

Bartosz Łyskanowski

Indeks: 198051

Grupa: 7a

Kod Terminu:

Temat projektu

34. Sklep - towary, dostawcy, finanse.

Opis projektu

Stworzenie bazy danych do utrzymywania informacji o sytuacji w sklepie: dostarczanych towarach od dostawców, klientach kupujących te towary i zatrudnionych pracownikach. Tworzona w celu lepszej organizacji danych i możliwości podejmowania lepszych decyzji przez właściciela sklepu na podstawie zestawień wykonanych z tej bazy danych.

Szczegółowy opis projektu

Klientem jest właściciel sklepu, który zlecił wykonanie tej bazy danych.

Użytkownicy bazy danych: właściciel sklepu, pracownicy - szczególnie sprzedawcy i klienci sklepu.

Opis funkcjonalności bazy danych:

Klient wymaga by w bazie danych znalazły się informacje o produktach (towarach), które są dostarczane do sklepu przez dostawców tirmi. Każdy produkt musi mieć przypisaną cenę, jak i jego liczbę na magazynie (by właściciel wiedział, kiedy domawiać większą ilość danego towaru). Zbiór encji "produkty" ma tym samym wspierać sprzedawców, którzy wykorzystują informacje o cenie produktów nabijając paragony fiskalne. Taki paragon to dowód transakcji, który również ma być dokumentowany, by właściciel mógł obliczyć swoje zyski, bądź straty, gdy nie wszystkie produkty się sprzedają.

Sklep posiada również swoją aplikację, do której mogą rejestrować się klienci, dane takich użytkowników mają również być przechowywane w tej bazie danych. Gdy klient zeskanuje kartę lojalnościową przy transakcji, to dana ta ma również trafić do bazy i powinna nabić się do transakcji 5% zniżka. Klienci mogą oglądać jakie towary są w sklepie dostępne w jakich cenach. Właściciel sklepu chce również wiedzieć od jakich dostawców i kiedy zamawia dane produkty oraz mieć nadzór nad jakością dostaw.

Przykładowe zapytania w języku naturalnym do bazy danych:

Właściciel chce wiedzieć, z której firmy zamówił najwięcej produktów typu "pieczywo".

Klient chce zobaczyć w aplikacji ile pieniędzy wydał łącznie na zakupy w sklepie przez ostatni miesiąc.

Właściciel sprawdza ile transakcji wykonali pracownicy o pozycji "kierownik".

Właściciel chce wiedzieć, czy istniały dostawy, których koszt dostawy przekraczał koszt towarów i zostały oznaczone jako nie w pełni poprawne.

Kierownik sprawdza, czy któregoś typu towaru brakuje na magazynie (ilość na magazynie = 0).

Właściciel zarządza stałe obniżenie cen produktów o typie "warzywa" o 10%, które znajdują się na liście pięciu najmniej przychodowych produktów.

Właściciel chce wiedzieć jaki procent transakcji wykonują niezarejestrowani klienci (Klucz obcy IS NULL w Transactions)

Scenariusze użycia :

1. Obsługa klienta przy kasie:

Pracownik : wyszukaj cenę zeskanowanego produktu

BD: wyszukaj ten produkt i dodaj do transakcji produkt i odpowiednią liczbę ile sztuk zostało kupione (proces się powtarza do zeskanowania wszystkich produktów)

Klient: Skanuje kartę lojalnościową.

BD: Zaznacz, że dany klient dokonał tej transakcji i nabij 5% zniżkę.

Pracownik: Zakończ transakcję i dodaj do bazy.

BD: Wstawia odpowiednie dane do "Transakcje" i "Transaction_products".

2.Właściciel się nudzi i sprawdza, czy pracownicy pracują.

Właściciel: sprawdza, czy pracownicy wypracowali normę tygodniową.

BD: Liczy różnicę czasu pracy, grupuje i sumuje ją i zwraca listę pracowników, u których suma jest mniejsza niż planowany tygodniowy wymiar pracy.

3.Pracownik przyjmuje dostawę towaru

Pracownik: Zaakceptuj dostawę i umieść odpowiedni komentarz.

BD: Ustawia, że dostawa przybyła i wstawia w odpowiednie pola komentarze pracownika.

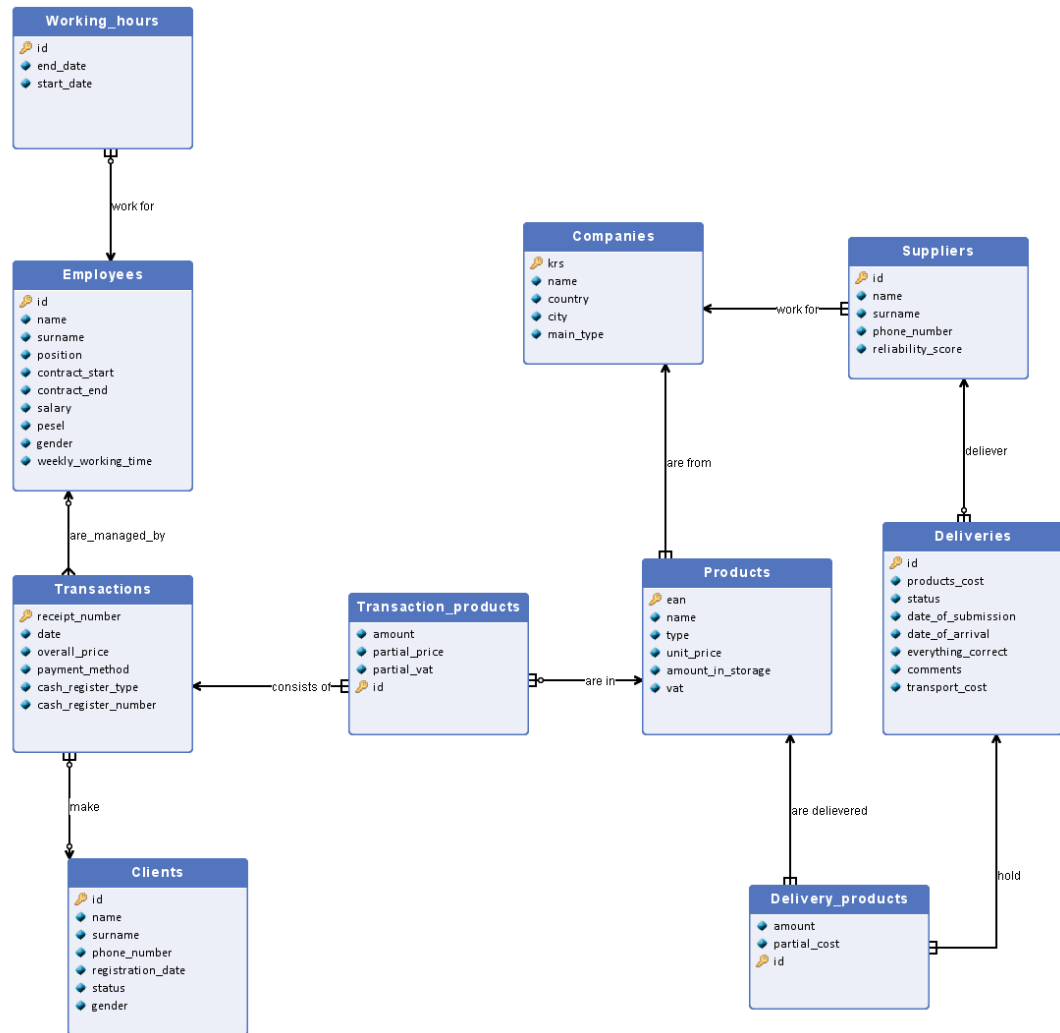
Pracownik: Zwiększ ocenę dostawcy, bo się nie spóźnił.

BD: wyszukuje tego dostawcę i odpowiednio zwiększa pole oceny dostawcy.

Wykluczenia:

W bazie nie będzie dokładnych informacji o firmach lub terminach pracy dostawców, ponieważ właściciel sklepu nie ma władzy nad nimi, więc w bazie będą znajdować się te firmy i ci dostawcy, którzy współpracują z właścicielem. Tak samo nie ma danych o niezarejestrowanych klientach.

ERD Diagram



Opis zbioru encji

Suppliers			
Zbiór dostawców, którzy dostarczają towary do sklepu z różnych firm. Może być ich do około 100. Mogą być usunięci, gdy przestaną już dostarczać towary, trzeba jednak zachować wtedy ostrożność, by usunąć odpowiednie encje z "deliveries", by zachować integralność bazy.			
Nazwa	Klucz główny	Typ/Dziedzina	Opis
id	Tak	liczba naturalna autoinkrement	nadane id każdemu dostawcy
name	Nie	string do 30 znaków	imię dostawcy
surname	Nie	string do 30 znaków	nazwisko dostawcy
phone_number	Nie	string odpowiadający prawdziwemu numerowi	numer kontaktowy do komunikacji z dostawcą
reliability_score	Nie	liczba naturalna [1,10]	ocena dostawcy na podstawie jak często się spóźnia itp.

Products			
Każdemu produktowi ze zbioru nadany jest unikalny EAN. W zbiorze znajdują się wszystkie produkty oferowane przez sklep, choć nie muszą być obecnie dostępne, wtedy ich ilość na magazynie wynosi 0. Nazwa produktu nie identyfikuje go jednoznacznie, np woda niegazowana może być z różnych firm i mieć różne ceny, stąd potrzebny EAN.			
Nazwa	Klucz główny	Typ/Dziedzina	Opis
ean	Tak	string kilkunastoznakowy (rzutowany kod kreskowy)	unikalny numer dla danego produktu European Article Number
name	Nie	string długość około 100	nazwa produktu np. Snickers
type	Nie	typ wyliczeniowy określający typ artykułu	Typ do którego dany produkt jest przypisany np. pieczywo, napój...
unit_price	Nie	liczba zmiennoprzecinkowa (0,10000) oznaczająca cenę za sztukę(z VAT)	Produkty kupowane na luz mają tu cenę na kilogram
amount_in_storage	Nie	liczba zmiennoprzecinkowa ile danego produktu jest na magazynie,	Produkty kupowane na luz mają tu liczbę kilogramów (stąd dziedziną jest liczba zmiennoprzecinkowa)
vat	Nie	procent, jaki stanowi VAT z danej ceny<0,1>	Właściciel chce wiedzieć, ile z przychodów musi oddać państwu

Deliveries

Zbiór zawiera wszystkie dostawy, które zasiliły stan sklepu. Każdej dostawie przypisane jest unikalne id. Encja trafia do zbioru w momencie złożenia zamówienia do dostawcy, a atrybuty związane z dostarczeniem do sklepu uzupełniane przy wyładunku. Można by usuwać encję po np. roku od złożenia zamówienia.

Nazwa	Klucz główny	Typ/Dziedzina	Opis
id	Tak	zbiór liter i znaków unikalny dla dostawy	kod ten jest unikalny dla każdej z dostaw
products_cost	Nie	liczba zmiennoprzecinkowa 0-200000 [zł]	ile właściciel zapłacił za produkty
status	Nie	enum : dostarczono, oczekuje, anulowano	odpowiedni status zamówienia
date_of_submission	Nie	data RRRR-MM-DD	kiedy właściciel zamówił dostawę
date_of_arrival	Nie	data RRRR-MM-DD (opcjonalny)	kiedy towar dojechał do magazynu (jak jeszcze nie dojechał to puste)
everything_correct	Nie	bool : true false	oznaczenie czy wszystko się zgadzało w dostawie
comments	Nie	łańcuch znaków do 1000 znaków (opcjonalny)	uwagi do dostaw, które np. późno dotarły, brakowało czegoś
transport_cost	Nie	liczba zmiennoprzecinkowa 0-20000 [zł]	ile właściciel zapłacił firmie za transport

Employees

W sklepie pracuje do 30 osób. Encja trafia do zbioru, gdy dana osoba zostaje zatrudniona. Osoby, które nie będą już pracować będą miały nieaktualny atrybut contract_end. Po kilku latach można byłoby usuwać niepracujących, ale wtedy razem z transakcjami, w których brali udział by zachować integralność bazy.

Nazwa	Klucz główny	Typ/Dziedzina	Opis
id	Tak	liczba naturalna autoinkrement	unikalny identyfikator pracownika (nie wszyscy mają nip lub pesel)
name	Nie	string do 30 znaków	imię pracownika
surname	Nie	string do 30 znaków	nazwisko pracownika
position	Nie	typ wyliczeniowy,np. kierownik, sprzedawca, sprzątacze,ochroniarz	pozycja pracownika w sklepie
contract_start	Nie	data w formacie RRRR- MM-DD	kiedy pracownik zaczął pracować w sklepie
contract_end	Nie	data w formacie RRRR- MM-DD	kiedy pracownikowi kończy się umowa
salary	Nie	liczba zakres 0-20000 [zł]	ile zarabia dany pracownik na miesiąc
pesel	Nie	string 11 znakowy prawdziwy pesel (opcjonalny)	pesel tylko dla pracowników z Polski (obcokrajowcy nie mają)
gender	Nie	char : M,K	M- oznacza płeć męską K kobietą
weekly_working_time	Nie	int : (20,70)	ilość godzin jaką musi przepracować pracownik przez tydzień (wcześniej umówiona)

Clients			
Zawiera zbiór wszystkich osób, które zarejestrowały się do strony sklepu i będą mogli dzięki temu mieć np. 5% tańsze zakupy przy zeskanowaniu aplikacji przy transakcji. Nie ma w tym zbiorze encji wszystkich osób, które robią zakupy w sklepie, bo niektórzy nie korzystają z konta. Gdy użytkownik się wyrejestruje to ustawiony zostanie status na "nieaktywny", później będzie można taką encję usunąć, ale dopiero gdy związane z nią transakcje się przedawnią, by nie naruszyć integralności bazy.			
Nazwa	Klucz główny	Typ/Dziedzina	Opis
id	Tak	liczba_naturalna autoinkrement	Każdemu klientowi, który zarejestrował się na stronie sklepu przypisuje się unikatowe id.
name	Nie	string do 30	imię klienta
surname	Nie	string do 30	nazwisko klienta
phone_number	Nie	string odpowiadający prawdziwemu numerowi tel. (opcjonalny)	numer telefonu podawany przy rejestracji
registration_date	Nie	timestamp format RRRR-MM-DD HH:MM:SS	data, od kiedy dany użytkownik jest zarejestrowany
status	Nie	enum : aktywny,nieaktywny	czy użytkownik nie wyrejestrował się z bazy, jeżeli nieaktywny to nie ma zniżki
gender	Nie	enum : M,K	M- oznacza płeć męską K kobiecą

Delivery_products			
Dostawa składa się z różnej ilości różnych produktów, stąd dodatkowy zbiór encji na przechowywanie ilości zakupionych produktów przez właściciela sklepu. stąd potrzeba klucza złożonego z dwóch kluczy obcych.			
Nazwa	Klucz główny	Typ/Dziedzina	Opis
id	Tak	liczba_naturalna autoinkrement	Pomocniczy klucz główny by nie używać podwójnego
amount	Nie	Liczba zmiennoprzecinkowa, ile danego towaru zamówił właściciel sklepu	Z powodu produktów na luz można kupić np. połowę produktu oznacza to wtedy pół kilo.
partial_cost	Nie	Liczba zmiennoprzecinkowa za kupioną ilość tego towaru	Tyle płaci właściciel za zakup tej ilości danego produktu

Transactions			
Zbiór wszystkich transakcji dokonanych w sklepie, zostają dodane przy wystawieniu paragonu, liczność może sięgać miliona. Można by je usuwać po np. kilku latach, gdy nie będą już potrzebne by ograniczyć liczebność, która przy powodzeniu sklepu może być bardzo duża. W sklepie występują kasy samoobsługowe, jak przystało na XXI w., więc klient może obsłużyć się sam.			
Nazwa	Klucz główny	Typ/Dziedzina	Opis
receipt_number	Tak	string na kilkanaście znaków	Unikalny numer dla każdego paragonu
date	Nie	timestamp format RRRR-MM-DD HH:MM:SS	dokładna data w której klient wykonał transakcje
overall_price	Nie	łączna cena transakcji zmiennoprzecinkowa liczba	suma jaką zapłacił klient
payment_method	Nie	typ wyliczeniowy : karta lub gotówka	sposób w jaki klient zapłacił w sklepie (przy kasie samoobsługowej można tylko kartą)
cash_register_type	Nie	typ wyliczeniowy : kasa samoobsługowa lub klasyczna	Klient może dokonać transakcji przy kasie samoobsługowej, jak i być obsłużonym
cash_register_number	Nie	typ int od 1 do ilości kas w sklepie	Numer kasy przy której została dokonana dana transakcja

Transaction_products			
Transakcja składa się z różnej ilości różnych produktów, stąd dodatkowy zbiór encji na przechowywanie ilości zakupionych produktów w trakcie transakcji. Można to identyfikować z wypisem produktów i ich ilością na paragonie fiskalnym (tylko że tu są z wszystkich wystawionych takich paragonów), stąd potrzeba klucza złożonego z dwóch kluczy obcych.			
Nazwa	Klucz główny	Typ/Dziedzina	Opis
id	Tak	liczba_naturalna autoinkrement	Pomocniczy klucz główny by nie używać podwójnego
amount	Nie	Liczba zmiennoprzecinkowa, ile danego towaru kupił klient	Z powodu produktów na luz można kupić np. połowę produktu oznacza to wtedy pół kilo.
partial_price	Nie	Liczba zmiennoprzecinkowa (Product.unit_price * amount)	Tyle płaci klient za zakup wziętej ilości danego produktu
partial_vat	Nie	Liczba zmiennoprzecinkowa	Tyle będzie wynosił VAT, za kupno tej ilości danego produktu.

		(Product.VAT * amount)	
--	--	------------------------	--

Companies			
Zbiór firm, które zajmują się dostawami towarów do naszego sklepu, identyfikowane po NIP-ie. Zbiór ten jest kluczowy, bo niektóre produkty o tej samej nazwie będą z różnych firm.			
Nazwa	Klucz główny	Typ/Dziedzina	Opis
krs	Tak	string 9 lub 14 cyfr, poprawny numer KRS firmy	Będzie ją identyfikował w zbiorze encji
name	Nie	string do 100 znaków	nazwa firmy
country	Nie	string kilkanaście znaków	nazwa państwa, w którym znajduje się siedziba firmy (niektóre produkty mogą być dostarczane spoza Polski)
city	Nie	string kilkanaście znaków	nazwa miasta, w którym znajduje się siedziba firmy
main_type	Nie	string kilkanaście znaków	typ, z którego dana firma produkuje najwięcej produktów np. pieczywo

Working_hours			
Zbiór encji zawiera godziny wejścia i wyjścia pracowników z budynku (na podstawie zeskanowania karty w momencie rozpoczęcia i zakończenia pracy). Na tej podstawie właściciel może sprawdzić, czy pracownicy wyrabiają normę, czy się spóźniają itp.			
Nazwa	Klucz główny	Typ/Dziedzina	Opis
id	Tak	numer wpisu do godzin pracy autoinkrement	sztuczny klucz do numerowania wejść
end_date	Nie	timestamp w formacie RRRR-MM-DD HH:MM:SS	dokładna data zakończenia pracy przez pracownika
start_date	Nie	timestamp w formacie RRRR-MM-DD HH:MM:SS	dokładna data rozpoczęcia pracy przez pracownika

Opis związków

Nazwa	Zbiór encji 1	Zbiór encji 2	Liczność związku	Opis
deliever	Suppliers	Deliveries	1 : 0..n	Związek przyporządkowuje dostawcy dostawy, które dostarczył do sklepu. Dostawca zawsze jest jeden, bo to on jest kierowcą ciężarówki. Mógł już wiele razy dostarczać różne towary, ale również mógł od czasu wpisania do bazy nic nie dostarczyć. Na podstawie tego związku można powiedzieć, z której firmy dotarło dane zamówienie (Zakładam, że dostawca może pracować tylko dla jednej firmy dostawczej).
are delivered	Products	Delivery_products	1 : 1..n	Związek przyporządkowuje produktom ile zostało ich dostarczonych w tej dostawie. Dostarczony produkt musi znajdować się w zbiorze sprzedawanych produktów, inaczej trzeba było by go utworzyć (by później dało się go kupić). Trzeba odpowiednio wtedy zwiększyć ilość tego produktu po dostawie.
hold	Delivery_products	Deliveries	1..n : 1	Związek przyporządkowuje dostawie zbiór wszystkich produktów z ich ilościami i ceną częściową (w dostawie musi być chociaż jeden produkt). Potrzebny by uniknąć redundacji z kupnem tego samego produktu kilka razy w jednej dostawie.

are from	Products	Companies	1..n : 1	<p>Związek przypisuje firmie produkty, które zostały przez nią wytworzone i dostarczone do sklepu. Produkt musi pochodzić ściśle z którejś z tych firm. Firma produkuje wiele towarów różnego typu i je dostarcza. W zbiorze encji Companies występują tylko firmy, które produkują dla naszego sklepu produkty, stąd brak opcjonalności. Nie ma też możliwości by obca firma produkowała produkt innej firmy. Możliwe też w ten sposób będzie ustalenie kraju pochodzenia produktu.</p>
make	Clients	Transactions	0..1 : 0..n	<p>Związek przyporządkowuje osobom zarejestrowanym, transakcje jakich dokonali w naszym sklepie (na początku klient nie będzie miał przypisanych żadnych transakcji stąd opcjonalność). Jeżeli użytkownik nie zeskanuje karty lojalnościowej sklepu przy zakupie to do zakupu nie będzie przypisanego klienta. Informacja jest potrzebna do nabicia zniżki za kartę lojalnościową.</p>
are managed by	Employees	Transactions	0..1 : 0..n	<p>Związek przyporządkowuje transakcje pracownikowi(chociaż pracownik nie musi być przypisany bo kasa może być samoobsługowa), który pracował wtedy na kasie. Pracownicy pełnią różne role (nie ma samych sprzedawców), więc nie każdy pracownik będzie przypisany jakimś transakcją. Informacja ta musi pojawić się na paragonie i gdyby doszło do jakiejś kradzieży, mamy zapisanego pracownika, który wtedy obsługiwał.</p>

consists of	Transactions	Transaction_products	1 : 1..n	Związek przyporządkowuje transakcji zbiór wszystkich produktów z ich ilościami i ceną częściową (w transakcji musi być chociaż jeden produkt). Potrzebny by uniknąć redundacji z kupnem tego samego produktu kilka razy w jednej transakcji.
are in	Transaction_products	Products	0..n : 1	Związek przyporządkowuje produktowi transakcje, w których został zakupiony (żaden klient mógł nie kupić danego towaru ze sklepu) i w jakiej ilości. Natomiast w zbiorze produkty musi wystąpić ten który zakupił klient, ponieważ w zbiorze encji products są wszystkie produkty, które oferuje sklep. Ten związek jest konieczny do powiązania produktów z transakcjami.
work for	Companies	Suppliers	1 : 1..n	Związek przyporządkowujący dostawców do danej firmy w której pracują. Zakładam, że dostawca może pracować tylko dla jednej firmy. W zbiorze przechowujemy tylko takich, których firma wystawiła do dostarczania produktów i tak samo tylko te firmy, z którymi współpracuje sklep.
work for	Employees	Working_hours	1 : 0..n	Związek przypisujący pracownika do okresu czasu, w którym pracował w sklepie. Oczywiście pracownik mógł jeszcze nie zacząć pracować w sklepie (został dopiero zatrudniony). Każdy wpis do godzin pracy oczywiście ma przypisanego pracownika, który pracuje przez ten okres czasu.

Schemat relacyjnej bazy danych

Companies(krs, name, country, city, main_type)

Suppliers(id, name, surname, phone_number, reliability_score, company_krs REF Companies)

Deliveries(id, product_cost, status, date_of_submission, date_of_arrival, everything_correct, comments, transport_cost, Supplier_id REF Suppliers)

Products(ean, name, type, unit_price, amount_in_storage, vat, company_krs REF Companies)

Delivery_products(id, delivery_id REF Deliveries, product_ean REF Products, amount, partial_cost)

Transactions(receipt_number, date, overall_price, payment_method, cash_register_type, cash_register_number, employee_id REF Employees, client_id REF Clients)

Transaction_products(id, receipt_number REF Transactions, product_ean REF Products, amount, partial_price, partial_vat)

Clients(id, name, surname, phone_number, registration_date, status, gender)

Employees(id, name, surname, position, contract_start, contract_end, salary, pesel, gender)

Working_hours(id, start_date, end_date, employee_id REF Employees)