

INTEGRALNOŚĆ DANYCH – REGUŁY I OGRANICZENIA

Andrzej Ptasznik

Warszawska Wyższa Szkoła Informatyki

aptaszni@wwsi.edu.pl

1. Zagadnienia programowe

Integralność danych pojawia się w kontekście tworzenia relacyjnej bazy danych, tylko na rozszerzonym poziomie zajęć. Bazy danych, od których na ogół wymagamy integralności danych, występują również na poziomie podstawowym.

Informatyka, IV etap edukacyjny, zakres podstawowy

2. Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. Uczeń:
 - 1) znajduje dokumenty i informacje w udostępnianych w Internecie bazach danych (np. bibliotecznych, statystycznych, w sklepach internetowych), ocenia ich przydatność i wiarygodność i gromadzi je na potrzeby realizowanych projektów z różnych dziedzin;
4. Opracowywanie informacji za pomocą komputera, w tym rysunków, tekstów, danych liczbowych, animacji, prezentacji multimedialnych i filmów. Uczeń:
 - 6) tworzy bazę danych, posługuje się formularzami, porządkuje dane, wyszukuje informacje stosując filtrowanie;
 - 7) wykonuje podstawowe operacje modyfikowania, i wyszukiwania informacji na relacyjnej bazie danych;

Informatyka, IV etap edukacyjny, zakres rozszerzony

2. Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. Uczeń:
 - 1) projektuje relacyjną bazę danych z zapewnieniem integralności danych;
 - 2) stosuje metody wyszukiwania i przetwarzania informacji w relacyjnej bazie danych (język SQL);
 - 3) tworzy aplikację bazodanową, w tym sieciową, wykorzystującą język zapytań, kwerendy, raporty; zapewnia integralność danych na poziomie pól, tabel, relacji;
 - 5) opisuje mechanizmy związane z bezpieczeństwem danych: szyfrowanie, klucz, certyfikat, zapora ogniowa.

Uwaga. Przedstawiony scenariusz zajęć wybiega poza zagadnienia programowe i powinien być adresowany do uczniów z lepszym przygotowaniem informatycznym i wykazujących większe zainteresowania pogłębieniem wiedzy z zakresu baz danych.

2. Temat zajęć

Integralność danych – reguły i ograniczenia

Zajęcia są poświęcone omówieniu problemów związanych z zapewnieniem integralności danych z wykorzystaniem Systemu Zarządzania Bazami Danych MS SQL Server 2008 R2 Express Edition.

Komentarz. Omówienie istoty stosowania reguł i ograniczeń na etapie implementacji bazy danych zrealizowane powinno być na prostym przykładzie, zawierającym odniesienia do różnych problemów związanych z zapewnieniem integralności danych.

3. Cele zajęć

W wyniku realizacji tych zajęć uczeń powinien umieć:

- identyfikować problemy związane z integralnością danych dla konkretnego przykładu;
- definiować w języku SQL odpowiednie postaci ograniczeń;
- sprawdzać działanie zdefiniowanych reguł;
- rozumieć i umieć wyjaśnić działanie reguł i ograniczeń.

Komentarz. Cele tych zajęć wykraczają poza podstawę programową.

4. Przygotowanie uczniów

Uczniowie przystępujący do tych zajęć powinni znać:

- podstawowe pojęcia z zakresu baz danych, takie jak: klucz podstawowy, klucz obcy, klucz potencjalny;
- podstawy tworzenia relacyjnych baz danych;
- podstawy języka SQL.

5. Metody pracy

W zajęciach są stosowane następujące metody pracy:

- generalnie, rozwiązywanie każdego rozważanego zagadnienia (problemu) składa się z sześciu etapów, które składają się na metodę rozwiązywania problemów z pomocą komputera; te etapy, to:
 - opis, dyskusja i zrozumienie sytuacji problemowej,
 - podanie specyfikacji problemów do rozwiązania,
 - zaprojektowanie rozwiązania (algorytmu),
 - implementacja (realizacja) rozwiązania w postaci programu komputerowego,
 - testowanie i ewaluacja rozwiązania komputerowego,
 - prezentacja sposobu otrzymania rozwiązania i samego rozwiązania;
- przygotowanie przez uczniów listy problemów związanych z integralnością danych, które mogą wystąpić w omawianym przykładzie bazy danych;
- samodzielne sporządzenie przez uczniów opisów sposobów rozwiązania poszczególnych problemów;
- zapisanie odpowiednich poleceń w języku SQL i ich testowanie;



- samodzielne testowanie poprawności zdefiniowanych reguł dla odpowiednio dobranych danych;
- prezentacja otrzymanych rozwiązań.

6. Formy pracy

Założone cele są realizowane za pomocą następujących form pracy:

- podczas burzy mózgów prowadzonej przez całą klasę lub w grupach uczniów – ma to doprowadzić do sformułowania problemów związanych z integralnością danych dla określonego przykładu bazy danych;
- każdy uczeń tworzy swoją bazę danych a w niej tabele, dla których będą definiowane ograniczenia;
- polecenia definiujące konkretne ograniczenia uczniowie piszą samodzielnie;
- testowanie działania zdefiniowanych reguł i ograniczeń jest realizowane indywidualnie przez poszczególnych uczniów;
- końcowym efektem pracy nad danym problemem jest prezentacja działania zdefiniowanych reguł.

7. Materiały pomocnicze

Nauczyciel i uczniowie korzystają z tablicy lub z tablicy z kartkami papieru, gdzie zapisują specyfikacje problemów, opisy sposobów rozwiązania, postaci poleceń w języku SQL.

Środki dydaktyczne

Uczniowie wykorzystują w czasie tych zajęć:

- podręcznik, np.: E. Gurbiel, G. Hardt-Olejniczak, E. Kołczyk, H. Krupicka, M. M. Sysło, *Informatyka, zakres rozszerzony*, WSiP, Warszawa 2004.
- materiały dotyczące baz danych, opracowane w projekcie Informatyka +;
- komputer i jego podstawowe oprogramowanie, w tym oprogramowanie MS SQL2008 R2 Express Edition oraz MS SQL Server Management Studio.

Komentarz. Oprogramowanie MS SQL Server 2008 R2 Express Edition jest darmowe; można je pobrać ze strony <http://www.microsoft.com/express/Database>.

8. Przebiegu zajęć (kolejnych lekcji)

Zajęcia, których celem jest wprowadzenia reguł i ograniczeń, zapewniających integralność danych, mogą być rozłożone na trzy kolejne lekcje. W niniejszej propozycji przyjęliśmy następujący tryb realizacji omawianego tematu:

- na początku, nauczyciel objaśnia istotę problemów związanych z zapewnieniem integralności danych oraz wyjaśnia rolę reguł i ograniczeń definiowanych w bazach danych (Lekcja 1);
- następnie uwaga zostaje skupiona na omówienie przykładowego fragmentu bazy danych i identyfikacji problemów, które powinny zostać rozwiązane (Lekcja 1);
- uczniowie przystępują do tworzenia bazy danych i przykładowych tabel (Lekcja 1);
- następnie uczniowie zajmują się kolejnymi problemami, dla których należy zdefiniować odpowiednie ograniczenia (Lekcja 2);

- kolejnym etapem jest zapoznanie się ze składnią poleceń języka SQL, które będą wykonywane przy definiowaniu reguł i ograniczeń (Lekcja 2);
- w ostatnim etapie uczniowie powinni wykonać odpowiednie polecenia, przygotowane na etapach wcześniejszych, i testować ich działanie (Lekcja 3);
- rozwiązywanie każdego z zagadnień składa się z sześciu etapów, wyżej wymienionych (w Metodach pracy).

Lekcja 1. Tworzenie przykładowej bazy danych i definiowanie problemów związanych z integralnością danych. Czas: 30 min.

W tej części lekcji należy przedstawić przykładowe tabele, które uczniowie utworzą w swojej bazie danych oraz omówić, jakie problemy związane z integralnością danych mogą wystąpić.

Osoby			
	Column Name	Data Type	Allow Nulls
	idosoby	int	<input type="checkbox"/>
	Nazwisko	varchar(64)	<input type="checkbox"/>
	Imie	varchar(32)	<input type="checkbox"/>
	Pesel	char(11)	<input type="checkbox"/>
	DataUrodzenia	date	<input type="checkbox"/>
	CzyKobieta	bit	<input type="checkbox"/>
	idmiasta	int	<input type="checkbox"/>
			<input type="checkbox"/>

Rysunek 1. Tabela Osoby

Miasta			
	Column Name	Data Type	Allow Nulls
	idmiasta	int	<input type="checkbox"/>
	Nazwa	varchar(64)	<input type="checkbox"/>
			<input type="checkbox"/>

Rysunek 2. Tabela Miasta

Na rysunkach 1 i 2 przedstawiono przykładowe tabele, w których dla poszczególnych kolumn określono typ danych, zdefiniowano klucze podstawowe (kolumna idosoby w tabeli Osoby i kolumna idmiasta w tabeli Miasta). Po omówieniu przeznaczenia przykładowych tabel uczniowie powinni przystąpić do tworzenia własnej bazy danych i w niej tabel, pokazanych na rysunkach 1 i 2.

Do realizacji zadania uczniowie wykorzystują SQL Server Management Studio. Każdy uczeń wykonuje polecenie:

CREATE DATABASE *Nazwiskolmie*

Tworzenie tabel można realizować poprzez napisanie polecenia w języku SQL. Dla tabeli Osoby polecenie to miałyby następującą postać;

```

CREATE TABLE Osoby (
    [idosoby] [int] IDENTITY(1,1) NOT NULL,
    [Nazwisko] [varchar](64) NOT NULL,
    [Imie] [varchar](32) NOT NULL,
    [Pesel] [char](11) NOT NULL,
    [DataUrodzenia] [date] NOT NULL,
    [CzyKobieta] [bit] NOT NULL,
    [idmiasta] [int] NOT NULL,
CONSTRAINT [PK_Osoby] PRIMARY KEY CLUSTERED
(
    [idosoby] ASC
),

```

lub z wykorzystaniem okienkowego projektanta tabel, uruchamianego z poziomu SQL Server Management Studio.

Można poprosić uczniów, by postarali się wskazać, jakie problemy związane z integralnością danych, mogą wystąpić w omawianym przykładzie. W dalszej części lekcji będziemy rozwiązywać następujące problemy:

- numer Pesel powinien być unikatowy w tabeli, chociaż nie jest kluczem podstawowym (klucz potencjalny);
- nazwa miasta w tabeli Miasta również powinna przyjmować wartości unikatowe;
- zadeklarowany typ char(11) dla numeru Pesel dopuszcza jako poprawne wszystkie ciągi znakowe o długości nie większej niż 11 znaków a powinno to być dokładnie 11 cyfr;
- kolumna idmiasta w tabeli Osoby może przyjmować tylko te wartości, które występują w tabeli miasta;
- sześć pierwszych cyfr numeru Pesel musi być zgodne z data urodzenia;
- przedostatnia cyfra numeru Pesel musi być zgodna z płcią danej osoby (kobiety – cyfra parzysta, mężczyźni – cyfra nieparzysta);
- nie można usuwać wierszy z tabeli Miasta, jeżeli istnieje choć jedna osoba powiązana z tym miastem (wartość kolumny id miasta).

Z tych przykładów powinna wypłynąć konkluzja, iż proces zapewnienia integralności danych jest złożony, a pierwszym etapem rozwiązania problemu jest wyszukanie zagrożeń.

Lekcja 2. Definiowanie ograniczeń. Czas: 60 min.

Zajęcia powinny rozpocząć się od przypomnienia problemów, jakie zidentyfikowano w przykładowych tabelach na pierwszej lekcji. Rozwiązanie kolejnych problemów będzie sprowadzało się do zdefiniowania odpowiednich ograniczeń, które zapewnią wymuszenie reguł gwarantujących integralność danych w omawianych tabelach.

Ograniczenie UNIQUE

Ograniczenie typu UNIQUE zapewnia wymuszenie unikalności danych w obrębie tabeli dla kolumn nie będących kluczem podstawowym. Ograniczenia dla danej tabeli definiujemy w języku SQL poleceniem **ALTER TABLE**. Przykładowe polecenia wymuszające unikalność numeru Pesel będzie miało postać:

```

ALTER TABLE Osoby
ADD CONSTRAINT UnikalnoscPesel UNIQUE NONCLUSTERED (pesel)

```

Polecenie dodaje do bazy danych ograniczenie typu UNIQUE o nazwie UnikalnoscPesel, które będzie powiązane z tabelą Osoby i wymusi przechowywanie w tabeli wierszy o unikatowych wartościach numeru Pesel. Przy próbie modyfikacji wiersza lub zapisania nowego z wartością Pesel, która już w tabeli istnieje, zostanie wygenerowany następujący komunikat o błędzie:

Problem zapewnienia unikalności nazwy miasta w tabeli miasta rozwiązujemy podobnie a przykładowe polecenie może mieć następującą postać:

```
ALTER TABLE Miasta
ADD CONSTRAINT UnikalnoscNazwyMiastal UNIQUE NONCLUSTERED (nazwa).
```

```
Error Message: Violation of UNIQUE KEY constraint 'UnikalnoscPesel'.
Cannot insert duplicate key in object 'dbo.Osoby'.
The statement has been terminated.
```

Ograniczenie CHECK

Ograniczenie typu CHECK jest wyrażeniem logicznym, które testuje modyfikowane dane, jeżeli wyrażenie w wyniku daje wartość false, to modyfikacja nie może być wykonana i generowany jest odpowiedni błąd. Ograniczenia dla danej tabeli definiujemy w języku SQL poleceniem ALTER TABLE. Przykładowe polecenia, wymuszające by numer Pesel składał się dokładnie z jedenastu cyfr, będzie miało postać:

```
ALTER TABLE Osoby
ADD CONSTRAINT PeselWarunek1
CHECK (pesel like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]').
```

Polecenie dodaje do bazy danych ograniczenie typu CHECK o nazwie PeselWarunek1, które będzie powiązane z tabelą Osoby i wymusi zapisywanie w tabeli wartości numeru Pesel zgodne z podanym wzorcem (11 cyfr). Przy próbie modyfikacji wiersza lub zapisania nowego z wartością Pesel, która nie spełnia zdefiniowanego ograniczenia, zostanie wygenerowany następujący komunikat o błędzie:

Ograniczenie PeselWarunek1 odnosi się jedynie do samej wartości tej kolumny, niezależnie od wartości danych w pozostałych kolumnach. Należy pamiętać, że numer Pesel musi być zgodny z datą urodzenia i płcią danej osoby. W celu zapewnienia tej zgodności należy zdefiniować kolejne ograniczenie:

```
Error Message: The UPDATE statement conflicted with the CHECK
constraint "PeselWarunek1". The conflict occurred in database
"Przyklady", table "dbo.Osoby", column 'Pesel'.
```

```
ALTER TABLE Osoby
ADD CONSTRAINT PeselWarunek2
CHECK (
    DataUrodzenia=cast('19'+substring(pesel,1,6) as date) and
CASE
    WHEN CzyKobieta=1 and substring(pesel,10,1) % 2=0 THEN 1
    WHEN CzyKobieta=0 and substring(pesel,10,1) % 2=1 THEN 1
    ELSE 0
END=1
)
```

Polecenie dodaje do bazy danych ograniczenie typu CHECK o nazwie PeselWarunek2, które będzie powiązane z tabelą Osoby i wymusi zapisywanie w tabeli wartości numeru Pesel zgodne z datą urodzenia i płcią danej osoby. Przy próbie modyfikacji wiersza lub zapisania nowego z wartością Pesel, która nie spełnia zdefiniowanego ograniczenia, zostanie wygenerowany następujący komunikat o błędzie:

Error Message: The UPDATE statement conflicted with the CHECK constraint "PesleWarunek2". The conflict occurred in database "Przykłady", table "dbo.Osoby".

Ograniczenie FOREIGN KEY

Ograniczenie typu FOREIGN KEY definiuje powiązanie klucza obcego tabeli z wartością klucza podstawowego innej tabeli. W omawianym przykładzie kluczem obcym jest kolumna idmiasta w tabeli Osoby, a powinna zostać powiązana z kluczem podstawowym idmiasta w tabeli Miasta. Zdefiniowanie ograniczenia FOREIGN KEY zapewnia, że wartość idmiasta w tabeli Osoby zawsze znajdzie odpowiadający jej wiersz w tabeli Miasta. Ograniczenia dla danej tabeli definiujemy w języku SQL poleceniem ALTER TABLE. Przykładowe polecenia, definiujące ograniczenie klucza obcego (integralność referencyjna), ma postać;

```
ALTER TABLE Osoby
ADD CONSTRAINT KluczObcyIdMiasta
FOREIGN KEY(idmiasta)
REFERENCES Miasta (idmiasta)
```

Polecenie dodaje do bazy danych ograniczenie typu FOREIGN KEY o nazwie KluczObcyIdMiasta, które będzie powiązane z tabelą Osoby i wymusi przyjmowanie tylko takich wartości dla kolumny idmiasta, które mają odpowiednik w tabeli Miasta. Przy próbie wykonania dowolnej operacji, która nie spełnia tego ograniczenia, zostanie wygenerowany następujący komunikat o błędzie:

Error Message: The DELETE statement conflicted with the REFERENCE constraint "FK_Osoby_Miasta". The conflict occurred in database "Przykłady", table "dbo.Osoby", column 'idmiasta'.

Komentarz: W opisanym przykładzie zgodność daty urodzenia z numerem Pesel jest zapewniona dla osób urodzonych w XX wieku, można zaproponować uczniom rozbudowę ograniczenia CHECK w taki sposób, żeby sprawdzało także poprawność dla osób urodzonych w XXI wieku.

Lekcja 3. Testowanie rozwiązań problemów integralności danych.

Czas: 30 min.

Po zdefiniowaniu ograniczeń dla przykładowych tabel uczniowie powinni zademonstrować ich działanie. W tym celu powinni przygotować zestaw poleceń modyfikujących dane (INSERT, UPDATE i DELETE) i zademonstrować reakcje bazy danych na próby wprowadzenia zmian, które są niezgodne z założonym poziomem integralności danych. Należy omówić z uczniami znaczenie problemów integralności danych w rzeczywistych systemach. Uczniom szczególnie zainteresowanym omawianymi problemami można zaproponować stworzenie ograniczenia, które sprawdzi poprawność liczby kontrolnej numeru Pesel. Opis algorytmu obliczającego liczbę kontrolną można znaleźć na stronie:

<http://www.mswia.gov.pl/portal/pl/381/32>