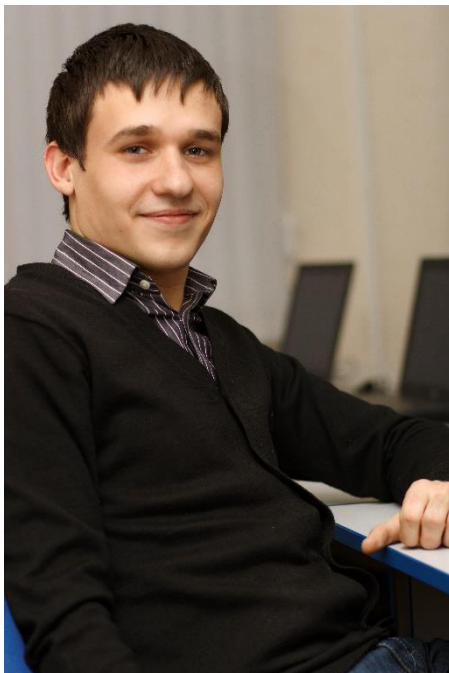


# SQL Базовый курс

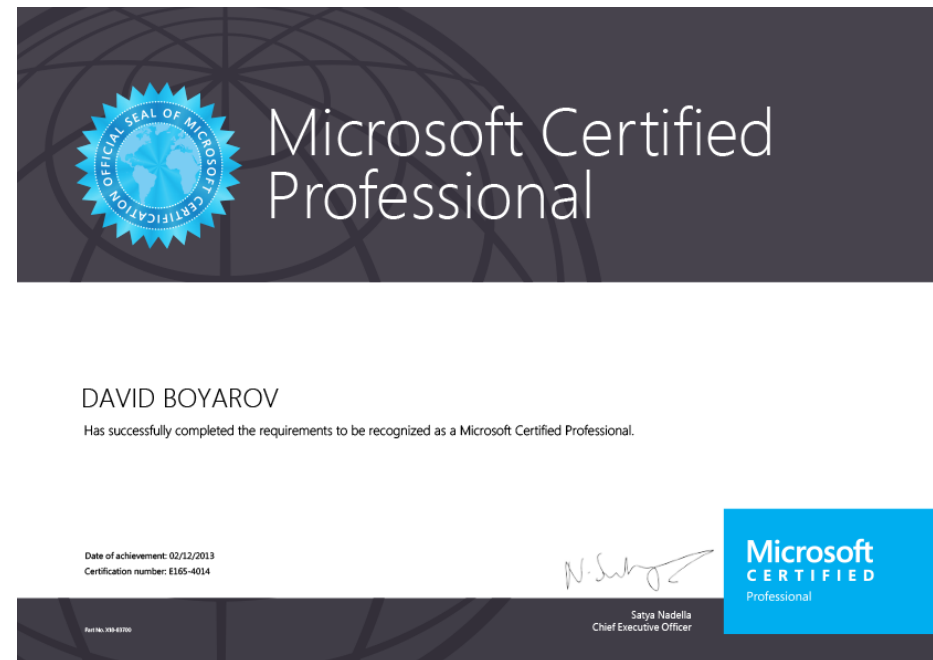
Транзакции

# SQL

## Автор курса



Давид Бояров



MCID: 9778145

# SQL

После урока обязательно



Повторите этот урок в видео формате на [ITVDN.com](http://itvdn.com)

Доступ можно получить через руководство вашего учебного центра



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://testprovider.com)

# SQL

Тема

## Транзакции

# ТРАНЗАКЦИЯ

## TRANSACTION

Транзакцией называется выполнение последовательности команд (SQL-конструкций) в базе данных, которая либо фиксируется при успешной реализации каждой команды, либо отменяется при неудачном выполнении хотя бы одной команды.

```
BEGIN TRANSACTION;
```

```
DECLARE @Id int;  
INSERT MyUserName VALUES ('TestName0', 'TestLName0');  
SET @Id = @@IDENTITY;  
INSERT MyUserTell VALUES (@Id, '(097) 2224455');
```

```
COMMIT TRANSACTION;
```

# ТРАНЗАКЦИЯ

## TRANSACTION

Откат транзакции – это действие, обеспечивающее аннулирование всех изменений данных, которые были сделаны в теле текущей незавершенной транзакции.

```
BEGIN TRANSACTION;
```

```
DECLARE @Id int;  
INSERT MyUserName VALUES ('TestName0', 'TestLName0');  
SET @Id = @@IDENTITY;  
INSERT MyUserTell VALUES (@Id, '(097) 2224455');
```

```
ROLLBACK TRANSACTION;
```

# ТРАНЗАКЦИЯ

## TRANSACTION

При выполнении транзакций несколькими пользователями одной базы данных могут возникать следующие проблемы:

- **Lost update**
- **Dirty reads**
- **Non-repeatable reads**
- **Phantom reads**

# ТРАНЗАКЦИЯ

## Lost update

**Lost update** – потерянное обновление. При одновременном изменении одного блока данных разными транзакциями одно из изменений теряется.

В обеих транзакциях изменяется значение поля f2, при этом одно из изменений теряется. Так что, f2 будет увеличено не на 45, а только на 20 или 25.

Это происходит потому, что:

1. Первая транзакция прочитала текущее состояние поля.
2. Вторая транзакция сделала свои изменения, основываясь на своих сохраненных в память данных.
3. Первая делает обновление поля, используя свои «старые» данные.

Транзакция 1	Транзакция 2
UPDATE tbl1 SET f2=f2+20 WHERE f1=1;	UPDATE tbl1 SET f2=f2+25 WHERE f1=1;



# ТРАНЗАКЦИЯ

## Dirty reads

**Dirty reads** – «грязное» чтение. Чтение данных, добавленных или изменённых транзакцией, которая впоследствии не подтвердится (откатится).

Первый пользователь начинает транзакцию, изменяющую данные. В это время другой пользователь (или создаваемая им транзакция) извлекает частично измененные данные, которые не являются корректными.

Транзакция 1	Транзакция 2
	SELECT f2 FROM tbl1 WHERE f1=1;
UPDATE tbl1 SET f2=f2+1 WHERE f1=1;	
	SELECT f2 FROM tbl1 WHERE f1=1;
ROLLBACK WORK;	

# ТРАНЗАКЦИЯ

## Non-repeatable reads

**Non-repeatable reads** – неповторяемое чтение. Ситуация, когда при повторном чтении в рамках одной транзакции ранее прочитанные данные оказываются изменёнными.

Первый пользователь начинает транзакцию, изменяющую данные. В это время другой пользователь начинает и завершает другую транзакцию. Первый пользователь при повторном чтении данных (например, если в его транзакцию входит несколько инструкций SELECT) получает другой набор записей.

Транзакция 1	Транзакция 2
	SELECT f2 FROM tbl1 WHERE f1=1;
UPDATE tbl1 SET f2=f2+1 WHERE f1=1;	
COMMIT;	
	SELECT f2 FROM tbl1 WHERE f1=1;

# ТРАНЗАКЦИЯ

## Phantom reads

**Phantom reads** – чтение фантомов. Ситуация, когда при повторном чтении в рамках одной транзакции одна и та же выборка дает разные множества строк.

Первый пользователь начинает транзакцию, выбирающую данные из таблицы. В это время другой пользователь начинает и завершает транзакцию, вставляющую или удаляющую записи. Первый пользователь получит другой набор данных, содержащий фантомы – удаленные или измененные строки.

Транзакция 1	Транзакция 2
	SELECT SUM(f2) FROM tbl1;
INSERT INTO tbl1 (f1,f2) VALUES (15,20);	
COMMIT;	
	SELECT SUM(f2) FROM tbl1;



От неповторяющегося чтения оно отличается тем, что результат повторного обращения к данным изменился не из-за изменения/удаления самих этих данных, а из-за появления новых (фантомных) данных.

# ТРАНЗАКЦИЯ

## TRANSACTION

Для решения этих проблем разработаны четыре уровня изоляции транзакции в SQL:

- **Read uncommitted.**
- **Read committed.**
- **Repeatable read.**
- **Serializable.**

# ТРАНЗАКЦИЯ

## TRANSACTION

**Read uncommitted.** Транзакция может считывать данные, с которыми работают другие транзакции. Применение этого уровня изоляции может привести ко всем перечисленным проблемам.

**Read committed.** Транзакция не может считывать данные, с которыми работают другие транзакции. Применение этого уровня изоляции исключает проблему «грязного» чтения.

**Repeatable read.** Транзакция не может считывать данные, с которыми работают другие транзакции. Другие транзакции также не могут считывать данные, с которыми работает эта транзакция. Применение этого уровня изоляции исключает все проблемы, кроме чтения фантомов.

**Serializable.** Транзакция полностью изолирована от других транзакций. Применение этого уровня изоляции полностью исключает все проблемы.

# ТРАНЗАКЦИЯ

## TRANSACTION

Поведение при различных уровнях изолированности

Уровень изолированности	Lost update (потерянное обновление)	Dirty reads («грязное» чтение)	Non-repeatable reads (неповторяемое чтение)	Phantom reads (чтение фантомов)
READ UNCOMMITTED		Разрешено	Разрешено	Разрешено
READ COMMITTED			Разрешено	Разрешено
REPEATABLE READ				Разрешено
SERIALIZABLE				

# ТРАНЗАКЦИЯ

## TRANSACTION

Уровни изолированности транзакций.  
Применение в SQL

```
BEGIN TRAN SET TRANSACTION ISOLATION LEVEL  
SERIALIZABLE  
DECLARE @OshibkiTabliciTourists int,  
@OshibkiTabliciInfoTourists int  
...  
ROLLBACK TRAN
```

# ТРИГГЕР

## TRIGGER

Триггер – это обработчик который можно выполнить во время выполнения операций `INSERT`, `UPDATE`, `DELETE`

Вместе с созданием триггера постоянно создаются две служебные таблицы: `inserted` и `deleted`

```
CREATE TRIGGER Production.ProductIsRationed
    ON Production.ProductInventory
    FOR INSERT, UPDATE
IF EXISTS
(
    SELECT *
    FROM inserted i
    JOIN deleted d
        ON d.ProductID = i.ProductID
        AND d.LocationID=i.LocationID
    WHERE (d.Quantity - i.Quantity) > d.Quantity/2
        AND d.Quantity - i.Quantity > 0
)
```



# Смотрите наши уроки в видео формате

ITVDN.com



Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://itvdn.com) для закрепления пройденного материала.

Все курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics



# Проверка знаний

TestProvider.com

TestProvider

Мы помогаем людям оценить себя

Главная Услуги и цены Центр Тестирования Поддержка О нас

Регистрация Войти

Поиск сертификата

Мы в социальных сетях

## Тестирование

Языки программирования и информационные технологии

**Microsoft**

C# ASP.NET MVC JavaScript Patterns OF Design SQL Architecture Guide WCF HTML&CSS XML SEO WPF HTML5&CSS3 JQuery XNA SharePoint GUI for Android Windows Azure Platform Microsoft Patterns&Practices TFS SCRUM ReSharper TDD WWF LINQ Entity Framework Windows Forms Refactoring Microsoft Expression Blend 4 Windows Phone 8 Windows 8 AppStore Visual Studio Tips&Tricks MSF MEF SilverLight AJAX MEF Service Oriented Architecture

Добро пожаловать на TestProvider.com!

Сайт перенесен на новую облачную платформу с использованием системы единой авторизации Single Sign On. Если вы хотите восстановить статистику по предыдущим экзаменам обратитесь в [службу поддержки](#). Для восстановления информации с предыдущей версии сайта, просба написать в службу поддержки Ваш старый и новый логины.

ITVDN PROMETRIC TEST CENTER CyberBionic Microsoft Partner Windows Azure Cloud Partner EBA

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](http://TestProvider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



# SQL

## Q&A

# Информационный видеосервис для разработчиков программного обеспечения

