

Создатель первой версии файла: Губайдулин Айдар, 174 группа (2013)

Ресурсы

[Сайт с заданием](#), книга Мартина Грабера “SQL”

Общие советы:

- **Начинайте сдавать лабы как можно раньше и не пропускайте сдачи.** Самое главное правило сдачи БД. На следующей неделе после ознакомительного занятия нужно прийти сдавать 1 лабу. Также стоит подготовить схему 2 лабы, чтобы Теймур указал на главные ошибки и на второй неделе вы её сдали. Так как в первый день придёт сдавать мало людей, вы точно успеете со второй. В другие же дни сдать за день две лабы просто нереально, желающих очень много. Если начать сдавать с первого дня и идти так до конца, не теряя недель, то баллы за сдачи будут гораздо выше и ближе к середине апреля вы всё закончите с оценкой Отл в зачётке. Также существует вариант сдачи всех 10 лаб в феврале - начале марта по договорённости с Теймуразовым. Такой способ ещё более предпочтителен, т.к. в феврале нагрузка минимальна и большинство времени всё равно идёт не на учёбу. А учиться лучше зимой, а не весной.
- На сдачу приходите нужно в самом начале пары (а, может, и раньше), т.к. если попасть в конец очереди и отвечать за 15 минут до конца, времени, необходимого на ответы на дополнительные вопросы, не будет. [КБТ и компания мало того что довольно строго ведёт очередь, но ещё и ограничивают запись в неё опоздавшим. Хотя, обычно, если у вас “нормальный” приоритет (ваша пара, без дисконта, сегодняшняя лаба) то вы успеете ответить]
- При подготовке к каждой лабе используйте Справку MSDN [Справка MSDN \(используйте поиск\)](#)
- Когда будете ставить MS SQL Server, нужно чётко придерживаться инструкций, данных КБТ. В противном случае вы не сможете сделать 7 лабу, т.к. при аутентификации искусственным пользователем возможно возникновение ошибки №18452. Возможно, существует решение проблемы, но во всех двух мне известных случаях приходилось брать ноут с нормально поставленным MS SQL у друзей. И лучше устанавливайте HE express версию, если собираетесь сдавать 10 лабу. (Не обязательно. 10 лаба сдается и без инструментов указанных в описании, даже без бд northwind. Хотя последняя конечно и удобна, легче будет создать свою тривиальную бд и заполнить её скриптом)
- **Ни в коем случае не пытайтесь переустановить MS SQL.** Удаления его из “Установки и удаления” программ не достаточно. MS SQL “опутывает” всю систему и, чтоб его удалить, можно, разве что, воспользоваться инструкцией с Майкрософта. Если пытаться всё делать самому, то могут остаться его куски, которые в дальнейшем будут препятствовать повторной установке. Если вы испортили свой сервер, то лучше просто создать ещё один.
- Попробуйте обогнать расписание сдач хотя бы на одну лабу. Уровень сложности сдачи падает раза в два.

1 лаба. Работа в среде интерактивного SQL

Pro-tip: Если выделить строчку в окне запроса и нажать “исполнить”, выполнится только она (при этом свойства подключения типа вывода информации о времени запроса, или execute as, будут работать если были исполнены в этом подключении)

pro-tip: F5 - “выполнить запрос”, F6 - проанализировать грамматику без выполнения,

PRO-tip: если вам надоело каждый раз удалять данные после неудачных запросов, перед любым

выражением создания\добавления поставьте BEGIN TRAN, после создания проверьте результат селектом, и в конце скрипта сделайте ROLLBACK

- При подготовке запросов важно, чтобы количество вызова оператора SELECT было минимальным. В идеале каждый запрос должен содержать ровно один SELECT. КБТ не очень любит вложенные запросы типа SELECT(SELECT(SELECT(...))). (мало того, он клялся и божился, что все запросы из этой лабы решаются в один селект)
- Хотя большинство запросов из вариантов можно найти в локалке, лучше написать их самому, т.к. запросы - основа понимания предмета. В любом случае, умение писать запросы необходимо в 4 лабе.
- Внимательно читайте условие запроса, ничего не додумывая и выполняя в точности то, что требуется.

Например, в 29 варианте 4 запрос гласит:

“Выбрать список продуктов, цены на которые в 1988г. менялись не менее 2 раз.”

Тут существует два “подводных камня”

- а. Если цена только установилась первый раз в 1988г., то это мы не считаем, т.к. цена *не изменилась*.
 - б. Если цена изменилась, но новая цена совпадает со старой (такое тоже, очевидно, возможно), то цена также *не изменилась*.
- Если, например, для какого-то отдела нужно вывести всех сотрудников, работающих в Нью-Йорке, а для некоторого отдела их нет, то запрос должен выдавать название отдела и 0 напротив него. Если он не выводит этот отдел вообще, то запрос потребуют переписать.
 - НИКОГДА не оставляйте в запросах служебные поля вроде id. Особенно если нужно вывести “все х кого-нибудь”. Если кроме него нету уникального поля, сделайте составное (например, если есть поля имя и фамилия, объедините их в одну строку и выведете). Так же желательно использовать именование столбцов в запросе(select что-то as толковое_имя)

Вопросы

1. Что такое SELECT, FROM, GROUP BY, ORDER BY, HAVING
2. Что такое ALL, DISTINCT
3. Чем отличаются INNER JOIN и OUTER JOIN
4. Что такое CROSS JOIN, какой более простой конструкцией его можно заменить.

Ответ: CROSS JOIN - это произведение двух таблиц. Гораздо более простой конструкцией является запятая. Т.е. SELECT (...) FROM tab1, tab2 - это всё равно, что SELECT(...) FROM tab1 CROSS JOIN tab2

5. Что будет, если ввести запрос SELECT * FROM table WHERE id = id

6. Что такое ISNULL

7. Что выведет COUNT(*), COUNT(), COUNT(8) в групповом запросе и почему.

8. Что выдаст SELECT * FROM table WHERE NULL=NULL

(Ответ: ничего, потому что по стандарту сравнение с NULL возвращает false, нужно использовать IS NULL)

9.

```
declare @TestTable as table(A int)
```

```
insert into @TestTable(A) values (1)
```

```
insert into @TestTable(A) values (7)
```

```
insert into @TestTable(A) values (5)
insert into @TestTable(A) values (NULL)
```

В - Пишут такую фигню и просят вывести среднее между 1,7,5.

О - Надо написать `AVG(CAST(A as FLOAT))` или `AVG(A*1.0)`
`select AVG(A*1.0) from @TestTable`

В - После спрашивают как считать COUNT с NULL и без NULL.

О - COUNT(*) и COUNT(A) соответственно

2 лаба. Проектирование схемы

Для сдачи этой работы нужно будет просто изобразить схему таблиц. Куда удобнее сразу сделать базу в SQL, чтобы сгенерировать схему автоматически встроенным средством.

- Не доверяйте простоте схемы в задании. Нужно продумать базу не только со стороны ответа на вопрос “кому и зачем” но ещё и “как это реализовать так чтобы было удобно”. Но и перебарщивать не стоит, чем сложнее схема - тем навороченнее запросы, и легче КБТ вас по ней завалить.
- В вашей базе данных не должно быть лишних таблиц. Ваша база должна коррелировать с реальной жизнью. То есть, проектируя базу, связанную с ЖКХ, заранее ознакомьтесь с этой сферой, с тарифами и услугами и т.д.
- Будьте готовы к вопросам типа: “Зачем в этой таблице нужен этот столбец?”, “Зачем нужна эта таблица?”
- **Самый главный вопрос** - “Для кого и зачем проектировалась эта база данных?”. Вы должны будете назвать цель и обосновать, зачем вам в вашей базе нужны все таблицы.
- По возможности не включайте в схему финансы в каком бы то ни было виде. Если вы попытаетесь создать базу бухгалтерии, то с огромной вероятностью вас завалят, т. к. нужно учитывать абсолютно всё: поставки, долги, неожиданные доходы и т.п. КБТ с лёгкостью вас завалит.
- То же относится к кадрам.
- Удобно делать схему с помощью, например, Database Design Tool (можно потом экспортировать в .sql) <http://gnuwin.epfl.ch/apps/DDT/en/>

3 лаба. Создание таблиц

- Нужно знать [типы данных](#)
- pro-tip: если в предыдущей лабе схему рисовали автоматом по созданной базе, то скрипты для этой тоже можно получить автоматически, в контекстном меню базы данных выбрав Task -> Generate Scripts; Его надо почистить от непонятных слов(
- При сдаче лабы вас попросят удалить все имеющиеся таблицы и запустить скрипт создания. Удобно в начале скрипта создания прописать удаление всех таблиц. (DROP TABLE <имя таблицы>). При этом нужно расположить команды удаления в нужном порядке, так как таблицы друг на друга ссылаются и это нужно учесть (сначала удаляйте таблицы, которые зависят от других, потом - “главные” таблицы).

- В таблицах нужно использовать ключевое слово IDENTITY. Если у вас этого не будет, вас отправят переделывать скрипт.

Вопросы:

1. Чем отличаются CHAR и VARCHAR?
2. Чем отличаются VARCHAR и NVARCHAR (зачем перед строковыми данными N'строка')?
3. Что будет, если убрать из объявления столбца PRIMARY KEY ключевое слово NOT NULL
 Ответ: ничего не будет, но по стандарту положено, чтобы этот столбец был NOT NULL
Следующий вопрос: убирает NOT. Ответ: Выдаст ошибку.
4. Чем отличаются PRIMARY KEY и UNIQUE?
5. Какая минимальная и максимальная может быть дата у datetime? (01.01.1000 - 31.12.9999) *Пиздишь, братан, минимальная дата - 01.01.1753. Ты написал для date*
6. Ещё одна суперзадача от КБТ: данные удаляются и добавляются снова. Возможны два варианта: всё заработало (Вам повезло, карма+1) и данные упали. В чём причина? Identity не сбросило. Как исправить? Я просто в INSERT явно указывал PRIMARY KEY.
7. Изменение параметров типа (или самого типа):

- a. Изменить тип данных поля в вашей базе.
- b. Что будет, если при создании таблицы с полем age SMALLINT написать age SMALLINT DEFAULT(30)?

Ответ: DEFAULT по умолчанию устанавливает всем строкам таблицы приписанное ему значение, если значения для соответствующего поля не вставляются в таблицу при помощи INSERT.

- c. Изменить ограничение наложенное, например, на тип VARCHAR. Т.е. если было VARCHAR(15), нужно сделать VARCHAR(30)

Ответ: ALTER TABLE CLIENT

ALTER COLUMN last_name VARCHAR(30);

- d. Изменить ограничение, наложенное при помощи CHECK, изменить значение DEFAULT.

Указание к ответу: в окне обозревателя можно посмотреть ограничения, относящиеся к любой таблице и полю в ней. Там же можно создать сценарий (скрипт) для его изменения. В итоге должно получиться что-то типа:

ALTER TABLE CLIENT

DROP CONSTRAINT [DF__CLIENT__age__2764BD12]

ALTER TABLE CLIENT

ADD CONSTRAINT [DF__CLIENT__age__2764BD12] DEFAULT (35) FOR age

8. Если попросят изменить тип поля, которое связано foreign key, возможны два решения: удалить constraint-ы, поменять везде типы, восстановить constraint-ы. Но КБТ это не понравится, он хочет копии таблиц с измененным типом и копирование данных туда

1. Что такое IDENTITY, UNIQUE, прочие constraint
2. Вопрос: Что такое каскадное удаление и обновление данных?

Ответ: используем:

REFERENCES referenced_table_name (ref_column)

ON DELETE CASCADE

ON UPDATE CASCADE

Более подробно: [Табличные ограничения \(constraint\)](#)

• 4 лаба. Манипулирование данными

- Для сдачи нужно знать всю теорию из лабы 1 (Работа в среде интерактивного SQL)
- Запросы на удаление и изменение в базе данных тоже должны быть сложными.
- В качестве доп.вопросов вам предложат написать новый запрос и/или исправить один из старых.
- IF поле = NULL -> минус балл. ТОЛЬКО is null.

5 лаба. Создание и использование представлений

pro-tip: строчки GO в генерируемых автоматически скриптах - разделение наборов команд внутри одного подключения. Если ругается, что какое-то выражение must be first in a batch, поставьте перед ним GO.

- [Информация о представлениях](#)
- В начале скрипта сделайте удаление представлений:
DROP VIEW <имя представления>

Вопросы:

1. Что такое представление. Для чего нужны представления.
2. Что такое WITH CHECK OPTION
3. Можно ли в представлении упорядочить записи?

Ответ. Несмотря на то, что предложение ORDER BY в представлениях недопустимо, записи можно упорядочить, используя конструкцию

SELECT TOP 100 PERCENT price FROM

В примере сортировка выполняется по цене price, вместо неё может быть что угодно, в том числе строки (фамилия, имя).

- Можно ли через представления изменять исходные таблицы?

Ответ: через некоторые можно, через некоторые нельзя. Подробнее по ссылке выше. Внимание! Вам нужно написать скрипты для выборок из вашего варианта и САМИМ придумать ЕЩЁ 3-4 выборки в предметной области (например, для поликлиники: "Определить процент смертности от заболевания 'кариес' "). Выборки должны быть более-менее сложными. Не ленитесь! От этого зависит ваша оценка за лабу. (А результаты - логичными. 100 смертность от кариеса или отрицательная успеваемость студентов - повод схлопотать минус балл, задержав сдачу на 10 минут, пока КБТ и компания будут над этим ржать. Seriously.)

4. Какие представления являются изменяемыми? Пожалуй, один из самых важных вопросов, т.к. то что написано в литературе, например, запрещает изменяемым представлениям x GROUP BY, что не верно. Некоторые представления могут быть изменяемыми, несмотря на то, что Грабер (автор книги по MS SQL) утверждает обратное. Готовьтесь к ответу на вопрос: "Какие созданные вами представления изменяемы?".

6 лаба. Управление транзакциями

- [Справка](#)
- [Страница на Википедии, которая поможет разобраться в этой лабе](#)
- Для сдачи нужно знать суть всех аномалий и уметь их объяснить. Если вы сдаёте лабу в срок (без дисконта) в надежде получить не менее 6, то про все аномалии, кроме потерянных изменений вас спрашивать не будут, т.к. только потерянные изменения являются ключевым моментом лабы и по ним можно очень много чего спросить.
- Внимание! На сайте представлен неправильный пример потерянных изменений. Вам лучше написать два скрипта. Один как с сайта. Другой - правильный (согласно определению потерянных изменений).

Пример:

Первый скрипт:

```
DROP TABLE tmp;
```

```
CREATE TABLE tmp (id int, tmp_value int)
```

```
INSERT INTO tmp VALUES (1,0)
```

```
go
```

```
SELECT * FROM tmp
```

```
SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED
```

```
BEGIN TRANSACTION
```

```
UPDATE tmp SET tmp_value =
```

```
    (SELECT experience FROM doctor WHERE doctor_id = 301)
```

```
WHERE id = 1;
```

```
go
```

```
SELECT * FROM tmp
```

```
UPDATE doctor SET experience = (SELECT tmp_value FROM tmp WHERE id = 1)
```

```
WHERE doctor_id = 301
```

```
COMMIT;
```

```
SELECT d.doctor_id, d.experience
```

```
FROM doctor d
```

```
WHERE doctor_id = 301;
```

Второй скрипт:

```
SELECT d.doctor_id, d.experience
```

```
FROM doctor d
```

```
WHERE doctor_id = 301;
```

```
SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED
```

```
BEGIN TRANSACTION
```

```
UPDATE doctor SET experience = experience + 1 WHERE doctor_id = 301
```

```
COMMIT;
```

```
go
SELECT d.doctor_id, d.experience
FROM doctor d
WHERE doctor_id = 301;
```

- Пример потерянных изменений

Одна транзакция читает данные, вторая читает данные.

Первая, скажем, увеличивает значение ячейки на 1 (Типа было 90, стало $90+1=91$) и заканчивается.

Вторая увеличивает значение на 1 ($90+1=91$) и заканчивается. В итоге у нас в ячейке записано 91, хотя должно быть $90+1=91$, $91+1=92$. Это потерянные изменения. Этот пример привел Теймуразов.

Вопросы:

1. Можно ли в одной транзакции запустить другую транзакцию?

Ответ: Можно

2. Что будет, если во второй транзакции (в неглавной) поменять READ UNCOMMITTED на READ COMMITTED

Ответ: ничего не будет

3. Вопрос: Как узнать, не выполняя COMMIT или ROLLBACK, количество открытых транзакций для данного пользователя?

Ответ: Используем системную переменную @@TRANSCOUNT (узнать ее значение можно, выполнив

SELECT @@TRANSCOUNT).

4. Наверное, одним из самых сложных вопросов является: “Почему же тогда все уровни изоляции блокируют потерянные изменения, а ты их мне продемонстрировал? Неужели во всех справочниках ошибки?”.

КБТ после незнания ответа сказал мне его, но я ничего не понял, причина кроется в несоответствии “классического” определения и определения из справочников. [May the Force be with you, Luke!](#) -

кстати, в итоге не помогло. Общий смысл таков - в справочниках - аномалия, а это, продемонстрированное нами - криво спроектированная бд.

5. Свойства транзакций

6. Что выдаст, если запустить

```
begin tran
```

```
begin tran
```

```
rollback
```

```
commit
```

Ответ - ошибку: The COMMIT TRANSACTION request has no corresponding BEGIN TRANSACTION.

7. Какой уровень изоляции стоит по дефолту? Ответ - READ COMMITTED

7 лаба. Управление доступом

Лучше сделать всю лабу в одном скрипте.

[Вводный материал](#)

Вопросы:

1. Как поменять пароль

Ответ:

Можно использовать EXEC sp_password

Можно использовать ALTER LOGIN

Можно изменить вручную, используя Management Studio

2. Что такое WITH GRANT OPTION
3. Что такое Роль - именованный набор привелегий
4. Что такое GRANT, REVOKE и DENY
5. Что будет, если для некоторого пользователя сделать подряд DENY, GRANT, REVOKE для некоторой привилегии?
 - a. Ответ: ничего не изменится, у пользователя не появится никаких новых привилегий.
6. Что будет, если для роли пользователя сделали DENY, а для пользователя потом GRANT? Для роли GRANT, для пользователя потом DENY?
 - a. Ответ:
 - i. DENY в любом случае пользуется приоритетом над GRANT -- не всегда, есть случаи когда DENY не глушит GRANT, google в помощь (вне роли делаешь сначала deny потом grant)
 - b.
7. Чем отличается LOGIN от USER

Ответ:
LOGIN - это доступ уровня сервера
USER - это доступ уровня базы
LOGIN и USER выполняется скриптом:
CREATE LOGIN "login_name" FROM WIND. Таким образом, если на сервере лежит несколько баз данных, мы можем создать кому-то LOGIN для доступа к нескольким из них. Внутри каждой базы этот пользователь должен иметь свой USER.
8. Доп. задача: пользователю, который аутентифицируется из WINDOWS разрешить создавать базы данных.

Решение заключается в присвоении этому пользователю стандартной роли dbcreator. Создание OWS
CREATE USER "user_name" FOR LOGIN "login_name"

 - a. Замечание. Крайне нежелательно вместо роли использовать команду
GRANT CREATE DATABASE TO "user_name"
 - b. т.к. это влечёт некоторые дальнейшие неудобства.
9. Доп. задача: пользователю test разрешить создавать новые таблицы.
10. Что такое dbo в названии таблиц базы данных (например dbo.Price)?
 - a. Ответ: схема
 - b. + сопутствующие вопросы что такое схема и для чего она нужна?
11. Доп. задача: написать скрипт, который даёт право пользователю создавать представления. Какие привилегии на базовые таблицы, из которых создаётся представление, он должен иметь.

8 лаба. Выборка метаданных

- Сделайте все скрипты в одном запросе. Так удобнее (вас все равно попросят это сделать).

Вопросы:

1. К первому скрипту: убрать из всех выборок системные таблицы. Если их нет, то попробуйте, например, создать в обозревателе объектов диаграмму БД. При выводе всех таблиц в первом скрипте не должно быть таблицы sysdiagrams.

Ответ: возможны несколько вариантов устранения проблемы

- a. Использование fn_listextendedproperty(...).
- b. Использование таблицы sys.extended_properties, заключающееся в исключении из выборки всех объектов, id которых содержатся в ней, т.е. исключаются все системные таблицы.

```
AND sysobjects.id NOT IN (SELECT major_id FROM sys.extended_properties)
```

2. Вывести список всех ролей в базе.

Ответ:

```
SELECT "name"
FROM sys.database_principals
WHERE type = 'R'
```

3. Попросят переписать, используя новые версии системных представлений (например, sys.objects вместо sys.sysobjects).

4.

```
AND a.xtype = 'U'
AND a.uid = (SELECT uid
              FROM sys.sysusers
              WHERE name=user_name())
```

```
ORDER BY 1, 2
```

5. Доп. вопрос: для каждой таблицы вывести имя столбца, для которого используется IDENTITY

Ответ:

```
SELECT a.name AS table_name, b.name AS column_name, is_identity
FROM sysobjects a, sys.columns b
WHERE a.id = b.object_id and
      is_identity=1
```

6. Доп. вопрос: вывести все ограничения CHECK, присутствующие в базе. Возможно несколько правильных вариантов ответа.

- a. Присоединить к таблице sysobjects таблицу syscomments и выбрать название ограничения и его описание:

```
SELECT name, text
FROM sysobjects SO
INNER JOIN syscomments SC ON SO.id = SC.id
WHERE xtype = 'C'
```

- b. Использование INFORMATION_SCHEMA:

```
SELECT *
FROM INFORMATION_SCHEMA.CHECK_CONSTRAINTS
```

7. Доп. вопрос: вывести identity и их параметры:

```
SELECT a.name AS table_name, b.name AS column_name, b.is_identity, seed_value, increment_value,
```

```

last_value
FROM sysobjects a, sys.columns b, sys.identity_columns c
WHERE a.id = b.object_id
AND a.xtype = 'U'
AND b.is_identity = 1
AND b.object_id = c.object_id
ORDER BY 1, 2

```

9 лаба. Создание и использование триггеров

- Перед сдачей лабы внимательно прочтите, что от вас требуется. Возможно, написание такого триггера, как на сайте, невозможно для вашей БД. В этом случае следует на предыдущей сдаче (или по почте) попросить КБТ перефразировать условия, которым должен удовлетворять триггер.

Вопросы:

1. Продемонстрируйте, как работает триггер при добавлении/удалении/изменении строки, а потом нескольких строк. Тут у большинства триггер падает и надо исправить за пару.
2. В каком порядке могут выполняться триггеры на одно и то же действие?

Ответ:

Можно задать первый и последний, остальные в рандомном порядке

3. Доп. вопрос: создать таблицу с двумя полями user_login и logon_date, в которой будет храниться информация о пользователях и времени их захода в базу. После чего написать триггер, который при входе каждого нового пользователя будет эту таблицу заполнять.

```

CREATE TABLE LOGIN_DATA
(
    "user_login" VARCHAR(40),
    "logon_date" DATETIME
);
CREATE TRIGGER logon_event
ON ALL SERVER
AFTER LOGON
AS
INSERT INTO [database_name].[schema_name].LOGIN_DATA VALUES(ORIGINAL_LOGIN(),
GETDATE())

```

[database_name].[schema_name].LOGIN_DATA главное не забудьте явно указать название базы и название схемы например: Chess.dbo.login_data

10 лаба. Использование индексов и средств оптимизации запросов

Вопросы:

Если у вас express версия, то вы скорее всего не сможете воспользоваться анализатором ядра СУБД, и у вас нет SQL Profiler. Решение: переустановить(я поставил MS SQL STUDIO ENTERPRISE 2008 R2 например) или возьмите ноут у друга:-D

1. На плане выполнения запроса необходимо, чтобы на всех Ваших индексах было seek , а не scan. Например, так может получиться, если Вы использовали в запросе функцию year (some_date) = some_year. Решение данной проблемы: необходимо использовать some_date between yyyy-mm-dd and yyyy-mm-dd (зависит от формата вашей даты)
2. Спрашивал, сколько полей задействовано в первом запросе и где это можно посмотреть (в плане выполнения запроса навести мышку, чтобы появилось оранжевое окошко, последняя строчка - то, что вам нужно)
3. Попросил открыть запросы из 4 лабы и для одного из них дал задание: переписать, чтобы время выполнения было оптимальным.
4. Что такое worktable? Ответ на msdn.