Министерство науки и высшей школы Российской Федерации

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Пермский государственный национальный исследовательский университет»

Механико-математический факультет

Кафедра математического обеспечения вычислительных систем

Отчёт о выполнении входного контроля  
по дисциплине «Технологии разработки распределённых приложений»

Работу выполнил  
студент группы ФИТ-1-2-2017 НБ,

4 курса  
Механико-математического факультета  
Ошев Кирилл Владимирович

Работу принял  
доцент кафедры математического   
обеспечения вычислительных систем   
механико-математического факультета,  
к.ф.-м.н., доц. Деменев Алексей Геннадьевич

Пермь, 2020

Оглавление

[1 Постановка задач 3](#_Toc52920619)

[2 Критерии оценивания 5](#_Toc52920620)

[3 Основная часть 6](#_Toc52920621)

[4 Заключение 8](#_Toc52920622)

# Постановка задач

Входной контроль проводится **в рамках самостоятельной работы** и заключается в **написании программы** на объектно-ориентированном языке, в которой необходимо реализовать импорт данных **из ненормализованной** структуры таблиц **базы данных в нормализованную**, а также **выгрузку в шаблоны** **табличного процессора**. Для этого необходимо:

1. Выбрать предметную область и **спроектировать реляционные** базы данных (БД): нормализованную (как минимум **пять таблиц в 3-й нормальной форме**) и соответствующую ненормализованную **(одна таблица)**.
2. Ненормализованная БД должна **храниться в** одном файле с одной таблицей в одном из форматов настольной СУБД**, бесплатной для преподавателей и учащихся в образовательных целях, либо на которую университет имеет действующую лицензию.** Следует (рекомендуется) **использовать** СУБД **отечественные (**например, Ред База Данных (Red Database) [1]**)** или **со свободной лицензией** — например, LibreOffice Base [2] (встроенные БД Firebird и HSQLDB) или SQLite [3]. **При отсутствии** такой возможности **(соответствующих умений)** допустимо **использование импортных** СУБД **с бесплатной лицензией** для университета в образовательных целях — например, Microsoft Access [4] (бесплатная лицензия учащимся и преподавателям при подписке Azure Dev Tools for Teaching [5]).
3. Нормализованная БД должна храниться в **корпоративной СУБД, бесплатной для преподавателей и учащихся в образовательных целях, либо на которую университет имеет действующую лицензию.** Следует (рекомендуется) **использовать** корпоративные СУБД **отечественные (**например, Standard-редакция Postgres Pro [6]**) или со свободной лицензией** — например, MariaDB [7], MySQL [8], PostgreSQL [9]. **При отсутствии** такой возможности **(соответствующих умений)** допустимо **использование импортных** корпоративных СУБД **с бесплатной лицензией** для университета в образовательных целях — например, бесплатные Express-редакции IBM DB2 [10], Microsoft SQL Server [11] или Oracle DataBase [12].
4. Реализовать экспорт данных из спроектированной нормализованной БД в **шаблон** **табличного процессора, бесплатного для преподавателей и учащихся в образовательных целях, либо на который университет имеет действующую лицензию.** Следует (рекомендуется) **использовать** табличные процессоры **отечественные или со свободной лицензией** — например, Calc из LibreOffice [13]. **При отсутствии** такой возможности **(соответствующих умений)** допустимо **использование импортных** табличных процессоров **с бесплатной лицензией** для университета в образовательных целях — например, Microsoft Excel (бесплатно в составе подписки Microsoft365 для образования [14]) или Spreadsheets (бесплатно в составе free-редакции WPS Office [15]).
5. **Набор данных** должен **формироваться в приложении**, а код по расстановке данных по электронной таблице – в макросе шаблона (в Calc — на LibreOffice Basic**,** на VBA — в случае использования Microsoft Excel), либо же в отдельном приложении, написанном на любом другом языке программирования, и вызываемым основным приложением.

Каждый студент **выполняет** **индивидуальное задание** **и представляет результаты преподавателю**, ведущему лабораторные (практические) занятия. Результаты задания оценивает преподаватель, с учётом распределения баллов из таблицы 1. За выполнение входного контроля студент может **максимально набрать десять баллов**. Считается, что студент **прошёл входной контроль**, если он получил **не менее пяти баллов** и **смог защитить работу**, чётко изложив алгоритм работы написанного им приложения.

# Критерии оценивания

Таблица 1. Начисление баллов за результаты выполнения индивидуального задания входного контроля дисциплины «Технологии разработки распределённых вычислений»

|  |  |
| --- | --- |
| **Требование к заданию** | **Максимальное количество баллов** |
| База данных находится в третьей нормальной форме. | 2 |
| Создание базы данных в корпоративной СУБД производится с помощью скрипта. | 1 |
| Приложение позволяет выполнять прием и передачу данных из ненормализованной БД в нормализованную без модификации данных. | 3 |
| Реализован экспорт данных из нормализованной БД в шаблон электронной таблицы. | 2 |
| Код по расстановке данных по электронной таблице написан в макросе шаблона (в Calc — на LibreOffice Basic, на VBA — в случае использования Microsoft Excel), либо же в отдельном приложении, написанном на любом другом языке программирования, и вызываемым основным приложением. | 2 |

# Основная часть

Для решения поставленной задачи были выбраны следующие технологии:

1. Для хранения данных из ненормализованной таблицы была использована настольная СУБД SQLite3. Создание и запись информации в БД производилась в консоли с помощью команд sqlite3.
2. Для хранения данных в виде совокупности таблиц, находящихся в третьей нормальной форме, была использована корпоративная СУБД MySQL. Для создания БД был использован сайт remotemysql.com.
3. Для написания программы взаимодействия и переноса информации между двумя СУБД был использован язык программирования Python и IDE PyCharm. Данный язык программирования был выбран из-за удобного синтаксиса, встроенной поддержки SQLite3, наличием пакета для поддержки MySQL, а также наличием пакета для выгрузки данных из БД в xslx (Excel) файл.

В качестве предметной области была выбрана выдача книг в библиотеке. В ненормализованной СУБД есть одна таблица, содержащая следующие столбцы: ФИО читателя, выданная книга, автор данной книги, жанр данной книги, телефонный номер читателя, дата выдачи. В этой нормализованной СУБД несколько таблиц:

1. Таблица для хранения информации о читателе. Содержит фамилию, имя и отчество (при наличии), которые хранятся уже не одним единым полем, а тремя, а также столбец номера телефона.
2. Таблица списка существующих жанров. Используется как источник данных для внешнего ключа «жанры», используемого в других таблицах (в данном случае данный ключ присутствует только в таблице «книги»).
3. Таблица находящихся в библиотеке книг. Здесь хранится её название и жанр (внешний ключ). Подразумевается, что у книги только один жанр, а не несколько.
4. Таблица для хранения авторов книг. Содержит в себе поле с инициалами и фамилией автора.
5. Таблица связи между авторами и книгами. Подразумевается, что у одной книги может быть несколько авторов. Данная таблица обеспечивает связь «многие ко многим».
6. Таблица связи между читателями и книгами. Один читатель может взять несколько книг, а одна книга может быть взята несколькими читателями. Данная таблица обеспечивает связь «многие ко многим».

Создаются данные таблицы прямо в программе, и удаляются в ней же.

Во время выполнения работы, было создано две программы, которая будет переносить данные из одной СУБД в другую: сначала версия, не использующая ООП, а затем версия, использующая его. Обе программы имеют одинаковую функциональность.

Рассмотрим программу, использующую ООП, поскольку использование данного подхода является требованием к заданию. В данной программе есть несколько классов:

1. Абстрактный класс для создания подключения к абстрактной базе данных.
2. Класс для подключения к СУБД SQLite. Наследуется от предыдущего.
3. Класс для подключения к СУБД MySQL. Наследуется от того же класса, что и предыдущий.

Далее для каждой сущности было создано по классу, реализующему хранение (при помощи полей) и перенос данных из ненормализованной БД в нормализованную СУБД (при помощи метода convertDB() ). Каждый из этих классов имеет одинаковую структуру: поля для хранения данных и один метод convertDB().

1. Класс для хранения и переноса данных о читателе.
2. Класс для хранения и переноса данных о жанрах книг.
3. Класс для хранения и переноса данных об авторах.
4. Класс для хранения и переноса данных о связи «многие ко многим» между авторами и написанными ими книгами.
5. Класс для хранения и переноса данных о связи «многие ко многим» между читателями и выданными им книгами.

Также для записи данных в Excel был создан класс ExcelWriter.

Принцип работы программы таков:

1. Подключение к ненормализованной настольной СУБД SQLite 3.
2. Обработка каждой сущности:
   1. создание объекта текущего класса хранения сущностей,
   2. запись данных в поля этого класса путём считывания из СУБД,
   3. вывод информации на консоль, содержащейся в этих полях, для проверки корректности.
3. Закрытие подключения к SQLite3.
4. Подключение к нормализованной корпоративной СУБД MySQL.
5. Обработка каждой сущности:
   1. создание таблицы для этой сущности,
   2. вызов функции convertDB() для заполнения созданной таблицы данными,
   3. вывод хранимых данных из СУБД с помощью SQL запроса в консоль для отслеживания корректности работы.
6. Выгрузка данных из СУБД MySQL в Excel файл “excel dump. xsls”.
7. Удаление всех созданных таблиц в целях повторной проверки работоспособности программы.
8. Закрытие подключения к MySQL.

# Заключение

Во время выполнения работы я научился:

1. основам ООП на языке программирования Python,
2. взаимодействию с СУБД SQLite и программой на Python,
3. взаимодействию с СУБД MySQL и программой на Python,
4. выгрузке информации из СУБД в Excel на Python,
5. переносу информации из одной СУБД в другую,
6. взаимодействию настольной и корпоративной СУБД.

Требование к заданию «Создание базы данных в корпоративной СУБД производится с помощью скрипта.» было выполнено частично, поскольку база данных уже должна быть заранее создана удалённо. Не был реализован код в макросе шаблона при переносе информации из СУБД в Excel. Требования к остальным заданиям входного контроля были выполнены.