

Integrita dat relační databáze - význam, princip zajištění integrity pomocí integritních omezení, způsoby nastavení v SQL, příklady

Integrita dat relační databáze - význam, princip zajištění integrity pomocí integritních omezení, způsoby nastavení v SQL, příklady.....	1
Integrita dat.....	2
1. Entitní integritní omezení.....	2
2. Referenční integritní omezení.....	2
3. Doménová integritní omezení.....	3
Příklad Osoby - Funkce.....	4

Integrita dat

- pravidla pro zajištění jednoznačnosti, správnosti a konzistence ukládaných dat
- Konzistence dat = dodržení všech integritních pravidel, které požadujeme
- zajistíme pomocí integritních omezení
- **Proč:**
 - zachycení chybných dat hned na vstupu, aby se chyba neprojevila při dalším zpracování
- **Typy:**
 - deklarativní - vyjádřená přímo v definici databáze, např. PK, FK, UNIQUE, CHECK
 - procedurální - implementovaná pomocí triggerů nebo uložených procedur

1. Entitní integritní omezení

- Jednoznačnost (PK)
- každá entita musí mít jeden **primární klíč** sloužící pro jednoznačné označení každého záznamu v entitě, který je unikátní, není NULL a může být:
 - jednoduchý - jeden atribut
 - složený - složen z více atributů
 - přirozený - atributy které přirozeně popisují entitu (např. rodné číslo)
 - umělý - uměle vytvořený databází (např. id)

2. Referenční integritní omezení

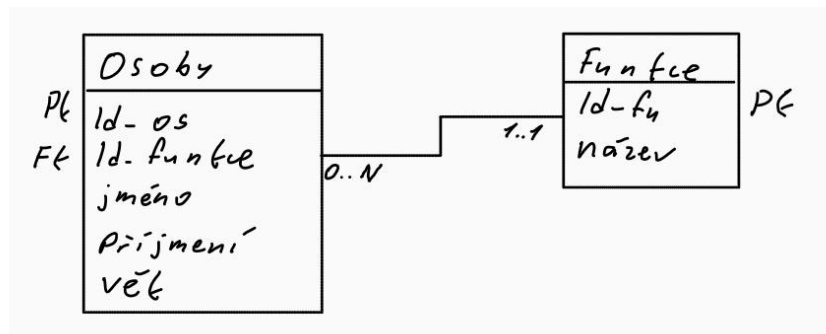
- na každé straně N musí existovat atribut, kterému říkáme cizí klíč FK, který obsahuje hodnotu PK v tabulce na straně 1 a nebo je NULL
- referenční integrita řeší situaci, kdy chceme smazat/upravit záznam na straně 1 - v hlavní tabulce a rozhoduje se co dělat se záznamy na straně N - v podřízené tabulce
- **Druhy akcí:**
 - **NO ACTION** - nepovolí záznam na straně 1 smazat, defaultní
 - **CASCADE** - smazat záznamy na straně N
 - **SET NULL** - nastaví FK na null
 - **SET DEFAULT** - nastaví FK na default

3. Doménová integritní omezení

- omezení atributů
- Typy omezení:
 - NULL, NOT NULL
`NOT NULL CHECK`
 - MIN, MAX - horní, dolní mez
 - CHECK - podmínky pro hodnoty atributu
`CHECK (vek >= 0 AND vek <= 120)`
 - UNIQUE
`UNIQUE`
 - DEFAULT - možnost nastavení standardní hodnotu
`DEFAULT 1`
 - možnost kontroly podle číselníku - (např. ENUM nebo referenční tabulka)

```
CREATE TABLE objednávky(  
    stav NVARCHAR(20) NOT NULL  
    CHECK (stav IN ('Nova', 'Zpracovana', 'Odeslana', 'Stornovana'))  
);
```
 - % wild card pro řetězce (např. LIKE v SQL)
`CHECK (prijmeni LIKE '[A-Z]%)`

Příklad Osoby - Funkce



1. Entitní integrita

- Existence PK

```
id_os int IDENTITY(1, 1) PRIMARY KEY
```

```
id_fu int IDENTITY(1, 1) PRIMARY KEY
```

2. Referenční integrita

- nastavit on update, on delete

```
ALTER TABLE osoby
ADD CONSTRAINT fk_os_fu
FOREIGN KEY (id_fu) REFERENCES funkce (id_fu)
ON DELETE no action
ON UPDATE no action;
```

3. Doménová integrita

```
CREATE TABLE funkce(
    id_fu INT IDENTITY(1, 1) PRIMARY KEY,
    nazev NVARCHAR(100) NOT NULL UNIQUE
);

CREATE TABLE osoby(
    id_os INT IDENTITY(1, 1) PRIMARY KEY,
    id_fu INT NOT NULL DEFAULT 1,
    jmeno NVARCHAR(50) NOT NULL CHECK (jmeno LIKE '[A-Z]%),
    prijmeni NVARCHAR(50) NOT NULL CHECK (prijmeni LIKE '[A-Z]%),
    vek INT CHECK (vek >= 0 AND vek <= 120)
);
```