

¹**Shamuratov Ulug‘bek Alisher uli**

*Toshkent Axborot Texnologiyalar Universiteti magistrant,
ulugbekshamuratov1219@gmail.com*

²**Ochilboyev Umidjon Ilxom o‘g‘li**

Toshkent Axborot Texnologiyalar Universiteti magistrant, canc41946@gmail.com

Do‘shanov Bekzod Davronbek o‘g‘li

*Toshkent Axborot Texnologiyalar Universiteti magistrant,
duschanovbekzod17@gmail.com*

Annotatsiya. Ushbu maqolada ma‘lumotlar bazasida tranzaksiyalarni boshqarish, SQL so‘rovlari orqali ularni yaratish va qayta ishlash jarayonlari yoritiladi. Amaliy qismda bank hisoblari o‘rtasida pul o‘tkazish misoli asosida tranzaksiyalarning ma‘lumotlar yaxlitligi, ishonchliligi va xavfsizligini ta‘minlashdagi ahamiyati ko‘rsatib beriladi.

Kalit so‘zlar: ma‘lumotlar bazasi, tranzaksiya, SQL so‘rovlari, COMMIT, ROLLBACK, ma‘lumotlar yaxlitligi.

Аннотация. В данной статье рассматриваются процессы управления транзакциями в базе данных, а также создание и обработка SQL-запросов. В практической части на примере перевода денежных средств между банковскими счетами показано значение транзакций в обеспечении целостности, надежности и безопасности данных.

Ключевые слова: база данных, транзакция, SQL-запросы, COMMIT, ROLLBACK, целостность данных.

Annotation. This article discusses transaction management in databases, as well as the creation and processing of SQL queries. The practical part demonstrates the importance of transactions in ensuring data integrity, reliability, and security through the example of transferring money between bank accounts.

Keywords: database, transaction, SQL queries, COMMIT, ROLLBACK, data integrity.

KIRISH

Bugungi kunda axborot tizimlari, bank dasturlari, elektron savdo platformalari, ta‘lim tizimlari va boshqa ko‘plab dasturiy mahsulotlar ma‘lumotlar bazasi asosida ishlaydi. Ma‘lumotlar bazasida saqlanayotgan axborotlarning to‘g‘ri, ishonchli va izchil bo‘lishi tizim samaradorligini ta‘minlovchi asosiy omillardan biri hisoblanadi. Ayniqsa, bir nechta foydalanuvchi bir vaqtning o‘zida ma‘lumotlar bilan ishlaydigan tizimlarda ma‘lumotlarning buzilishi, noto‘g‘ri o‘zgarishi yoki tugallanmagan holatda saqlanib qolishi jiddiy muammolarga olib kelishi mumkin. Ma‘lumotlar bazasida bunday muammolarning

oldini olish uchun tranzaksiyalar mexanizmidan foydalaniladi. Tranzaksiya bir yoki bir nechta SQL so‘rovlaridan tashkil topgan mantiqiy jarayon bo‘lib, u barcha amallarni yagona birlik sifatida bajarilishini ta‘minlaydi. Agar tranzaksiya tarkibidagi barcha amallar muvaffaqiyatli bajarilsa, natija ma‘lumotlar bazasida saqlanadi. Agar jarayon davomida xatolik yuz bersa, bajarilgan o‘zgarishlar bekor qilinadi va ma‘lumotlar avvalgi holatiga qaytariladi.

Tranzaksiyalar ayniqsa moliyaviy amallarni bajarishda muhim ahamiyatga ega. Masalan, bank hisoblari o‘rtasida pul o‘tkazish jarayonida bir hisobdan mablag‘ yechilib, ikkinchi hisobga qo‘shilishi kerak. Agar bu jarayonning bir qismi bajarilib, ikkinchi qismi bajarilmay qolsa, ma‘lumotlar bazasida nomuvofiqlik yuzaga keladi. Shu sababli tranzaksiyalar ma‘lumotlar yaxlitligini, ishonchliligini va xavfsizligini ta‘minlashda asosiy vositalardan biri hisoblanadi. Ushbu maqolada tranzaksiyalarni boshqarishda so‘rovlar yaratish va qayta ishlash jarayoni nazariy hamda amaliy jihatdan yoritiladi. Amaliy qismda bank hisoblari o‘rtasida pul o‘tkazish misoli asosida `START TRANSACTION`, `COMMIT` va `ROLLBACK` buyruqlarining qo‘llanilishi ko‘rib chiqiladi. Natijada tranzaksiyalar yordamida ma‘lumotlar bazasida xatoliklarning oldini olish va ma‘lumotlar izchilligini saqlash imkoniyatlari tahlil qilinadi.

NAZARIY QISM

Ma‘lumotlar bazasida tranzaksiya tushunchasi ma‘lumotlar ustida bajariladigan bir nechta amallarni yagona mantiqiy jarayon sifatida boshqarishni anglatadi. Tranzaksiya odatda INSERT, UPDATE, DELETE yoki SELECT kabi SQL so‘rovlaridan tashkil topadi va ularning barchasi birgalikda bajariladi. Agar tranzaksiya tarkibidagi barcha amallar muvaffaqiyatli yakunlansa, o‘zgarishlar ma‘lumotlar bazasida saqlanadi. Aksincha, jarayon davomida xatolik yuzaga kelsa, barcha o‘zgarishlar bekor qilinadi va ma‘lumotlar avvalgi holatiga qaytariladi. Shu jihatdan tranzaksiyalar ma‘lumotlar bazasida ishonchlilik, xavfsizlik va yaxlitlikni ta‘minlovchi asosiy mexanizmlardan biri hisoblanadi.

Tranzaksiyalar ayniqsa ko‘p foydalanuvchili tizimlarda muhim ahamiyatga ega. Chunki bitta tizimdan bir vaqtning o‘zida bir nechta foydalanuvchi foydalanishi mumkin. Masalan, bank tizimida bir foydalanuvchi pul o‘tkazayotgan vaqtda boshqa foydalanuvchi ham o‘z hisobidan operatsiya bajarishi mumkin. Bunday holatda ma‘lumotlar bir-biriga zid bo‘lib qolmasligi, noto‘g‘ri yangilanmasligi yoki qisman saqlanib qolmasligi kerak. Tranzaksiyalar aynan shu muammoni hal qiladi, ya‘ni bir nechta so‘rovlarning to‘liq bajarilishini yoki umuman bajarilmasligini nazorat qiladi. Ma‘lumotlar bazasida tranzaksiyalarni boshqarishning asosiy maqsadi ma‘lumotlar yaxlitligini saqlashdan iborat. Ma‘lumotlar yaxlitligi deganda bazada saqlanayotgan axborotlarning to‘g‘ri, izchil va mantiqiy bog‘langan holatda bo‘lishi tushuniladi. Masalan, pul o‘tkazish jarayonida yuboruvchining hisobidan mablag‘ yechilib, qabul qiluvchining hisobiga qo‘shilishi kerak. Agar faqat birinchi amal bajarilib, ikkinchi amal bajarilmasa, ma‘lumotlar bazasida xatolik yuzaga keladi. Tranzaksiya bunday holatlarning oldini olib, barcha amallarni bitta jarayon sifatida boshqaradi.

Tranzaksiyalar ACID tamoyillari asosida ishlaydi. ACID — bu tranzaksiyaning ishonchli bajarilishini ta‘minlovchi to‘rtta muhim xususiyatlar majmuasidir. Ular atomarlik,

izchillik, izolyatsiya va doimiylik tamoyillaridan iborat. Ushbu tamoyillar ma’lumotlar bazasida har qanday murakkab jarayonning xavfsiz va tartibli bajarilishini ta’minlaydi.

ACID xususiyati	Mazmuni	Oddiy misol
Atomicity	Tranzaksiya tarkibidagi barcha amallar to’liq bajariladi yoki umuman bajarilmaydi	Pul bir hisobdan yechilib, ikkinchi hisobga qo’shilishi kerak
Consistency	Tranzaksiya bajarilgandan keyin ma’lumotlar bazasi to’g’ri holatda qoladi	Hisob balansi manfiy bo’lib qolmasligi kerak
Isolation	Bir vaqtning o’zida bajarilayotgan tranzaksiyalar bir-biriga xalaqit bermaydi	Ikki foydalanuvchi bir hisobga bir vaqtda noto’g’ri ta’sir qilmaydi
Durability	Tranzaksiya tasdiqlangandan keyin natija bazada saqlanib qoladi	Pul o’tkazish muvaffaqiyatli yakunlansa, natija yo’qolmaydi

Tranzaksiyaning atomarlik xususiyati amallarning bo’linmasligini bildiradi. Ya’ni tranzaksiya ichidagi bir nechta SQL so’rovlari alohida-alohida emas, balki yagona jarayon sifatida qaraladi. Masalan, bank hisoblari o’rtasida pul o’tkazishda birinchi hisobdan pul yechish va ikkinchi hisobga pul qo’shish amallari bir-biriga bog’liq. Agar ulardan biri bajarilmasa, butun tranzaksiya bekor qilinishi kerak. Bu esa ma’lumotlarning noto’g’ri holatda saqlanib qolishining oldini oladi. Izchillik xususiyati esa tranzaksiya bajarilgandan keyin ma’lumotlar bazasi belgilangan qoidalar va cheklovlarga mos holatda qolishini ta’minlaydi. Masalan, bank tizimida hisobdagi mablag’ manfiy bo’lib qolmasligi kerak. Agar foydalanuvchi hisobida 500 000 so’m bo’lsa, u 1 000 000 so’m o’tkazishga urinishi mumkin emas. Bunday holatda tizim tranzaksiyani bajarishdan oldin shartlarni tekshirishi yoki xatolik yuz bersa, bajarilgan o’zgarishlarni bekor qilishi lozim.

Izolyatsiya xususiyati ko’p foydalanuvchili tizimlarda juda muhimdir. Bir vaqtning o’zida bir nechta tranzaksiya bajarilganda, ular bir-birining natijasiga salbiy ta’sir qilmasligi kerak. Masalan, ikkita foydalanuvchi bir vaqtning o’zida bitta mahsulotni xarid qilmoqchi bo’lsa, tizim mahsulot miqdorini noto’g’ri kamaytirib yubormasligi kerak. Izolyatsiya darajalari orqali tranzaksiyalarning bir-biriga qanday ta’sir qilishi nazorat qilinadi. Doimiylik xususiyati tranzaksiya muvaffaqiyatli yakunlangandan keyin uning natijasi ma’lumotlar bazasida saqlanib qolishini anglatadi. Ya’ni COMMIT buyrug’i bajarilgandan so’ng tranzaksiya natijasi tasdiqlanadi va tizimda nosozlik yuz bersa ham, ushbu o’zgarishlar yo’qolmasligi kerak. Bu xususiyat bank, elektron to’lov va buyurtma tizimlari uchun juda muhim hisoblanadi.

Tranzaksiyalarni boshqarishda SQL tilidagi maxsus buyruqlardan foydalaniladi. Bu buyruqlar tranzaksiyani boshlash, tasdiqlash, bekor qilish yoki oraliq nuqtaga qaytarish imkonini beradi. Eng ko’p ishlatiladigan buyruqlar START TRANSACTION, COMMIT, ROLLBACK va SAVEPOINT hisoblanadi.

SQL buyrug‘i	Vazifasi
START TRANSACTION	Tranzaksiyani boshlaydi
COMMIT	Tranzaksiya natijalarini ma’lumotlar bazasiga saqlaydi
ROLLBACK	Tranzaksiyani bekor qiladi va ma’lumotlarni avvalgi holatga qaytaradi
SAVEPOINT	Tranzaksiya ichida oraliq saqlash nuqtasini yaratadi
ROLLBACK TO SAVEPOINT	Tranzaksiyani belgilangan oraliq nuqtagacha qaytaradi

Tranzaksiyani boshlash uchun START TRANSACTION buyrug‘i ishlatiladi. Ushbu buyruqdan keyin yozilgan SQL so‘rovlari tranzaksiya tarkibiga kiradi. Tranzaksiya yakunida barcha amallar to‘g‘ri bajarilgan bo‘lsa, COMMIT buyrug‘i orqali natijalar bazaga saqlanadi. Agar xatolik yuz bersa yoki amallarni bekor qilish zarur bo‘lsa, ROLLBACK buyrug‘i ishlatiladi. SAVEPOINT esa katta tranzaksiyalar ichida ma’lum bir oraliq nuqtani saqlab qo‘yish imkonini beradi. Tranzaksiyalarni boshqarishda so‘rovlarni to‘g‘ri yaratish ham katta ahamiyatga ega. Chunki noto‘g‘ri yozilgan SQL so‘rovi nafaqat xatolik keltirib chiqaradi, balki ma’lumotlar bazasining izchilligiga ham zarar yetkazishi mumkin. Masalan, UPDATE so‘rovida WHERE sharti yozilmasa, jadvaldagi barcha yozuvlar o‘zgarib ketishi mumkin. Shu sababli tranzaksiya ichida bajariladigan har bir so‘rov aniq, tekshirilgan va maqsadga muvofiq bo‘lishi kerak.

So‘rovlarni qayta ishlash jarayoni bir necha bosqichdan iborat bo‘ladi. Dastlab foydalanuvchi yoki dastur tomonidan SQL so‘rovi yuboriladi. Keyin ma’lumotlar bazasini boshqarish tizimi ushbu so‘rovni tahlil qiladi, uning sintaktik va mantiqiy jihatdan to‘g‘riligini tekshiradi. Shundan so‘ng so‘rov bajariladi va natija hosil qilinadi. Agar so‘rov tranzaksiya ichida bajarilayotgan bo‘lsa, uning natijasi darhol doimiy saqlanmasligi mumkin. Natija faqat COMMIT buyrug‘i bajarilgandan keyingina bazaga yakuniy tarzda yoziladi. Tranzaksiyalar bank tizimlari, elektron savdo platformalari, ombor boshqaruvi, ta’lim tizimlari va boshqa ko‘plab sohalarda qo‘llaniladi. Bank tizimida tranzaksiya pul o‘tkazmalarining xavfsizligini ta’minlaydi. Elektron savdo tizimida buyurtma berish, to‘lovni amalga oshirish va mahsulot miqdorini kamaytirish amallari tranzaksiya orqali boshqariladi. Ombor tizimida mahsulot kirim-chiqimi noto‘g‘ri hisoblanib qolmasligi uchun tranzaksiyalar ishlatiladi. Ta’lim tizimida esa talabalar baholari, to‘lovlari yoki ro‘yxatdan o‘tish jarayonlari tranzaksiyalar yordamida ishonchli boshqarilishi mumkin.

Qo‘llanilish sohasi	Tranzaksiya vazifasi
Bank tizimi	Pul o‘tkazmalarini xavfsiz bajarish
Elektron savdo	Buyurtma, to‘lov va mahsulot sonini birgalikda boshqarish
Ombor tizimi	Mahsulot kirim-chiqimini to‘g‘ri hisoblash
Ta’lim tizimi	Talaba baholari va to‘lovlarni izchil saqlash
Mehmonxona yoki	Bron qilish jarayonida joylar bandligini nazorat qilish

aviachipta tizimi	
-------------------	--

Tranzaksiyalar noto‘g‘ri boshqarilsa, tizimda turli muammolar paydo bo‘lishi mumkin. Masalan, juda uzun davom etadigan tranzaksiyalar boshqa foydalanuvchilarning so‘rovlarini kutishga majbur qiladi. Bu esa tizim sekinlashishiga olib keladi. Shuningdek, bir nechta tranzaksiya bir-birini kutib qolsa, bloklanish holati yuzaga kelishi mumkin. Shu sababli tranzaksiyalarni imkon qadar qisqa, aniq va maqsadli tuzish kerak. Har bir tranzaksiya faqat zarur bo‘lgan amallarni o‘z ichiga olishi, ortiqcha so‘rovlar bilan murakkablashtirilmasligi lozim.

Umuman olganda, tranzaksiyalar ma‘lumotlar bazasida so‘rovlarni xavfsiz va ishonchli qayta ishlash uchun zarur mexanizm hisoblanadi. Ular ma‘lumotlar yaxlitligini ta‘minlaydi, xatolik yuz berganda o‘zgarishlarni bekor qilish imkonini beradi va ko‘p foydalanuvchili tizimlarda tartibli ishlashni ta‘minlaydi. Shu sababli tranzaksiyalarni boshqarish bo‘yicha bilimlar ma‘lumotlar bazasi bilan ishlovchi dasturchilar, administratorlar va axborot tizimlari mutaxassislari uchun muhim nazariy va amaliy ahamiyatga ega.

Amaliy qism

Ushbu amaliy qismda bank hisoblari o‘rtasida pul o‘tkazish jarayoni misolida tranzaksiyalarni boshqarish ko‘rib chiqiladi. Amaliy ishda accounts jadvali orqali mijozlarning hisob raqamlari va balanslari saqlanadi, transactions jadvali orqali esa amalga oshirilgan pul o‘tkazmalari tarixi qayd etiladi. Jarayon davomida pul yuboruvchi hisobidan mablag‘ yechiladi, qabul qiluvchi hisobiga mablag‘ qo‘shiladi va tranzaksiya natijasi alohida jadvalga yoziladi.

Avval bank mijozlarining hisob ma‘lumotlarini saqlash uchun accounts jadvali yaratiladi.

```
CREATE TABLE accounts (  
  account_id INT PRIMARY KEY,  
  full_name VARCHAR(100),  
  balance DECIMAL(12,2)  
);
```

Pul o‘tkazmalari tarixini saqlash uchun esa transactions jadvali yaratiladi.

```
CREATE TABLE transactions (  
  transaction_id INT PRIMARY KEY,  
  sender_id INT,  
  receiver_id INT,  
  amount DECIMAL(12,2),  
  transaction_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  status VARCHAR(30)  
);
```

Boshlang‘ich ma‘lumot sifatida ikki nafar mijozning hisob raqami va balanslari kiritiladi.

```
INSERT INTO accounts (account_id, full_name, balance)  
VALUES
```

```
(1, 'Ali Karimov', 1500000.00),
(2, 'Vali Rahimov', 500000.00);
```

Boshlang‘ich holatni ko‘rish uchun quyidagi so‘rov bajariladi.

```
SELECT * FROM accounts;
```

Natija:

account_id	full_name	balance
1	Ali Karimov	1500000.00
2	Vali Rahimov	500000.00

Ushbu holatda Ali Karimovning hisobida 1 500 000 so‘m, Vali Rahimovning hisobida esa 500 000 so‘m mavjud. Endi Ali Karimov hisobidan Vali Rahimov hisobiga 300 000 so‘m pul o‘tkazish tranzaksiya orqali bajariladi.

```
START TRANSACTION;
UPDATE accounts
SET balance = balance - 300000
WHERE account_id = 1;
UPDATE accounts
SET balance = balance + 300000
WHERE account_id = 2;
INSERT INTO transactions
(transaction_id, sender_id, receiver_id, amount, status)
VALUES
(1, 1, 2, 300000, 'Muvaffaqiyatli');
COMMIT;
```

Ushbu kodda tranzaksiya START TRANSACTION buyrug‘i orqali boshlanadi. Birinchi UPDATE so‘rovi Ali Karimov hisobidan 300 000 so‘mni kamaytiradi. Ikkinchi UPDATE so‘rovi esa Vali Rahimov hisobiga 300 000 so‘m qo‘shadi. Shundan so‘ng transactions jadvaliga pul o‘tkazmasi haqida ma‘lumot yoziladi. Barcha amallar to‘g‘ri bajarilgani uchun COMMIT buyrug‘i orqali o‘zgarishlar ma‘lumotlar bazasida saqlanadi.

Tranzaksiyadan keyingi hisob holatini tekshirish uchun quyidagi so‘rov bajariladi.

```
SELECT * FROM accounts;
```

Natija:

account_id	full_name	balance
1	Ali Karimov	1200000.00
2	Vali Rahimov	800000.00

Natijadan ko‘rinib turibdiki, Ali Karimov hisobidan 300 000 so‘m yechildi va Vali Rahimov hisobiga 300 000 so‘m qo‘shildi. Demak, tranzaksiya muvaffaqiyatli bajarildi.

Endi pul o‘tkazmasi haqidagi yozuv transactions jadvalida saqlanganini tekshiramiz.

```
SELECT * FROM transactions;
```

Natija:

transaction_id	sender_id	receiver_id	amount	transaction_date	status
1	1	2	300000.00	2026-05-10 12:30:00	Muvaffaqiyatli

15-May, 2026-yil

Bu natija pul o'tkazish jarayoni nafaqat hisob balanslariga ta'sir qilganini, balki tranzaksiya tarixi sifatida ham bazaga yozilganini ko'rsatadi.

Keyingi holatda yuboruvchining hisobida yetarli mablag' bo'lmagan vaziyat ko'rib chiqiladi. Ali Karimov hisobida 1 200 000 so'm mavjud bo'lsa-da, u 2 000 000 so'm o'tkazishga urinadi. Bunday holatda tranzaksiya bajarilmasligi va ma'lumotlar avvalgi holatda qolishi kerak.

```
START TRANSACTION;  
UPDATE accounts  
SET balance = balance - 2000000  
WHERE account_id = 1 AND balance >= 2000000;  
UPDATE accounts  
SET balance = balance + 2000000  
WHERE account_id = 2;  
ROLLBACK;
```

Ushbu kodda birinchi UPDATE so'rovida balance >= 2000000 sharti berilgan. Bu shart yuboruvchining hisobida yetarli mablag' bor-yo'qligini tekshiradi. Ali Karimov hisobida 2 000 000 so'm mablag' mavjud bo'lmagani uchun birinchi amal bajarilmaydi. Shu sababli tranzaksiya ROLLBACK buyrug'i orqali bekor qilinadi.

XULOSA

Ushbu maqolada ma'lumotlar bazasida tranzaksiyalarni boshqarishda so'rovlar yaratish va qayta ishlash jarayoni nazariy hamda amaliy jihatdan ko'rib chiqildi. Nazariy qismda tranzaksiya tushunchasi, uning ma'lumotlar yaxlitligi va ishonchliligini ta'minlashdagi o'rni, shuningdek, COMMIT, ROLLBACK, START TRANSACTION kabi buyruqlarning vazifalari yoritildi. Amaliy qismda bank hisoblari o'rtasida pul o'tkazish jarayoni misol sifatida tanlandi. Bunda yuboruvchi hisobidan mablag' yechish, qabul qiluvchi hisobiga mablag' qo'shish va tranzaksiya tarixini saqlash kabi amallar bajarildi. Tranzaksiya muvaffaqiyatli yakunlanganda COMMIT buyrug'i yordamida o'zgarishlar ma'lumotlar bazasiga saqlandi. Mablag' yetarli bo'lmagan holatda esa ROLLBACK buyrug'i orqali noto'g'ri o'zgarishlarning oldi olindi.

Olib borilgan amaliy ish shuni ko'rsatdiki, tranzaksiyalar ma'lumotlar bazasida bir nechta bog'liq amallarni yagona jarayon sifatida boshqarishga xizmat qiladi. Bu esa ma'lumotlarning noto'g'ri, qisman yoki izchilliksiz saqlanib qolishining oldini oladi. Ayniqsa, bank, elektron savdo, ombor va to'lov tizimlarida tranzaksiyalarni to'g'ri qo'llash ma'lumotlar xavfsizligi va tizim barqarorligini ta'minlashda muhim ahamiyatga ega. Shunday qilib, tranzaksiyalarni boshqarish bo'yicha bilim va ko'nikmalar ma'lumotlar bazasi bilan ishlovchi mutaxassislar uchun zarur hisoblanadi. SQL so'rovlarini to'g'ri yaratish, tranzaksiya natijalarini nazorat qilish va xatolik yuz berganda o'zgarishlarni bekor qilish mexanizmlaridan foydalanish axborot tizimlarining ishonchli ishlashini ta'minlaydi.

FOYDALANILGAN ADABIYOTLAR:

1. Silberschatz A., Korth H. F., Sudarshan S. Database System Concepts. 7th edition. New York: McGraw-Hill Education, 2019.
2. Elmasri R., Navathe S. B. Fundamentals of Database Systems. 7th edition. Pearson Education, 2016.
3. Connolly T., Begg C. Database Systems: A Practical Approach to Design, Implementation, and Management. 6th edition. Pearson Education, 2015.
4. Date C. J. An Introduction to Database Systems. 8th edition. Addison-Wesley, 2004.
5. PostgreSQL Global Development Group. PostgreSQL Documentation: Transactions. Rasmiy hujjatlarda tranzaksiyalar COMMIT, ROLLBACK va SAVEPOINT kabi mexanizmlar orqali boshqarilishi ko‘rsatilgan. ([PostgreSQL](#))
6. PostgreSQL Global Development Group. PostgreSQL Documentation: Transaction Isolation. Ushbu manbada tranzaksiyalarning izolyatsiya darajalari va bir vaqtda bajariladigan amallarni boshqarish masalalari yoritilgan. ([PostgreSQL](#))
7. Oracle Corporation. MySQL 8.4 Reference Manual: START TRANSACTION, COMMIT, and ROLLBACK Statements. MySQL hujjatlarida START TRANSACTION tranzaksiyani boshlash uchun standart SQL sintaksisi ekani ko‘rsatilgan. ([dev.mysql.com](#))
8. Oracle Corporation. MySQL Reference Manual: Transactional and Locking Statements. MySQL tranzaksiyalarni START TRANSACTION, COMMIT va ROLLBACK operatorlari orqali qo‘llab-quvvatlaydi. ([dev.mysql.com](#))
9. Garcia-Molina H., Ullman J. D., Widom J. Database Systems: The Complete Book. 2nd edition. Pearson, 2008.
10. Ramakrishnan R., Gehrke J. Database Management Systems. 3rd edition. McGraw-Hill, 2003.