ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

Государственное образовательное учреждение

Высшего профессионального образования

РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ТОРГОВО-ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ

КЕМЕРОВСКИЙ ИНСТИТУТ (ФИЛИАЛ)

ФАКУЛЬТЕТ ЗАОЧНОГО ОБУЧЕНИЯ

Кафедра вычислительной техники и информационных технологий

**Контрольная работа**

**по дисциплине**

**“Базы данных”**

**по теме: “Нормализация таблиц в реляционной модели базы данных”**

Выполнил:

студент группы ПИс-061

(сокращенная форма обучения)

Жилкова Ольга Анатольевна

г. Кемерово 2007 г.

Содержание

[1 Нормализация таблиц в реляционной модели БД](#_Toc268501985)

[1.1 Понятие “Нормализация"](#_Toc268501986)

[1.2 Первая нормальная форма](#_Toc268501987)

[1.3 Вторая нормальная форма](#_Toc268501988)

[1.4 Третья нормальная форма](#_Toc268501989)

[1.5 Четвертая нормальная форма](#_Toc268501990)

[1.6 Пятая нормальная форма.](#_Toc268501991)

[2. Реляционная алгебра над учебной базой](#_Toc268501992)

[3. База данных для предметной области “Учебные пособия"](#_Toc268501993)

[Литература](#_Toc268501994)

## 1 Нормализация таблиц в реляционной модели БД

## 1.1 Понятие “Нормализация"

Нормализация - это формализованная процедура, в процессе выполнения которой атрибуты данных (поля) группируются в таблицы, а таблицы, в свою очередь, - в базы данных. Цели нормализации следующие:

Исключить дублирование информации в таблицах.

Обеспечить возможность изменений в структуре таблиц.

Уменьшить влияние структурных изменений базы данных на работу приложений, обеспечивающих пользователям доступ к данным.

Процесс нормализации состоит из нескольких этапов. Первые три из них, выполняемых чаще всего, были описаны в 1972 году доктором Коддом.

## 1.2 Первая нормальная форма

Каждое поле таблицы должно представлять уникальный тип информации. Это правило означает, что необходимо избавиться от повторяющихся полей и разделить составные поля на отдельные элементы данных. Правила построения первой нормальной формы требуют, чтобы все таблицы данных были плоскими и не содержали повторяющихся данных в различных строках. Под плоской понимается таблица, имеющая только два измерения: длина (число записей или строк) и ширина (число полей или столбцов). Ее ячейки не могут содержать больше одного значения. Если хотя бы одна ячейка таблицы содержит больше одного значения, для представления ее содержимого уже требуется третье измерение - глубина. Плоские таблицы и плоские файлы баз данных очень похожи тем, что имеют только два измерения. Однако в плоском файле содержится лишь одна таблица и не накладываются ограничения на содержимое ее ячеек.

Примером ненормализованных данных могут выступать данные, представленные в таблице 1.1 Таким способом обычно составляется расписание движения транспорта.

Таблица 1.1 - Ненормализованные данные

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Судно | Название | Рейс | Погрузка | Прибытие из | Прибытие | Порт | Отправление | Прибытие | Порт | Отправление |
| 526 | Japan Bear | 9203W | 5/31/92 | SFO | 6/6/92 | HNL | 6/8/92 | 7/15/92 | OSA | 7/18/92 |
| 603 | Korea Bear | 9203W | 5/05/92 | OAK | 6/19/92 | OSA | 6/21/92 | 6/25/92 | INC | 6/28/92 |
| 531 | China Bear | 9204W | 6/20/92 | LAX | 7/10/92 | PAP | 7/11/92 | 8/28/92 | SYD | 9/2/92 |
| 528 | Japan Bear | 9204W | 8/20/92 | SFO | 8/27/92 | HNL | 8/29/92 | 9/30/92 | OSA | 10/2/92 |

Поскольку суда останавливаются во многих портах, столбцы Прибытие, Порт и Отправление повторяются для каждой остановки. Такая структура записи данных не подходит для реляционной базы данных. запись приведенной информации не соответствует требованиям первой нормальной формы, поскольку содержит повторяющуюся группу столбцов. Эту таблицу необходимо разделить на две: Порты и рейсы судов, не содержащие повторяющихся групп, как показано в таблицах 1.2 и 1.3

Таблица 1.2 - Таблица “Рейсы судов”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Судно | Название | Рейс | Погрузка | Прибытие из |
| 528 | Japan Bear | 9203W | 5/31/92 | SFO |
| 603 | Korea Bear | 9203W | 6/5/92 | OAK |
| 531 | China bear | 9204W | 6/20/92 | LAX |
| 528 | Japan bear | 9204W | 8/20/92 | SFO |

Таблица 1.3 - Таблица “Порты”

|  |  |  |
| --- | --- | --- |
| Прибытие | Порт | Отправление |
| 6/6/92 | HNL | 6/8/92 |
| 6/19/92 | OSA | 6/21/92 |
| 7/10/92 | PAP | 7/11/92 |
| 8/27/92 | HNL | 8/29/92 |
| 7/15/92 | OSA | 7/18/92 |
| 6/25/92 | INC | 6/28/92 |
| 8/28/92 | SYD | 9/2/92 |
| 9/30/92 | OSA | 10/2/92 |

Теперь необходимо установить связь между таблицами Порты и Рейсы судов. В столбце рейс указывается текущий год, номер рейса за этот год, а также направление рейса (например, 9204W - это четвертый рейс за 1992 год в западном направлении). Таким образом, для связи между таблицами следует применять поля Судно и Рейс. Использовать какой-либо один из этих способов недостаточно, поскольку одно судно может делать несколько рейсов в течение года, а в одном направлении могут отправляться сразу несколько судов. Поскольку для удовлетворения требований первой нормальной формы придется создать новую таблицу Порты, необходимо отсортировать ее столбцы в порядке значимости. Первыми, как правило, размещаются столбцы, используемые для установки связи. При этом они располагаются в той последовательности, в какой они входят в составной первичный ключ. Данные показаны в таблице 1.4

Таблица 1.4 - Таблица “Порты”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Судно | Рейс | Порт | Прибытие | Отправление |
| 528 | 9203W | HNL | 6/6/92 | 6/8/92 |
| 603 | 9203W | OSA | 6/19/92 | 6/21/92 |
| 531 | 9204W | PAP | 7/10/92 | 7/11/92 |
| 528 | 9204W | HNL | 8/27/92 | 8/29/92 |
| 528 | 9203W | OSA | 7/15/92 | 7/18/92 |
| 603 | 9203W | INC | 6/25/92 | 6/28/92 |
| 531 | 9204W | SYD | 8/28/92 | 9/2/92 |
| 528 | 9204W | OSA | 9/30/92 | 10/2/92 |

Теперь необходимо определить ключевые поля таблицы Порты, что дает возможность точно идентифицировать ее записи. Обязательно необходимо создать первичный ключ, поскольку от этой таблицы могут зависеть многие другие. Необходимо добавить столбцы Судно и рейс, так как они обеспечивают связь с данными таблицы Рейсы судов, также добавить поле Порт для создания совершенного уникального ключа (столбы Судно и Рейс могут содержать повторяющиеся значения). Комбинации Судно+Рейс+Порт представляет собой составной первичный ключ, значение которого однозначно идентифицирует запись. Значения этого ключа не повторяются, поскольку учтена возможность дважды делать остановку в одном порту (придвижении туда и обратно). Так, если судно возвращается с востока, рейс помечается суффиксом “Е".

## 1.3 Вторая нормальная форма

Для приведения таблиц ко второй нормальной форме необходимо обеспечить полную зависимость столбцов, которые не являются ключевыми, от первичного ключа, а если этот ключ составной, то от каждого его элемента. Под полной зависимостью понимается возможность однозначного определения значения каждого неключевого поля с помощью значения первичного ключа. Если для однозначного определения используется составной первичный ключ, то это правило применяется к каждому значению из полей, входящих в составной ключ. Всегда, когда это возможно, в качестве первичного ключа следует использовать самые простые данные, имеющие “естественные" уникальные значения. Перед переходом ко второй нормальной форме необходимо привести данные к первой нормально форме. В процессе создания второй нормальной формы большая часть повторяющихся данных, оставшихся в таблице после приведения ее к первой нормальной форме, будет удалена.

Для создания в таблице Рейсы судов однозначного ключа придется использовать составной ключ (Судно+Рейс). Поскольку номер и название судна могут повторяться. Поля Судно и Название не зависят от первичного ключа, так как полем Рейс ничего не определяется. Название судна указывается в каждом рейсе. Так, например, название Japan Bear появляется дважды. Все эти недостатки нарушают правила второй нормальной формы. Возникает необходимость разбиения таблицы Рейсы судов еще на две: Рейсы и Суда. Каждый корабль описывается одной строкой в таблице суда, а одна строка таблицы Рейсы описывает рейс одного судна (с целью упрощения построения базы данных восточные и западные направления рассматриваются как отдельные рейсы). Как и в таблице Порты, для установления соответствия между рейсами и судами необходимо создать ключ, поэтому необходимо добавить поле номеров судов в таблицу Рейсы. Таблицы Суда и Рейсы показаны в таблицах 1.5 и 1.6

Таблица 1.5 - Таблица “Суда”

|  |  |
| --- | --- |
| Судно | Название |
| 528 | Japan Bera |
| 603 | Korea Bear |
| 531 | China bear |

Таблица 1.6 - Таблица “Рейсы”

|  |  |  |  |
| --- | --- | --- | --- |
| Судно | Рейс | Погрузка | Прибытие из |
| 528 | 9203W | 5/31/92 | SFO |
| 603 | 9203W | 6/5/92 | OAK |
| 531 | 9204W | 6/20/92 | LAX |
| 528 | 9204W | 8/20/92 | SFO |

После определения первичных ключей для каждой таблицы можно проверить, чтобы все данные, включенные в таблицы, относились к соответствующим объектам. Другими словами, следует убедиться, что каждое поле функционально зависит от первичного ключа, который идентифицирует исходный объект таблицы. Для каждого значения первичного ключа значения в столбцах данных должны относиться к объекту таблицы и полностью его описывать. Это правило используется двояко. Во-первых, в таблице не должно быть данных, не относящихся к объекту, определяемому первичным ключом. Во-вторых, данные в таблице должны полностью описывать объект.

## 1.4 Третья нормальная форма

В третьей нормальной форме столбцы, не являющиеся ключевыми, зависят от первичного ключа таблицы и не зависят от всех остальных столбцов. Должна быть возможность изменять значения любого поля (не входящего в первичный ключ) без воздействия на данные других полей. Прежде чем перейти к третьей нормальной форме, необходимо привести свои данные к первой, а затем - ко второй.

Таблицы суда и Рейсы уже представлены в третьей нормальной форме, поскольку они содержат повторяющихся групп столбцов и существует зависимость неключевых столбцов этих таблиц от ключевых. В таблице Порты нет повторяющихся групп полей, а неключевые поля Прибытие и Отправление зависят от составного ключа (Судно + Рейс + Порт) и не зависят друг от друга. (Дата прибытия никак не влияет на дату отправления, поскольку интервал между ними определяется временем погрузки товаров на судно, доступностью стоянок, погрузочных кранов, а также погодными условиями) Таким образом, таблица Порты удовлетворяет требованиям первой, второй и третьей нормальных форм.

Например, необходимо ввести в базу данных еще и служащих, членов команд кораблей (капитанов, инженеров и т.п.). Для этого необходимо добавить в таблицу Суда коды служащих, являющиеся первичными ключами таблицы Служащие, показанные в таблице 1.7

Таблица 1.7 - Таблица с транзитивным отношением между судами и служащими команды.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Судно | Название | Капитан | Старший помощник | Первый помощник |
| 528 | Japan Bear | 01023 | 01155 | 01367 |
| 603 | Korea Bear | 00955 | 01203 | 00823 |
| 531 | China Bear | 00721 | 00912 | 01251 |

Такая таблица нарушает одно из правил третьей нормальной формы, поскольку никто из назначенных на какое-либо судно служащих от этого судна не зависит непосредственно (т.е. добавлямое неключевое поле не зависит от первичного ключа). Капитан, его помощник или старший помощник могут выполнять свои обязанности на любом судне. Каждого офицера можно в случае необходимости перевести на другой корабль или оставить на суше до следующего рейса.

Можно попробовать добавить столбец с номерами служащих в таблицу Рейсы. Но и это не решит всех проблем, так как судно может прибыть в порт с одним командным составом, а отбыть с совершенно другим. Кроме этого, возникает необходимость определить членов команды, остающихся на судне во время его стоянки в порту.

Все эти вопросы решаются с помощью таблицы Команды, приведенной в таблице 1.8 повторяющиеся значения полей Порт (порт отправления) и Отправление в (порт прибытия) создают записи для тех членов команды, которые несут ответственность за судно во время его пребывания в порту. Таблица Команды относится к категории связующих таблиц, поскольку все поля связаны с первичными ключами основных таблиц (Суда, Рейсы, Порты и Служащие) или являются частью этих таблиц.

Таблица 1.8 - Таблица “Команды”

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Судно | Рейс | Порт | Отправление в | Капитан | Старший помощник | Первый помощник |
| 528 | 9203W | SFO | HNL | 01023 | 01156 | 01367 |
| 528 | 9203W | HNL | HNL | 01023 | 01156 | 01367 |
| 528 | 9203W | HNL | OSA | 01023 | 01156 | 01367 |
| 528 | 9203W | OSA | OSA | 01023 | 01156 | 01367 |
| 528 | 9203W | OSA | INC | 01023 | 01156 | 01367 |

Теперь все таблицы являются плоскими, не содержат повторяющейся информации, за исключением данных, используемых в ключах. Все эти таблицы удовлетворяют требованиям первой, второй и третьей нормальной форм.

## 1.5 Четвертая нормальная форма

Чтобы база данных находилась в четвертой нормальной форме, необходимо, чтобы независимые элементы данных, между которыми существует связь типа многие-ко-многим, не хранились в одной таблице. Таблица команды не находится в четвертой нормальной форме, поскольку связи многие-ко-многим установлены между ее полем судно и полями с кодами членов команды.

Многие разработчики приложений баз данных игнорируют четвертую и пятую нормальные формы в своих программных продуктах, поскольку считают их весьма специфическими. Результатом этого зачастую является создание базы данных неправильной структуры, хотя это совсем не означает, что она не будет функционировать.

## 1.6 Пятая нормальная форма

Пятая нормальная форма требует обеспечения возможности точного восстановления исходной таблицы из таблиц, на которых она основана. Построение пятой нормальной формы требует удовлетворения требований третьей нормальной формы и, при наличии связей многие-ко-многим, соответствия правилам четвертой.

Таблица Рейсы очень похожа на таблицу Порты. Столбец прибытие из аналогичен столбцу Порт, а столбец Погрузка - столбцу Отправление. именно по этой причине можно переместить данные таблицы Рейсы в таблицу порты и удалить таблицу Рейсы. новая таблица порты показана в таблице 1.9 Значения столбца Прибытие для записей, внесенных в таблицу Порты из таблицы рейсы, не определены (не существуют), так как в таблице Рейсы были указаны только даты отравления.

Таблица 1.9 - Таблица “Порты”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Судно | Рейс | Порт | Прибытие | Отправление |
| 528 | 9203W | HNL | 6/6/92 | 6/8/92 |
| 528 | 9203W | OSA | 6/19/92 | 6/21/92 |
| 528 | 9204W | PAP | 7/10/92 | 7/11/92 |
| 528 | 9204W | HNL | 8/27/92 | 8/29/92 |
| 528 | 9203W | OSA | 7/15/92 | 7/18/92 |
| 603 | 9203W | INC | 6/25/92 | 6/28/92 |
| 531 | 9204W | SYD | 8/28/92 | 9/2/92 |
| 528 | 9204W | OSA | 9/30/92 | 10/2/92 |
| 528 | 9203W | SFO |  | 5/31/92 |
| 603 | 9203W | OAK |  | 6/5/92 |
| 531 | 9204W | LAX |  | 6/20/92 |
| 528 | 9204W | SFO |  | 6/20/92 |

Невозможно восстановить исходную таблицу из объединенных таблиц Рейсы и Порты, поскольку не сможете отличить строку отправления от других строк по значениям ее полей в таблице. Чтобы отличить строки отправлений, можно было бы использовать значения Null в поле Прибытие, однако значение Null должно быть зарезервированным для условия “данные недоступны". Необходимо устранить любые двусмысленности, которые могут привести к появлению значений Null, и преобразовать таблицу в пятую нормальную форму, добавив односимвольное поле Тип, определяющее прибытие или отправление. В показанной таблице 1.10 Порты коды Е и S представляют соответственно погрузку (Embarkation) и ожидаемое прибытие (Scheduled). Могут также использоваться коды M для заправки (Maintenance) и R - для обратного рейса (Return voyage).

Таблица 1.10 - Таблица “Порты”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Судно | Рейс | Порт | Тип | Прибытие | Отправление |
| 528 | 9203W | HNL | S | 6/6/92 | 6/8/92 |
| 528 | 9203W | OSA | S | 6/19/92 | 6/21/92 |
| 528 | 9204W | PAP | S | 7/10/92 | 7/11/92 |
| 528 | 9204W | HNL | S | 8/27/92 | 8/29/92 |
| 528 | 9203W | OSA | S | 7/15/92 | 7/18/92 |
| 603 | 9203W | INC | S | 6/25/92 | 6/28/92 |
| 531 | 9204W | SYD | S | 8/28/92 | 9/2/92 |
| 528 | 9204W | OSA | S | 9/30/92 | 10/2/92 |
| 528 | 9203W | SFO | E |  | 5/31/92 |
| 603 | 9203W | OAK | E |  | 6/5/92 |
| 531 | 9204W | LAX | E |  | 6/20/92 |
| 528 | 9204W | SFO | E |  | 6/20/92 |

## 2. Реляционная алгебра над учебной базой

R1 - список абитуриентов, сдававших репетиционные вступительные экзамены;

R2 - список абитуриентов, сдававших вступительные экзамены на общих основаниях;

R3 - список абитуриентов, принятых в институт.

Необходимо написать ответ на запрос в виде формулы реляционной алгебры. Запрос: “Список абитуриентов, которые поступали два раза и поступили в вуз".

Таблица 2.1 - Отношение R1

|  |  |  |  |
| --- | --- | --- | --- |
| Обозначение записи | ФИО абитуриента | Номер и серия паспорта | № Школы |
| a | Жилкова О.А.  | 32 05 4237 | № 31 |
| b | Богач Д.О.  | 34 07 4385 | № 42 |
| с | Конопелько О.П.  | 37 08 4282 | № 56 |
| d | Кочкина Т.В.  | 38 02 3458 | № 52 |
| e | Докучаев Ю.А.  | 58 02 3718 | № 62 |
| f | Богданова Ю.В.  | 38 72 4290 | № 48 |
| g | Сидорова С.И.  | 39 52 4870 | № 45 |
| h | Сидоров А.А.  | 38 59 3295 | № 46 |
| l | Тарабрина Л.В.  | 40 58 2598 | № 49 |

Таблица 2.2 - Отношение R2

|  |  |  |  |
| --- | --- | --- | --- |
| Обозначение записи | ФИО абитуриента | Номер и серия паспорта | № Школы |
| a | Жилкова О.А.  | 32 05 4237 | № 31 |
| b | Богач Д.О.  | 34 07 4385 | № 42 |
| d | Кочкина Т.В.  | 38 02 3458 | № 52 |
| h | Сидоров А.А.  | 38 59 3295 | № 46 |
| m | Тарабрин В.В.  | 35 92 4058 | №48 |
| n | Голоушкина В.А.  | 38 92 4259 | № 52 |
| o | Токарева М.А.  | 39 98 4085 | № 46 |
| p | Круглова Т.Ю.  | 32 58 3498 | № 47 |

Таблица 2.3 - Отношение R3

|  |  |  |  |
| --- | --- | --- | --- |
| Обозначение записи | ФИО абитуриента | Номер и серия паспорта | № Школы |
| a | Жилкова О.А.  | 32 05 4237 | № 31 |
| b | Богач Д.О.  | 34 07 4385 | № 42 |
| d | Кочкина Т.В.  | 38 02 3458 | № 52 |
| h | Сидоров А.А.  | 38 59 3295 | № 46 |
| p | Круглова Т.Ю.  | 32 58 3498 | № 47 |
| с | Конопелько О.П.  | 37 08 4282 | № 56 |

Операция реляционной алгебры - “пересечение".

R1 (a,b,c,d,e,f,g,h,l) R2 (a,b,d,h,m,n,o,p) =R3 (a,b,d,h)

## 3. База данных для предметной области “Учебные пособия"

Ненормализованное представление информации в виде схемы.

Дисциплины

Пособия

Специальности

Приведение к третьей нормальной форме.

первая нормальная форма - каждое поле таблицы представляет уникальный тип информации;

Таблица “Дисциплины" - номер дисциплины, наименование дисциплины, количество часов;

Таблица “Пособия” - номер пособия, ФИО автора, Номер дисциплины;

Таблица “Специальности” - номер специальности, наименование специальности;

вторая нормальная форма - каждая таблица должна иметь первичный ключ, который может состоять из одного Ии нескольких полей.

Таблица “Дисциплины" - номер дисциплины;

Таблица “Пособия” - номер пособия;

Таблица “Специальности” - номер специальности.

третья нормальная форма - столбцы, не являющиеся ключевыми, зависят от первичного ключа таблицы и не зависят от всех остальных столбцов. Для связывания таблиц “Дисциплины" и “Специальности” необходимо создать дополнительную таблицу “Дисциплины-Специальности".

Таблица “Дисциплины-Специальности" - номер дисциплины, номер специальности.

Дисциплины

Пособия

Специальности

Дисциплины -

Специальности

Данные таблиц.

Таблица 3.1 - Таблица “Дисциплины”

|  |  |  |
| --- | --- | --- |
| Номер дисциплины | Наименование дисциплины | Количество часов |
| 1 | Информатик | 132 |
| 2 | Экономика | 180 |
| 3 | Базы данных | 72 |
| 4 | Основы бухгалтерского учета | 86 |
| 5 | Основы программирования | 92 |
| 6 | Теория вероятностей и математическая статистика | 146 |
| 7 | Мировая экономика | 112 |
| 8 | Компьютерные сети | 98 |

Таблица 3.2 - Таблица “Пособия”

|  |  |  |
| --- | --- | --- |
| Номер пособия | ФИО автора | Номер дисциплины |
| 1 | Джон Вейкас | 3 |
| 2 | Роджер Дженнингс | 3 |
| 3 | Вирджиния Андерсон | 1 |
| 4 | Попов А.А.  | 1 |
| 5 | Булатов А.С.  | 2 |
| 6 | Бендина Н.В.  | 4 |
| 7 | Видяпин В.И.  | 2 |
| 8 | Дурович А.П.  | 4 |
| 9 | Коуров Л.В.  | 1 |
| 10 | Кашанин Т.В.  | 7 |
| 11 | Гмурман В.Е.  | 6 |
| 12  | Кенин А.М.  | 8 |
| 13 | Питер Эйткен | 5 |
| 14 | Подбельский В.В.  | 5 |
| 15 | Вендров А.М.  | 7 |
| 16 | Рапаков Г.Г.  | 8 |
| 17 | Якушева Г.В.  | 6 |
| 18 | Комягина В.Б.  | 4 |
| 19 | Бердиченко Е.В.  | 7 |

Таблица 3.3 - Таблица “Специальности”

|  |  |
| --- | --- |
| Номер дисциплины | Наименование дисциплины |
| 101170 | Прикладная информатика в экономике |
| 220135 | Программное обеспечение ВТ и АС |
| 11370 | Бухгалтерский учет |
| 13568 | Экономическая теория |
| 73809 | Администрирование компьютерных сетей |

Таблица 3.4 - Таблица “Дисциплины - Специальности"

|  |  |
| --- | --- |
| Номер специальности | Номер дисциплины |
| 101170 | 1 |
| 101170 | 5 |
| 101170 | 6 |
| 101170 | 2 |
| 101170 | 7 |
| 220135 | 1 |
| 220135 | 5 |
| 220135 | 6 |
| 220135 | 3 |
| 11370 | 1 |
| 11370 | 4 |
| 11370 | 2 |
| 13568 | 2 |
| 13568 | 6 |
| 13568 | 7 |
| 13568 | 1 |
| 73809 | 3 |
| 73809 | 5 |
| 73809 | 8 |

Создание таблиц. Для создания запросов выбрана СУБД ACCESS.

CREATE дисциплины (number integer,

name\_disz varchar (100),

hour integer);

CREATE пособия (number integer,

author varchar (100),

diszipl integer);

CREATE специальности (number varchar (10),

name\_spez varchar (100));

CREATE дисциплины\_специальности (number\_spez varchar (100),

number\_disz integer).

Заполнение таблиц данными. Для примера показаны вставки по одной записи.

INSERT INTO дисциплины (number, name\_disz, hour) VALUES (1, “Информатика”, 132);

INSERT INTO специальности (number, name\_spez) VALUES (“101170", “Прикладная информатика в экономике”);

INSERT INTO пособия (number, autor, diszipl) VALUES (1, “Джон Вейкас”, 3);

INSERT INTO дисциплины\_специальности (number\_spez, number\_disz) VALUES (“101170”, 1)

Запрос1 - Для номера специальности “220135" вывести наименование этой специальности, наименования дисциплин для этой специальности, у которых количество часов больше 90 и меньше 140, а также авторов пособий для этих дисциплин.

SELECT специальности. number AS "Номер специальности",

специальности. name\_spez AS "Специальность",

дисциплины. name\_disz AS "Дисциплина",

дисциплины. hour AS "Количество часов",

пособия. author AS "Автор пособия"

FROM специальности, дисциплины, пособия,

дисциплины\_специальности

WHERE дисциплины\_специальности. number\_disz=дисциплины. number And

дисциплины\_специальности. number\_spez=специальности. number And

пособия. diszipl=дисциплины. number And

специальности. number="220135" And

дисциплины. hour Between 90 And 140

ORDER BY дисциплины. name\_disz, пособия. author;



Запрос 2 - Вывести для каждой специальности номер специальности, наименование специальности, количество дисциплин для каждой специальности, у которых количество часов больше 90 и меньше 150.

SELECT специальности. number AS "Номер специальности",

специальности. name\_spez AS "Специальность",

COUNT (дисциплины\_специальности. number\_disz) AS "Количество

дисциплин"

FROM специальности, дисциплины\_специальности, дисциплины

WHERE дисциплины\_специальности. number\_spez=специальности. number

And дисциплины\_специальности. number\_disz=дисциплины. number

And дисциплины. hour Between 90 And 150

GROUP BY специальности. number, специальности. name\_spez

ORDER BY специальности. name\_spez;



## Литература

1. Базы данных: теория и практика: Учебник для вузов/ Б.Я. Советов., В.В. Цехановский, В.Д. Черотовской. - М.: Высш. шк., 2005. - 463 с. ил.
2. Вейскас Д. эффективная работа с Microsoft Access 97 - СПб: ЗАО “Издательство “Питер””, 1999. - 976 с.: ил.
3. В.В. Корнеев, А.Ф. Гареев, С.В. Васютин, В.В. Райх. Базы данных. Интеллектуальная обработка информации. - М.: Издатель Момачева С.В., Издательство Нолидж, 2001. - 496 с., ил.
4. Дженнингс, Роджер. Использование Microsoft Access 2000. Специальное издание.: Пер. с англ.: Уч. пос. - М.: Издательский дом “Вильямс", 2000. - 1152 с.: ил. - Парал. тит. англ.