

23. Бази от данни. Релационен модел на данните.

1. Релационен модел на данните: домейн, релация, кортеж, атрибути

Релационният модел се основава на математическото понятие n -членна релация. Всяка релация е множество от елементи, които се състоят от n компонента и се наричат n -торки. Чрез една релация се моделира даден клас от обекти, а всяка n -торка от релацията представлява конкретен обект от този клас.

Домейн наричаме именувано множество, разглеждано като множество от допустими стойности на някаква величина.

В контекста на БД **релация** е двумерна таблица, в която се съхраняват данните, т.е. релация е таблица от елементи, в която редовете са n -торки, а стълбовете съдържат елементи от един и същ домейн. С други думи, релацията е множество от уникални кортежи.

Атрибутите са означенията (имената) на колоните на релацията.

- Пример: title

С всеки атрибут е асоцииран домейн

- Например title: string

Кортежи са редове в релацията, които съдържат конкретните стойности на всеки компонент

- Пример: (Star wars, 1977, 124, Color)
- Следва се вече определената последователност от атрибутите от релационната схема
- Релацията представлява множество от кортежи
- Компонентите на всеки кортеж принадлежат на домейна за съответния атрибут

Релационният модел изисква всеки компонент на кортеж да бъде атомарен, т.е. се позволяват само прости типове (като integer, string), а сложните типове като списъци и масиви са забранени.

2. Схема на релация; схема на релационна база от данни

Името на релацията и множеството от атрибутите ѝ формират **схемата на релацията**.

Изписваме схемата на релацията като посочваме името, следвано от имената на атрибутите в скоби:

- Пример: Movies(title, year, length, filmType)
- Въпреки че атрибутите в релационна схема са множество, а не списък, ние трябва да укажем стандартна подредба на атрибутите един спрямо друг.

Базата от данни представлява колекция от релации.

Съвкупността от всички схеми на релации в една база от данни се нарича **схема на базата от данни**.

Множеството от кортежи за дадена релация наричаме **екземпляр на релацията**. Една релация се променя с времето. Схемата представлява структурата (или дефиницията) на релацията, а екземпляр е текущо състояние на релацията.

Кардиналността на един екземпляр на релация е равна на броят на кортежите в нея.

3. Реализация на релационната база от данни - не сме го говорили? - питай

Процес на проектиране?:

1. Определяме данните, които ще се съхраняват
2. Определяме връзките между обектите
3. Определяме ключовете и свързващите колони
4. Определяме ограниченията върху обектите и връзките между тях
5. Отстраняване на евентуални недостатъци (излишества)
6. Реализиране на базата от данни

4. Видове операции върху реляционната база от данни

Data definition language - подезик за описание/промяна на схемата

- Create, Drop, Alter
- Пример:

```
CREATE TABLE employees (  
    id            INTEGER    PRIMARY KEY,  
    first_name    CHAR(50)   NULL,  
    last_name     CHAR(75)   NOT NULL,  
    date_of_birth DATE       NULL  
);  
DROP TABLE employees;  
ALTER TABLE employees ADD place_of_birth CHAR(50);  
ALTER TABLE employees DROP COLUMN place_of_birth;
```

- Когато се изпълни DDL операция, промените се прилагат незабавно

Data manipulation language - подезик за работа с данните:

- Select, Insert, Update, Delete
- След като се изпълни DML операция трябва да се направи COMMIT, за да станат промените постоянни.

```
SELECT books.title, count(*) AS Authors  
FROM books  
JOIN book_authors  
ON books.isbn = book_authors.isbn  
WHERE books.price > 100.00  
GROUP BY books.title;  
  
INSERT INTO my_table (field1, field2, field3) VALUES ('test', 'N', NULL);  
  
UPDATE my_table SET field1 = 'updated value' WHERE field2 = 'N';  
  
DELETE FROM my_table WHERE field2 = 'N';
```

Data control language?????

5. Реляционна алгебра

Операциите се извършват върху релациите.

Видове реляционна алгебра

- Ядро на реляционната алгебра - релациите са множества от кортежи
- Разширена реляционна алгебра - релациите са мултимножества от кортежи (съдържат повторения на кортежи)

Реляционната алгебра е междинен език за изчисление на заявки и представлява множество от правила за обработка на алгебрични изрази?

Основни класове операции:

- Операции върху множества (обединение, сечение, разлика)
- Отстраняващи части от релациите (селекция и проекция)
- Комбиниращи кортежи от 2 релации
- Преименуване

За примери ще използваме базата данни:

Movie(Title, Year, length, inColor, studioName, producerC#)

StarsIn(MovieTitle, MovieYear, StarName)

Теоретико множествени операции:

а. **Обединение** (бинарна, комутативна, асоциативна) $R \cup S$

- Прилага се върху релации, които имат съвместими реляционни схеми, т.е. имат еднакъв брой атрибути, които си съответстват
- Резултат: нова релация със същата реляционна схема, където кортежите са обединение от кортежите на двете релации

b. Сечение (бинарна, комутативна, асоциативна) - $R \cap S$

- Прилага се върху релации, които имат съвместими/еквивалентни релационни схеми, т.е. имат еднакъв брой атрибути, които си съответстват
- Резултат: нова релация със същата релационна схема, където кортежите са само общите кортежи на двете релации

c. Разлика (бинарна) $R - S$

- Прилага се върху релации, които имат съвместими/еквивалентни релационни схеми, т.е. имат еднакъв брой атрибути, които си съответстват
- Резултат: нова релация със същата релационна схема, която съдържа всички кортежи от първата релация, които не се съдържат във втората релация

d. Проекция (унарна) $\pi_{\langle attr list \rangle}(R)$

- $\langle attr list \rangle$ е списък атрибути (колони)
- Пример: $\pi_{title, year, length}(Movie)$
- Резултат: Релация, която съдържа само атрибутите посочени в $\langle attr list \rangle$. Ако има повтарящи се кортежи в новата таблица, те се премахват)

e. Select or Restrict (унарна, комутативна) $\sigma_{\langle predicate \rangle}(R)$

- $\langle predicate \rangle$ - условен израз съдържащ:
 - $\langle attribute \rangle \langle op \rangle \langle attribute \rangle$
 - $\langle attribute \rangle \langle op \rangle \langle constant \rangle$
 - $Op \in \{=, \neq, <, >, \leq, \geq\}$
- Резултат: нова релация със същата релационна схема, но съдържаща само кортежите, които отговарят на условия израз
- Това е **вертикална рестрикция**, защото новата релация е със същите атрибути, но броят на кортежите ще е ограничен, т.е. вертикално ще бъде по-къса в общия случай

f. Декартово произведение (бинарна, комутативна, асоциативна) $R \times S$

- Резултат: нова релация с релационна схема, която е обединение на схемите на R и S множеството от всички двойки, при които първият елемент е произволен елемент от R , а вторият - от S .
- При еднакви имена на атрибути в двете релации, използваме имена от типа: $\langle \text{име на релацията} \rangle . \langle \text{име на атрибут} \rangle$

g. Тета съединение (бинарна) $R \bowtie_C S$

- Резултат: нова релация с релационна схема, която е обединение на схемите на R и S , съдържаща кортежите от декартовото произведение $R \times S$, които удовлетворяват C .

h. Еквисъединение - тета съединение, при което условието C включва само съвпадение по атрибутите

i. Естествено съединение $R \bowtie S$

- Свързва по всички атрибути с еднакви имена
- Автоматично отстранява повтарящи се колони

j. Преименуване

- $R_1 := \rho_{R_1(A_1, \dots, A_n)}(R_2)$ превръща R_1 в релация с атрибути A_1, \dots, A_n и същите кортежи като R_2 .

Класификация - основни (Всички други могат да се получат от тях) и допълнителни:

Основни са: селекция, проекция, обединение, разлика, декартово произведение и преименуване

Допълнителни:

- Сечение $R \cap S = R - (R - S)$
- Тета съединение $R \bowtie_C S = \sigma_C(R \times S)$
- Естествено съединение $= R \bowtie S = \pi_L(\sigma_C(R \times S))$, където
 - L - списък с атрибутите в R и тези от S , които не са в R .
 - C - условие от вида $R.A_1 = S.A_1 \text{ AND } R.A_2 = S.A_2 \text{ AND } \dots \text{ AND } R.A_n = S.A_n$, където A_1, \dots, A_n са всички атрибути в схемите на R и S .

Приоритет на операторите:

1. Унарни оператори - селекция, прожекция, преименуване
2. Декартово произведение и селекция
3. Сечение
4. Обединение и разлика
5. Използване на скоби