MA’LUMOTLAR BAZALARIDA INDEKSLASH MEXANIZMLARI VA ULARNING ISHLASH SAMARADORLIGIGA TA’SIRI

Sevinch Jumabaeva
Farg‘ona davlat texnika universiteti
Email: sevinchjumabaeva25@gmail.com

Dilshodov Abrorjon Dilshodjon o‘g‘li
Farg‘ona davlat texnika universiteti
Axborot tizimlari va texnologiyalari
kafedrasi katta o‘qituvchisi

Annotatsiya. Maqolada ma’lumotlar bazalarida indekslardan foydalanishning nazariy va amaliy asoslari, ularning SQL so‘rovlari bajarilish tezligiga ta’siri hamda axborot tizimlarining umumiy ishlash samaradorligini oshirishdagi o‘rni tahlil qilinadi. Zamonaviy dasturiy tizimlarda ma’lumotlar hajmining ortishi qidiruv, saralash, filtrlash va jadvallarni bog‘lash jarayonlarini optimallashtirishni talab qiladi. Shu bois indekslar relatsion ma’lumotlar bazalarida muhim fizik optimallashtirish mexanizmi sifatida qaraladi. Tadqiqot natijasida indekslash strategiyasi real so‘rovlar, jadval hajmi, ma’lumotlar taqsimoti va tizim yuklamasi asosida ishlab chiqilgandagina yuqori samaradorlik berishi asoslandi.

Kalit so‘zlar: indeks, ma’lumotlar bazasi, SQL, B-tree indeks, hash indeks, kompozit indeks, query execution plan, ishlash samaradorligi.

Kirish

Axborot tizimlarining rivojlanishi bilan ma’lumotlarni saqlash va qayta ishlash jarayonlariga qo‘yiladigan talablar ortib bormoqda. Bank, ta’lim, logistika, elektron savdo, tibbiyot va davlat boshqaruvi kabi sohalarda axborot tizimlari katta hajmdagi ma’lumotlar bilan ishlaydi. Bunday sharoitda foydalanuvchi yuborgan so‘rovga qisqa vaqt ichida javob qaytarish tizim sifatining asosiy ko‘rsatkichlaridan biri hisoblanadi.

Ma’lumotlar bazasida so‘rovlar tezligi jadval tuzilishi, yozuvlar soni, so‘rov shakli, server resurslari va indekslardan foydalanish darajasiga bog‘liq. Indekslar jadvaldagi yozuvlarga tezkor murojaat qilish imkonini beruvchi yordamchi ma’lumot

tuzilmasidir. Silberschatz, Korth va Sudarshan indekslarni ma’lumotlarga tezkor kirish yo‘lini ta’minlovchi fizik tuzilma sifatida izohlaydi [1]. Elmasri va Navathe esa indekslashni relatsion ma’lumotlar bazalarida fizik loyihalash va optimallashtirishning muhim tarkibiy qismi sifatida ko‘rsatadi [2].

Maqolaning dolzarbligi shundaki, amaliy dasturiy tizimlarda sekin ishlash muammosi ko‘pincha faqat dastur kodiga emas, balki ma’lumotlar bazasi tuzilmasi va indekslash strategiyasiga ham bog‘liq bo‘ladi. Indekslar to‘g‘ri tanlansa, qidiruv va jadval bog‘lanishlari tezlashadi; noto‘g‘ri tanlansa, xotira sarfi ortadi va yozish operatsiyalari sekinlashadi.

Tadqiqot maqsadi va metodologiyasi

Ushbu maqolaning maqsadi ma’lumotlar bazalarida indekslarning ishlash mexanizmini, SQL so‘rovlari bajarilishiga ta’sirini va umumiy tizim samaradorligini oshirishdagi o‘rnini ilmiy-amaliy jihatdan yoritishdan iborat. Tadqiqotda indeks tushunchasi, indeks turlari, ularning afzalliklari, cheklovlari va samarali indekslash qoidalari tahlil qilindi.

Maqolada nazariy tahlil, qiyosiy yondashuv va amaliy misol asosida izohlash metodlaridan foydalanildi. Nazariy asos sifatida ma’lumotlar bazasi tizimlari bo‘yicha Silberschatz, Korth, Sudarshan [1], Elmasri va Navathe [2], Garcia-Molina, Ullman va Widom [3], Ramakrishnan va Gehrke [4] asarlari tahlil qilindi. Amaliy samaradorlik masalalarida Kleppmann [7], Schwartz, Zaitsev va Tkachenko [8], Winand [9] hamda Özsu va Valduriez [10] yondashuvlari asos qilib olindi.

Tadqiqotda indekslar faqat nazariy tushuncha sifatida emas, balki real SQL so‘rovlari, query execution plan, jadval hajmi, o‘qish va yozish operatsiyalari nisbati bilan bog‘liq holda ko‘rib chiqildi. Bu yondashuv xalqaro ilmiy maqolalarda talab qilinadigan mantiqiy izchillik, dalillilik va amaliy yo‘naltirilganlik mezonlariga mos keladi.

Indeks tushunchasi va uning vazifalari

Indeks — bu jadvaldagi ma’lumotlarga tezkor murojaat qilish uchun yaratiladigan maxsus yordamchi tuzilma. Uni kitob oxiridagi ko‘rsatkichga qiyoslash mumkin: kerakli mavzuni topish uchun barcha sahifalarni ketma-ket o‘qish shart emas,

ko‘rsatkich orqali mavzu joylashgan sahifa tez aniqlanadi. Ma’lumotlar bazasida ham indeks jadvaldagi barcha satrlarni to‘liq tekshirish zaruratini kamaytiradi.

Indekslar jadvaldagi bir yoki bir nechta ustunlar asosida yaratiladi. Masalan, talabalar jadvalida identifikator, guruh nomi yoki qabul yili bo‘yicha qidiruv tez-tez bajarilsa, ushbu ustunlarga indeks yaratish mumkin. Indeks jadvaldagi ma’lumot mazmunini o‘zgartirmaydi, balki ma’lumotga kirish yo‘lini optimallashtiradi. Date ma’lumotlar bazasida mantiqiy model va fizik amalga oshirish farqini alohida ajratadi [5]. Indekslar aynan fizik optimallashtirish vositasidir.

Indekslarning asosiy vazifalari qidiruvni tezlashtirish, saralash va filtrlashni yengillashtirish, JOIN operatsiyalarini optimallashtirish hamda unikal indekslar orqali ma’lumotlar yaxlitligini qo‘llab-quvvatlashdan iborat. Indekssiz jadvalda tizim ko‘pincha full table scan, ya’ni butun jadvalni ketma-ket tekshirish usulidan foydalanadi. Katta hajmdagi jadvallarda bunday usul sezilarli vaqt va resurs talab qiladi.

Indekslarning asosiy turlari

Ma’lumotlar bazalarida indekslarning bir nechta turlari qo‘llaniladi. Oddiy indeks bitta ustun asosida yaratiladi va shu ustun bo‘yicha qidiruvni tezlashtiradi. Unikal indeks esa ustundagi qiymatlarning takrorlanmasligini ta’minlaydi. Masalan, foydalanuvchi loginlari, pasport raqamlari yoki elektron pochta manzillari uchun unikal indekslardan foydalanish ma’lumotlar yaxlitligini saqlashga yordam beradi.

B-tree indeks relatsion ma’lumotlar bazalarida eng keng tarqalgan indeks turidir. U muvozanatlangan daraxtsimon tuzilma asosida ishlaydi va tenglik, oraliq qidiruv, saralash hamda taqqoslash operatsiyalarida samarali natija beradi. Garcia-Molina, Ullman va Widom indeks tuzilmalarini so‘rovlarni bajarishdagi muhim optimallashtirish vositalari qatorida ko‘rsatadi [3].

Hash indeks aniq qiymat bo‘yicha qidiruvda samarali bo‘ladi. U hash funksiyasi yordamida qiymatni ma’lum manzil bilan bog‘laydi. Masalan, foydalanuvchi ID raqami yoki loginini aniq topish talab etilganda hash indeks tez natija berishi mumkin.

Biroq hash indeks qiymatlar orasidagi tartibni saqlamaydi, shu sababli oraliq qidiruv va saralashda B-tree indeksga nisbatan cheklangan imkoniyatga ega.

Kompozit indeks ikki yoki undan ortiq ustun asosida yaratiladi. Bunday indekslar murakkab so‘rovlar ko‘p bajariladigan tizimlarda muhim ahamiyatga ega. Masalan, talabalarni guruh va qabul yili bo‘yicha qidirish kerak bo‘lsa, (group_name, admission_year) indeksidan foydalanish mumkin. Winand kompozit indekslarda ustunlar ketma-ketligi so‘rov samaradorligiga kuchli ta’sir qilishini ta’kidlaydi [9].

Indekslarning sql so‘rovlar samaradorligiga ta’siri

SQL so‘rovlarida indekslar asosan WHERE, JOIN, ORDER BY va GROUP BY operatorlari bilan bog‘liq holda ishlatiladi. WHERE shartida indekslangan ustun qatnashsa, ma’lumotlar bazasi kerakli yozuvlarni topish uchun butun jadvalni ko‘zdan kechirmaydi. Bu ayniqsa millionlab yozuvlarga ega jadvallarda muhim amaliy natija beradi.

JOIN operatsiyalarida indekslar jadvallar orasidagi mos yozuvlarni tezroq topishga yordam beradi. Relatsion ma’lumotlar bazalarida jadvallar bir-biri bilan kalit ustunlar orqali bog‘lanadi. Agar bog‘lanishda qatnashayotgan ustunlar indekslangan bo‘lsa, tizim mos yozuvlarni topish uchun kamroq resurs sarflaydi. Ramakrishnan va Gehrke relatsion tizimlarda indekslarning jadval bog‘lanishlari jarayonidagi ahamiyatini alohida ko‘rsatadi [4].

ORDER BY operatorida indekslangan ustunlardan foydalanish saralash xarajatini kamaytiradi. GROUP BY operatorida ham indekslar qiymatlarni tartibli ko‘rib chiqish orqali guruhlash jarayonini yengillashtirishi mumkin. Ammo indeks mavjud bo‘lishi uning albatta ishlatilishini anglatmaydi. Ma’lumotlar bazasi optimizatori indeksdan foydalanish yoki foydalanmaslikni jadval hajmi, statistik ma’lumotlar va so‘rov murakkabligiga qarab belgilaydi.

Query Execution Plan indekslarning haqiqiy ishlashini tekshirishda muhim vositadir. U so‘rov qanday bajarilayotganini, qaysi indeks tanlanganini va jadval to‘liq skan qilingan-qilinmaganini ko‘rsatadi. Schwartz, Zaitsev va Tkachenko MySQL samaradorligini tahlil qilishda execution plan orqali indekslardan foydalanishni tekshirishni muhim amaliy bosqich sifatida ko‘rsatadi [8].

Amaliy tahlil

Talabalar jadvali misolida students(id, full_name, group_name, admission_year, birth_year) tuzilmasini ko‘rib chiqamiz. Agar tizimda talabalarni guruh bo‘yicha qidirish tez-tez bajarilsa, group_name ustuniga indeks yaratish maqsadga muvofiq bo‘ladi. Bunday indeks SELECT * FROM students WHERE group_name = “622-22” kabi so‘rovlar bajarilishini tezlashtiradi.

Agar qidiruv ko‘pincha guruh va qabul yili bo‘yicha birgalikda bajarilsa, kompozit indeks samaraliroq bo‘ladi. Masalan, (group_name, admission_year) indeksini yaratish WHERE group_name = “622-22” AND admission_year = 2022 shartli so‘rovda yaxshi natija berishi mumkin. Ammo faqat admission_year ustuni bo‘yicha qidiruv bajarilganda ushbu indeksning samarasi kamayishi ehtimoli bor.

Yana bir amaliy holat sifatida students va results jadvallarini bog‘lashni ko‘rish mumkin. Agar results jadvalidagi student_id ustuni indekslangan bo‘lsa, talabalar va ularning natijalarini bog‘lovchi JOIN so‘rovi tezroq bajariladi. Bu elektron jurnal, baholash tizimi va ta’lim boshqaruv platformalarida amaliy ahamiyatga ega. Harrington relatsion loyihalashda jadval bog‘lanishlari va kalit maydonlar to‘g‘ri tanlanishi tizim samaradorligiga ta’sir qilishini ta’kidlaydi [6].

Muhokama: afzalliklar, cheklovlar va tavsiyalar

Indekslarning asosiy afzalligi qidiruv tezligini oshirishidir. Katta hajmdagi jadvalda kerakli yozuvni tez topish foydalanuvchiga javob qaytarish vaqtini qisqartiradi. Bundan tashqari, indekslar saralash, filtrlash, JOIN va ayrim guruhlash operatsiyalarida samaradorlikni oshiradi. To‘g‘ri indekslash server resurslaridan oqilona foydalanishga yordam beradi.

Biroq indekslarning cheklovlari ham mavjud. Har bir indeks qo‘shimcha xotira egallaydi. Jadvalga yangi yozuv qo‘shilganda, mavjud yozuv yangilanganda yoki o‘chirilganda indekslar ham yangilanishi kerak. Shu sababli ortiqcha indekslar INSERT, UPDATE va DELETE operatsiyalarini sekinlashtirishi mumkin. Agar ustundagi qiymatlar juda ko‘p takrorlansa, indeks kutilgan natijani bermasligi ham mumkin.

Samarali indekslash uchun real SQL so‘rovlari tahlil qilinishi kerak. Qaysi ustunlar WHERE shartida, qaysilari JOIN jarayonida, qaysilari ORDER BY yoki GROUP BY operatorlarida ishlatilayotgani aniqlanishi lozim. Kompozit indekslarda ustunlar tartibi ehtiyotkorlik bilan tanlanadi, har bir muhim indeks esa Query Execution Plan orqali tekshiriladi. Kleppmann ma’lumotlar intensiv tizimlarda samaradorlik saqlash modeli, indekslash, tranzaksiyalar va tizim arxitekturasining uyg‘unligiga bog‘liq ekanini ko‘rsatadi [7]. Özsu va Valduriez taqsimlangan ma’lumotlar bazalarida indekslash, so‘rovlarni optimallashtirish va ma’lumotlarni joylashtirish o‘zaro bog‘liq jarayonlar ekanini ta’kidlaydi [10].

Xulosa

Indekslar ma’lumotlar bazalarida qidiruv, saralash, filtrlash va jadvallarni bog‘lash jarayonlarini tezlashtiruvchi muhim optimallashtirish vositasidir. Ular katta hajmdagi ma’lumotlar bilan ishlovchi axborot tizimlarida javob qaytarish vaqtini qisqartiradi, server resurslaridan foydalanishni yaxshilaydi va foydalanuvchi tajribasini oshiradi.

Tahlillar shuni ko‘rsatadiki, B-tree indekslar ko‘p hollarda universal va samarali yechim bo‘lib xizmat qiladi, hash indekslar aniq qiymat bo‘yicha qidiruvda foydali, kompozit indekslar esa murakkab ko‘p ustunli so‘rovlarni optimallashtirishda muhim o‘rin tutadi. Shu bilan birga, indekslar qo‘shimcha xotira egallaydi va yozish operatsiyalariga ortiqcha yuklama berishi mumkin.

Umuman olganda, indekslash strategiyasi real so‘rovlar, jadval hajmi, ma’lumotlar taqsimoti, o‘qish va yozish operatsiyalari nisbati hamda tizim arxitekturasi bilan uyg‘un holda ishlab chiqilgandagina yuqori samaradorlik beradi. To‘g‘ri indekslash zamonaviy relatsion ma’lumotlar bazalarida ishlash samaradorligini ta’minlovchi muhim ilmiy-amaliy omildir.

Foydalanilgan adabiyotlar

1. Abror, D. (2025). RELATSION VA NORELYATSION (NoSQL) MA’LUMOTLAR BAZALARINING TAQQOSLANISHI. INTERNATIONAL JOURNAL OF INTEGRATED SCIENCES, 1(1).

2. Sherdil o’g’li, M. B., & Dilshodjon o’g’li, D. A. (2025). MA’LUMOTLAR BAZASIDA INDEKSLASH VA SO ‘ROVLARNI OPTIMALLASHTIRISH. SCIENTIFIC AND PRACTICAL RESEARCH OF THE 21ST CENTURY, 1(1).

3. Jaxongir o’g’li, K. S., Nuriddinjon o’g’li, O. O., & Dilshodjon o'g'li, D. A. (2025). MA’LUMOTLAR BAZASIDAN FOYDALANISHDA KENG TARQALGAN XATOLAR VA ULARNI BARTARAF ETISH. EDUCATION AND SCIENCE YESTERDAY AND TODAY, 1(1).

4. Dilshodjon ogli, D. A. (2025). MA'LUMOTLAR BAZALARIDA SUN'IY INTELEKT YORDAMIDA NATIJALARNI BASHORAT QILISH. Shokh Articles Library, 1(1).

5. Nodirbek, S., & Abrorbek, D. (2025). MA'LUMOTLAR BAZASINING XAVFSIZLIGI VA KIRISH NAZORATI. SCIENTIFIC AND PRACTICAL RESEARCH OF THE 21ST CENTURY, 1(1).

6. Dilshodjon o’g’li, D. A. (2025). NOSQL TIZIMLARI VA KATTA HAJMDAGI MA’LUMOTLARNI BOSHQARISHDAGI AHAMIYATI. SCIENTIFIC AND PRACTICAL RESEARCH OF THE 21ST CENTURY, 1(1).

7. Feruzbek, M., & Abrorbek, D. (2025). SQL VA NOSQL MA’LUMOTLAR BAZALARINI TAQQOSLASH: ARXITEKTURA, SAMARADORLIK VA QO ‘LLANISH JIHATLARI. SCIENTIFIC AND PRACTICAL RESEARCH OF THE 21ST CENTURY, 1(1).

8. Shohzamon, Y., & Abrorbek, D. (2025). MA’LUMOTLAR BAZASI VA FAYL TIZIMLARI O ‘RTASIDAGI FARQ. SCIENTIFIC AND PRACTICAL RESEARCH OF THE 21ST CENTURY, 1(1).

9. Abrorbek, D. (2025). MA’LUMOTLAR BAZASIDA XAVFSIZLIK VA SQL INJECTION. EDUCATION AND SCIENCE YESTERDAY AND TODAY, 1(1).

10. Sevinch, H., & Abrorbek, D. (2025). SQL SELECT SO ‘ROVLARI BILAN ISHLASH. MA’LUMOTLARNI SARALASH. WHERE, AND, OR, NOT OPERATORLARIDAN FOYDALANISH. EDUCATION AND SCIENCE YESTERDAY AND TODAY, 1(1).

11. Ozodbek, X., Qosimjon, S., Otabek, A., & Abrorjon, D. (2025). TAQSIMLANGAN MA’LUMOTLAR BAZALARI VA ULARNING KOMPYUTER TARMOQLARIDAGI AHAMIYATI. SCIENTIFIC AND PRACTICAL RESEARCH OF THE 21ST CENTURY, 1(1).

12. UniPublish, M. C. H. J. (2025). Ma’lumotlar bazasida saralash va filtrlashning nazariy asoslari. Taraqqiyot Spektri, 1(12), 86-91.