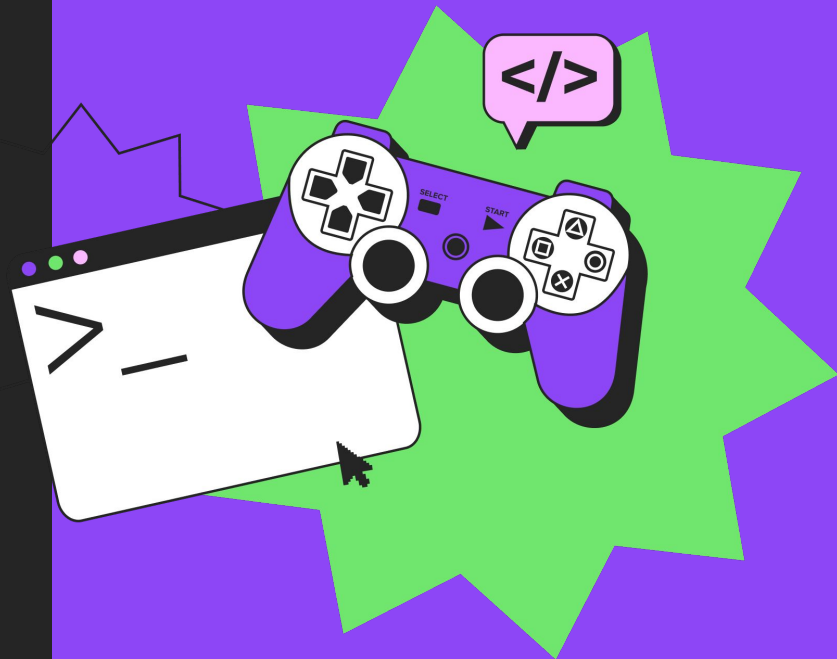


# Базы данных и SQL

## Урок 6

Транзакции, ACID. временные таблицы, управляющие конструкции, циклы





## План курса

1

Инструменты для работы с базами данных

2

SQL – создание объектов, изменение данных, логические операторы

3

SQL – выборка данных, сортировка, агрегатные функции

4

SQL – объединение таблиц union, соединение - join, подзапросы

5

SQL – оконные функции

6

Транзакции, ACID. Временные таблицы, управляющие конструкции, циклы

7

Будущий урок

8

Будущий урок

9

Будущий урок

10

Будущий урок

11

Будущий урок

12






Будущий урок



# Содержание урока



## Что будет на уроке сегодня

-  Понятие транзакции, свойства ACID.
-  Понятие временной таблицы
-  Переменные
-  Оператор IF
-  Оператор While



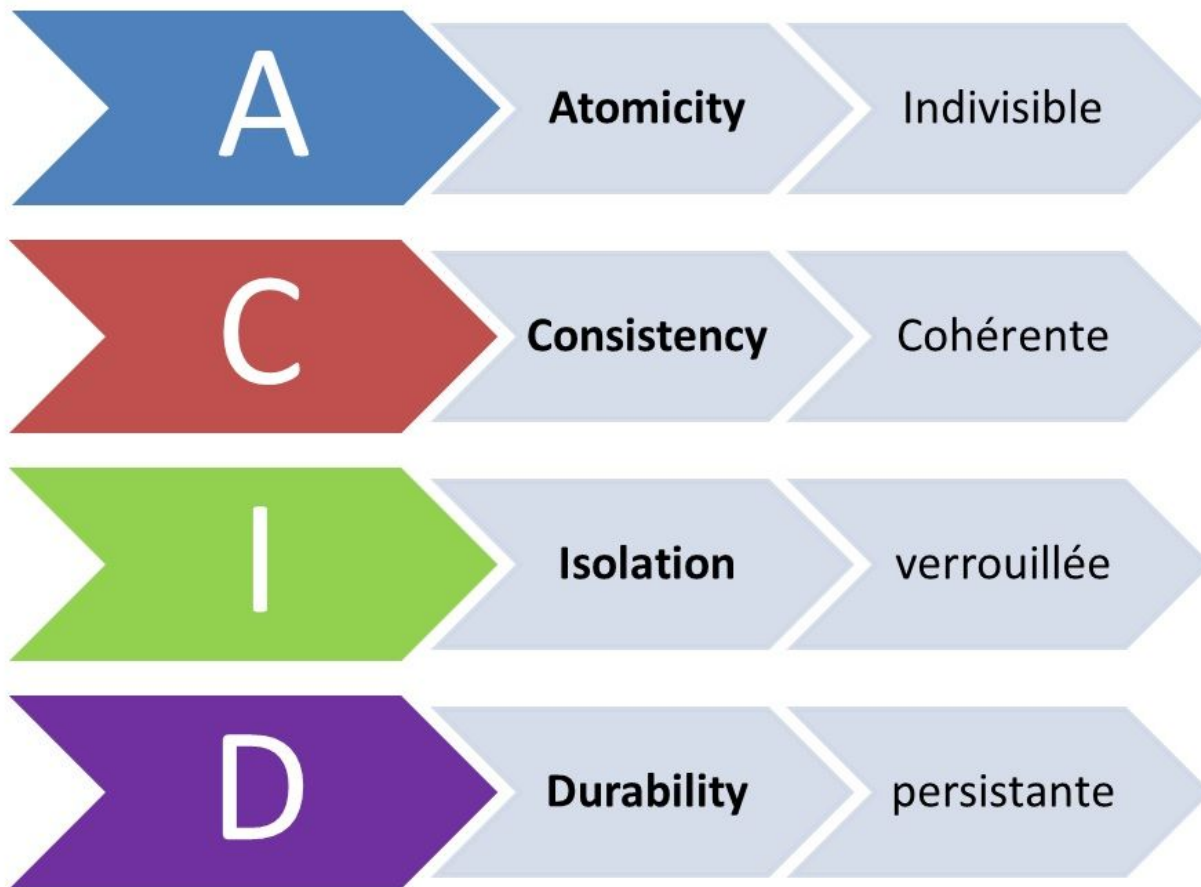
## Определение

**Транзакция** — это набор последовательных операций с базой данных, соединенных в одну логическую единицу.

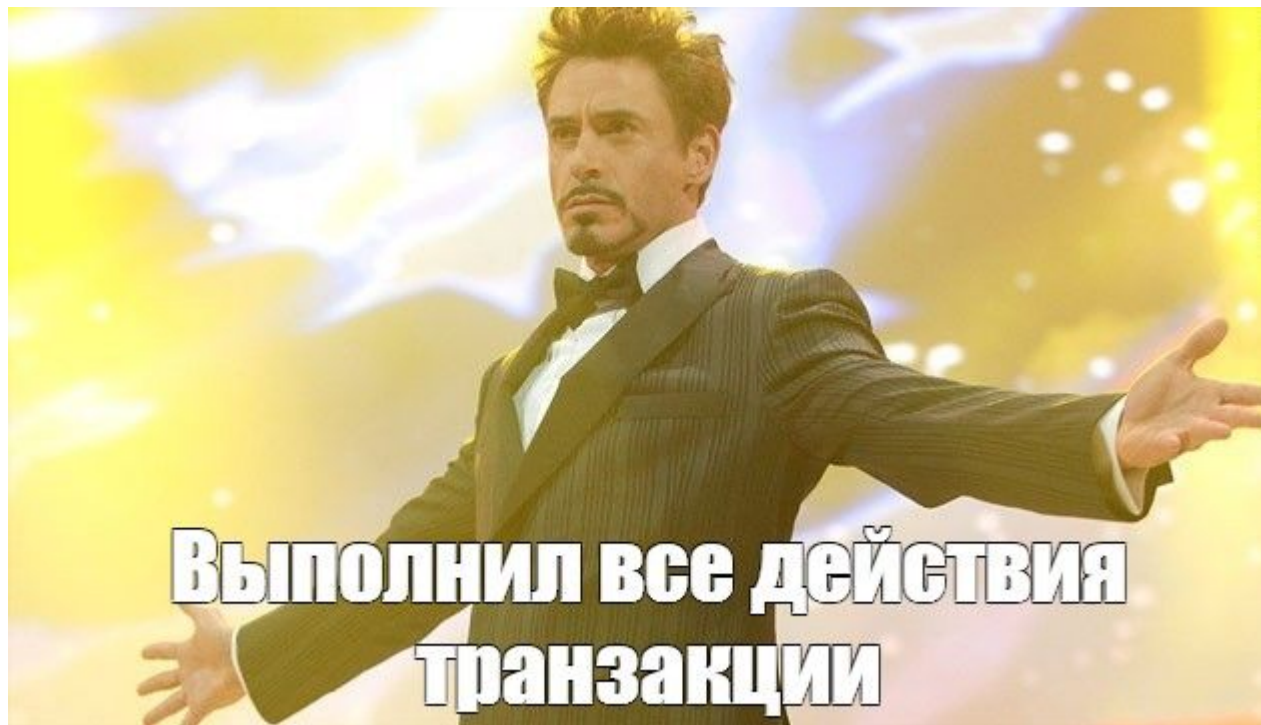
**Транзакции делят на два вида:**

**Неявные** транзакции, которые предусмотрены на уровне базы данных. Например, БД задает отдельную инструкцию INSERT, UPDATE или DELETE как единицу транзакции.

**Явные** транзакции — их начало и конец явно обозначаются такими инструкциями, как BEGIN TRANSACTION, COMMIT или ROLLBACK.



## Atomicity или атомарность (A)

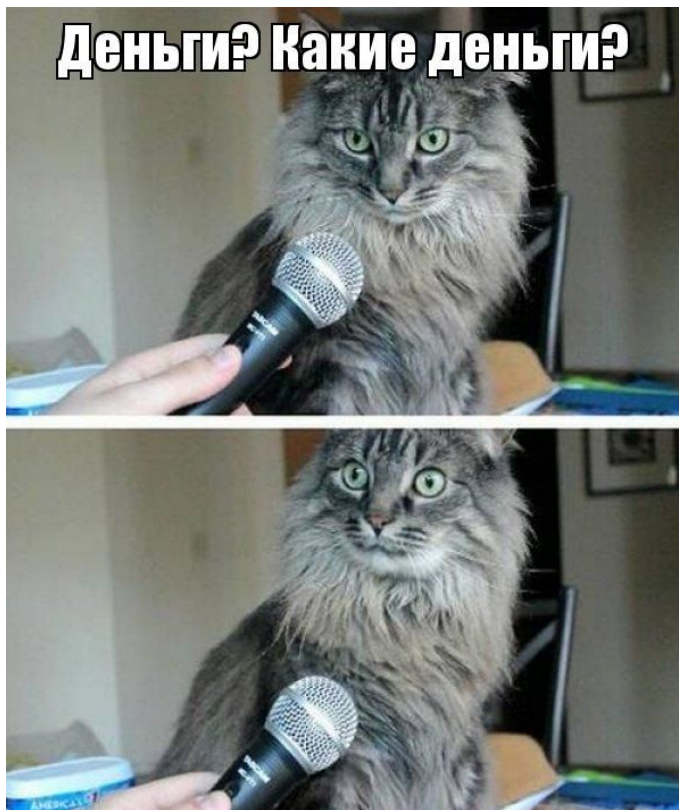


## Consistency или согласованность (C)





## Isolation или изоляция (I)





## Решение эффектов

### Read uncommitted

- ✓ Эффект потерянного обновления
- ✗ Эффект грязного чтения
- ✗ Эффект неповторяемого чтения
- ✗ Эффект чтения фантомов



## Решение эффектов

### Read committed

- ✓ Эффект потерянного обновления
- ✓ Эффект грязного чтения
- ✗ Эффект неповторяемого чтения
- ✗ Эффект чтения фантомов



## Решение эффектов

### Repeatable read

- ✓ Эффект потерянного обновления
- ✓ Эффект грязного чтения
- ✓ Эффект неповторяемого чтения
- ✗ Эффект чтения фантомов



## Решение эффектов

### Serializable

- ✓ Эффект потерянного обновления
- ✓ Эффект грязного чтения
- ✓ Эффект чтения фантомов
- ✓ Эффект неповторяемого чтения



## **Durability или долговечность (D)**

Долговечность означает, что если транзакция выполнена, и даже если в следующий момент произойдет сбой в системе, результат сохранится.



## Синтаксис

```
1 START TRANSACTION;  
2 SELECT total FROM accounts WHERE user_id = 2;  
3 UPDATE accounts SET total = total - 3000 WHERE user_id = 2;  
4 UPDATE accounts SET total = total + 3000 WHERE user_id IS NULL;  
5 COMMIT;
```



## Временные таблицы

```
1 CREATE TEMPORARY TABLE table_name(  
2     column_1_definition,  
3     column_2_definition,  
4     ... ,  
5     table_constraints  
6 );
```





## Переменная MySQL

1. Задание переменной:

```
1 SET @variable_name := value;  
2 SET @counter := 100;
```

2. Задание переменной, используя оператор SELECT:

```
1 SELECT @variable_name := value;
```



**Примеры переменных MySQL:**  
получаем самый дорогой продукт в таблице products и назначает цену пользовательской переменной @msrp:

```
1 SELECT
2     @msrp:=MAX(msrp)
3 FROM
4     products;
```

	@msrp:=MAX(msrp)
▶	214.3



## Оператор IF

```
1 IF(expr,if_true_expr,if_false_expr)
```

```
1 SELECT IF(400<2000, "YES", "NO");
```

```
1 SELECT IF(1 = 2,'true','false'); -- false  
2 SELECT IF(1 = 1,' true','false'); -- true
```



## Отображение N/A вместо NULL: изначально только NULL

```
1 SELECT
2     customer Number, customer Name, state, country
3 FROM
4     customers;
```

	customerNumber	customerName	state	country
▶	103	Atelier graphique	NULL	France
	112	Signal Gift Stores	NV	USA
	114	Australian Collectors, Co.	Victoria	Australia
	119	La Rochelle Gifts	NULL	France
	121	Baane Mini Imports	NULL	Norway
	124	Mini Gifts Distributors Ltd.	CA	USA
	125	Havel & Zbyszek Co	NULL	Poland
	128	Blauer See Auto, Co.	NULL	Germany



## Отображение N/A вместо NULL

```
1 SELECT
2     customerNumber,
3     customerName,
4     IF(state IS NULL, 'N/A', state) state,
5     country
6 FROM
7     customers;
```

	customerNumber	customerName	state	country
▶	103	Atelier graphique	N/A	France
	112	Signal Gift Stores	NV	USA
	114	Australian Collectors, Co.	Victoria	Australia
	119	La Rochelle Gifts	N/A	France
	121	Baane Mini Imports	N/A	Norway
	124	Mini Gifts Distributors Ltd.	CA	USA
	125	Havel & Zbyszek Co	N/A	Poland
	128	Blauer See Auto, Co.	N/A	Germany



## Оператор IF вместе с агрегатными функциями

```
1 SELECT
2     SUM(IF(status = 'Shipped', 1, 0)) AS Shipped,
3     SUM(IF(status = 'Cancelled', 1, 0)) AS Cancelled
4 FROM
5     orders;
```

	Shipped	Cancelled
▶	303	6



## Оператор COUNT IF

```
1 SELECT DISTINCT
2     status
3 FROM
4     orders
5 ORDER BY status;
```

	status
▶	Cancelled
	Disputed
	In Process
	On Hold
	Resolved
	Shipped



## Оператор COUNT IF

```
1 SELECT
2     COUNT(IF(status = 'Cancelled', 1, NULL)) Cancelled,
3     COUNT(IF(status = 'Disputed', 1, NULL)) Disputed,
4     COUNT(IF(status = 'In Process', 1, NULL)) 'In Process',
5     COUNT(IF(status = 'On Hold', 1, NULL)) 'On Hold',
6     COUNT(IF(status = 'Resolved', 1, NULL)) 'Resolved',
7     COUNT(IF(status = 'Shipped', 1, NULL)) 'Shipped'
8 FROM
9     orders;
```

	Cancelled	Disputed	In Process	On Hold	Resolved	Shipped
▶	6	3	6	4	4	303





## Синтаксис процедуры:

```
1 CREATE PROCEDURE procedure_name [ (parameter datatype [, parameter datatype]) ]  
2  
3 BEGIN  
4     declaration_section  
5     executable_section  
6 END;
```

**procedure\_name** - наименование процедуры в MySQL.

**parameter** - необязательный. Один или несколько параметров передаются в процедуру. При создании процедуры могут быть объявлены три типа параметров:

**IN** - Параметр может ссылаться на процедуру. Значение параметра не может быть перезаписано процедурой.

**OUT** - Параметр не может ссылаться на процедуру, но значение параметра может быть перезаписано процедурой.

**IN OUT** - Параметр может ссылаться на процедуру, и значение параметра может быть перезаписано процедурой.

**declaration\_section** - место в процедуре, где вы объявляете локальные переменные.

**executable\_section** - место в процедуре, в котором вы создаете код процедуры.



## Процедура: пример

```
mysql> DELIMITER &&
mysql> CREATE PROCEDURE Filter ()
-> BEGIN
-> SELECT * FROM data.record WHERE Country LIKE '%ia';
-> END &&
Query OK, 0 rows affected (0.11 sec)

mysql> DELIMITER ;
```

```
mysql> CALL Filter();
+-----+-----+-----+-----+
| id | Name   | City      | Country |
+-----+-----+-----+-----+
| 6  | Huzaifa | Saint_20  | Russia  |
| 7  | Kamran  | Leban%on  | Lebia   |
| 8  | Imama   | Sydney_20 | Australia |
+-----+-----+-----+-----+
3 rows in set (0.04 sec)

Query OK, 0 rows affected (0.07 sec)
```



## WHILE

```
1 [ label_name: ] WHILE condition DO
2
3     { ... statements ... }
4
5 END WHILE [ label_name ];
```

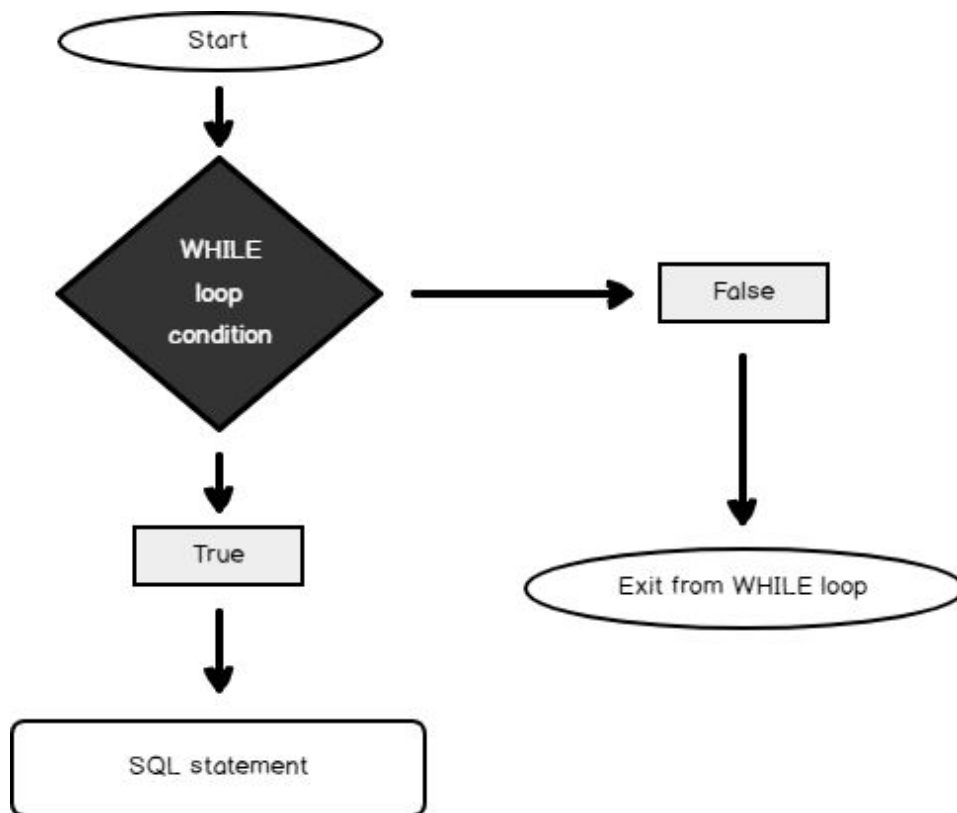
**label\_name** - необязательный. Это наименование связано с циклом WHILE.

**condition** - условие которое проверяется при каждой итерации цикла WHILE. Если condition примет значение TRUE, тело цикла выполняется. Если условие принимает значение FALSE, цикл WHILE прекращается.

**statements** - код, выполняемый при каждом, проходе через цикл WHILE.



# WHILE





## WHILE: пример на цикл

```
C:\Program Files\MySQL\MySQL Server 5.1\bin\mysql.exe
mysql>
mysql> SELECT magazine_incoming.id_incoming, products.name, products.author, magazine_incoming.quantity
-> FROM magazine_incoming, products
-> WHERE magazine_incoming.id_product=products.id_product;
```

id_incoming	name	author	quantity
1	Стихи о любви	Андрей Вознесенский	10
1	Собрание сочинений, том 2	Андрей Вознесенский	5
1	Собрание сочинений, том 3	Андрей Вознесенский	7
1	Русская поэзия	Николай Заболоцкий	10
1	Машенька	Владимир Набоков	10
1	Доктор Живаго	Борис Пастернак	8
1	Мертвые души	Николай Гоголь	8
1	Три сестры	Антон Чехов	8
1	Беглянка	Владимир Даль	8
2	Наши	Сергей Довлатов	10
2	Приглашение на казнь	Владимир Набоков	10
2	Лолита	Владимир Набоков	6
2	Темные аллеи	Иван Бунин	10
2	Дар	Владимир Набоков	10
2	Идиот	Федор Достоевский	10
2	Братья Карамазовы	Федор Достоевский	10
2	Ревизор	Николай Гоголь	10
2	Гранатовый браслет	Александр Куприн	10
3	Сын вождя	Юлия Вознесенская	10
3	Эмигранты	Алексей Толстой	10
3	Горе от ума	Александр Грибоедов	10
3	Анна Каренина	Лев Толстой	10
3	Повести и рассказы	Николай Лесков	10
3	Антоновские яблоки	Иван Бунин	10

```
24 rows in set (0.22 sec)

mysql> _
```



## WHILE: результат по каждой поставке отдельно

```
1
2 SELECT magazine_incoming.id_incoming, products.name, products.author,
   magazine_incoming.quantity
3 FROM magazine_incoming, products
4 WHERE magazine_incoming.id_product=products.id_product AND
   magazine_incoming.id_incoming=1;
5
6 SELECT magazine_incoming.id_incoming, products.name, products.author,
   magazine_incoming.quantity
7 FROM magazine_incoming, products
8 WHERE magazine_incoming.id_product=products.id_product AND
   magazine_incoming.id_incoming=2;
9
10 SELECT magazine_incoming.id_incoming, products.name, products.author,
   magazine_incoming.quantity
11 FROM magazine_incoming, products
12 WHERE magazine_incoming.id_product=products.id_product AND
   magazine_incoming.id_incoming=3;
```

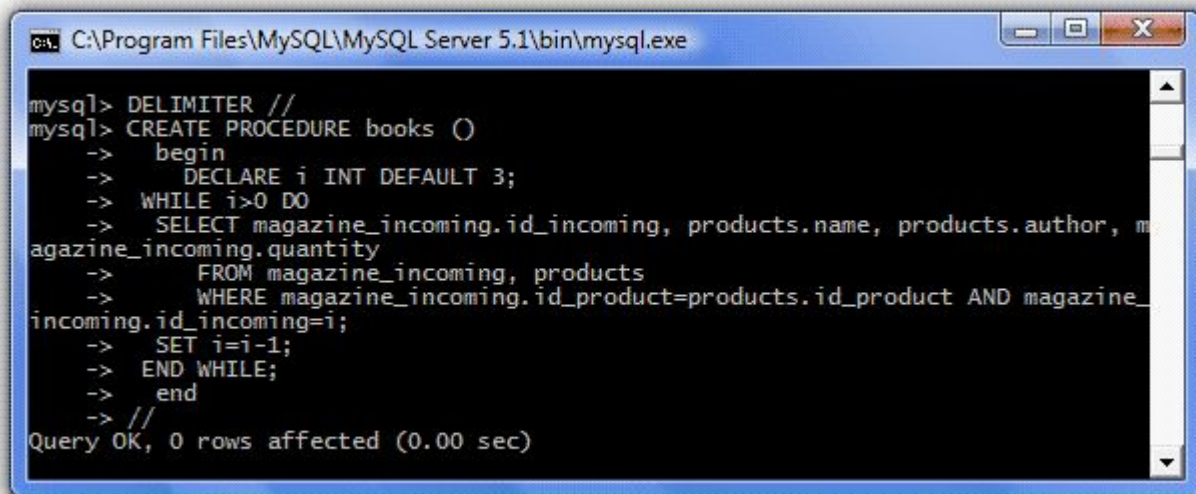


## Запишем запрос короче, используя while:

```
1 DECLARE i INT DEFAULT 3;
2 WHILE i>0 DO
3     SELECT magazine_incoming.id_incoming, products.name, products.author, magazine_incoming.quantity
4     FROM magazine_incoming, products
5     WHERE magazine_incoming.id_product=products.id_product AND magazine_incoming.id_incoming=i;
6     SET i=i-1;
7 END WHILE;
```



## Процедура с помощью цикла while



```
C:\Program Files\MySQL\MySQL Server 5.1\bin\mysql.exe

mysql> DELIMITER //
mysql> CREATE PROCEDURE books ()
-> begin
->     DECLARE i INT DEFAULT 3;
->     WHILE i>0 DO
->         SELECT magazine_incoming.id_incoming, products.name, products.author, m
magazine_incoming.quantity
->         FROM magazine_incoming, products
->         WHERE magazine_incoming.id_product=products.id_product AND magazine_
incoming.id_incoming=i;
->         SET i=i-1;
->     END WHILE;
-> end
-> //
Query OK, 0 rows affected (0.00 sec)
```



## Вызов процедуры через CALL()

```
C:\Program Files\MySQL\MySQL Server 5.1\bin\mysql.exe
mysql> CALL books ()//
```

id_incoming	name	author	quantity
3	Сын вождя	Юлия Вознесенская	10
3	Эмигранты	Алексей Толстой	10
3	Горе от ума	Александр Грибоедов	10
3	Анна Каренина	Лев Толстой	10
3	Повести и рассказы	Николай Лесков	10
3	Антоновские яблоки	Иван Бунин	10

6 rows in set (0.03 sec)

id_incoming	name	author	quantity
2	Наши	Сергей Довлатов	10
2	Приглашение на казнь	Владимир Набоков	10
2	Лолита	Владимир Набоков	6
2	Темные аллеи	Иван Бунин	10
2	Дар	Владимир Набоков	10
2	Идиот	Федор Достоевский	10
2	Братья Карамазовы	Федор Достоевский	10
2	Ревизор	Николай Гоголь	10
2	Гранатовый браслет	Александр Куприн	10

9 rows in set (0.10 sec)

id_incoming	name	author	quantity
1	Стихи о любви	Андрей Вознесенский	10
1	Собрание сочинений, том 2	Андрей Вознесенский	5
1	Собрание сочинений, том 3	Андрей Вознесенский	7
1	Русская поэзия	Николай Заболоцкий	10
1	Машенька	Владимир Набоков	10
1	Доктор Живаго	Борис Пастернак	8
1	Мертвые души	Николай Гоголь	8
1	Три сестры	Антон Чехов	8
1	Беглянка	Владимир Даль	8

9 rows in set (0.20 sec)

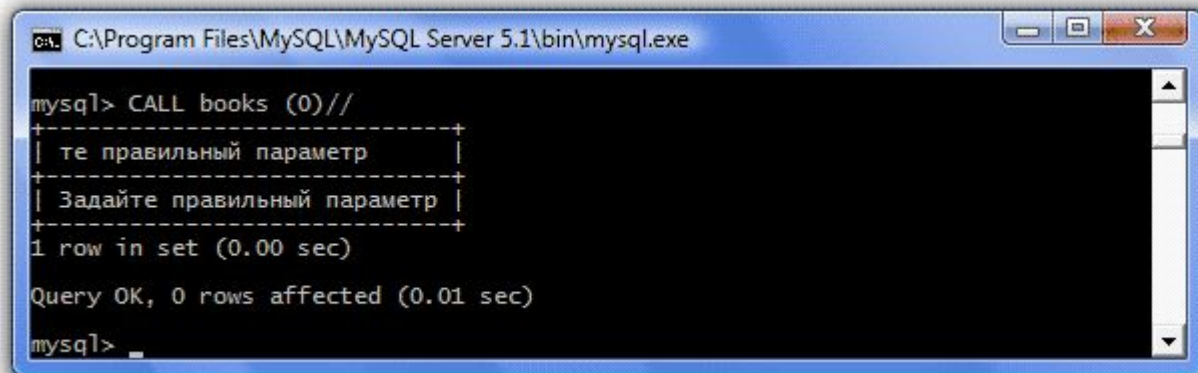


## Модификация процедуры:

```
1
2 CREATE PROCEDURE books (IN num INT)
3   begin
4     DECLARE i INT DEFAULT 0;
5     IF (num>0) THEN
6       WHILE i < num DO
7         SELECT magazine_incoming.id_incoming, products.name, products.author,
magazine_incoming.quantity
8         FROM magazine_incoming, products
9         WHERE magazine_incoming.id_product=products.id_product AND magazine_incoming.id_incoming=i;
10        SET i=i+1;
11      END WHILE;
12    ELSE
13      SELECT  'Задайте правильный параметр';
14    END IF;
15  end
16 //
```



## Модификация процедуры:



```
C:\Program Files\MySQL\MySQL Server 5.1\bin>mysql.exe






mysql> CALL books (0)//
+-----+
| те правильный параметр |
+-----+
| Задайте правильный параметр |
+-----+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)

mysql> _
```



## Итоги занятия:

-  Понятие транзакции, свойства ACID.
-  Понятие временной таблицы
-  Переменные
-  Оператор IF
-  Оператор While



**Спасибо за внимание!**