**Hurtownie danych – Spr. 1.**

PWr. WIZ, Data: 7.03.2022

|  |  |  |
| --- | --- | --- |
| Student | Email: 256305@student.pwr.edu.pl | Ocena |
| Indeks | 256305 |  |
| Imię | Grzegorz |
| Nazwisko | Dzikowski |

Zestaw składa się z 1 zadania. Jeżeli nie potrafisz rozwiązać zadania, to próbuj podać, chociaż częściowe rozwiązanie lub uzasadnienie przyczyny braku rozwiązania. Pamiętaj o podaniu nr. indeksu oraz imienia i nazwiska.

SZBD: **MS SQL 2017 lub 2019**

Baza danych: **?**

Zad. 1.

Proszę przeanalizować konceptualny model danych „Usługi” (Rys. 1.), który jest niekompletny, ale klasy i relacje miedzy nimi mogą reprezentować rozpatrywany wycinek rzeczywistości. Następnie należy wykonać następujące zadania:

1. Zweryfikować model danych w kontekście podanego zbioru reguł i ograniczeń dziedzinowych modyfikując zbiór reguł i ograniczeń (uzupełniając ich definicję) lub korygując konceptualny model danych (uzasadniając!)
2. Przedstawić uzupełnioną i poprawioną wersję modelu danych (kompletny diagram klas UML)
3. Utworzyć logiczny/fizyczny model danych w postaci skryptu w języku DDL SQL (uwzględniając reguły i ograniczenia dziedzinowe), starając się zachować zgodność ze standardem języka SQL (pomijając, o ile to możliwe, natywne konstrukcje implementacji języków SQL)
4. Utworzyć bazę danych w systemie MS SQL 2017 lub 2019, która jest fizycznym modelem danych modelowanego wycinka rzeczywistości
5. Wprowadzić klika rekordów do każdej tabeli sprawdzając poprawność implementacji (zarówno poprawne dane, jak i niezgodne z obowiązującymi regułami – komentując i wyjaśniając uzyskane komunikaty z systemu SZBD)

Reguły i graniczenia dziedzinowe:

Reg/01 – klient może wielokrotnie robić zakupy w tym samym sklepie

Reg.02 – W sklepie może robić zakupy dowolny klient

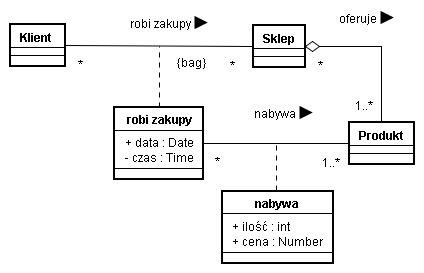
Reg.03 – Każdy zakup realizowany jest przez klienta w sklepie w określonym dniu i godzinie

Reg/04 – Sklep musi oferować co najmniej jeden produkt

Reg/05 – Ten sam produkt (rodzaj) może być oferowany przez wiele sklepów

Reg/06 – Każdy sklep może indywidualnie zaproponować cenę i ilość oferowanego produktu

Reg/07 - …



Rysunek 1. Konceptualny model danych „Usługi”

**Rozwiązanie:**

Reg/01 – Klient może wielokrotnie robić zakupy w tym samym sklepie

Reg/02 – W sklepie może robić zakupy dowolny klient

Reg/03 – Każdy zakup realizowany jest przez jednego klienta w sklepie w określonym dniu i godzinie

Reg/04 – Każdy sklep musi oferować co najmniej jeden produkt

Reg/05 – Ten sam produkt (rodzaj) może być oferowany przez wiele sklepów

Reg/06 – Każdy sklep może indywidualnie zaproponować cenę i ilość oferowanego produktu

Reg/07 – Każdy zakup musi posiadać przynajmniej jeden produkt w ilości większej od zera

Reg/08 – Data każdego zakupu nie może być w przyszłosci

Reg/09 – Jeden klient może robić zakupy w różnych sklepach

Reg/10 - Każdy klient musi posiadać typ (osoba fizyczna, firma lub instytucja) oraz posiadać nazwę (max 255 znaków), adres dostawy (max 1028 znaków) i numer telefonu (max 30 znaków)

Reg/11 – Każdy klient może posiadać email (max 255 znaków)

Reg/12 – Każdy sklep musi posiadać nazwę (max 255 znaków), adres (max 1028 znaków) oraz numer telefonu (max 30 znaków)

Reg/13 – Każdy produkt musi zawierać unikalny numer (liczba), nazwę (max 255 znaków), cenę (liczba zmiennoprzecnikowa), ilość (liczba) oraz kategorię

Reg/14 – Każdy produkt może posiadać długość (liczbę), szerokość (liczbę), głębokość (liczbę) w cm i masę w gramach (liczbę)

Reg/14 – Każda kategoria posiada unikalny numer (liczbę) i nazwę (max 255 znaków)

Reg/15 – Każda nazwa kategorii jest unikalna

Diagram

Description automatically generated

Diagram

Description automatically generated

CREATE TABLE "Typy Klientow" (

nazwa\_typu varchar(100) NOT NULL,

PRIMARY KEY (nazwa\_typu));

CREATE TABLE Kategorie (

IdKat INTEGER NOT NULL PRIMARY KEY IDENTITY(1,1),

nazwa varchar(255) NOT NULL UNIQUE);

CREATE TABLE Klienci (

IdKli INTEGER NOT NULL PRIMARY KEY IDENTITY(1,1),

nazwa varchar(255) NOT NULL,

adres varchar(1028) NOT NULL,

email varchar(255),

nr\_telefonu int NOT NULL,

typ\_klienta varchar(100) NOT NULL,

FOREIGN KEY(typ\_klienta) REFERENCES "Typy Klientow"(nazwa\_typu));

CREATE TABLE Sklepy (

IdS INTEGER NOT NULL PRIMARY KEY IDENTITY(1,1),

nazwa\_sklepu varchar(255),

lokalizacja varchar(1028),

nr\_telefonu int);

CREATE TABLE "Zrobione Zakupy" (

IdZakup INTEGER NOT NULL PRIMARY KEY IDENTITY(1,1),

IdKlient int NOT NULL,

IdSklep int NOT NULL,

data\_zakupu date NOT NULL,

godzina\_zakupu time NOT NULL,

FOREIGN KEY(IdKlient) REFERENCES Klienci(IdKli),

FOREIGN KEY(IdSklep) REFERENCES Sklepy(IdS));

CREATE TABLE Produkty (

SklepIdS int NOT NULL,

IdP INTEGER NOT NULL PRIMARY KEY IDENTITY(1,1),

nazwa\_produktu varchar(255) NOT NULL,

ilosc int NOT NULL,

cena float(10) NOT NULL,

wysokosc int,

szerokosc int,

glebokosc int,

masa int,

Kategoria int NOT NULL,

FOREIGN KEY(SklepIdS) REFERENCES Sklepy(IdS),

FOREIGN KEY(Kategoria) REFERENCES Kategorie(IdKat));

CREATE TABLE "Produkty W Zakupie" (

IdN INTEGER NOT NULL PRIMARY KEY IDENTITY(1,1),

ProduktyIdP int NOT NULL,

ZakupId int NOT NULL,

ilosc\_produktu int,

cena\_jednostkowa float(10),

FOREIGN KEY(ProduktyIdP) REFERENCES Produkty(IdP),

FOREIGN KEY(ZakupId) REFERENCES "Zrobione Zakupy"(IdZakup));

CREATE UNIQUE INDEX Klienci\_IdKli

ON Klienci (IdKli);

CREATE UNIQUE INDEX Sklepy\_IdS

ON Sklepy (IdS);

CREATE UNIQUE INDEX Produkty\_IdP

ON Produkty (IdP);

CREATE UNIQUE INDEX Kategorie\_IdKat

ON Kategorie (IdKat);

CREATE UNIQUE INDEX "Typy Klientow\_nazwa\_typu"

ON "Typy Klientow" (nazwa\_typu);

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Insert w bazę danych z niewłaściwymi danymi (NULL) nie działa

Graphical user interface, text, application, email

Description automatically generated

Niewłaściwy typ klienta wyrzuca błąd constaintów

Graphical user interface, text, application, email

Description automatically generated

**Wnioski:**

Projekt bazy danych, która wykonujemy na Hurtownie Danych, różnią się od tych na poprzednich kursach, głównie poprzez skupienie na proces, nie na dane. Musiałem trochę inaczej podejść do projektowania systemu w tym zadaniu.

Dialekt MSSQL ma kilka różnic w stosunku do MySQL, np. zamiast AutoIncrement – IDENTITY. Dodatkowo, Management jest dużo potężniejszym narzędziem niż wszystkie inne, które kiedykolwiek używałem. Miałem problemy z połączeniem z MSSQL z maszyny wirtualnej, więc pewnie w przyszłości bym postawił lokalną kopię, ale na potrzeby zajęć pozostawie go na maszynie.

*Uwaga:*

* Sprawozdanie bez wniosków końcowych nie będzie sprawdzane i tym samym ocena jest negatywna!
* Plik ze sprawozdaniem powinien mieć nazwę **Spr01HD-index-Nazwisko - 2022** i format **pdf**
* Oprogramowanie MS SQL 2017 lub 2019
* Model konceptualny danych należy przygotować wykorzystując narzędzie wspierające język UML np. Visual Paradigm