Изпитна тема №8

Бази данни

Въведение в базите от данни. Типове данни. Основни команди. Моделиране на релационни бази от данни. Връзки между таблици. Ограничения в таблици.

Базите данни представляват организирна колекция от данни, обикновено съхранявана в електронен вариант. Това ни позволява да въвждаме, организириме и получаваме данните бързо. Базите данни често се намират на сървър (database server). Това позволява те да бъдат достъпвани от повече от един потребител и предоставя по-добра производителност. Потребителите обикновено не достъпват базите данни директно, а използават системи за управление на бази от данни (database management system - DBMS). Това представлява програма, която ни позволява да работим със съхранените данни. Основните видове бази данни са плосък тип (flat type) бази данни, йерархични бази данни и релационни бази данни.

При създаването на таблица в база данни е необходмо да бъдат зададени име и тип на всяка колона в нея. Типът данни, указва какъв вид данни ще бъдат съхранявани и съответно как трябва да се заимодейства с тях. В MSSQL има 3 основни типа данни – текст, числа и дати. Текст под различни форми може да бъде записан в колони, които са от тип: “char”, “varchar”, “text”. Типът “char” може да съдържа фиксиран брой символи. Дължината се определя, чрез число, което се поставя в скоби след името на типа. “varchar” се различава по това, че при него, дължината не е фиксирана и единственото, което се определя е максималната дължина. “text” е подобен на “varchar”, но при него има възможност за по-голям по размер текст. В SQL има възможност да се запазва и уникод (Unicode) текст. Това се случва, като пред името на типа се добави буквата “n”. Например “varchar” в уникод вариант ще бъде “nvarchar”. Числовите типове в SQL са следните: “bit”, “tinyint“, “smallint“, “int“, “bigint“, “numeric“, “decimal“, “smallmoney“, “money“. Типовете: “datetime“,“ smalldatetime“, “date“, “time“, “datetimeoffset“, “datetime2“ могат да бъдат използвани за колони, в които ще бъдат съхранявани дати.

Командите в SQL могат да бъдат разпределени в 4 категории - eзик за дефиниране на данни (Data Definition Language - DDL), език за манипулиране на данни (Data Manipulation Language - DML), език за управление на данни (Data Control Language - DCL) и език за контрол на транзакциите (Transaction Control Language - TCL).

Командите от езикът за дефиниране на данни (DDL) позволяват да създаваме, редактираме и изтриваме обекти в базите данни. Основните команди са “CREATE”, “ALTER”, “DROP”. С командата “CREATE” се създават разлини обекти. С “ALTER” имаме взъможност да редактираме обекта, а чрез “DROP”, те могат да бъдат изтрити. В следния пример се създава таблица Потребители (Users) с 4 колони. След това тя бива редактиране, като редакцията е добавяна не още една колона. След създаването и редакцията таблицата бива изтрита.

Text

Description automatically generatedGraphical user interface

Description automatically generated with medium confidence



Командите от езика за манипулиране на данни (DML) позволяват да извършваме различни действия като четене (SELECT), добавяне (INSERT), редактиране (UPDATE) и изтриване (DELETE). За да извлечем данни от таблица използваме командата “SELECT”. В следния пример се демонстрира как можем да покажем съдържанието на таблицата Hotels. При използване не “SELECT” може да бъде извършвана и филтрация на данните, чрез клаузата “WHERE”.

Table

Description automatically generated with medium confidenceGraphical user interface

Description automatically generated

Командите “UPDATE”, “INSERT” и “DELETE” съответно се използват за редактиране, въвеждане на нова информация и изтриване.

Релационните бази данни са тип бази данни, които съхраняват и предоставят данни, които са свързани помежду си. В релационните бази данни всеки ред от таблица представлява запис, със уникален идентификатор, наричан ключ. Колоните на всяка таблица съдържат атрибути за данните, а при всеки запис, обикновено има стойност за определения атрибут. Всяка таблица има атрибут, наречен първичен ключ (primary key), чиято стойност е уникалния идентификатор на реда. Таблиците в релационните бази данни могат да бъдат свързани, чрез външни ключове (foreign keys). Външните ключове представляват референция към първичен ключ на друга таблица.

Основната цел на прилагането на техниката за нормализиране е да се намали излишъкът и зависимостта на данните. Нормализацията ни помага да разделим големи таблици на множество малки таблици, като дефинираме логическа връзка между тези таблици. Чрез нормализация се увеличава последователността на данните,тъй като се избягва дублирането на данни, като съхранява данните само на едно място. Помага при групирането на подобни или свързани данни по същата схема, като по този начин се получава по-доброто групиране на данни. Релационните бази данни имат 6 различни нормални форми, но най-широко използвана е 3-та нормална форма.

За да бъде една база данни в 1-ва нормална форма е необходимо да няма повтарящи се данни и да няма колони, които са групирани.

За да бъде една таблица във 2-ра нормална форма тя трябва да бъде вече в първа нормална форма и всички колони трябва да зависят единствено от първичния ключ.

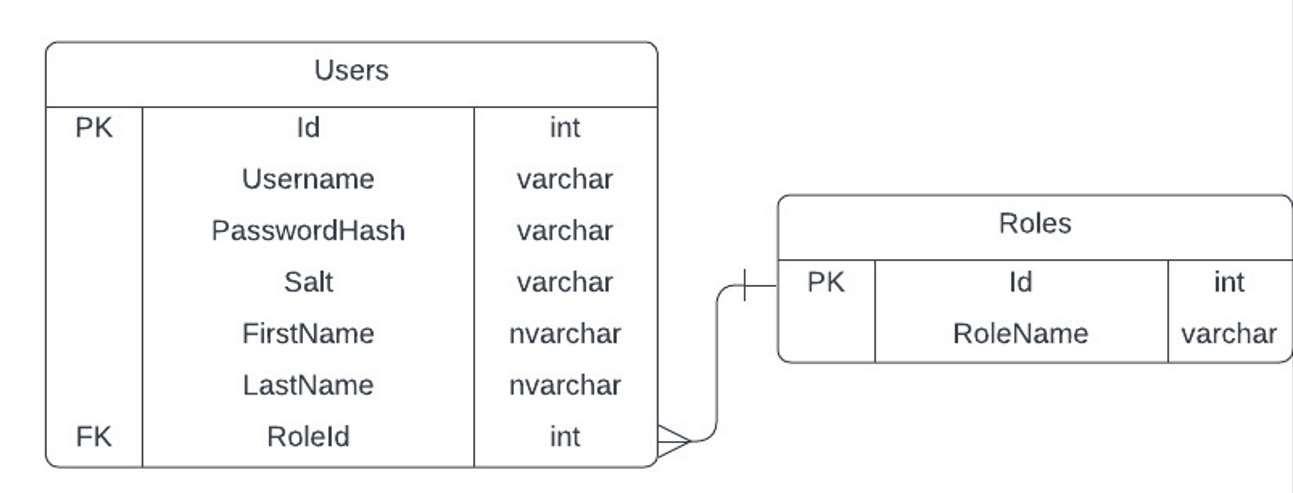
За 3-та нормална форма е необходима като за начало 2-ра нормална форма да бъде изпълнена и колоните без ключ да са независими една от друга. Всяка колона без ключ трябва да зависи единствено от първичния ключ.

За Бойс-Код нормално форма базата данни трябва да бъде в 3-та нормална форма и при всяка функционална зависимост между А и В, А трябва да бъде супер ключ.

За 4-та нормална форма, базата трябва да бъде във Бойс-Код нормална форма и да няма две или повече независими данни, описващо съответния обект.

За да бъде в 5-та нормална форма, трябва да отговаря на условията за четвърта нормална форма и може да бъде разделена на множество таблици, без загуба на данни.

В релационните бази данни релации се сформират, чрез външни ключове. Основинте видове връзки могат да бъдат 3 вида: едно към много, едно към едно и много към много. Връзка едно към много е най-честно срещаната. Тя асоциира ред от родителксата таблица с много редове от таблицата наследник. Връзката едно към едно означава, че на един запис от едната таблица отговаря точно един запис от другата. Връзката много към много позволява много редове от родителската таблица да бъдат свързани с много редове от таблицата наследник. В диаграмата се демонстрира връзка едно към много. Един ред от таблицата Users може да бъде асоцииран с един ред от таблицата Roles, но един ред от таблицата Roles може да бъде асоцииран с много редове от таблицата Users.



Oграниченията (constraints) се използват за определяне на правила за данните в таблица. Ограниченията се използват за ограничаване на типа данни, които могат да влязат в таблица. Това гарантира точността и надеждността на данните в таблицата. Ако има разлика между ограничението и подадените данни, действието се прекъсва. Ограниченията могат да бъдат на ниво колона или на ниво таблица. Ограниченията на ниво колона се прилагат за колона, а ограниченията на ниво таблица се прилагат за цялата таблица.

Най -използваните ограничения, използвани в SQL са “NOT NULL”, което гарантира, че във съответната колона не може да попадне стойност “NULL”. “UNIQUE” гарантира, че данните в колоната, за която е поставено ограничение, ще има уникални данни. Използва се за бързо създаване и извличане на данни от базата данни. Първичния ключ, който идентифицира всеки ред от таблицата също се категоризира като ограничение. Вторичният ключ също е ограничение, което предотвратява действия, които могат да унищожат връзките между таблиците. Ограничението “CHECK” подигорява това, че в таблицата ще бъдат записани данни, които отговарят на конкретни условия. С помощта на „DEFAULT“ може да бъде зададена стойност по подразбиране на съответната колона. В следната заявка се демонстрира по какъв начин могат да бъдат поставени първичен и вторичен ключ. „INDEX“ се използва за бързо създаване и извличане на данни от базата данни.

Обобщен план (Опорни думи)

1. Въведение в базите от данни
   * Организирана колекция от данни. Flat Type, Йерархични, Релационни видове.
   * Database server, СУБД (DBMS)
2. Типове данни
   * Числови – bit, int, tinyint, decimal и тн.
   * Текстови – varchar, char, text и тн.
     + Unicode – nvarchar, ncahr и тн.
   * Дати – date, datetime и тн.
3. Основни команди.
   * DML – SELECT, INSERT, UPDATE, DELETE и тн.
   * DDL - CREATE, ALTER, TRUNCATE и тн.
   * TCL
   * DCL
4. Моделиране на релационни бази от данни.
   * Релационни бази данни, атрибути, primary key, foreign key
5. Връзки между таблици.
   * 1NF- няма повтарящи се колони, всяка колона е уникална.
   * 2NF- 1-ва форма, има primary key, всичко зависи от него.
   * 3NF- 2-ра форма, не са преходни зависими от primary key.
   * BCNF- 3-та форма, зависимост между А и В, А е суперключ.
   * 4NF- БК форма, няма две или повече независими данни, описващи обект.
   * 5NF- - 4-та форма, може да бъде разделена, без загуба на данни.
6. Връзки между таблици
   * Едно към едно
   * Едно към много
   * Много към много
7. Ограничения в таблици.
   * NOT NULL
   * UNIQUE
   * PRIMARY KEY
   * FOREIGN KEY, CHECK, DEFAULT, INDEX

Казус

Цялостно изграждане на база данни.