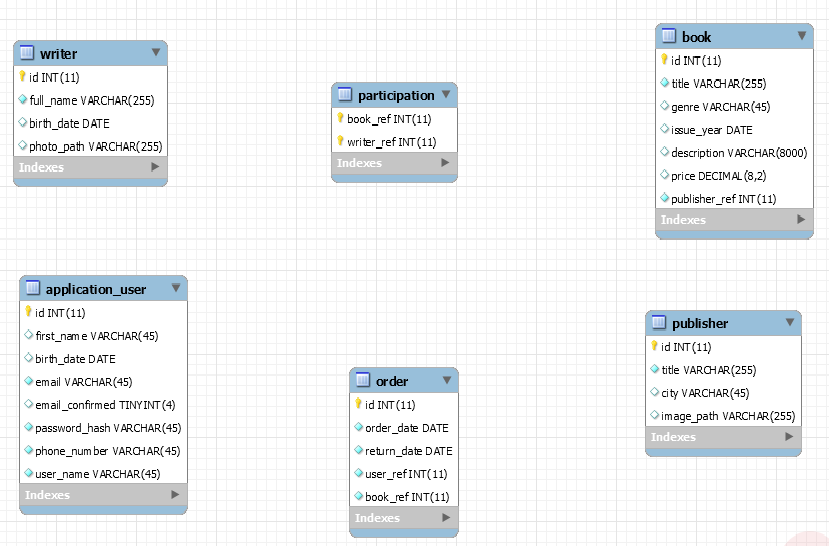
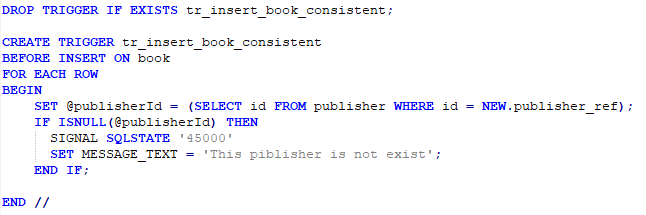


Рисунок 3.2 – EER-диаграмма для базы данных с типом таблиц – MyISAM

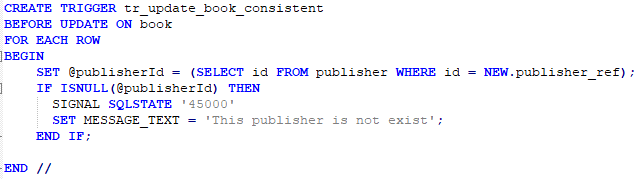
Так как таблицы типа “MyISAM” не поддерживают ограничения внешних ключей, целостность данных обеспечивают соответствующие триггеры:

Для таблицы book:

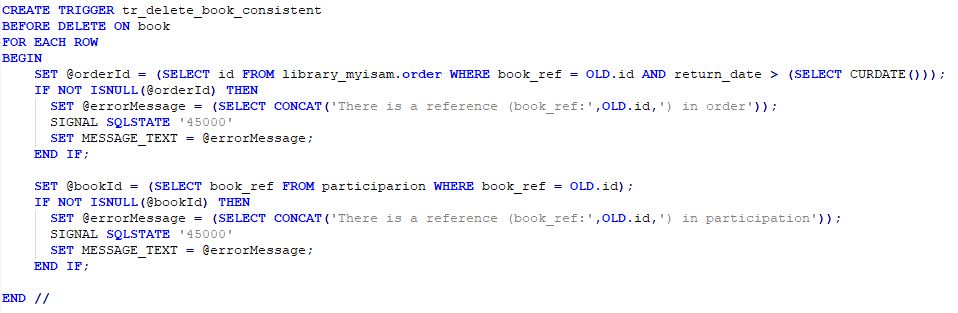
Триггер обеспечивающий целостность связи с таблицей publisher при добавлении новой записи:



Триггер обеспечивающий целостность связи с таблицей publisher при обновлении существующей записи:

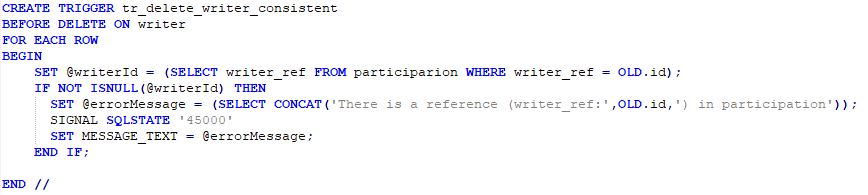


Триггер обеспечивающий целостность связи с таблицами order и participation при удалении записей:



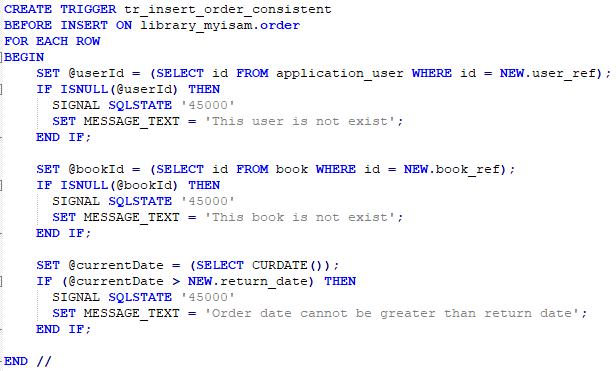
Для таблицы writer:

Триггер обеспечивающий целостность связи с таблицей participation при удалении записей:

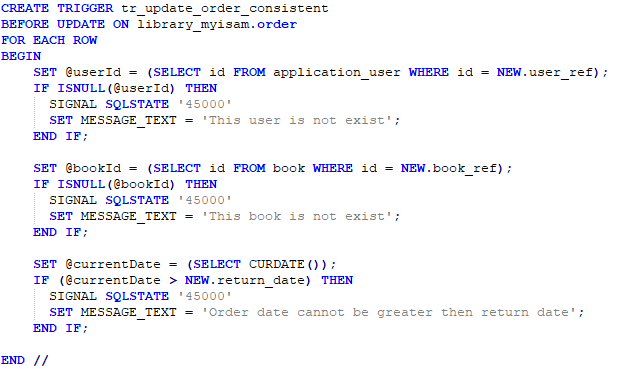


Для таблицы order:

Триггер обеспечивающий целостность связи с таблицами application\_user и book при добавлении новой записи:

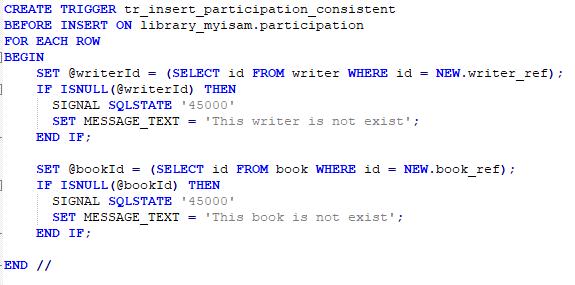


Триггер обеспечивающий целостность связи с таблицами application\_user и book при обновлении существующей записи:

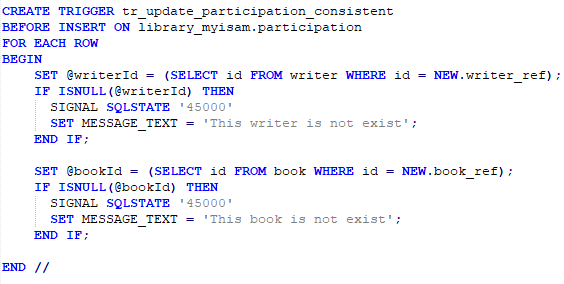


Для таблицы participation:

Триггер обеспечивающий целостность связи с таблицами writer\_user и book при добавлении новой записи:

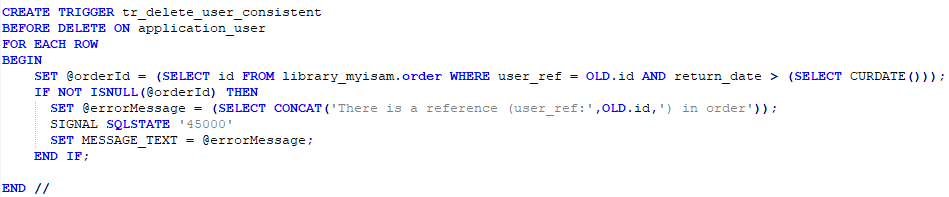


Триггер обеспечивающий целостность связи с таблицами writer\_user и book при обновлении существующей записи:



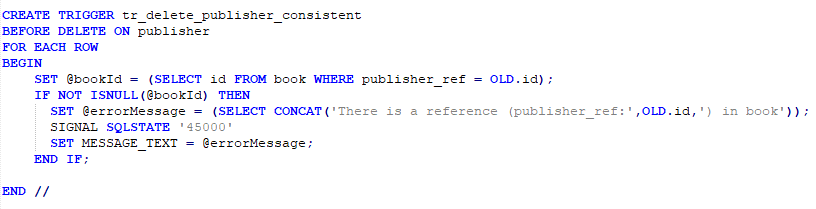
Для таблицы application\_user:

Триггер обеспечивающий целостность связи с таблицей order при удалении записи:



Для таблицы publisher:

Триггер обеспечивающий целостность связи с таблицей book при удалении записи:



* 1. Реализация бизнес-функций информационной системы на стороне сервера MySQL

Для реализации основных бизнес-функций информационной системы были использованы процедуры, функции и представления.

* функция «Заказ книги»:

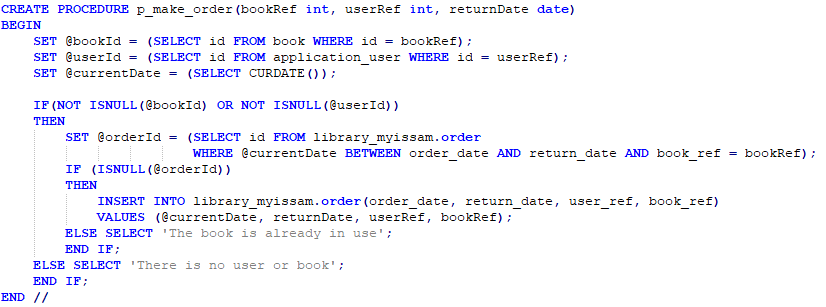


Рисунок 3.3 – Код процедуры для функции «Заказ книги»

* функция «Поиск информации о книге»:

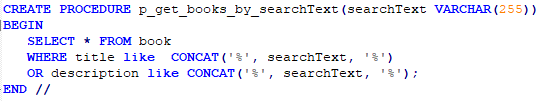


Рисунок 3.4 – Код процедуры для функции «Поиск информации о книге»

* функция «Поиск информации о писателе»:

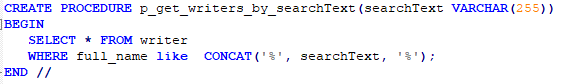


Рисунок 3.5 – Код процедуры для функции «Поиск информации о писателе»

* функция «Поиск информации о издательстве»:

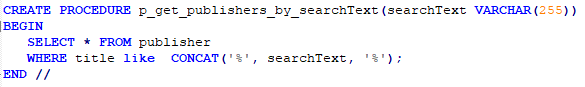


Рисунок 3.6 – Код процедуры для функции «Поиск информации о писателе»

* функция «Поиск доступных книг»:

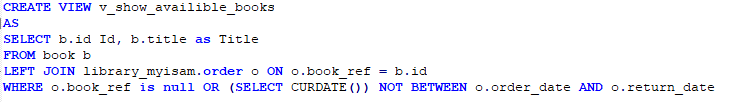


Рисунок 3.7 – Код представления для функции «Поиск доступных книг»

ВЫВОДЫ

В ходе выполнения междисциплинарной курсовой работы было выполнено проектирование высоконагруженной информационной системы «Библиотека».

Высоконагруженная информационная система была спроектирована и разработана в среде для визуального проектирования баз данных «MySQL Workbench» - инструмент, который позволяет наглядно представить модель базы данных в графическом виде, а также является удобным редактором SQL запросов, позволяющий сразу же отправлять их к серверу и получать ответ в виде таблицы.

Для устранения из базы избыточных функциональных зависимостей между атрибутами (полями таблиц) была проведена нормализация базы данных. Все таблицы находятся не менее, чем в 3й нормальной форме (Отношение находится в 3НФ, когда находится во 2НФ и каждый не ключевой атрибут не транзитивно зависит от первичного ключа.)

Целостность данных была сохранена, путем написания триггеров, которые также обеспечивают ограничение внешних ключей для таблиц типа «MyISAM». Для таблиц типа «InnoDB» ограничение внешних ключей сохранено путем установки опции «RESTRICT» на внешние ключи и добавлением свойства «NOT NULL» к соответствующим атрибутам.

Обработка данных в ИС «Библиотека» производится в многопользовательском режиме с разграничением прав доступа для обычного пользователя и зарегистрированного.

Разработанное информационная система выполняет следующие функции:

* регистрация пользователей;
* просмотр информации о книгах;
* просмотр информации о авторах;
* просмотр информации о издательствах;
* оформление заказа;
* просмотр списка заказанных книг;
* изменение информации о пользователях;

Веб-сайт «Библиотека» имеет простой и интуитивно понятный для взаимодействия интерфейс, который отображает все основные функции системы и не требует навыков программирования, что позволяет пользователям заказывать книги в библиотеке онлайн, тем самым экономит клиенту время.

Данное программное обеспечение позволяет упростить работу по сопровождению онлайн библиотеки. Данная программа обладает всеми необходимыми компонентами для выполнения необходимых задач в рамках заданной предметной областью.

ПЕРЕЧЕНЬ ССЫЛОК

1. Методические указания к междисциплинарного курсового проекта по для студентов направления подготовки 6.050101 - Компьютерные науки [Электронное издание] / Сост .: Ю.В. Мищеряков, З.А. Имангулова., Л.В. Колесник - Харьков: ХНУРЭ, 2016. - 30 с.
2. Справочник по MySQL [Электронный ресурс] - Режим доступа URL: https://dev.mysql.com
3. Справочник по Workbench [Электронный ресурс] - Режим доступа URL: https:// dev.mysql.com/doc/workbench/en/
4. Работа с базами данных: учебное пособие / сост. О. Н. Евсеева, А. Б. Шамшев. – Ульяновск: УлГТУ, 2009. – 170 с.

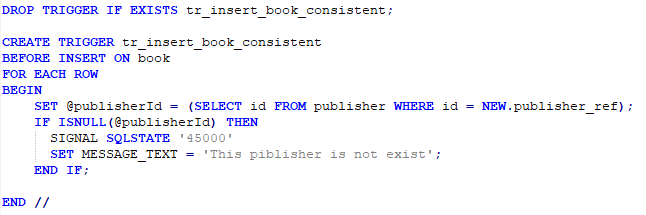
ПРИЛОЖЕНИЕ А

Разработка триггеров для поддержки целостности данных

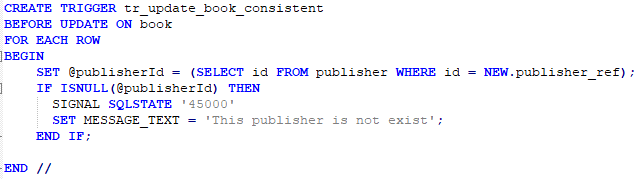
Так как таблицы типа “MyISAM” не поддерживают ограничения внешних ключей, целостность данных обеспечивают соответствующие триггеры:

Для таблицы book:

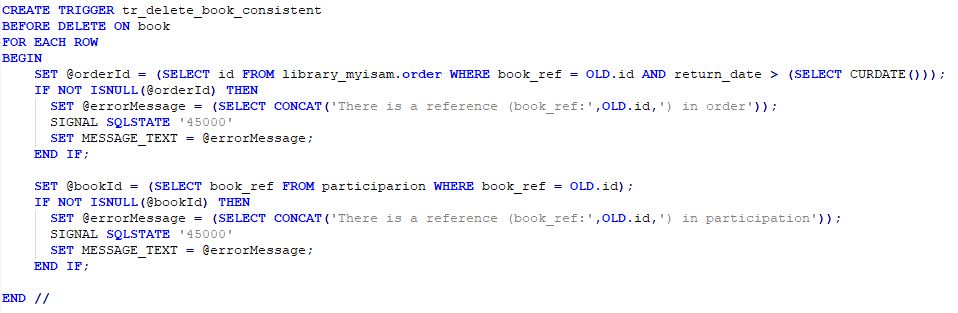
Триггер обеспечивающий целостность связи с таблицей publisher при добавлении новой записи:



Триггер обеспечивающий целостность связи с таблицей publisher при обновлении существующей записи:

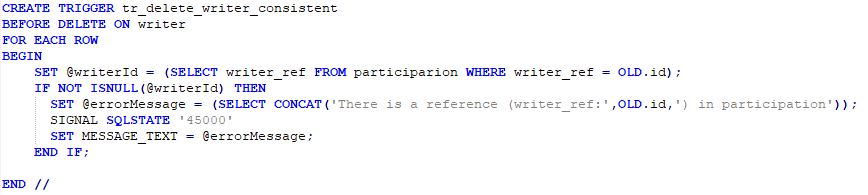


Триггер обеспечивающий целостность связи с таблицами order и participation при удалении записей:



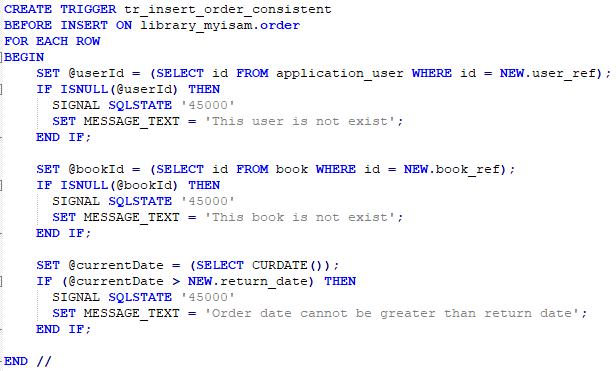
Для таблицы writer:

Триггер обеспечивающий целостность связи с таблицей participation при удалении записей:

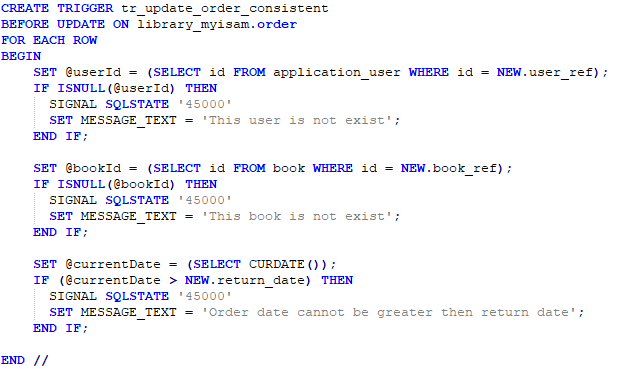


Для таблицы order:

Триггер обеспечивающий целостность связи с таблицами application\_user и book при добавлении новой записи:

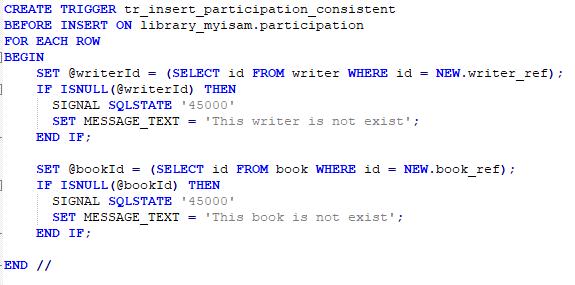


Триггер обеспечивающий целостность связи с таблицами application\_user и book при обновлении существующей записи:

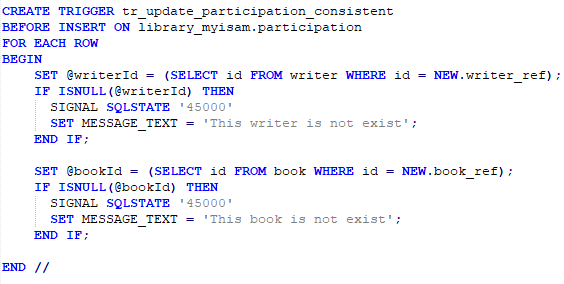


Для таблицы participation:

Триггер обеспечивающий целостность связи с таблицами writer\_user и book при добавлении новой записи:

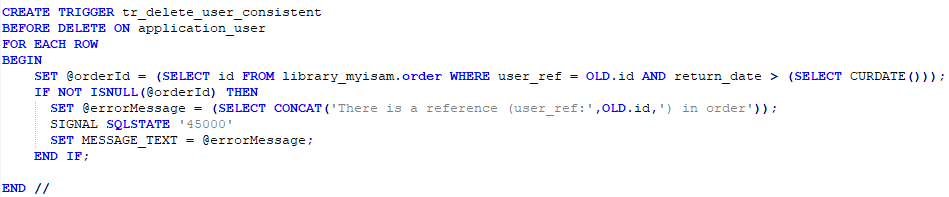


Триггер обеспечивающий целостность связи с таблицами writer\_user и book при обновлении существующей записи:



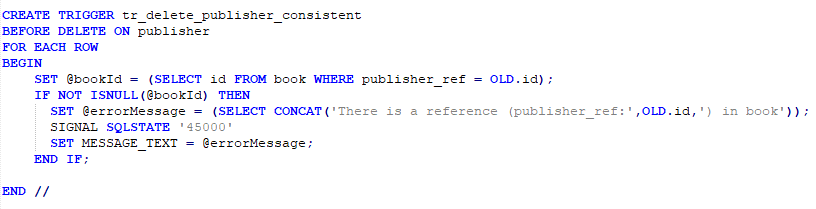
Для таблицы application\_user:

Триггер обеспечивающий целостность связи с таблицей order при удалении записи:



Для таблицы publisher:

Триггер обеспечивающий целостность связи с таблицей book при удалении записи:



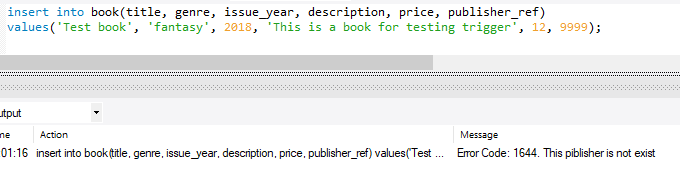


Рисунок А.1 – Результат выполнения вставки записи в таблицу «book» с ссылкой на несуществующее издательство.

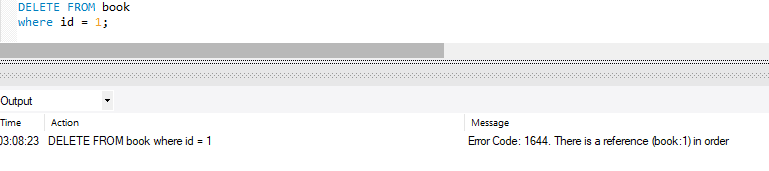


Рисунок А.2 – Результат выполнения удаления таблицы «book», при имеющейся ссылке на эту запись в таблце «order»

ПРИЛОЖЕНИЕ Б

Для реализации основных бизнес-функций информационной системы были использованы процедуры, функции и представления.

* функция «Заказ книги»:

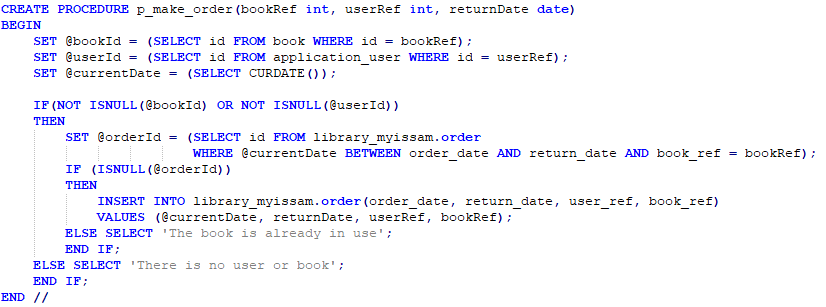


Рисунок Б.1 – Код процедуры для функции «Заказ книги»

* функция «Поиск информации о книге»:

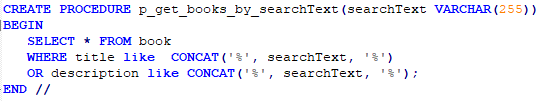


Рисунок Б.2 – Код процедуры для функции «Поиск информации о книге»

* функция «Поиск информации о писателе»:

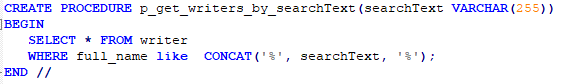


Рисунок Б.3 – Код процедуры для функции «Поиск информации о писателе»

* функция «Поиск информации о издательстве»:

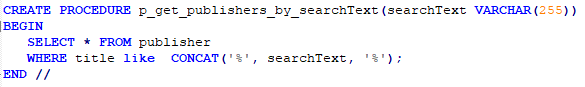


Рисунок Б.4 – Код процедуры для функции «Поиск информации о писателе»

* функция «Поиск доступных книг»:

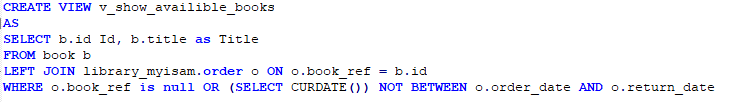
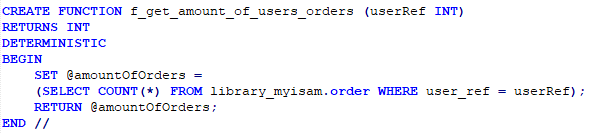


Рисунок Б.5 – Код представления для функции «Поиск доступных книг»

* функция «Просмотр кол-ва заказанных книг у пользователя»



* Рисунок Б.6 – Код функции для функции «Просмотр кол-ва заказанных книг у пользователя»

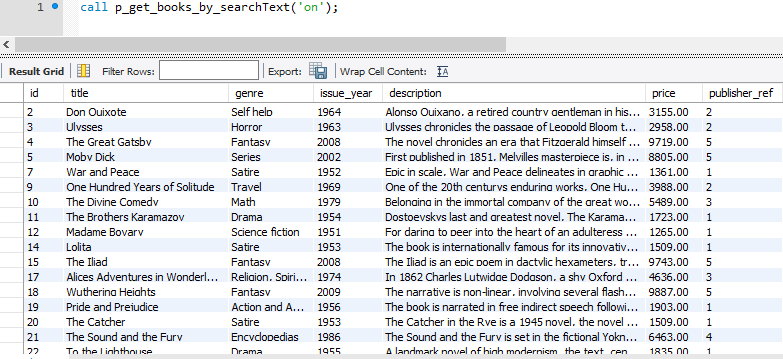


Рисунок Б.7 – Результат выполнения процедуры «Поиск доступных книг»

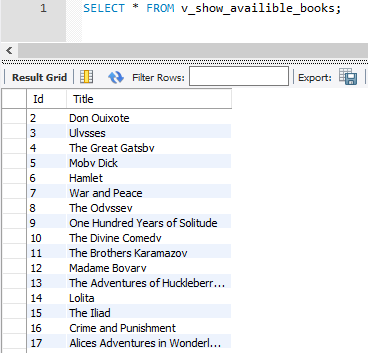


Рисунок Б.8 – Результат выполнения представления «Поиск доступных книг»

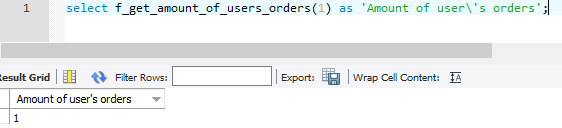


Рисунок Б.9 – Результат выполнения функции «Просмотр кол-ва заказанных книг у пользователя»