

KODDNING RELYATSION MODELII

Tojimamatov Israil Nurmamatovich*Farg'ona davlat universiteti amaliy matematika va informatika kafedrası kata o'qituvchisi,**Email: israiltojimatov@gmail.com**Farg'ona davlat universiteti talabasi***Omonaliyeva Elinur Umidbek qizi***Email: omonaliyevaelinur644@gmail.com*

Annotatsiya: Ushbu ishda **koddning relatsion modeli** va uning zamonaviy ma'lumotlar bazalaridagi o'rni o'rganiladi. Relatsion model ma'lumotlarni **jadval shaklida** saqlash, ular orasidagi bog'lanishlarni aniqlash va ma'lumotlar ustida mantiqan to'g'ri amallarni bajarishni ta'minlash tamoyiliga asoslangan. Modelning asosiy tushunchalari sifatida relatsiya (jadval), tupel (qator), atribut (ustun), domain (atribut qiymatlari to'plami), primary key (asosiy kalit) va foreign key (chet kalit) keltiriladi. Relatsion model dasturchilar uchun qulay interfeys yaratadi, ma'lumotlar yaxlitligi va ishonchliligini ta'minlaydi hamda SQL tili orqali CRUD operatsiyalarini bajarish imkonini beradi. Ishda relatsion modelning **Talabalar va Kurslar** misoli asosida amaliy qo'llanilishi ko'rsatildi, jumladan, bog'lanish jadvali orqali many-to-many munosabatlar tashkil etilishi, primary va foreign key orqali ma'lumotlar integriteti saqlanishi va SQL so'rovlari yordamida ma'lumotlarni olish tartibi tushuntirildi. Shuningdek, relatsion modelning tranzaksiyalarni boshqarish va ACID printsiplari asosida ishlash xususiyatlari ham yoritildi. Natijada, koddning relatsion modeli dasturchilar uchun ma'lumotlar bazasini samarali loyihalash va boshqarishning asosiy vositasi sifatida taqdim etiladi.

Ka'lit so'zlar: Relatsion model, interfeys, MySQL, PostgreSQL, Oracle, SQL Server, tranzaksiya, tupel, atribut.

Annotation: This work studies **Codd's relational model** and its role in modern database systems. The relational model is based on the principles of storing data in **tables**, defining relationships between them, and performing logically consistent operations on the data. Key concepts of the model include relation (table), tuple (row), attribute (column), domain (set of possible values for an attribute), primary key, and foreign key. The relational model provides developers with a convenient interface, ensures data integrity and reliability, and allows performing CRUD operations using SQL. The study demonstrates the practical application of the relational model using the example of **Students and Courses**, including the creation of a junction table to manage many-to-many relationships, maintaining data integrity through primary and foreign keys, and retrieving data via SQL queries. Additionally, the model's support

for transaction management and operations based on ACID principles is highlighted. As a result, Codd's relational model is presented as a fundamental tool for developers to efficiently design and manage databases.

Keywords: Relational model, interface, MySQL, PostgreSQL, Oracle, SQL Server, transaction, tuple, attribute.

Аннотация: В данной работе рассматривается **реляционная модель Кода** и её роль в современных системах управления базами данных. Реляционная модель основана на принципах хранения данных в **таблицах**, определения связей между ними и выполнения логически корректных операций с данными. Основные концепции модели включают отношение (таблицу), кортеж (строку), атрибут (столбец), домен (множество допустимых значений для атрибута), первичный ключ и внешний ключ. Реляционная модель предоставляет разработчикам удобный интерфейс, обеспечивает целостность и надёжность данных, а также позволяет выполнять операции CRUD с использованием SQL. В работе демонстрируется практическое применение реляционной модели на примере **Студентов и Курсов**, включая создание связующей таблицы для управления отношениями «многие-ко-многим», поддержание целостности данных с помощью первичных и внешних ключей, а также получение данных с помощью SQL-запросов. Кроме того, освещаются возможности модели по управлению транзакциями и работе на основе принципов ACID. В результате реляционная модель Кода представляется как фундаментальный инструмент для разработчиков для эффективного проектирования и управления базами данных.

Ключевые слова: реляционная модель, интерфейс, MySQL, PostgreSQL, Oracle, SQL Server, транзакция, кортеж, атрибут.

Koddning relatsion modeli — bu ma'lumotlar bazasini tashkil qilish va boshqarish uchun matematik asosga ega model bo'lib, u ma'lumotlarni jadval shaklida saqlashni va ular ustida mantiqan to'g'ri amallarni bajarishni ta'minlaydi. Ushbu modelni 1970-yillarda Edgar F. Codd ishlab chiqqan va shu sababli ko'pincha "Codd modeli" deb ataladi. Uning asosiy g'oyasi shuki, ma'lumotlar qat'iy tuzilishga ega bo'lishi, ular orasida bog'lanishlar aniq belgilanishi va dasturchi yoki foydalanuvchi ma'lumotlarni osongina so'rovlar yordamida qayta ishlay olishi kerak.

Relatsion modelda ma'lumotlar bir nechta asosiy tushunchalar orqali ifodalanadi. Relatsiya — bu ma'lumotlar jadvali, ya'ni ma'lum bir mavzuga oid yozuvlar to'plami. Tupel — jadvaldagi bir qator, ya'ni bitta yozuv. Atribut — jadvaldagi ustun bo'lib, ma'lum bir ma'lumot turini ifodalaydi. Har bir atribut qabul qilishi mumkin bo'lgan qiymatlar to'plami esa domain deb ataladi. Har bir yozuvni noyob aniqlash uchun primary key ishlatiladi, boshqa jadvaldagi primary keyga ishora

qiluvchi atribut esa foreign key bo‘lib, shu orqali jadvalar orasida bog‘lanishlar hosil qilinadi.

Relations model har bir obyekt (masalan, talaba, kurs yoki mahsulot) jadval sifatida ifodalanadi, uning xususiyatlari ustunlar, yozuvlar esa tupel bo‘ladi. Misol uchun, Talabalar jadvali quyidagicha bo‘lishi mumkin: StudentID, Ism, Familiya, Yoshi. Bu yerda StudentID primary key bo‘lib, har bir talaba yozuvini noyob identifikatsiya qiladi. Kurslar jadvali esa CourseID, Kurs nomi, O‘qituvchi ustunlaridan iborat bo‘lib, CourseID primary key hisoblanadi. Talaba va kurslar orasidagi ko‘p-ko‘p (many-to-many) bog‘lanish uchun alohida bog‘lanish jadvali ishlatiladi: masalan, StudentID = 1, CourseID = 101. Shu tarzda StudentID Talabalar jadvali, CourseID Kurslar jadvali bilan foreign key orqali bog‘langan bo‘ladi.

Relations modelning katta afzalliklaridan biri bu yaxlitlik (integrity) va normalizatsiya tamoyillari orqali ma‘lumotlar bazasini toza va tartibli saqlashdir. Normalizatsiya jarayoni yordamida ma‘lumotlar takrorlanishi kamaytiriladi, redundant yozuvlar yo‘q qilinadi va bog‘lanishlar mantiqan to‘g‘ri tashkil etiladi. Masalan, agar talabalar jadvalida har bir kursni to‘liq yozib borsak, ma‘lumot takrorlanadi. Buning o‘rniga, kurslar jadvali va bog‘lanish jadvali orqali bu muammo hal qilinadi.

Relations modelda ma‘lumotlar **yaxlitligi (integrity)** muhim rol o‘ynaydi. Primary key va foreign key cheklovlari, NOT NULL, UNIQUE kabi constraintlar orqali noto‘g‘ri yoki takroriy yozuvlarning bazaga kiritilishi oldi olinadi. Shu bilan birga, relations model tranzaksiyalarni qo‘llab-quvvatlaydi, ya‘ni bir necha amallar ketma-ketligi bir butun sifatida bajariladi (ACID xossalari: Atomicity, Consistency, Isolation, Durability). Masalan, bank tizimida pul o‘tkazmasi operatsiyasi ikki yozuvni yangilaydi: bir hisobdan mablag‘ni yechish, ikkinchisiga qo‘shish. Bu operatsiya to‘liq bajarilishi yoki umuman bajarilmasligi kerak, agar jarayonda xatolik yuz bersa, MBBT tranzaksiyani bekor qiladi.

Relations modelni dasturchi uchun amaliy jihatdan tushuntiradigan bo‘lsak, bu **jadvalaro orasidagi bog‘lanishlarni tuzish, SQL orqali so‘rovlar yozish, constraintlar orqali ma‘lumotlarni validatsiya qilish va tranzaksiyalarni boshqarish** demakdir. Relyatsion model bugungi kunda eng keng tarqalgan bo‘lib, MySQL, PostgreSQL, Oracle, SQL Server kabi tizimlar aynan shu tamoyil asosida ishlaydi. Shu tarzda dasturchi har bir obyekttni jadval, uning atributlarini ustun, yozuvlarni tupel sifatida ko‘rib, ular ustida CRUD (Create, Read, Update, Delete) amallarini xavfsiz va samarali bajaradi.

Relations model shuningdek tranzaksiyalarni boshqarishni ham qo‘llab-quvvatlaydi. Tranzaksiya — bu bir nechta mantiqan bog‘langan amallar ketma-ketligi bo‘lib, u to‘liq bajarilishi yoki umuman bajarilmasligi kerak. Masalan, bank tizimida pul o‘tkazmasi operatsiyasida mablag‘ bir hisobdan yechilib, ikkinchisiga qo‘shiladi. ACID printsiplari (Atomicity, Consistency, Isolation, Durability) asosida

tranzaksiyalar ishonchli amalga oshiriladi. Shu bilan birga, relatsion MBBTlar ma'lumotlarni zaxiralash va tiklash mexanizmlari orqali nosozlik yoki avariya holatlarida ma'lumot yo'qotilishini oldini oladi.

Relatsion modelni dasturchi uchun amaliy jihatdan tushuntiradigan bo'lsak, bu jadval strukturasini loyihalash, atribut va yozuvlarni aniqlash, primary va foreign key orqali bog'lanishlar yaratish, normalizatsiya tamoyillari asosida ma'lumotlarni tozalash va SQL orqali CRUD (Create, Read, Update, Delete) amallarini bajarish demakdir. Misol uchun, talabalar va kurslar bilan ishlash tizimida SQL so'rovlari yordamida barcha kurslar ro'yxatini, ma'lum talabaning kurslarini yoki kursga yozilgan talabalarni olish mumkin.

Bugungi kunda relatsion model MySQL, PostgreSQL, Oracle, SQL Server kabi tizimlarning asosi bo'lib xizmat qiladi va dasturchilar uchun qulay interfeys hamda samarali so'rovlarni bajarish imkonini beradi. Shu sababli, relatsion modelni tushunish va uni amaliy tizimlarda to'g'ri qo'llash dasturchilar uchun zarur bo'lgan ko'nikma hisoblanadi, chunki u ma'lumotlar bazasi dizayni, ma'lumotlar yaxlitligi va ishlash samaradorligini ta'minlaydi.

Xulosa: Koddning relatsion modeli zamonaviy ma'lumotlar bazalarining asosiy tamoyili bo'lib, ma'lumotlarni jadval shaklida saqlash, ular orasidagi bog'lanishlarni aniqlash va mantiqan to'g'ri amallarni bajarish imkonini beradi. Modelning asosiy tushunchalari — relatsiya, tupel, atribut, domain, primary key va foreign key — ma'lumotlar yaxlitligi va ishonchliligini ta'minlaydi. Relatsion model dasturchilarga qulay interfeys yaratadi, SQL orqali CRUD operatsiyalarini bajarish imkonini beradi va tranzaksiyalarni ACID printsiplari asosida boshqarishni ta'minlaydi. Amaliy misollar orqali (Talabalar va Kurslar) ko'rsatildiki, bog'lanish jadvali orqali many-to-many munosabatlar tashkil qilinadi va ma'lumotlarning takrorlanishi oldi olinadi. Shu bilan birga, relatsion model ma'lumotlar bazasini samarali loyihalash, boshqarish va himoyalashning muhim vositasi ekanligi tasdiqlanadi.

Foydalanilgan adabiyotlar:

1. Codd, E. F. (1970). *A Relational Model of Data for Large Shared Data Banks*. Communications of the ACM, 13(6), 377–387.
2. Elmasri, R., & Navathe, S. (2016). *Fundamentals of Database Systems* (7th edition). Pearson.
3. Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). *Database System Concepts* (7th edition). McGraw-Hill.
4. Date, C. J. (2004). *An Introduction to Database Systems* (8th edition). Addison-Wesley.
5. Tojimamatov, I., & Mirsiddiqova, M. (2025). BERILGANLAR BAZASIDA HAYOTIY SIKL. Модели и методы в современной науке, 4(6), 66-70.

6. Tojimamatov, I., & Siddiqova, G. (2025). TRANZAKSIYALARNI TAQSIMLANGAN TARZDA QAYTA ISHLASH MODELLARI. Современные подходы и новые исследования в современной науке, 4(6), 30-35.
7. Нурмаматович, Т. И., & Рахила, А. (2025). НА ОСНОВЕ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ПОВЫШЕНИЕ УСТОЙЧИВОСТИ К ПОЛОМКАМ И АВАРИЯМ. YANGI O 'ZBEKISTON, YANGI TADQIQOTLAR JURNALI, 2(8), 197-204.
8. Tojimamatov, I., & Ahmataliyeva, S. (2025). BERILGANLARNI MARKAZLASHGAN TARZDA BOSHQARISH TAMOIYILLARI. Академические исследования в современной науке, 4(21), 59-64.
9. Tojimamatov, I., & Adxamova, C. (2025). AMALIY TIZIMLARDA BERILGANLAR BAZASINI BOSHQARISH TIZIMLARI O'RNII. Академические исследования в современной науке, 4(21), 77-82.
10. Tojimamatov, I., & Fazliddinov, X. (2025). BERILGANLAR BAZASI ADMINISTRATORI VA UNING XUSUSYATLAR. Академические исследования в современной науке, 4(21), 90-95.
11. Karimberdiyevich, O. M., Nurmamatovich, T. I., & Abdulaziz o'g'li, Y. M. (2024). BIG DATA SOHASIDAGI XALQARO LOYIHALAR. IZLANUVCHI, 1(1), 39-45.
12. Karimberdiyevich, O. M., Abdulaziz o'g'li, Y. M., & Zarifjon o'g'li, X. N. (2024). DATA MINING METODLARI VA BOSQICHLARI. YANGI O 'ZBEKISTON, YANGI TADQIQOTLAR JURNALI, 1(4), 303-311.
13. Nurmamatovich, T. I. (2024). BERILGANLARNING TARMOQ MODELLARI: ODDIY VA MURAKKAB TARMOQ TUZILISHLARI.