



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ *Робототехники и комплексной автоматизации*

КАФЕДРА *Системы автоматизированного проектирования (РК-6)*

## **ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

по дисциплине: «Базы данных»

Студент Базан Ольга Владимировна

Группа РК6-44Б

Тип задания Лабораторная работа

Тема лабораторной работы Банковские счета

Студент \_\_\_\_\_ **Базан О. В.**  
*подпись, дата* *фамилия, и.о.*

Преподаватель \_\_\_\_\_ **Пивоварова Н. В.**  
*подпись, дата* *фамилия, и.о.*

*Москва, 2022 г.*

## Оглавление

Описание предметной области .....	3
Инфологическая модель предметной области.....	4
Логическая модель базы данных .....	5
Физическое проектирование базы данных.....	6
Тестовые данные .....	11
Простые запросы.....	12
Сложные запросы.....	14
Процедура .....	17

## Описание предметной области

### Вариант 17. Банковские счета.

В базе данных банка хранятся данные о счетах клиентов.

О *клиентах* известно: уникальный номер договора, фамилия, номер телефона, дата рождения, домашний адрес, дата заключения договора, дата расторжения договора. Каждый клиент может открыть любое количество счетов.

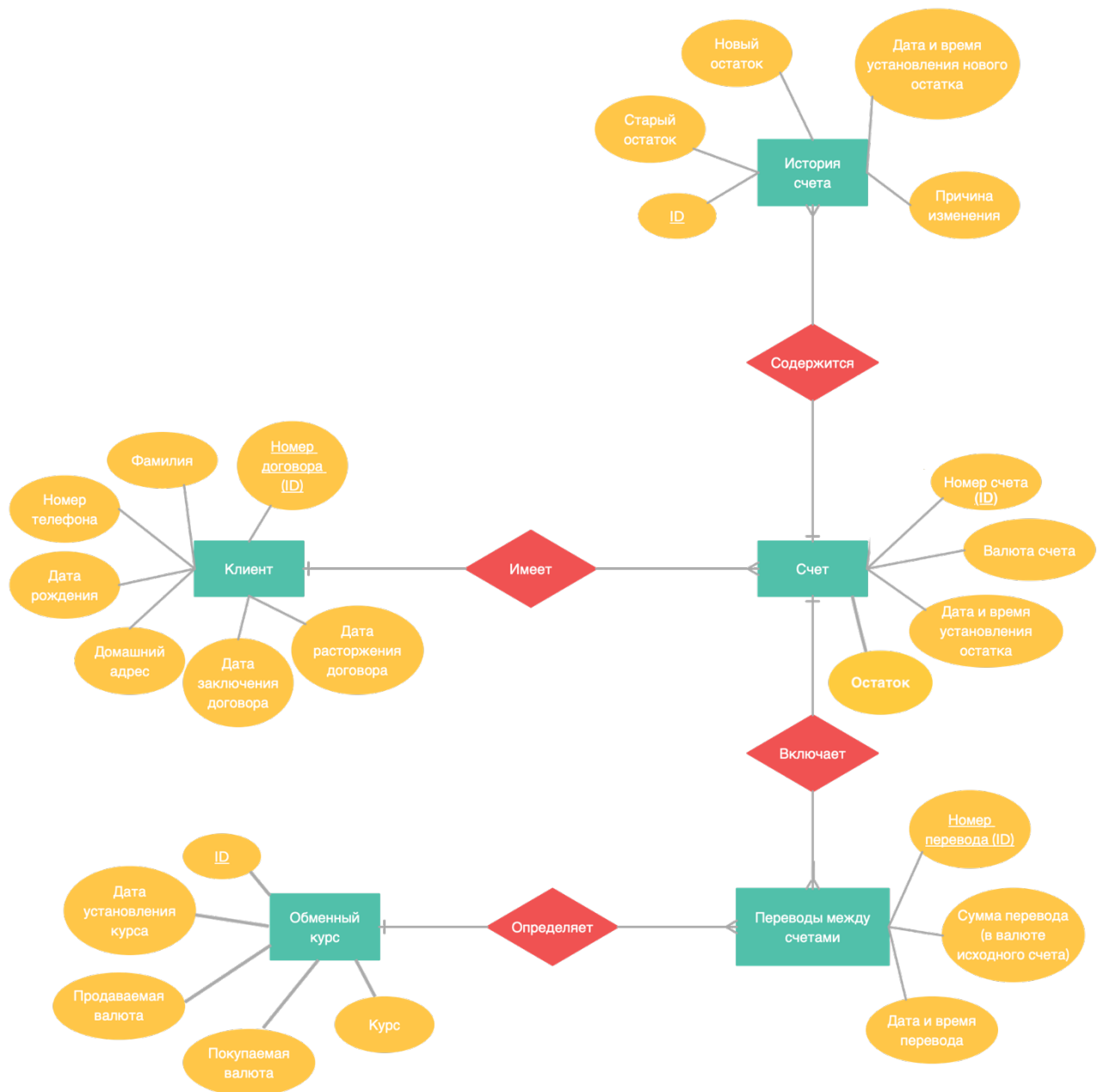
По каждому *счету* известно: уникальный 12-значный номер, валюта счета, остаток на счете, дата и время установления остатка.

Сохраняется также *история счетов* в форме: старый остаток, новый остаток, дата и время установления нового остатка, причина изменения. Причинами изменения остатка на счете могут быть пополнение счета, перевод денег со счета на счет, снятие наличности.

Отдельно сохраняется *история переводов* между счетами одного и того же клиента в форме: сумма перевода в валюте исходного счета, дата и время перевода, номер исходного счета, номер целевого счета, номер исходного счета, номер целевого счета. *Внутренний курс банка* на каждую дату также сохраняется в базе данных.

При обращении клиента оператор банка находит его в базе данных и далее может выполнить любую операцию по переводу денег со счета на счет, пополнение остатка на счете или списание денежных сумм со счета.

## Инфологическая модель предметной области



## Логическая модель базы данных

client\_t

client_id	surname	phone_number	birth_date	address	start_of_deal	end_of_deal
Pk						

account\_t

account_id	currency	balance	balance_dt	client_id
Pk				Fk

transaction\_t

transaction_id	amount_of_money	transaction_dt	account_id_from	account_id_to	exchange_rate_id
Pk			Fk	Fk	Fk

exchange\_rate

exchange_rate_id	exchange_rate_dt	sold_currency	bought_currency	rate
Pk				

account\_history

account_history_id	prev_balance	new_balance	account_history_dt	action_h	account_id
Pk					Fk

# Физическое проектирование базы данных

База данных bank

CREATE DATABASE BANK;

## Таблица client\_t








Column	Datatype		PK	NN	UQ	B...	UN	ZF	AI	G	Default / Expression
 client_id	INT	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 surname	VARCHAR(50)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 phone_number	CHAR(12)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 birth_date	DATE	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 address	VARCHAR(200)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 start_of_deal	DATE	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 end_of_deal	DATE	↕	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис 1.1. Параметры таблицы client\_t

Index	Type
PRIMARY	PRIMARY
	↕

Рис 1.2. Индексы таблицы client\_t

## Таблица account\_t






Column	Datatype		PK	NN	UQ	B...	UN	ZF	AI	G	Default / Expression
 account_id	CHAR(12)	↕	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 currency	VARCHAR(10)	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 balance	FLOAT	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 balance_dt	DATETIME	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 client_id	INT	↕	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Рис. 2.1 Параметры таблицы account\_t

Foreign Key	Referenced Table	Foreign key details 'client_id'	
client_id	`bank`.`client_t`	Column	Referenced Column
<click to edit>		<input type="checkbox"/> account_id	
		<input type="checkbox"/> currency	
		<input type="checkbox"/> balance	
		<input type="checkbox"/> balance_dt	
		<input checked="" type="checkbox"/> client_id	client_id

Рис. 2.2 Внешние ключи таблицы account\_t

Index	Type
PRIMARY	PRIMARY
client_id_idx	INDEX

Рис. 2.3 Индексы таблицы account\_t

### Таблица exchange\_rate






Column	Datatype	PK	NN	UQ	B...	UN	ZF	AI	G	Default / Expression
 exchange_rate_id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
 exchange_rate_dt	DATE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 sold_currency	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 bought_currency	VARCHAR(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 rate	DOUBLE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Рис. 3.1 Параметры таблицы exchange\_rate

Index	Type
PRIMARY	PRIMARY

Рис. 3.2 Индексы таблицы exchange\_rate

## Таблица transaction\_t







Column	Datatype	PK	NN	UQ	B...	UN	ZF	AI	G	Default / Expression
 transaction_id	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
 amount_of_m...	FLOAT	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 transaction_dt	DATETIME	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 account_id_f...	CHAR(12)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 account_id_to	CHAR(12)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 exchange_rat...	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

Рис. 4.1 Параметры таблицы transaction\_t

Foreign Key	Referenced Table		Foreign key details 'account_id_from'	
account_id_from	`bank`.`account_t`	<input type="checkbox"/>	Column	Referenced Column
account_id_to	`bank`.`account_t`	<input type="checkbox"/>	<input type="checkbox"/> transaction_id	<input type="checkbox"/>
exchange_rate_id	`bank`.`exchange_rate`	<input type="checkbox"/>	<input type="checkbox"/> amount_of_money	<input type="checkbox"/>
<click to edit>		<input type="checkbox"/>	<input type="checkbox"/> transaction_dt	<input type="checkbox"/>
		<input type="checkbox"/>	<input checked="" type="checkbox"/> account_id_from	account_id
		<input type="checkbox"/>	<input type="checkbox"/> account_id_to	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/> exchange_rate_id	<input type="checkbox"/>

Рис. 4.2 Внешние ключи таблицы transaction\_t

Foreign Key	Referenced Table		Foreign key details 'account_id_to'	
account_id_from	`bank`.`account_t`	<input type="checkbox"/>	Column	Referenced Column
account_id_to	`bank`.`account_t`	<input type="checkbox"/>	<input type="checkbox"/> transaction_id	<input type="checkbox"/>
exchange_rate_id	`bank`.`exchange_rate`	<input type="checkbox"/>	<input type="checkbox"/> amount_of_money	<input type="checkbox"/>
<click to edit>		<input type="checkbox"/>	<input type="checkbox"/> transaction_dt	<input type="checkbox"/>
		<input type="checkbox"/>	<input type="checkbox"/> account_id_from	<input type="checkbox"/>
		<input type="checkbox"/>	<input checked="" type="checkbox"/> account_id_to	account_id
		<input type="checkbox"/>	<input type="checkbox"/> exchange_rate_id	<input type="checkbox"/>

Рис. 4.3 Внешние ключи таблицы transaction\_t



Foreign Key	Referenced Table		Foreign key details 'exchange_rate_id'	
account_id_from	`bank`.`account_t`	↕	Column	Referenced Column
account_id_to	`bank`.`account_t`	↕	<input type="checkbox"/> transaction_id	↕
exchange_rate_id	`bank`.`exchange_rate`	↕	<input type="checkbox"/> amount_of_money	↕
<click to edit>		↕	<input type="checkbox"/> transaction_dt	↕
			<input type="checkbox"/> account_id_from	↕
			<input type="checkbox"/> account_id_to	↕
			<input checked="" type="checkbox"/> exchange_rate_id	exchange_rate_id ↕

Рис. 4.4 Внешние ключи таблицы transaction\_t

Index	Type
PRIMARY	PRIMARY
exchange_rate_id_idx	INDEX
account_id_from_idx	INDEX
account_id_to_idx	INDEX

Рис. 4.5 Индексы таблицы transaction\_t

## Таблица account\_history







Column	Datatype	PK	NN	UQ	B...	UN	ZF	AI	G	Default / Expression
 account_hist...	INT	↕ <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
 prev_balance	FLOAT	↕ <input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 new_balance	FLOAT	↕ <input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 account_hist...	DATETIME	↕ <input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 action_h	VARCHAR(45)	↕ <input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
 account_id	CHAR(12)	↕ <input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Рис. 5.1 Параметры таблицы account\_history

Foreign Key	Referenced Table		Foreign key details 'account_history_ibfk_1'	
account_history_ibfk_1	`bank`.`account_t`	↕	Column	Referenced Column
<click to edit>		↕	<input type="checkbox"/> account_history_id	↕
			<input type="checkbox"/> prev_balance	↕
			<input type="checkbox"/> new_balance	↕
			<input type="checkbox"/> account_history_dt	↕
			<input type="checkbox"/> action_h	↕
			<input checked="" type="checkbox"/> account_id	account_id ↕

Рис. 5.2 Внешние ключи таблицы account\_history

Index	Type	
PRIMARY	PRIMARY	↕
account_id_idx	INDEX	↕

*Рис. 5.3 Индексы таблицы account\_history*

## Тестовые данные

Для проверки результатов запросов внесем в базу тестовые данные

Таблица client\_h

client_id	surname	phone_number	birth_date	address	start_of_deal	end_of_deal
1	Астафьев	+79856743254	1978-03-23	г. Москва, Дубининская улица, 63с2, кв 52	2009-04-21	2023-01-23
2	Васильев	+79098764531	1973-09-08	г. Химки, Спортивная улица, 10	2020-05-22	2023-01-22
3	Иванова	+79096547888	2001-06-13	г. Москва, 1-й Колобовский переулок, 15/6, кв 15	2020-03-07	NULL
4	Гордеева	+79047832333	1998-07-24	Раменский городской округ, деревня Малое Саврасово, 96А	2020-05-21	NULL
5	Кузьмин	+79058764553	2004-05-16	г. Москва, Хитровский переулок, 2/8с5, кв 203	2020-01-12	NULL

Таблица account\_h

account_id	currency	balance	balance_dt	client_id	^
173477383888	RUB	74210	2020-12-07 12:02:17	3	
265533352627	USD	1500	2020-12-07 12:02:17	3	
273849984837	RUB	0	2022-12-20 13:27:15	2	
362663773362	EUR	0	2020-03-10 10:02:51	1	
372727772839	RUB	170000	2020-09-13 14:07:31	4	
777362199000	RUB	0	2022-11-13 14:53:11	1	
848726364789	RUB	73000	2022-12-30 18:12:16	5	
874326554335	USD	0	2020-03-10 09:53:22	1	

Таблица exchange\_rate

exchange_rate_id	exchange_rate_dt	sold_currency	bought_currency	rate
1	2010-06-13	USD	RUB	31.17
2	2010-06-13	RUB	USD	0.032082130253449
3	2010-06-13	EUR	RUB	38.11
4	2010-06-13	RUB	EUR	0.026239832065075
5	2020-03-01	USD	RUB	66.61
6	2020-03-01	RUB	USD	0.01501276084672
7	2020-03-01	EUR	RUB	66.7
8	2020-03-01	RUB	EUR	0.014992503748126
9	2020-12-07	USD	RUB	74.21
10	2020-12-07	RUB	USD	0.013475272874276
11	2020-12-07	EUR	RUB	74.13
12	2020-12-07	RUB	EUR	0.013489815189532
13	2020-03-01	USD	EUR	0.998650674662669

Таблица transaction\_t

transaction_id	amount_of_money	transaction_dt	account_id_from	account_id_to	exchange_rate_id
1	150000	2010-06-13 12:01:13	777362199000	874326554335	2
2	300	2020-03-01 11:00:43	874326554335	362663773362	13
3	1000	2020-12-07 12:02:17	265533352627	173477383888	9

Таблица account\_history

account_history_id	prev_balance	new_balance	account_history_dt	action_h	account_id
1	0	200000	2009-04-21 13:09:21	Пополнение	777362199000
2	200000	50000	2010-06-13 12:01:13	Перевод	777362199000
3	0	4812.32	2010-06-13 12:01:13	Пополнение	874326554335
4	4812.32	4512.32	2020-03-01 11:00:43	Перевод	874326554335
5	0	299.59	2020-03-01 11:00:43	Пополнение	362663773362
6	50000	0	2022-11-13 14:53:11	Снятие	777362199000
7	4512.32	0	2020-03-10 09:53:22	Снятие	362663773362
8	0	100000	2020-08-21 16:21:20	Пополнение	273849984837
9	100000	0	2022-12-20 13:27:15	Снятие	273849984837
10	0	2500	2020-10-12 10:03:41	Пополнение	265533352627
11	2500	1500	2020-12-07 12:02:17	Перевод	265533352627
12	0	74210	2020-12-07 12:02:17	Пополнение	173477383888
13	0	170000	2020-09-13 14:07:31	Пополнение	372727772839
14	0	23000	2020-05-10 08:53:19	Пополнение	848726364789
15	23000	73000	2022-12-30 18:12:16	Пополнение	848726364789

## Простые запросы

1. Показать, сколько клиентов заключили контракты в каждом месяце первого полугодия 2020 года.

**SELECT**

```
    MONTH(start_of_deal) AS deal_month,  
    COUNT(*) AS clients_count  
FROM client_t  
WHERE YEAR(start_of_deal) = 2020 AND MONTH(start_of_deal) <= 6  
GROUP BY MONTH(start_of_deal)
```

*Результат выполнения запроса*

deal_month	clients_count
5	2
3	1
1	1

2. Показать все сведения о клиентах, расторгнувших контракты за последние 5 дней.

```
SELECT *  
FROM client_t  
WHERE CURDATE() - end_of_deal < 5
```

*Результат выполнения запроса*

client_id	surname	phone_number	birth_date	address	start_of_deal	end_of_deal
1	Астафьев	+79856743254	1978-03-23	г. Москва, Дубининская улица, 63с2, кв 52	2009-04-21	2023-01-23
2	Васильев	+79098764531	1973-09-08	г. Химки, Спортивная улица, 10	2020-05-22	2023-01-22

3. Показать количество счетов у каждого клиента.

**SELECT**

```
    client_id,  
    COUNT(account_id) AS accounts_count  
FROM account_t  
GROUP BY client_id
```

*Результат выполнения запроса*

client_id	accounts_count
1	3
2	1
3	2
4	1
5	1

4. Найти максимальный по модулю остаток средств на счетах.

```
SELECT MAX(balance) AS max_balance
FROM account_t
```

Результат выполнения запроса

max_balance
170000

5. Показать, сколько раз изменялись остатки на каждом из счетов в марте 2020 года.

```
SELECT
    account_id,
    COUNT(*) AS changing_count
FROM account_history
WHERE YEAR(account_history_dt) = 2020 AND MONTH(account_history_dt) = 3
GROUP BY account_id
```

Результат выполнения запроса

account_id	changing_count
362663773362	2
874326554335	1

6. Показать все сведения о клиентах с номерами в диапазоне от XXXXXXXXXXXX до YYYYYYYYYYYY.

```
SELECT *
FROM client_t
WHERE phone_number BETWEEN +79000000000 AND +79060000000
```

Результат выполнения запроса

client_id	surname	phone_number	birth_date	address	start_of_d...	end_of_deal
4	Гордеева	+79047832333	1998-07-24	Раменский городской округ, деревня Малое...	2020-05-21	NULL
5	Кузьмин	+79058764553	2004-05-16	г. Москва, Хитровский переулок, 2/8с5, кв 203	2020-01-12	NULL

## Сложные запросы

1. Показать отчет о действиях (пополнение, снятие, перевод) со счетами в 2020 году по форме:  
номер счета, фамилия владельца счета, количество действий.

**SELECT**

```
account_id AS 'Номер счета',  
surname AS 'Фамилия владельца счета',  
COUNT(action_h) AS 'Количество действий'
```

**FROM** client\_t

**JOIN** account\_t

**USING**(client\_id)

**JOIN** account\_history

**USING**(account\_id)

**WHERE** YEAR(account\_history\_dt) = 2020

**GROUP BY** account\_id

*Результат выполнения запроса*

Номер счета	Фамилия владельца счета	Количество действий
362663773362	Астафьев	2
874326554335	Астафьев	1
273849984837	Васильев	1
173477383888	Иванова	1
265533352627	Иванова	2
372727772839	Гордеева	1
848726364789	Кузьмин	1

2. Показать количество переводов с каждого из счетов клиента по фамилии XXXXX.

**SELECT**

```
account_id,  
COUNT(transaction_id) AS transactions_count
```

**FROM** client\_t **AS** c

**JOIN** account\_t **AS** a

**USING**(client\_id)

**LEFT JOIN** transaction\_t **AS** t

**ON** a.account\_id = t.account\_id\_from

**WHERE** surname = 'Астафьев'

**GROUP BY** account\_id

*Результат выполнения запроса*

account_id	transactions_count
362663773362	0
777362199000	1
874326554335	1

3. Показать все сведения о самом молодом клиенте банка.

```
SELECT *  
FROM client_t  
WHERE birth_date = (SELECT MAX(birth_date)  
                    FROM client_t)
```

*Результат выполнения запроса*

client_id	surname	phone_number	birth_date	address	start_of_d...	end_of_deal
5	Кузьмин	+79058764553	2004-05-16	г. Москва, Хитровский переулок, 2/8с5, кв 203	2020-01-12	NULL

4. Показать все сведения о счетах, с которых ни разу не переводились деньги на другие счета.

```
SELECT a.*  
  
FROM account_t AS a  
LEFT JOIN transaction_t AS t  
ON a.account_id = t.account_id_from  
  
WHERE transaction_id IS NULL
```

*Результат выполнения запроса*

account_id	currency	balance	balance_dt	client_id
173477383888	RUB	74210	2020-12-07 12:02:17	3
273849984837	RUB	0	2022-12-20 13:27:15	2
362663773362	EUR	0	2020-03-10 10:02:51	1
372727772839	RUB	170000	2020-09-13 14:07:31	4
848726364789	RUB	73000	2022-12-30 18:12:16	5

5. Показать все сведения о счетах, с которых ни разу не переводились деньги в марте 2020 года.

```
SELECT a.*  
  
FROM account_t AS a  
LEFT JOIN (SELECT *  
           FROM transaction_t  
           WHERE transaction_dt LIKE '2020-03%') AS t
```

```

ON a.account_id = t.account_id_from
WHERE transaction_id IS NULL

```

Результат выполнения запроса

account_id	currency	balance	balance_dt	client_id
173477383888	RUB	74210	2020-12-07 12:02:17	3
265533352627	USD	1500	2020-12-07 12:02:17	3
273849984837	RUB	0	2022-12-20 13:27:15	2
362663773362	EUR	0	2020-03-10 10:02:51	1
372727772839	RUB	170000	2020-09-13 14:07:31	4
777362199000	RUB	0	2022-11-13 14:53:11	1
848726364789	RUB	73000	2022-12-30 18:12:16	5

6. Показать все сведения о клиенте, чаще всех совершавшего переводы в 2020 году.

```

CREATE VIEW transactions_count AS
SELECT
    c.*,
    COUNT(transaction_id) AS tr_cnt

FROM client_t AS c
JOIN account_t AS a
    USING(client_id)
JOIN transaction_t AS t
    ON a.account_id = t.account_id_from

WHERE YEAR(transaction_dt) = 2020
GROUP BY client_id;

```

```

SELECT
    client_id,
    surname,
    phone_number,
    birth_date,
    address,
    start_of_deal,
    end_of_deal
FROM transactions_count
WHERE tr_cnt = (SELECT MAX(tr_cnt)
                FROM transactions_count)

```

Результат выполнения запроса

client_id	surname	phone_number	birth_date	address	start_of_d...	end_of_deal
1	Астафьев	+79856743254	1978-03-23	г. Москва, Дубининская улица, 63с2, кв 52	2009-04-21	2023-01-23
3	Иванова	+79096547888	2001-06-13	г. Москва, 1-й Колобовский переулок, 15/6, к...	2020-03-07	NULL



## Процедура

Разработать процедуру, которая будет принимать на вход номер перевода. В процедуре реализовать изменение остатков на соответствующих счетах.

### План процедуры

1. Определение входного параметра процедуры – номера перевода  
t\_num INT
2. Объявление обработчика ошибок  
DECLARE done INT DEFAULT 0;
3. Извлечение данных из полей amount\_of\_money, transaction\_dt, account\_id\_from, account\_id\_to таблицы transaction\_t и соответствующего им курса валют rate из таблицы exchange\_rate

```
SELECT
    amount_of_money,
    transaction_dt,
    account_id_from,
    account_id_to,
    rate
INTO
    a_money,
    t_dt,
    a_from,
    a_to,
    r
FROM transaction_t
JOIN exchange_rate
    USING(exchange_rate_id)
WHERE transaction_id = t_num;
```

4. Транзакция, обновляющая поля таблицы account\_t  
START TRANSACTION;

- 1) Обновление данных для счета, с которого переводятся деньги

```
UPDATE account_t
SET
    balance = balance - a_money,
    balance_dt = t_dt
WHERE account_id = a_from;
```

- 2) Обновление данных для счета, на который переводятся деньги

```
UPDATE account_t
SET
    balance = balance + a_money * r,
    balance_dt = t_dt
WHERE account_id = a_to;
```

## Текст оператора создания процедуры

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `account_t Updating_balance`(t_num INT)
BEGIN
    /* Переменные, куда извлекаются данные */
    DECLARE a_money FLOAT;
    DECLARE t_dt DATETIME;
    DECLARE a_from CHAR(12);
    DECLARE a_to CHAR(12);
    DECLARE r DOUBLE DEFAULT 1;

    /* Флаг для обработчика ошибок */
    DECLARE done INT DEFAULT 0;

    /* Обработчик ошибок */
    DECLARE CONTINUE HANDLER FOR SQLSTATE '02000' SET done = 1;

    /* Извлечение данных в соответствующие переменные */
    SELECT
        amount_of_money,
        transaction_dt,
        account_id_from,
        account_id_to,
        rate
    INTO
        a_money,
        t_dt,
        a_from,
        a_to,
        r
    FROM transaction_t JOIN exchange_rate USING(exchange_rate_id)
    WHERE transaction_id = t_num;

    /* Транзакция для обновления данных в таблице account_t */
    START TRANSACTION;
    /* Обновление данных для счета, с которого переводятся деньги */
    UPDATE account_t
    SET
        balance = balance - a_money,
        balance_dt = t_dt
    WHERE account_id = a_from;
    /* Обновление данных для счета, на который переводятся деньги */
    UPDATE account_t
    SET
        balance = balance + a_money * r,
        balance_dt = t_dt
    WHERE account_id = a_to;
    /* Если транзакция выполнена без ошибок, то сохраняем текущее состояние БД */
    IF done = 0 THEN COMMIT;
    /* Если при выполнении транзакции произошла ошибка, то откатываем БД до ее
    состояния перед началом выполнения транзакции */
    ELSE ROLLBACK;
    END IF;
END
```

Для демонстрации работы процедуры добавим в таблицу transaction\_t новую строку

transaction_id	amount_of_mon...	transaction_dt	account_id_from	account_id_to	exchange_rate_id
1	150000	2010-06-13 12:01:13	777362199000	874326554335	2
2	300	2020-03-01 11:00:43	874326554335	362663773362	13
3	1000	2020-12-07 12:02:17	265533352627	173477383888	9
4	100	2023-01-03 16:15:44	265533352627	173477383888	9

Вызовем процедуру, передав в нее номер недавно добавленной транзакции

**CALL** account\_t\_updating\_balance(4)

После вызова процедуры получаем изменение остатков на соответствующих счетах

account_id	currency	balance	balance_dt	client_id
173477383888	RUB	81631	2023-01-03 16:15:44	3
265533352627	USD	1400	2023-01-03 16:15:44	3
273849984837	RUB	0	2022-12-20 13:27:15	2
362663773362	EUR	0	2020-03-10 10:02:51	1
372727772839	RUB	170000	2020-09-13 14:07:31	4
777362199000	RUB	0	2022-11-13 14:53:11	1
848726364789	RUB	73000	2022-12-30 18:12:16	5
874326554335	USD	0	2020-03-10 09:53:22	1

# Триггер

Разработать триггер, который будет срабатывать при каждом добавлении записи о переводе. В триггере реализовать изменение остатков на соответствующих счетах.

## Текст оператора создания триггера

```
CREATE DEFINER = CURRENT_USER TRIGGER `transaction_t_AFTER_INSERT`
AFTER INSERT ON `transaction_t`
FOR EACH ROW
BEGIN
    /* Переменная, куда извлекаются данные */
    DECLARE r DOUBLE DEFAULT 1;

    /* Флаг для обработчика ошибок */
    DECLARE done INT DEFAULT 0;

    /* Обработчик ошибок */
    DECLARE CONTINUE HANDLER FOR SQLSTATE '02000'
    SET done = 1;

    /* Извлечение соответствующего переводу rate в переменную r */
    SELECT rate
    INTO r
    FROM transaction_t
    JOIN exchange_rate
        USING(exchange_rate_id)
    WHERE transaction_id = new.transaction_id

    IF done = 0 THEN

        /* Обновление данных для счета, с которого переводятся деньги */
        UPDATE account_t
        SET
            balance = balance - new.amount_of_money,
            balance_dt = new.transaction_dt
        WHERE account_id = new.account_id_from;

        /* Обновление данных для счета, на который переводятся деньги */
        UPDATE account_t
        SET
            balance = balance + new.amount_of_money * r,
            balance_dt = new.transaction_dt
        WHERE account_id = new.account_id_to;

    END IF;
END
```

Для демонстрации работы триггера добавим в таблицу transaction\_t новую строку

transaction_id	amount_of_mon...	transaction_dt	account_id_from	account_id_to	exchange_rate_id
1	150000	2010-06-13 12:01:13	777362199000	874326554335	2
2	300	2020-03-01 11:00:43	874326554335	362663773362	13
3	1000	2020-12-07 12:02:17	265533352627	173477383888	9
4	200	2023-01-13 00:00:00	265533352627	173477383888	9

Заметим, что обновление остатков в таблице account\_t произошло автоматически

account_id	currency	balance	balance_dt	client_id
173477383888	RUB	89052	2023-01-13 00:00:00	3
265533352627	USD	1300	2023-01-13 00:00:00	3
273849984837	RUB	0	2022-12-20 13:27:15	2
362663773362	EUR	0	2020-03-10 10:02:51	1
372727772839	RUB	170000	2020-09-13 14:07:31	4
777362199000	RUB	0	2022-11-13 14:53:11	1
848726364789	RUB	73000	2022-12-30 18:12:16	5
874326554335	USD	0	2020-03-10 09:53:22	1