

**QBE TILI ORQALI TASAVVUR, PROTSEDURA VA FUNKSIYALARNI
YARATISH**

QBE LANGUAGE FOR CREATING VIEWS, PROCEDURES AND FUNCTIONS

**ЯЗЫК QBE ДЛЯ СОЗДАНИЯ ПРЕДСТАВЛЕНИЙ, ПРОЦЕДУР И
ФУНКЦИЙ**

Sobirjonov Behzod Qahramon o'g'li

Farg'ona davlat universiteti Axborot texnologiyalari kafedrası o'qituvchisi
behzodbekqahramonovich@gmail.com

Numonjonov Ozodbek Nodirjon o'g'li

*Farg'ona davlat universiteti Axborot tizimlari va texnologiyalari
yo'nalishi 2-kurs talabasi*
numonjonovozodbek720@gmail.com

Nishonov Abbosbek Qahramonjon o'g'li

*Farg'ona davlat universiteti Axborot tizimlari va texnologiyalari
yo'nalishi 2-kurs talabasi*
abbosbeknishonov00631@gmail.com

Annotatsiya: Ushbu ilmiy-uslubiy maqolada QBE (Query By Example) tili orqali ma'lumotlar bazalarida tasavvurlar (views), protseduralar (procedures) va funksiyalar (functions)ni yaratishning nazariy asoslari hamda amaliy jihatlari chuqur tahlil qilinadi. QBE — bu foydalanuvchilarga SQL sintaksisini qo'lda yozmasdan, qulay vizual interfeys orqali so'rovlar yaratish, ularni sinovdan o'tkazish va natijalarni interaktiv ko'rish imkonini beruvchi texnologiyadir.

Maqolada QBE tilining paydo bo'lish tarixi, uning ma'lumotlar bazasi boshqaruv tizimlarida (xususan, Microsoft Access, FoxPro, Oracle Forms) tutgan o'rni va afzalliklari yoritilgan. Shuningdek, maqolada foydalanuvchilar uchun SQL buyruqlarini avtomatik shakllantirish jarayoni, QBE muhitida ma'lumotlarni filtrlash, birlashtirish va tahlil qilish imkoniyatlari amaliy misollar bilan tushuntirilgan.

Tadqiqot natijalari shuni ko'rsatadiki, QBE orqali yaratilgan tasavvurlar, protseduralar va funksiyalar foydalanuvchilarga dasturlash tajribasisiz ham ma'lumotlar ustida murakkab operatsiyalarni bajarish imkonini beradi. QBE tili o'quv jarayonlarida, axborot tizimlari loyihalashda va ma'lumotlarni tahlil qilishda samarali vosita sifatida qo'llanilishi mumkin.

Kalit so'zlar: QBE, Query By Example, Access, SQL, view, procedure, function, ma'lumotlar bazasi, vizual interfeys, ma'lumotlarni tahlil qilish, dasturlash.

Abstract: This scientific-methodological article provides a comprehensive analysis of the theoretical foundations and practical applications of the QBE (Query By Example) language in creating views, procedures, and functions within database management systems. QBE is an intuitive visual query language that allows users to interact with databases and create complex queries without the need to manually write SQL syntax.

The article discusses the origin of the QBE concept, its advantages over traditional SQL-based systems, and its wide usage in platforms such as Microsoft Access, FoxPro, and Oracle Forms. Furthermore, it demonstrates how QBE automatically generates SQL commands, processes data filtering and combination tasks, and enables real-time analysis through a user-friendly interface.

Practical examples and visual demonstrations show that QBE simplifies the process of creating logical structures (views), reusable code blocks (procedures), and computational modules (functions) in databases. It can be effectively used in educational settings, information system development, and data analysis tasks.

Keywords: QBE, Query By Example, Access, SQL, view, procedure, function, database, visual interface, query building, data analysis.

Аннотация: В данной научно-методической статье подробно рассматриваются теоретические и практические аспекты использования языка QBE (Query By Example) при создании представлений (views), процедур (procedures) и функций (functions) в системах управления базами данных. QBE — это интуитивно понятный визуальный язык запросов, который позволяет пользователям взаимодействовать с базой данных без необходимости вручную писать SQL-код.

В статье описываются история возникновения QBE, его преимущества перед традиционными языками SQL, а также области применения в таких системах, как Microsoft Access, FoxPro и Oracle Forms. Особое внимание уделено механизмам автоматической генерации SQL-запросов, фильтрации данных и их объединению с использованием графического интерфейса.

Практические примеры показывают, что использование QBE значительно упрощает процесс создания логических представлений, процедур и функций, что делает этот язык эффективным инструментом для образовательных целей, проектирования информационных систем и анализа данных.

Ключевые слова: QBE, Query By Example, Access, SQL, представление, процедура, функция, база данных, визуальный интерфейс, анализ данных, построение запросов.

Kirish

Zamonaviy axborot texnologiyalari davrida ma'lumotlar bazalarini boshqarish tizimlari (MBBT) turli sohalarda — ta'lim, tibbiyot, iqtisodiyot, ishlab chiqarish va boshqalarda muhim ahamiyat kasb etadi. Ma'lumotlar bazalari bilan samarali ishlashda foydalanuvchidan SQL (Structured Query Language) tilini mukammal bilish talab qilinadi. Biroq barcha foydalanuvchilar ham dasturlash yoki SQL sintaksisini o'rganishga qiziqmaydi yoki bunga ehtiyoj sezmaydi. Shu sababli intuitiv va vizual vositalar asosida ishlovchi interfeyslarning paydo bo'lishi ma'lumotlar bilan ishlashni ancha soddalashtirdi.

Shunday qulay vositalardan biri — QBE (Query By Example) tilidir. QBE tili foydalanuvchilarga SQL so'rovlarini bevosita yozmasdan, maxsus jadval ko'rinishidagi interfeys orqali tuzish imkonini beradi. Bu esa ma'lumotlar bazasi bilan muloqotni grafik muhitda tashkil etish imkonini berib, foydalanuvchi uchun qulaylik va tezkorlikni ta'minlaydi. QBE dastlab IBM kompaniyasi tomonidan 1970-yillarda ishlab chiqilgan bo'lib, keyinchalik Microsoft Access, FoxPro, dBase, Oracle Forms kabi tizimlarda keng tatbiq etilgan.

QBE tilining asosiy afzalligi shundaki, u SQL kodini avtomatik tarzda generatsiya qiladi va shu orqali foydalanuvchining dasturlash ko'nikmasiga ehtiyojni kamaytiradi. Shu sababli, bu til o'quv jarayonlarida, tahliliy ishlarda, shuningdek, kichik va o'rta korxonalar faoliyatida ma'lumotlar bazasi bilan ishlashni osonlashtiruvchi vosita sifatida keng qo'llaniladi.

QBE yordamida foydalanuvchi so'rovlar (queries), tasavvurlar (views), protseduralar (procedures) va funksiyalar (functions) yaratishi, ma'lumotlarni filtrlash, tahlil qilish va yangilash amallarini avtomatlashtirishi mumkin. Ayniqsa, ma'lumotlarni mantiqiy tuzilmalarga ajratish, hisobotlar tayyorlash va hisob-kitoblarni tezkor bajarish jarayonlarida QBE tilining samaradorligi yuqoridir.

Shuningdek, QBE tili ta'lim jarayonida ham alohida ahamiyatga ega — u SQL so'rovlarini o'rgatish jarayonida vizual yondashuvni ta'minlaydi, bu esa talabalarga so'rovlarning mantiqiy tuzilishini oson tushunishga yordam beradi. Shu jihatdan, QBE dasturchilar va oddiy foydalanuvchilar o'rtasidagi bilim farqini qisqartiruvchi muhim vosita sifatida qaraladi.

Ushbu maqolada QBE tilining nazariy asoslari, uning amaliy qo'llanilishi, shuningdek, tasavvurlar (views), protseduralar (procedures) va "funksiyalar (functions)"ni yaratish usullari batafsil yoritiladi. Maqola doirasida Microsoft Access misolida QBE tilining imkoniyatlari, afzalliklari va amaliy natijalari tahlil qilinadi.

Asosiy qisim

Query By Example (QBE) — bu foydalanuvchiga SQL kabi deklarativ so'zlashuvni qo'lda yozmasdan, ma'lumotlar bazasidan so'rovlar yaratish imkonini beruvchi grafik yoki interaktiv so'rov quruvchi muhitdir. QBE g'oyasi 1970-yillarda IBM ilmiy tadqiqotlarida shakllangan va ma'lumotlar bazalari bilan ishlashni soddalashtirishga qaratilgan. Qisqa muddat ichida u ta'lim jarayonida, kichik va o'rta biznes ilovalarida, hamda ofis avtomatizatsiyasi vositalarida ommalashdi (masalan, Microsoft Access, dBASE, va ba'zi BI vositalarida QBE-ga o'xshash interfeys elementlari mavjud).

QBE ning maqsadi — so'rovlar tuzish jarayonini “namuna kiritish” (example) orqali vizualizatsiya qilish va foydalanuvchi kiritgan parametrlar asosida kerakli SQL so'rovini avtomatik hosil qilishdir. Shu bois QBE interaktivlik, tushunarlik va tez o'rganish xususiyatlari bilan ajralib turadi.

QBE tizimining ish jarayoni bir nechta asosiy bosqichga bo'linadi:

1. Modelni ko'rsatish (Representation) — foydalanuvchiga jadval va ustunlar grafik tarzda ko'rsatiladi; bu yerda har bir maydon (field)ga misol qiymatlar yoki shartlar kiritiladi.
2. Namuna kiritish (Example input) — foydalanuvchi kerakli ustunlarga “namuna” yoki shartlarni yozadi (masalan, Like "A*" yoki > 70).
3. So'rovni generatsiya qilish (Query generation) — tizim foydalanuvchi kiritgan namuna asosida SQL sintaksisida so'rov (SELECT, UPDATE, DELETE va hokazo) hosil qiladi.
4. So'rovni bajarish (Execution) — hosil qilingan SQL so'rovi DBMS tomonidan bajariladi.
5. Natijani vizual ko'rsatish (Result visualization) — natija jadval yoki hisobot shaklida foydalanuvchuga taqdim etiladi.

Bu jarayonda muhim jihat — QBE tizimi foydalanuvchi kiritmalarini validatsiya qiladi, turli tipdagi operatorlarni (LIKE, BETWEEN, IN va hokazo) mos SQL konstruksiyalariga aylantiradi, hamda ba'zan indekslardan foydalanish yoki joinlarni optimallashtirish uchun qo'shimcha yo'riqnomalar qo'shishi mumkin.

QBE tizimi bir necha funksional komponentlardan tashkil topadi:

- Query Designer (grafik oynasi): jadval va ustunlar ko'rsatiladi; foydalanuvchi ustunlarga shart va ifodalarni kiritadi.
- Expression Builder: hisoblangan ustunlar va matematik/lojiiy ifodalar yozish uchun mo'ljallangan modul.
- Action Queries: UPDATE, DELETE yoki INSERT kabi amal bajaruvchi so'rovlarni yaratishga imkon beradi.

- Macro/Procedure Builder: bir nechta so‘rov va boshqa amallarni ketma-ket bajarish uchun makro yoki protsedura yaratish.
- View/Report saver: so‘rovni View yoki Report sifatida saqlash va keyin qayta ishlatish imkoniyati.
- SQL Preview/Debug (ba’zi tizimlarda): hosil bo‘lgan SQL kodni ko‘rish va tahrirlash imkoniyati — ayniqsa ilg‘or foydalanuvchilar uchun foydalidir.

Interfeys dizayni foydalanuvchi uchun doimiy ravishda izchil bo‘lishi kerak; bu elementlar yordamida QBE oddiy foydalanuvchilardan tortib, o‘rta darajadagi analitiklargacha bo‘lgan auditoriyalarga xizmat qiladi.

QBE yordamida View (tasavvur) yaratish

View — ko‘pincha ma’lumotni qayta ishlash va hisobotlar uchun ishlatiladigan virtual jadval. QBE yordamida view yaratishning afzalliklari:

- Oddiyligi: foydalanuvchi murakkab JOIN va filtrlarni vizual usulda belgilaydi; tizim esa to‘g‘ri SQL shakliga keltiradi.
- Modullik: view lar boshqa so‘rovlar va reportlarda takroran ishlatilishi mumkin.
- Biznes mantiqini ajratish: murakkab filtrlar va kalkulyatsiyalarni view ichida saqlash orqali frontendni soddalashtirish mumkin.

Ammo praktika shuni ko‘rsatadiki, QBE orqali yaratilgan viewlar har doim samarali ishlamaydi — ayniqsa view ichida ko‘p darajali subquerylar yoki noyob optimizatsiya talab qilinadigan vaziyatlarda. Shu sababli viewlar dizayn qilinayotganda indekslar va performans tahlili ham amalga oshirilishi lozim.

Protsedura (Procedure) va makroslarni yaratish

QBE muhitida protsedura yaratish ko‘pincha ikki turga bo‘linadi:

1. Action Query asosidagi protseduralar: bir nechta UPDATE/DELETE/APPEND so‘rovlarini ketma-ket bajarish.
2. Makro yoki skript: tizimning makro funksiyasi orqali bir qator amallarni avtomatlashtirish (foydalanuvchi tugmani bosin yoki jadval yangilanganda avtomatik ishga tushsin).

Protseduralar uchun amaliy maslahatlar:

1. Protseduralarni yaratishdan oldin har doim backup nusxasini oling
2. Makrolarda foydalanuvchiga xatoliklar haqida ma’lumot beruvchi logging va error-handling qo‘shing.
3. Protsedurani imkoniyat bo‘lsa parametrik qiling — bu uni qayta ishlatishni osonlashtiradi.

Funksiyalar va hisoblangan ustunlar

QBE muhitida funksiyalar odatda Expression Builder yordamida yaratiladi; ular quyidagi vazifalarni bajaradi:

- Reying, ball, ko'rsatkichlarni hisoblash.
- Matnlarni formatlash va shartli qiymatlarni qaytarish.
- Bir nechta ustunlarga asoslangan murakkab ifodalarni oddiy ko'rinishda taqdim etish.

Yaxshi amaliyotlar:

➤ Murakkab yoki ko'p foydalaniladigan formulalarni DBMS darajasida (ya'ni User-Defined Function) yaratish tavsiya etiladi — bu performans va qayta ishlatish nuqtai nazaridan samaraliroq.

➤ Hisoblangan ustunlarda imkon qadar indekslarni qo'llashni rejalashtiring (ba'zi DBMSlar materialize qilingan view yoki computed column indekslash imkonini beradi).

QBE va SQL — taqqoslash va o'zaro to'ldiruvchanlik

QBE va SQL bir-birini to'ldiradi: QBE — tez, intuitiv va o'rganishga qulay; SQL esa — yanada kuchli, optimallashtirilgan va murakkab so'rovlarni yozish imkonini beradi. Quyidagi holatlar mavjud:

- QBE dan foydalanish joylari: tez prototiplash, ta'lim, ofis ilovalari, oddiy hisobotlar.
- SQL dan foydalanish joylari: murakkab ETL jarayonlari, katta hajmli ma'lumotlar, optimallashtirish va tizim darajasidagi monitoring.

Har ikkisi ham bir loyihada hamkorlikda ishlatilishi mumkin: QBE orqali yaratilgan so'rovlarni tahlil qilib, keyin ular asosida qo'lda yozilgan optimallashtirilgan SQL versiyasini ishlab chiqish amaliyoti keng tarqalgan.

Performance (ishlash tezligi) va optimizatsiya masalalari

QBE interfeysi ortida hosil bo'lgan SQL har doim eng samarali bo'lmisligi mumkin. Shuning uchun quyidagi jihatlarga e'tibor berish zarur:

1. Indexlardan foydalanish: where shartlarida ishlatiladigan ustunlarga indeks qo'yish.
2. JOIN strategiyasi: QBE avtomatik JOINlar keltirib chiqarishi mumkin — ularning tiplari (INNER, LEFT, RIGHT) va ketma-ketligi performansga ta'sir qiladi.
3. Materialize qilingan viewlar: agar view juda qimmatli hisob-kitoblar olib borayotgan bo'lsa, uni materialize qilish (saqlangan natija bilan) samarali bo'ladi.
4. Limit va paging: katta natijalarni bir zumda yuklamaslik uchun pagination yoki TOP/LIMIT operatorlaridan foydalanish.
5. SQL Preview va tahrir: QBE tizimi hosil qilgan SQLni ko'rib chiqish va optimallashtirish uchun SQL preview funksiyasidan foydalanish.

Amaliy tavsiya: murakkab so'rovlar uchun avval QBE orqali prototip tuzib, so'ngra hosil bo'lgan SQLni DBMS-explain (query plan) bilan tekshiring va optimallashtiring.

Xavfsizlik va ruxsatlar (Security & Permissions)

QBE interfeysi og'ir xavfsizlik muammolarining manbai bo'lishi mumkin, chunki foydalanuvchilar ko'p holatda SQLni ko'rmagan holda ma'lumotlarga oson kirishadi. Shu bois:

- Ruxsatlarni qat'iy belgilash: foydalanuvchi rollariga ko'ra view va so'rovlarni cheklash.
- Parametrizatsiya: ma'lumotlarni tahlil qiluvchi so'rovlar parametrli bo'lishi va SQL injection xavfini kamaytirish. (QBE tizimlari odatda parametrlarni avtomatik tasdiqlaydi, ammo ehtiyot chorasi sifatida qo'lda tekshirish tavsiya etiladi.)
- Audit va logging: kim qaysi so'rovni bajarishini va qaysi o'zgartirishlarni qilganini yozib borish.
- Ma'lumotlarni masklash: sezgir maydonlar (PIN, pasport raqami, tibbiy ma'lumot) uchun maskalash yoki anonimlash usullarini qo'llash.

QBE ma'lumotlar bazasi ta'limida juda foydali vositadir:

- Vizual tushuncha: talabalar JOIN, GROUP BY, filtr va hesaplashlarni avval vizual tarzda ko'rib, so'ngra SQLga o'tishadi.
- Tez prototiplash: talabalar tezda so'rovlar yozib, natijalarni tahlil qilishni o'rganadi.
- Amaliy mashqlar: QBE bilan yaratilgan misollarni SQL ga tarjima qilish orqali talabalar ikkala yondashuvni ham o'rganadi.

Ta'lim jarayonida tavsiya: QBE orqali masalani yechib bo'lgach, hosil bo'lgan SQLni ko'rsatish va uning optimallashtirilishini tahlil qilish.

Amaliy misollar**1.Oddiy SELECT View**

Maqsad: Balli 70 dan yuqori bo'lgan va guruh nomi "A" bilan boshlanadigan talabalar ro'yxatini olish.

QBE:

- FIO: -
- Guruh: Like "A*"
- Ball: > 70

Hosil bo'lgan SQL:

```
CREATE VIEW HighScoringAGroup AS
```

```
SELECT FIO, Guruh, Ball
```

```
FROM Talabalar
```

```
WHERE Guruh LIKE 'A%' AND Ball > 70;
```

2.Update Action Query / Protsedura

Maqsad: Balli 50 dan past bo'lgan talabalarning Holat ustunini "Past" deb belgilash.

QBE (Update Query):

- Holat: "Past"
- WHERE Ball < 50

Hosil bo'lgan SQL:

UPDATE Talabalar

SET Holat = 'Past'

WHERE Ball < 50;

3. Hisoblangan ustun (Function)

Maqsad: Reytingni hisoblash — $\text{Reyting} = \text{Ball} * 1.2 + \text{Qo'shimchaBall}$

QBE ustuni:

- Reyting: $[\text{Ball}] * 1.2 + [\text{Qo'shimchaBall}]$

Yuqoridagi misollarni kengroq korxona sharoitida ko'rib chiqishda, misolning har bir bosqichi uchun tranzaksiya nazorat, backup va logging qo'shimcha qismlarini ham rejalashtirish lozim.

QBE ning cheklovlari — real hayotdagi muammolar

➤ Murakkab biznes mantiqlari: QBE oddiy mantiqlar uchun juda yaxshi, lekin murakkab statistika, analitika yoki ma'lumotlarni qayta ishlash jarayonlari uchun yetarli emas.

➤ Massiv ma'lumotlarda samaradorlik: GUI qatlamini katta hajmdagi yozuvlar bilan birgalikda ishlatish tizimni sekinlashtirishi mumkin.

➤ Integratsiya va CI/CD: QBE asosan GUI markazida bo'lganligi sababli kod bazasiga asoslangan versiya nazorati va CI/CD jarayonlariga moslashishi qiyin bo'lishi mumkin.

➤ Platforma bog'liqligi: QBE implementatsiyasi DBMSga xos bo'lishi mumkin — bir platformadan boshqasiga ko'chirishda so'rovlar va viewlarni qayta moslash zarur.

QBE g'oyasi o'zini yangilab, quyidagi yo'nalishlarda rivojlanmoqda

Natural Language Query (NLQ) integratsiyasi: foydalanuvchi tabiiy tilda so'rov bersa, tizim uni QBE/SQL ga aylantiradi.

➤ Visual Query Builders + BI integratsiyasi: QBE elementlari analytics va BI platformalariga integratsiya qilinmoqda, bu esa foydalanuvchiga interaktiv dashboardlar va real vaqt tahlillarini yaratishga imkon beradi.

➤ Auto-optimization va ML yordamida query tuning: hosil qilingan SQLni avtomatik optimallashtirish va so'rov rejalari asosida takliflar berish.

➤ Cloud va multi-tenant arxitekturalar: QBE interfeyslarini bulut xizmatlariga moslashtirish va ko'p foydalanuvchili muhitlarda xavfsizlikni ta'minlash.

XULOSA:

Ushbu maqolada QBE (Query By Example) tilining nazariy va amaliy jihatlari batafsil tahlil qilindi. QBE tilining asosiy afzalliklari — foydalanuvchilarga SQL kodini bilmasdan so'rovlar, tasavvurlar (views), protseduralar (procedures) va funksiyalar (functions) yaratish imkonini berishi, interaktiv grafik interfeys orqali ma'lumotlar bilan tezkor va intuitiv ishlashdir. Maqolada QBE tilining Access kabi ma'lumotlar bazasi tizimlarida amaliy qo'llanilishi misollar bilan ko'rsatildi, jumladan: yuqori baholangan talabalarni chiqarish, past baholilarning holatini avtomatik yangilash va talabaning reytingini hisoblash.

QBE tilining afzalliklari nafaqat foydalanuvchi qulayligi bilan, balki ta'lim jarayonida ham o'z aksini topadi. Talabalar QBE yordamida so'rovlarning mantiqiy tuzilishini vizual tarzda o'rganadi, bu esa SQL va ma'lumotlar bazasi tushunchalarini tez va samarali egallashga yordam beradi. Shu bilan birga, QBE tili katta hajmdagi murakkab so'rovlar yoki ilg'or SQL funksiyalarini qo'lda ishlashda cheklavlarga ega bo'lishi mumkinligi ham qayd etildi.

Maqolada keltirilgan amaliy misollar orqali QBE yordamida yaratilgan tasavvurlar, protseduralar va funksiyalarning samaradorligi ko'rsatildi. Xususan, foydalanuvchi grafik interfeys orqali ma'lumotlarni filtrlash, tartiblash va hisoblashni tezkor amalga oshirishi, shuningdek, natijalarni real vaqt rejimida ko'rishi mumkinligi ta'kidlandi.

Shuni ta'kidlash joizki, QBE tili ma'lumotlar bazalari bilan ishlashni soddalashtiruvchi samarali vosita sifatida, ayniqsa kichik va o'rta hajmdagi ma'lumotlar bazalarida, shuningdek ta'lim va o'quv jarayonlarida keng qo'llanishi mumkin. Shu bilan birga, murakkab ma'lumotlar bazalari va professional tizimlar uchun SQLning to'liq imkoniyatlaridan foydalanish zarur bo'lganda QBE o'rnini to'liq bosa olmaydi.

Xulosa qilib aytganda, QBE tili — foydalanuvchilarni SQL sintaksisidan ozod qiluvchi, ma'lumotlar bilan vizual va interaktiv ishlashni ta'minlovchi qulay vosita bo'lib, uning samarali qo'llanilishi nafaqat amaliy ishlar, balki ta'lim jarayonini ham sezilarli darajada soddalashtiradi.

Foydalanilgan adabiyotlar

1. Date C.J. An Introduction to Database Systems – 8th Edition, Pearson, 2003. – P. 45–78.
2. Pratt P.J., Adamski J.J. Concepts of Database Management – Cengage Learning, 2019. – P. 112–145.
3. Rob P., Coronel C. Database Systems: Design, Implementation, and Management – Cengage, 2020. – P. 200–235.

4. Microsoft Docs. Query By Example (QBE) Overview – <https://learn.microsoft.com> – (murojaat qilingan sana: 2025-yil 1-noyabr).
5. TutorialsPoint. QBE in MS Access – <https://www.tutorialspoint.com> – (murojaat qilingan sana: 2025-yil 1-noyabr).
6. GeeksforGeeks. SQL View, Procedure, Function Concepts – <https://www.geeksforgeeks.org> – (murojaat qilingan sana: 2025-yil 2-noyabr).
7. W3Schools. MS Access Query Examples – <https://www.w3schools.com> – (murojaat qilingan sana: 2025-yil 2-noyabr).