



**POLITECHNIKA LUBELSKA
WYDZIAŁ ELEKTROTECHNIKI
I INFORMATYKI**

**KIERUNEK STUDIÓW
INFORMATYKA**

Przedmiot: Wprowadzenie do systemów baz danych

Raport z wykonania projektu pt.

Baza danych dla sprzedaży produktów marki Apple

Autor:
*Karol Trzpil
Volodymyr Tsukanov
Vasileuskaya Yana
Warhol Daniel*

Lublin, 2024

ROZDZIAŁ 1. OPIS WYBRANEGO OBSZARU RZECZYWISTOŚCI ORAZ WSKAZANIE PROBLEMU, KTÓRY ZOSTANIE ROZWIĄZANY PRZY POMOCY SYSTEMU INFORMATYCZNEGO WYKORZYSTUJĄCEGO PROJEKTOWANĄ BAZĘ DANYCH

Modelowaną rzeczywistość tworzy sklep internetowy, w którym oferowane są różnego rodzaju produkty związane z marką Apple. Sklep prowadzi globalną działalność biznesową, co oznacza, że jego klienci pochodzą z miast zlokalizowanych w krajach znajdujących się na różnych kontynentach. Ze względu na dużą różnorodność klientów, a co za tym idzie możliwą mobilność dokonują oni zakupów nie tylko w macierzystym kraju, ale również podczas pobytu za granicą. Właściciel sklepu dokonał segmentacji krajów według różnych regionów sprzedaży. Oznacza to, że każdy kraj został przypisany do jednego regionu sprzedaży.

W celu ułatwienia klientom dotarcia do określonych produktów podczas wizyty w sklepie internetowym, zostały one sklasyfikowane według różnych kategorii oraz podkategorii produktowych. Każda podkategoria jest przypisana tylko do jednej kategorii, a każdy produkt należy tylko do jednej podkategorii.

Sklep oferuje obecnie swoim klientom różne kanały sprzedaży, takie jak strona webowa (tradycyjna i mobilna) oraz aplikacja mobilna. Składając zamówienie, klient może wybrać jeden z kilku sposobów dostawy bądź odbioru produktu oraz opłacenia zamówienia. Mając swój własny profil w sklepie, klient może monitorować bieżący status jego zamówienia.

Każde zamówienie jest identyfikowane przez unikalny numer. Produkt może być sprzedany po cenie katalogowej bądź też po cenie promocyjnej, wynikającej z faktu otrzymania przez klienta rabatu na ten produkt, wyrażonego w procentach lub kwotą. W zależności od wartości zamówienia, przesyłka może być darmowa lub konieczne jest jej opłacenie przez klienta. Data wysłania zamówionych produktów jest określana indywidualnie dla każdego zamówienia.

ROZDZIAŁ 2. SFORMUŁOWANIE CELU BUDOWY SYSTEMU INFORMATYCZNEGO, WYKORZYSTUJĄCEGO PROJEKTOWANĄ BAZĘ DANYCH, ORAZ PREZENTACJA WYMAGAŃ FUNKCJONALNYCH STAWIANYCH SYSTEMOWI

Sklep internetowy „Pear”

System informatyczny nosi nazwę „Pear”. Jego głównym celem jest umożliwienie sprzedaży poprzez stronę webową, a także dostarczenie informacji na temat różnego rodzaju produktów marki Apple. System będzie także dawał możliwość zadania pytania na temat poszczególnego produktu.

Wymagania funkcjonalne dla klienta

1. Założenie konta w systemie
2. Możliwość zadania pytania na temat produktu
3. Przeglądanie listy produktów
4. Składanie zamówienia
5. Dokonanie zapłaty
6. Wybranie szczegółów dostawy
7. Podgląd informacji o produktach
8. Podgląd zamówień
9. Śledzenie zamówień
10. Dodawanie oraz usuwanie produktów w koszyku
11. Możliwość zwrotu produktów
12. Zmiana szczegółów dostawy
13. Możliwość anulowania zamówienia

Wymagania funkcjonalne dla administratora systemu

14. Logowanie do serwisu
15. Zarządzanie produktami
16. Dodawanie użytkowników systemu
17. Blokowanie dostępu do systemu
18. Odblokowanie dostępu do systemu
19. Zmiana hasła użytkownika
20. Edycja zamówienia
21. Edycja informacji o produkcie
22. Edycja kategorii i podkategorii
23. Odpowiedź na pytania klientów i pracowników

Wymagania funkcjonalne dla pracownika

- 24. Generowanie raportu ze sprzedaży
- 25. Wysyłanie wiadomości do klienta
- 26. Złożenie zamówienia w imieniu klienta
- 27. Przyjęcie zwrotu produktu
- 28. Podgląd dostępnych towarów w sklepach
- 29. Odpowiedź na pytania klientów
- 30. Zadawanie pytań administratorom



Fundusze Europejskie
Wiedza Edukacja Rozwój



**Rzeczpospolita
Polska**

Unia Europejska
Europejski Fundusz Społeczny



ROZDZIAŁ 3. SZCZEGÓŁOWY OPIS PROCESÓW REALIZOWANYCH W WYBRANEJ RZECZYWISTOŚCI, ZALEŻNOŚCI MIĘDZY NIMI ORAZ OBIEKTÓW W NICH UCZESTNICZĄCYCH

Funkcjonalność klienta

1. Założenie konta w systemie

Klient ma możliwość zarejestrowania się w serwisie podając adres email oraz opcjonalnie numer telefonu oraz ustawieniu hasła. Klient może potwierdzić rejestrację konta za pomocą linku wysłanego na e-mail bądź kodem wysłanym na numer telefonu.

Obiekty rzeczywistości: klient, administrator, konto.

2. Możliwość zadania pytania na temat produktu

Klient ma możliwość zadania pytania na temat danego produktu dotyczącego tego produktu, jego wysyłki, bądź dostępności w magazynie/sklepie.

Obiekty rzeczywistości: klient, pracownik, administrator, produkt, pytanie.

3. Przeglądanie listy produktów

Klient ma możliwość przeglądu produktów podzielonych na różnego rodzaju kategorie i podkategorie. Możliwe zawężanie grupy produktów poprzez wybór kategorii lub podkategorii lub specyfikacji.

Obiekty rzeczywistości: klient, produkt, kategorie, podkategorie.

4. Składanie zamówienia

Klient ma możliwość złożenia zamówienia produktów w koszyku na podany adres zamieszkania, bądź umówić się na odbiór w wybranym punkcie. Klient wybiera również formę płatności (kartą, gotówką przy odbiorze, raty, PayPal).

Obiekty rzeczywistości: klient, zamówienie, adres, płatność.

5. Dokonanie zapłaty

Klient może dokonać płatności za pomocą wybranego przez siebie rodzaju płatności.

Obiekty rzeczywistości: klient, zamówienie, płatność.

6. Wybranie szczegółów dostawy

Klient może wybrać szczegóły swojego zamówienia takie jak rodzaj odbioru, dostawcę spośród podanych,

Obiekty rzeczywistości: klient, zamówienie, dostawa.

7. Podgląd informacji o produktach

Klient może przeglądać informacje zamieszczone na stronie internetowej opisujące poszczególne produkty (specyfikacja, opis produktu, zawartość opakowania/produktu).

Obiekty rzeczywistości: klient, produkt.

8. Podgląd zamówień

Klient ma możliwość podglądu dokonanych przez siebie zamówień na stronie internetowej, ma możliwość sprawdzenia poprawności produktów, ilości oraz ceny.

Obiekty rzeczywistości: klient, zamówienie.

9. Śledzenie zamówień

Klient ma możliwość śledzenia postępu jego zamówienia poprzez wgląd w etapy dokonanego zamówienia, jego ostatnią lokalizację, możliwość odbioru.

Obiekty rzeczywistości: klient, zamówienie, produkt ?

10. Dodawanie oraz usuwanie produktów w koszyku

Klient ma możliwość dodawania oraz usuwania poszczególnych produktów z koszyka, zmiany ilości produktów, ostatecznie złożenia zamówienia na produkty w koszyku.

Obiekty rzeczywistości: klient, koszyk, produkty, zamówienie.

11. Możliwość zwrotu produktów

Klient ma możliwość dokonania zwrotu zamówionych przez siebie produktów po wyborze powodu zwrotu oraz wypełnieniu formularza. Zwrot musi zostać zatwierdzony przez pracownika następnie klient odzyska wydane pieniądze.

Obiekty rzeczywistości: klient, pracownik, zwrot, produkt.

12. Zmiana szczegółów dostawy

Klient ma możliwość zmiany adresu dostawy lub rodzaju odbioru jeśli przesyłka jest na odpowiednim do tego etapie.

Obiekty rzeczywistości: klient, adres, zamówienie.

13. Możliwość anulowania zamówienia

Klient ma możliwość anulowania poszczególnego zamówienia. Po anulowaniu dokonany zostanie zwrot pieniędzy jeżeli płatność została wykonana online.

Obiekty rzeczywistości: klient, zamówienie, płatność.

Funkcjonalność administratora systemu

14. Logowanie do serwisu

Administrator loguje się do systemu specjalnym kontem. Logowanie jest dwuetapowe, drugim etapem jest podanie kodu otrzymanego w wiadomości SMS. W ten sposób administrator uzyskuje pełną funkcjonalność konta.

Obiekty rzeczywistości: administrator, konto.

15. Zarządzanie produktami

Administrator ma możliwość edycji przynależności poszczególnych produktów do kategorii i podkategorii.

Obiekty rzeczywistości: administrator, produkt, kategorie, podkategorie.

16. Dodawanie użytkowników systemu

Administrator ma możliwość manualnego dodawania użytkowników do systemu w razie problemów klientów czy też zatrudnienia nowych pracowników.

Obiekty rzeczywistości: administrator, konto, klient, pracownik.

17. Blokowanie dostępu do systemu

Administrator ma możliwość zablokowania dostępu do systemu poszczególnym klientom na określony czas.

Obiekty rzeczywistości: administrator, klient.

18. Odblokowanie dostępu do systemu

Administrator może w dowolnym momencie odblokować system dla poszczególnym klientom.

Obiekty rzeczywistości: administrator, klient.

19. Zmiana hasła użytkownika

Administrator może dokonać zmiany hasła dla każdego użytkownika po ówczesnym otrzymaniu zgłoszenia problemu z hasłem.

Obiekty rzeczywistości: administrator, klient, pracownik, konto.

20. Edycja zamówienia

Administrator może dokonać edycji zamówienia jeżeli zostanie o to poproszony przez klienta. Może zmienić zamówiony produkt, ilość produktów, ???.

Obiekty rzeczywistości: administrator, zamówienie, produkt.

21. Edycja informacji o produkcie

Administrator może zmieniać informacje dotyczące produktu (cena, opis, dostępność).

Obiekty rzeczywistości: administrator, produkt.

22. Edycja kategorii oraz podkategorii

Administrator może dodawać oraz usuwać kategorie i podkategorie, edytować już istniejące kategorie i podkategorie.

Obiekty rzeczywistości: administrator, kategorie, podkategorie.

23. Odpowiedź na pytania klientów i pracowników

Administrator może odpowiadać na pytania zadawane przez klientów dotyczące produktów, może również odpowiadać na pytania zadawane przez pracowników dotyczące systemu.

Obiekty rzeczywistości: administrator, klient, pracownik, pytanie.

Funkcjonalność pracownika

24. Generowanie raportu ze sprzedaży

Pracownik ma możliwość wygenerowania raportu ze sprzedaży dokonanych w danym miesiącu.

Obiekty rzeczywistości: pracownik, produkt.

25. Wysyłanie wiadomości do klienta

Pracownik ma możliwość wysyłania wiadomości do klienta o treści dotyczącej wyłącznie sprzedaży i obsługi produktu lub strony.

Obiekty rzeczywistości: pracownik, klient, produkt, wiadomość.

26. Złożenie zamówienia w imieniu klienta

Pracownik ma możliwość złożenia zamówienia produktów wybranych przez klienta. Klient nie musi posiadać konta w systemie żeby pracownik mógł złożyć zamówienie.

Obiekty rzeczywistości: pracownik, klient, produkt.

27. Przyjęcie zwrotu produktu

Pracownik ma możliwość przyjęcia zamówionego przez klienta produktu po wypełnieniu przez klienta formularza, przedstawienia potwierdzenia zakupu tego produktu, oraz przeglądu technicznego tego produktu. Pracownik ma prawo odmówić wykonania zwrotu, jeżeli powyższe warunki nie są spełnione.

Obiekty rzeczywistości: pracownik, klient, zwrot, produkt.

28. Podgląd dostępnych towarów w sklepach

Pracownik ma możliwość przeglądu produktów znajdujących się w wybranym sklepie

Akcje: 1. Wybór sklepu (Pracownik wybiera docelowy sklep przez wyszukiwanie)

2. Przeglądanie listy dostępnych produktów w sklepie (Pracownik wybiera docelową kategorię/podkategorię i przegląda listę produktów)

Obiekty rzeczywistości: pracownik, sklep, produkt.

29. Odpowiedź na pytania klientów

Pracownik ma możliwość dodawania odpowiedzi na zadane przez klienta pytanie.

Obiekty rzeczywistości: pracownik, klient, produkt, pytanie, odpowiedź.

30. Zadawanie pytań administratorom

Pracownik może zadawać pytania związane z ogólnym systemem (wprowadzanie kategorii i podkategorii, edycje produktów) skierowanych do administratorów.

Obiekty rzeczywistości: pracownik, administrator, pytanie.

ROZDZIAŁ 4. MODEL KONCEPTUALNY PROJEKTOWANEJ BAZY DANY

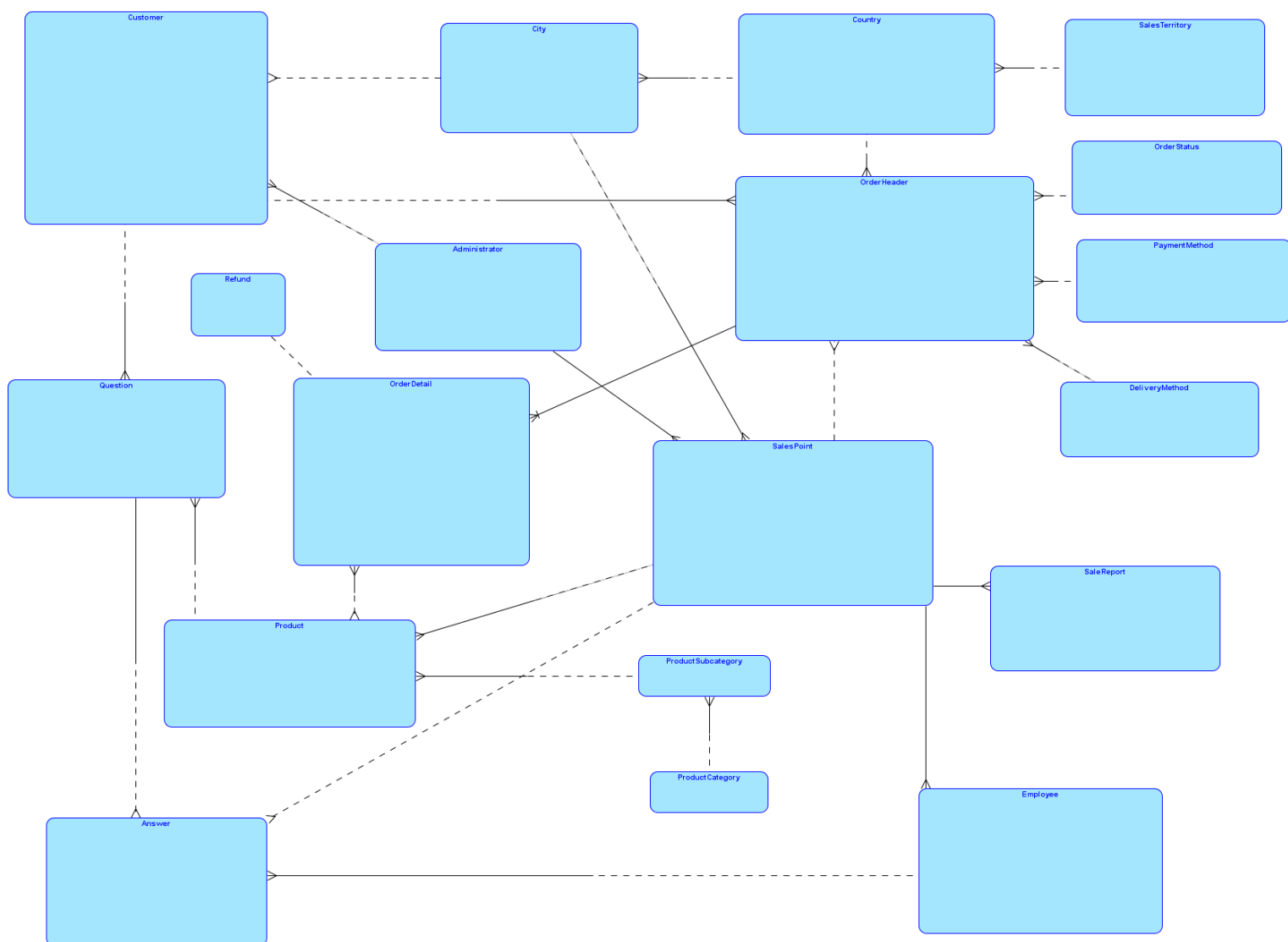


Tabela 1. Zawiera nazwy i opisy encji.

Nazwa encji	Opis encji
Customer	Informacje o użytkowniku, posiadającym konto (profil) w sklepie internetowym, obejmujące email użytkownika i hasło dostępu
City	Miasto, w którym mieszka użytkownik lub znajduje się sklep
Country	Kraj, w którym mieszka użytkownik lub znajduje się sklep
SalesTerritory	Kontynent w którym mieszka użytkownik lub znajduje się sklep
Refund	Zwrot pieniędzy, wymiana towaru
Administrator	Osoba mająca dostęp do sklepów lub danych użytkownika
OrderHeader	Dostawie na zamówieniu
OrderStatus	Status zamówienia (przesyłka w drodze, czeka na odbiór, została odebrana)
PaymentMethod	Metoda płatności (karta, gotówka)
DeliveryMethod	Metoda dostarczenia przesyłki



Question	Pytania, które może zadać użytkownik, dotyczące konkretnego produktu
OrderDetail	Informacja o zamówieniu (Procent rabatu, ilość towarów, cena itp.)
SalesPoint	Adres, numer telefonu konkretnego sklepu
SaleReport	Protokół wypełniany przez sprzedawcę po sprzedaży towaru (data, imię i nazwisko oraz dane wypełniającego (sprzedającego))
Product	Towar posiadający nazwę i kod
Answer	Odpowiedź wypełnia sprzedawca (pracownik), ale w imieniu sklepu, w którym pracuje
ProductSubcategory	Podkategoria produktu, posiada tylko nazwę
ProductCategory	Kategoria produktu, posiada tylko nazwę
Employee	Pracuje w konkretnym sklepie, odpowiada na pytania i wypełnia raporty sprzedaży

ROZDZIAŁ 5. MODEL ZWIĄZKÓW ENCJI PROJEKTOWANEJ BAZY DANYCH

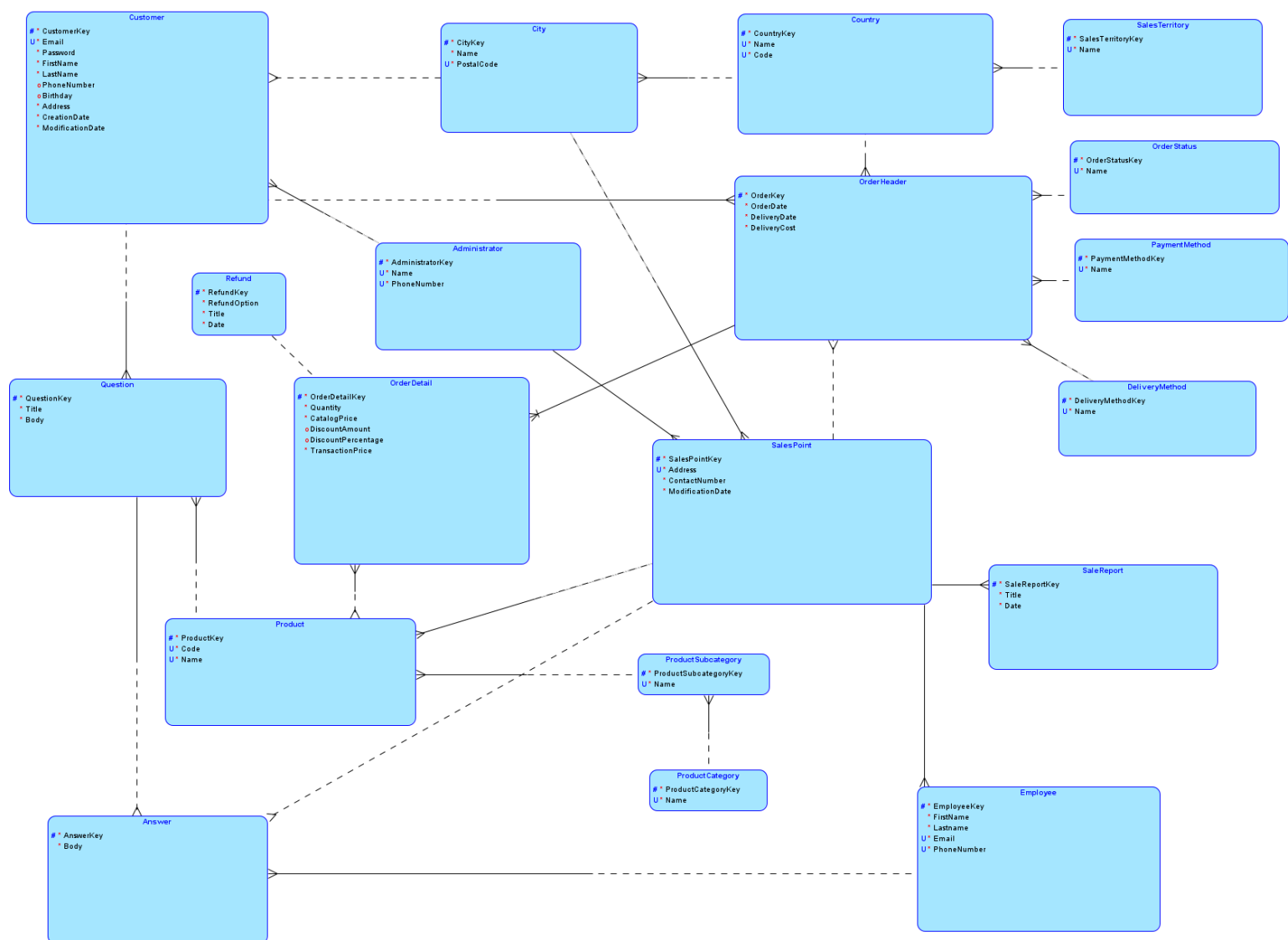


Tabela 1. Zawiera nazwy i opisy encji.

Nazwa encji	Opis encji
Customer	Informacje o użytkowniku, posiadającym konto (profil) w sklepie internetowym, obejmujące email użytkownika i hasło dostępu
City	Miasto, w którym mieszka użytkownik lub znajduje się sklep
Country	Kraj, w którym mieszka użytkownik lub znajduje się sklep
SalesTerritory	Kontynent w którym mieszka użytkownik lub znajduje się sklep
Refund	Zwrot pieniędzy, wymiana towaru
Administrator	Osoba mająca dostęp do sklepów lub danych użytkownika
OrderHeader	Dostawie na zamówieniu
OrderStatus	Status zamówienia (przesyłka w drodze, czeka na odbiór, została odebrana)
PaymentMethod	Metoda płatności (karta, gotówka)
DeliveryMethod	Metoda dostarczenia przesyłki



Question	Pytania, które może zadać użytkownik, dotyczące konkretnego produktu
OrderDetail	Informacja o zamówieniu (Procent rabatu, ilość towarów, cena itp.)
SalesPoint	Adres, numer telefonu konkretnego sklepu
SaleReport	Protokół wypełniany przez sprzedawcę po sprzedaży towaru (data, imię i nazwisko oraz dane wypełniającego (sprzedającego))
Product	Towar posiadający nazwę i kod
Answer	Odpowiedź wypełnia sprzedawca (pracownik), ale w imieniu sklepu, w którym pracuje
ProductSubcategory	Podkategoria produktu, posiada tylko nazwę
ProductCategory	Kategoria produktu, posiada tylko nazwę

Tabela 2. Zawiera nazwy i opisy atrybutów encji..

Nazwa atrybutu	Typ danych	Wartość wymagana	Wartość unikatowa	Opis atrybutu
[Customer] CustomerKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególnego klienta.
[Customer] FirstName	Ciąg znaków (50 Znaków)	Tak	Nie	Imię poszczególnego klienta.



[Customer] LastName	Ciąg znaków (50 Znaków)	Tak	Nie	Nazwisko poszczególnego klienta.
[Customer] Email	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalny adres e-mail poszczególnego klienta.
[Customer] PhoneNumber	Ciąg znaków (11 Znaków)	Nie	Nie	Opcjonalny numer telefonu poszczególnego klienta.
[Customer] Address	Ciąg znaków (80 Znaków)	Tak	Nie	Adres zamieszkania poszczególnego klienta.
[Customer] Birthday	Data	Nie	Nie	Opcjonalna data urodzenia poszczególnego klienta.
[Customer] Password	Ciąg znaków (50 Znaków)	Tak	Nie	Hasło poszczególnego klienta.
[Customer] CreationDate	Data	Tak	Nie	Data utworzenia konta poszczególnego klienta.
[Customer] ModificationDate	Data	Tak	Nie	Data modyfikacji konta poszczególnego klienta.
[City] CityKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególnego miasta.
[City] Name	Ciąg znaków (50 Znaków)	Tak	Nie	Nazwa poszczególnego miasta.
[City] PostalCode	Ciąg znaków (50 Znaków)	Tak	Tak	Kod pocztowy poszczególnego miasta.
[Country] CountryKey	Liczba całkowita (Smallint)	Tak	Tak	Unikalny identyfikator poszczególnego państwa.

[Country] Name	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalna nazwa poszczególnego państwa.
[Country] Code	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalny kod poszczególnego państwa.
[SalesTerritory] SalesTerritoryKey	Liczba całkowita (Smallint)	Tak	Tak	Unikalny identyfikator poszczególnego regionu sprzedaży.
[SalesTerritory] Name	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalna nazwa poszczególnego regionu sprzedaży.
[Administrator] AdministratorKey	Liczba całkowita (Smallint)	Tak	Tak	Unikalny identyfikator poszczególnego konta administratora.
[Administrator] Name	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalna nazwa poszczególnego konta administratora.
[Administrator] PhoneNumber	Ciąg znaków (11 Znaków)	Tak	Tak	Unikalny numer telefonu poszczególnego konta administratora.
[OrderHeader] OrderKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególnego zamówienia.
[OrderHeader] OrderDate	Data	Tak	Nie	Data złożenia zamówienia przez klienta.
[OrderHeader] DeliveryDate	Data	Tak	Nie	Przewidywana data dostawy poszczególnego zamówienia.
[OrderHeader] DeliveryCost	Liczba rzeczywista max 99999999.99	Tak	Nie	Dodatkowy koszt dostawy poszczególnego zamówienia.



[OrderStatus] OrderStatusKey	Liczba całkowita (Smallint)	Tak	Tak	Unikalny identyfikator poszczególnego statusu zamówienia.
[OrderStatus] Name	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalna nazwa poszczególnego statusu zamówienia.
[PaymentMethod] PaymentMethodKey	Liczba całkowita (Smallint)	Tak	Tak	Unikalny identyfikator poszczególnych metody płatności.
[PaymentMethod] Name	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalna nazwa poszczególnych metody płatności.
[Refund] RefundKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególnego zwrotu.
[Refund] Title	Ciąg znaków (50 Znaków)	Tak	Nie	Tytuł zwrotu napisany przez klienta.
[Refund] RefundOption	Znak	Tak	Nie	Opcja zwrotu do wyboru przez klienta.
[Refund] Date	Data	Tak	Nie	Data zgłoszenia zwrotu przez klienta.
[OrderDetail] OrderDetailKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególnego koszyka.
[OrderDetail] Quantity	Liczba całkowita max. 999	Tak	Nie	Ilość poszczególnego produktu wybranego przez klienta.
[OrderDetail] CatalogPrice	Liczba rzeczywista max. 99999999.99	Tak	Nie	Cena poszczególnego produktu wybranego przez klienta.



[OrderDetail] DiscountAmount	Liczba rzeczywista max. 99999999.99	Nie	Nie	Kwota rabatu otrzymanego na poszczególne produkty.
[OrderDetail] DiscountPercentage	Liczba rzeczywista max. 0.99	Nie	Nie	Procent rabatu z jakiego jest obliczana kwota rabatu na poszczególny produkt.
[OrderDetail] TransactionPrice	Liczba rzeczywista max. 99999999.99	Tak	Nie	Kwota całej transakcji obliczana z sumy ilości, ceny oraz rabatu poszczególnych produktów.
[DeliveryMethod] DeliveryMethodKey	Liczba całkowita (Smallint)	Tak	Tak	Unikalny identyfikator poszczególniej metody dostawy.
[DeliveryMethod] Name	Ciąg znaków (50 Znaków)	Tak	Tak	Nazwa poszczególniej metody dostawy.
[Question] QuestionKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególnego pytania.
[Question] Title	Ciąg znaków (50 Znaków)	Tak	Nie	Tytuł pytania napisany przez klienta.
[Question] Body	Ciąg znaków (250 Znaków)	Tak	Nie	Ciało pytania napisane przez klienta.
[Product] ProductKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególnego produktu.
[Product] Code	Ciąg znaków (10 Znaków)	Tak	Tak	Unikalny kod poszczególnego produktu.
[Product] Name	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalna nazwa poszczególnego produktu.



[SalesPoint] SalesPointKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególnego punktu sprzedaży.
[SalesPoint] Address	Ciąg znaków (80 Znaków)	Tak	Tak	Unikalny adres poszczególnego punktu sprzedaży.
[SalesPoint] ContactNumber	Ciąg znaków (11 Znaków)	Tak	Nie	Numer kontaktowy poszczególnego punktu sprzedaży.
[SalesPoint] ModificationDate	Data	Tak	Nie	Data modyfikacji informacji o poszczególnym punkcie sprzedaży.
[SaleReport] SaleRaportKey	Liczba całkowita (Smallint)	Tak	Tak	Unikalny identyfikator poszczególnego raportu ze sprzedaży w danym punkcie sprzedaży.
[SaleReport] Title	Ciąg znaków (50 Znaków)	Tak	Nie	Tytuł poszczególnego raportu ze sprzedaży.
[SaleReport] Date	Data	Tak	Nie	Data poszczególnego raportu ze sprzedaży
[Answer] AnswerKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególniej odpowiedzi na pytanie.
[Answer] Body	Ciąg znaków (250 Znaków)	Tak	Nie	Ciało napisane przez pracownika poszczególniej odpowiedzi na pytanie.
[ProductSubcategory] ProductSubcategoryKey	Liczba całkowita (Smallint)	Tak	Tak	Unikalny identyfikator poszczególniej podkategorii produktów.



[ProductSubcategory] Name	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalna nazwa poszczególnej podkategorii produktów.
[ProductCategory] ProductCategoryKey	Liczba całkowita (Smallint)	Tak	Tak	Unikalny identyfikator poszczególnej kategorii produktów.
[ProductCategory] Name	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalna nazwa poszczególnej kategorii produktów.
[Employee] EmployeeKey	Liczba całkowita (Integer)	Tak	Tak	Unikalny identyfikator poszczególnego konta pracownika.
[Employee] FirstName	Ciąg znaków (50 Znaków)	Tak	Nie	Imię poszczególnego pracownika.
[Employee] Lastname	Ciąg znaków (50 Znaków)	Tak	Nie	Nazwisko poszczególnego pracownika.
[Employee] Email	Ciąg znaków (50 Znaków)	Tak	Tak	Unikalny adres e-mail poszczególnego pracownika.
[Employee] PhoneNumber	Ciąg znaków (11 Znaków)	Tak	Tak	Unikalny numer telefonu poszczególnego pracownika.

Tabela 3.1 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	City → Customer
Opis	<p>Miasto nie musi mieć przypisanego Klienta.</p> <p>Każde miasto może mieć wiele klientów.</p> <p>Klient może mieć przypisane tylko jedno miasto.</p> <p>Nie każdy klient musi mieć przypisane miasto.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek opcjonalny

Tabela 3.2 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Country → City
Opis	<p>Państwo nie musi posiadać miasta.</p> <p>Państwo może posiadać wiele miast.</p> <p>Każde miasto musi należeć do jakiegoś państwa.</p> <p>Każde miasto może należeć tylko do jednego państwa.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy



Tabela 3.3 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	SalesTerritory → Country
Opis	<p>Terytorium sprzedaży nie musi posiadać państwa.</p> <p>Terytorium sprzedaży może posiadać wiele państw.</p> <p>Każde państwo musi należeć do jakiegoś terytorium sprzedaży.</p> <p>Każde państwo może należeć tylko do jednego terytorium sprzedaży.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.4 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Customer → Question
Opis	<p>Klient nie musi posiadać zapytania.</p> <p>Klient może posiadać wiele zapytań.</p> <p>Każde zapytanie musi należeć do jakiegoś klienta.</p> <p>Każde zapytanie może należeć tylko do jednego klienta.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.5 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Administrator → Customer
Opis	Administrator nie musi mieć przypisanego klienta. Administrator może posiadać wielu przypisanych klientów. Każdy klient musi należeć do jakiegoś administratora. Każdy klient może należeć do jednego administratora.
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.6 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Customer → OrderHeader
Opis	Klient nie musi posiadać zamówienia. Klient może posiadać wiele zamówień. Każde zamówienie musi należeć do jakiegoś klienta. Każde zamówienie może należeć tylko do jednego klienta.
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy



Tabela 3.7 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Country → OrderHeader
Opis	<p>Państwo nie musi posiadać zamówienia.</p> <p>Państwo może posiadać wiele zamówień.</p> <p>Każde zamówienie musi należeć do jakiegoś państwa.</p> <p>Zamówienie może posiadać tylko jedno państwo.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.8 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	OrderStatus → OrderHeader
Opis	<p>Status zamówienia nie musi posiadać zamówienia.</p> <p>Status zamówienia może posiadać wiele zamówień.</p> <p>Każde zamówienie musi posiadać jakiś status zamówienia.</p> <p>Każde zamówienie może posiadać tylko jeden status zamówienia.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy



Tabela 3.9 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	PaymentMethod → OrderHeader
Opis	<p>Metoda zapłaty nie musi posiadać zamówienia.</p> <p>Metoda zapłaty może posiadać wiele zamówień.</p> <p>Każde zamówienie musi posiadać jakąś metodę zapłaty.</p> <p>Każde zamówienie może posiadać tylko jedną metodę zapłaty.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.10 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	DeliveryMethod → OrderHeader
Opis	<p>Metoda dostawy nie musi posiadać zamówienia.</p> <p>Metoda dostawy może posiadać wiele zamówień.</p> <p>Każde zamówienie musi posiadać jakąś metodę dostawy.</p> <p>Każde zamówienie może posiadać tylko jedną metodę dostawy.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.11 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	SalesPoint → OrderHeader
Opis	<p>Punkt sprzedaży nie musi mieć przypisanego zamówienia.</p> <p>Każdy punkt sprzedaży może mieć wiele zamówień.</p> <p>Zamówienie może mieć przypisany tylko jeden punkt sprzedaży.</p> <p>Nie każde zamówienie musi mieć przypisany punkt sprzedaży.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek opcjonalny

Tabela 3.12 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	OrderHeader → OrderDetail
Opis	<p>Każde zamówienie musi posiadać jakiś koszyk.</p> <p>Zamówienie może mieć wiele koszyków.</p> <p>Każdy koszyk musi należeć do jakiegoś zamówienia.</p> <p>Każdy koszyk może należeć tylko do jednego zamówienia.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy



Tabela 3.13 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Product → OrderDetail
Opis	<p>Produkt nie musi należeć do koszyka.</p> <p>Produkt może należeć do wielu koszyków.</p> <p>Każdy koszyk musi posiadać jakiś produkt.</p> <p>Każdy koszyk może posiadać wiele produktów .</p>
Stopień	Związek binarny
Typ	N:M
Istnienie	Związek obowiązkowy

Tabela 3.14 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Refund → OrderDetail
Opis	<p>Zwrot nie musi należeć do koszyka.</p> <p>Zwrot może należeć tylko do jednego koszyka.</p> <p>Każdy koszyk może posiadać jeden zwrot.</p>
Stopień	Związek binarny
Typ	1:1
Istnienie	Związek opcjonalny



Tabela 3.15 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Product → Question
Opis	<p>Produkt nie musi posiadać zapytania.</p> <p>Produkt może posiadać wiele zapytań.</p> <p>Każde zapytanie musi należeć do jakiegoś produktu.</p> <p>Każde zapytanie może posiadać tylko jeden produkt.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.16 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Question → Answer
Opis	<p>Zapytanie nie musi posiadać odpowiedzi.</p> <p>Zapytanie może posiadać wiele odpowiedzi.</p> <p>Każda odpowiedź musi należeć do jakiegoś zapytania.</p> <p>Każda odpowiedź może posiadać tylko jedno zapytanie.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy



Tabela 3.17 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	SalesPoint → Product
Opis	<p>Punkt sprzedaży nie musi posiadać produktu.</p> <p>Punkt sprzedaży może posiadać wiele produktów.</p> <p>Każdy produkt musi należeć do jakiegoś punktu sprzedaży.</p> <p>Każdy produkt może należeć tylko do jednego punktu sprzedaży.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.18 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	ProductSubcategory → Product
Opis	<p>Podkategoria produktu nie musi posiadać produktu.</p> <p>Podkategoria produktu może posiadać wiele produktów.</p> <p>Każdy produkt musi należeć do jakiejś podkategorii produktów.</p> <p>Każdy produkt może należeć tylko do jednej podkategorii produktów.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy



Tabela 3.19 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	City → SalesPoint
Opis	<p>Miasto nie musi posiadać punktu sprzedaży.</p> <p>Miasto może posiadać wiele punktów sprzedaży.</p> <p>Każdy punkt sprzedaży musi należeć do jakiegoś miasta.</p> <p>Każdy punkt sprzedaży może należeć tylko do jednego miasta.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.20 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Administrator → SalesPoint
Opis	<p>Administrator musi mieć przypisany punkt sprzedaży.</p> <p>Administrator może posiadać wiele punktów sprzedaży.</p> <p>Każdy punkt sprzedaży musi należeć do jakiegoś administratora.</p> <p>Każdy punkt sprzedaży może należeć tylko do jednego administratora.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy



Tabela 3.21 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	ProductCategory → ProductSubcategory
Opis	Kategoria produktów nie musi posiadać podkategorii produktów. Kategoria produktów może posiadać wiele podkategorii produktów. Każda podkategoria produktów musi należeć do jakiejś kategorii. Każda podkategoria produktów może należeć tylko do jednej kategorii.
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.22 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	SalesPoint → SaleReport
Opis	Punkt sprzedaży musi posiadać raportu sprzedaży. Punkt sprzedaży może posiadać wiele raportu sprzedaży. Każdy raport sprzedaży musi należeć do jakiegoś punktu sprzedaży. Każdy raport sprzedaży może należeć tylko do jednego punktu sprzedaży.
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.23 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	SalesPoint → Employee
Opis	<p>Punkt sprzedaży nie musi posiadać pracownika.</p> <p>Punkt sprzedaży może posiadać wielu pracowników.</p> <p>Każdy pracownik musi należeć do jakiegoś punktu sprzedaży.</p> <p>Każdy pracownik może należeć tylko do jednego punktu sprzedaży.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

Tabela 3.24 Zawiera Relacje i jej opis.

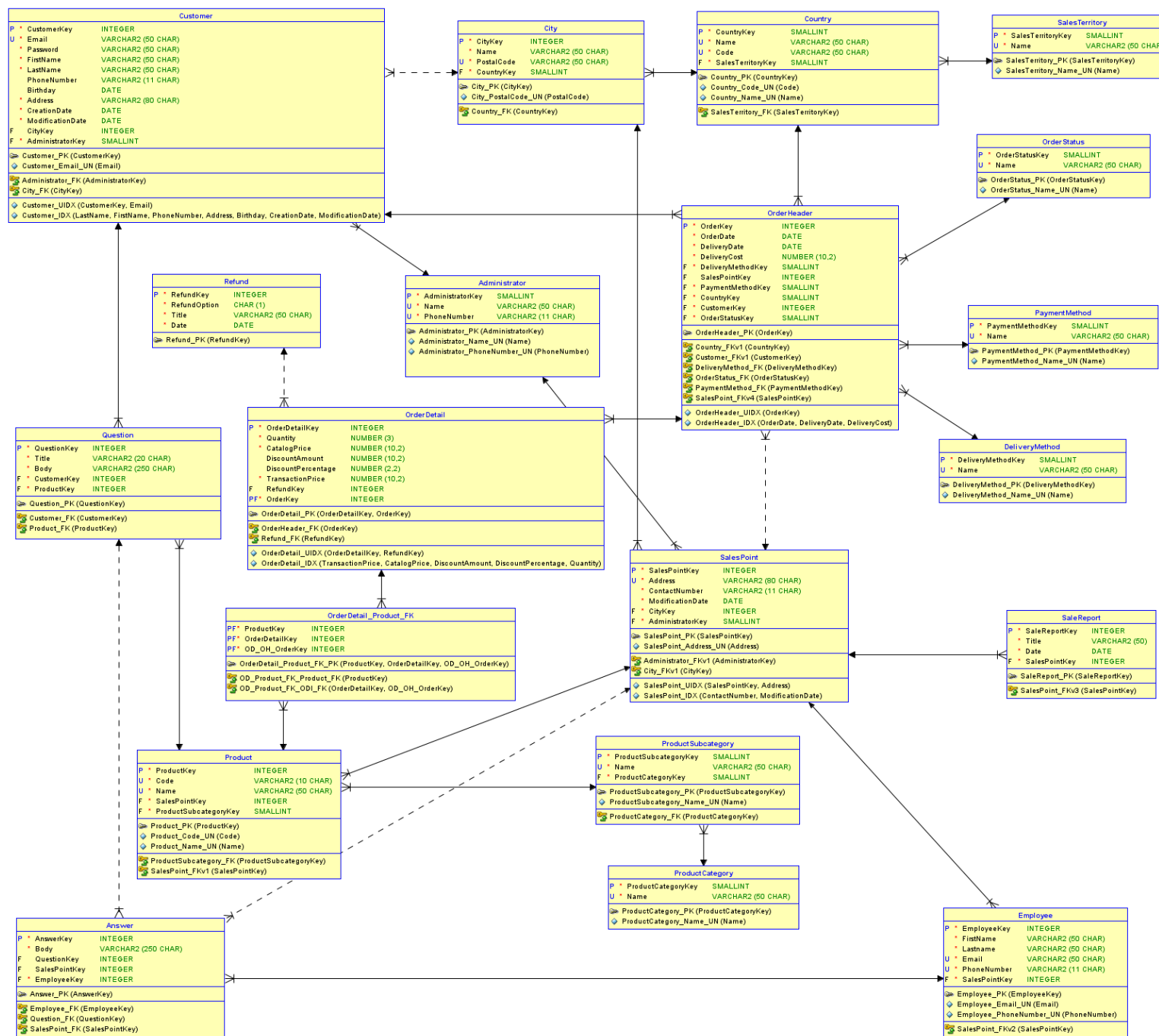
Tytuł	Opis
Nazwy encji	SalesPoint → Answer
Opis	<p>Punkt sprzedaży nie musi posiadać odpowiedzi.</p> <p>Każdy punkt sprzedaży może mieć wiele odpowiedzi.</p> <p>Każda odpowiedź może mieć przypisany tylko jeden punkt sprzedaży.</p> <p>Nie każda odpowiedź musi mieć przypisany punkt sprzedaży.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek opcjonalny



Tabela 3.25 Zawiera Relacje i jej opis.

Tytuł	Opis
Nazwy encji	Employee → Answer
Opis	<p>Pracownik nie musi posiadać odpowiedzi.</p> <p>Pracownik może posiadać wiele odpowiedzi.</p> <p>Każda odpowiedź musi należeć do jakiegoś pracownika.</p> <p>Każda odpowiedź może należeć tylko do jednego pracownika.</p>
Stopień	Związek binarny
Typ	1:M
Istnienie	Związek obowiązkowy

ROZDZIAŁ 6. MODEL RELACYJNY PROJEKTOWANEJ BAZY DANYCH



ROZDZIAŁ 7. KOD SQL – TWORZENIE BAZY DANYCH

Przedstaw instrukcje do tworzenia poszczególnych tabel bazy danych.

Przedstaw instrukcje do tworzenia więzów integralności zarówno na poziomie poszczególnych kolumn jak i całych tabel.

Przedstaw instrukcje do tworzenia związków między tabelami.

Przedstaw instrukcje do tworzenia indeksów dla poszczególnych tabel.

```
CREATE TABLE administrator (  
    administratorkey SMALLINT NOT NULL,  
    name          VARCHAR2(50 CHAR) NOT NULL,  
    phonenumber   VARCHAR2(11 CHAR) NOT NULL  
);
```

```
ALTER TABLE administrator ADD CONSTRAINT administrator_pk PRIMARY KEY ( administratorkey  
);
```

```
ALTER TABLE administrator ADD CONSTRAINT administrator_name_un UNIQUE ( name );
```

```
ALTER TABLE administrator ADD CONSTRAINT administrator_phonenumber_un UNIQUE (  
    phonenumber );
```

```
CREATE TABLE answer (  
    answerkey      INTEGER NOT NULL,  
    body           VARCHAR2(250 CHAR) NOT NULL,  
    questionkey    INTEGER,  
    salespointkey  INTEGER,  
    employeekey    INTEGER NOT NULL  
);
```

```
ALTER TABLE answer ADD CONSTRAINT answer_pk PRIMARY KEY ( answerkey );
```

```
CREATE TABLE city (  
    citykey  INTEGER NOT NULL,  
    name     VARCHAR2(50 CHAR) NOT NULL,  
    postalcode VARCHAR2(50 CHAR) NOT NULL,
```



Fundusze Europejskie
Wiedza Edukacja Rozwój



**Rzeczpospolita
Polska**

Unia Europejska
Europejski Fundusz Społeczny



```
countrykey SMALLINT NOT NULL  
);
```

```
ALTER TABLE city ADD CONSTRAINT city_pk PRIMARY KEY ( citykey );
```

```
ALTER TABLE city ADD CONSTRAINT city_postalcode_un UNIQUE ( postalcode );
```

```
CREATE TABLE country (  
    countrykey    SMALLINT NOT NULL,  
    name          VARCHAR2(50 CHAR) NOT NULL,  
    code          VARCHAR2(50 CHAR) NOT NULL,  
    salesterritorykey SMALLINT NOT NULL  
);
```

```
ALTER TABLE country ADD CONSTRAINT country_pk PRIMARY KEY ( countrykey );
```

```
ALTER TABLE country ADD CONSTRAINT country_code_un UNIQUE ( code );
```

```
ALTER TABLE country ADD CONSTRAINT country_name_un UNIQUE ( name );
```

```
CREATE TABLE customer (  
    customerkey    INTEGER NOT NULL,  
    email          VARCHAR2(50 CHAR) NOT NULL,  
    password       VARCHAR2(50 CHAR) NOT NULL,  
    firstname      VARCHAR2(50 CHAR) NOT NULL,  
    lastname       VARCHAR2(50 CHAR) NOT NULL,  
    phonenumber    VARCHAR2(11 CHAR),  
    birthday       DATE,  
    address        VARCHAR2(80 CHAR) NOT NULL,  
    creationdate   DATE NOT NULL,  
    modificationdate DATE NOT NULL,  
    citykey        INTEGER,
```



```
        administratorkey SMALLINT NOT NULL
    );
```

```
CREATE UNIQUE INDEX customer_uidx ON
    customer (
        customerkey
        ASC,
        email
        ASC );
```

```
CREATE INDEX customer_idx ON
    customer (
        lastname
        ASC,
        firstname
        ASC,
        phonenumber
        ASC,
        address
        ASC,
        birthday
        ASC,
        creationdate
        DESC,
        modificationdate
        DESC );
```

```
ALTER TABLE customer ADD CONSTRAINT customer_pk PRIMARY KEY ( customerkey );
```

```
ALTER TABLE customer ADD CONSTRAINT customer_email_un UNIQUE ( email );
```

```
CREATE TABLE deliverymethod (
```

```
    deliverymethodkey SMALLINT NOT NULL,  
    name          VARCHAR2(50 CHAR) NOT NULL  
);
```

```
ALTER TABLE deliverymethod ADD CONSTRAINT deliverymethod_pk PRIMARY KEY (  
deliverymethodkey );
```

```
ALTER TABLE deliverymethod ADD CONSTRAINT deliverymethod_name_un UNIQUE ( name );
```

```
CREATE TABLE employee (  
    employeekey  INTEGER NOT NULL,  
    firstname    VARCHAR2(50 CHAR) NOT NULL,  
    lastname     VARCHAR2(50 CHAR) NOT NULL,  
    email        VARCHAR2(50 CHAR) NOT NULL,  
    phonenumber  VARCHAR2(11 CHAR) NOT NULL,  
    salespointkey INTEGER NOT NULL  
);
```

```
ALTER TABLE employee ADD CONSTRAINT employee_pk PRIMARY KEY ( employeekey );
```

```
ALTER TABLE employee ADD CONSTRAINT employee_email_un UNIQUE ( email );
```

```
ALTER TABLE employee ADD CONSTRAINT employee_phonenumber_un UNIQUE ( phonenumber  
);
```

```
CREATE TABLE orderdetail (  
    orderdetailkey  INTEGER NOT NULL,  
    quantity        NUMBER(3) NOT NULL,  
    catalogprice    NUMBER(10, 2) NOT NULL,  
    discountamount   NUMBER(10, 2),  
    discountpercentage NUMBER(2, 2),  
    transactionprice NUMBER(10, 2) NOT NULL,  
    refundkey        INTEGER,
```



```
        orderkey          INTEGER NOT NULL
    );
```

```
CREATE UNIQUE INDEX orderdetail_uidx ON
    orderdetail (
        orderdetailkey
        ASC,
        refundkey
        ASC );
```

```
CREATE INDEX orderdetail_idx ON
    orderdetail (
        transactionprice
        ASC,
        catalogprice
        ASC,
        discountamount
        ASC,
        discountpercentage
        ASC,
        quantity
        ASC );
```

```
ALTER TABLE orderdetail ADD CONSTRAINT orderdetail_pk PRIMARY KEY ( orderdetailkey,
                                                                    orderkey );
```

```
CREATE TABLE orderdetail_product_fk (
    productkey          INTEGER NOT NULL,
    orderdetailkey INTEGER NOT NULL,
    od_oh_orderkey INTEGER NOT NULL
);
```



```
ALTER TABLE orderdetail_product_fk
    ADD CONSTRAINT orderdetail_product_fk_pk PRIMARY KEY ( productkey,
                                                         orderdetailkey,
                                                         od_oh_orderkey );
```

```
CREATE TABLE orderheader (
    orderkey          INTEGER NOT NULL,
    orderdate         DATE NOT NULL,
    deliverydate      DATE NOT NULL,
    deliverycost      NUMBER(10, 2) NOT NULL,
    deliverymethodkey SMALLINT NOT NULL,
    salespointkey     INTEGER,
    paymentmethodkey  SMALLINT NOT NULL,
    countrykey        SMALLINT NOT NULL,
    customerkey       INTEGER NOT NULL,
    orderstatuskey    SMALLINT NOT NULL
);
```

```
CREATE UNIQUE INDEX orderheader_uidx ON
    orderheader (
        orderkey
    ASC );
```

```
CREATE INDEX orderheader_idx ON
    orderheader (
        orderdate
    DESC,
        deliverydate
    DESC,
        deliverycost
    ASC );
```

```
ALTER TABLE orderheader ADD CONSTRAINT orderheader_pk PRIMARY KEY ( orderkey );
```

```
CREATE TABLE orderstatus (  
    orderstatuskey SMALLINT NOT NULL,  
    name          VARCHAR2(50 CHAR) NOT NULL  
);
```

```
ALTER TABLE orderstatus ADD CONSTRAINT orderstatus_pk PRIMARY KEY ( orderstatuskey );
```

```
ALTER TABLE orderstatus ADD CONSTRAINT orderstatus_name_un UNIQUE ( name );
```

```
CREATE TABLE paymentmethod (  
    paymentmethodkey SMALLINT NOT NULL,  
    name          VARCHAR2(50 CHAR) NOT NULL  
);
```

```
ALTER TABLE paymentmethod ADD CONSTRAINT paymentmethod_pk PRIMARY KEY (  
paymentmethodkey );
```

```
ALTER TABLE paymentmethod ADD CONSTRAINT paymentmethod_name_un UNIQUE ( name );
```

```
CREATE TABLE product (  
    productkey      INTEGER NOT NULL,  
    code            VARCHAR2(10 CHAR) NOT NULL,  
    name            VARCHAR2(50 CHAR) NOT NULL,  
    salespointkey   INTEGER NOT NULL,  
    productsubcategorykey SMALLINT NOT NULL  
);
```

```
ALTER TABLE product ADD CONSTRAINT product_pk PRIMARY KEY ( productkey );
```

```
ALTER TABLE product ADD CONSTRAINT product_code_un UNIQUE ( code );
```

```
ALTER TABLE product ADD CONSTRAINT product_name_un UNIQUE ( name );
```

```
CREATE TABLE productcategory (  
    productcategorykey SMALLINT NOT NULL,  
    name                VARCHAR2(50 CHAR) NOT NULL  
);
```

```
ALTER TABLE productcategory ADD CONSTRAINT productcategory_pk PRIMARY KEY (  
productcategorykey );
```

```
ALTER TABLE productcategory ADD CONSTRAINT productcategory_name_un UNIQUE ( name );
```

```
CREATE TABLE productsubcategory (  
    productsubcategorykey SMALLINT NOT NULL,  
    name                VARCHAR2(50 CHAR) NOT NULL,  
    productcategorykey   SMALLINT NOT NULL  
);
```

```
ALTER TABLE productsubcategory ADD CONSTRAINT productsubcategory_pk PRIMARY KEY (  
productsubcategorykey );
```

```
ALTER TABLE productsubcategory ADD CONSTRAINT productsubcategory_name_un UNIQUE (  
name );
```

```
CREATE TABLE question (  
    questionkey INTEGER NOT NULL,  
    title      VARCHAR2(20 CHAR) NOT NULL,  
    body       VARCHAR2(250 CHAR) NOT NULL,  
    customerkey INTEGER NOT NULL,  
    productkey  INTEGER NOT NULL  
);
```

```
ALTER TABLE question ADD CONSTRAINT question_pk PRIMARY KEY ( questionkey );
```

```
CREATE TABLE refund (
    refundkey      INTEGER NOT NULL,
    refundoption CHAR(1) NOT NULL,
    title    VARCHAR2(50 CHAR) NOT NULL,
    "Date"      DATE NOT NULL
);
```

```
ALTER TABLE refund ADD CONSTRAINT refund_pk PRIMARY KEY ( refundkey );
```

```
CREATE TABLE salereport (
    salereportkey INTEGER NOT NULL,
    title    VARCHAR2(50) NOT NULL,
    "Date"      DATE NOT NULL,
    salespointkey INTEGER NOT NULL
);
```

```
ALTER TABLE salereport ADD CONSTRAINT salereport_pk PRIMARY KEY ( salereportkey );
```

```
CREATE TABLE salespoint (
    salespointkey  INTEGER NOT NULL,
    address        VARCHAR2(80 CHAR) NOT NULL,
    contactnumber  VARCHAR2(11 CHAR) NOT NULL,
    modificationdate DATE NOT NULL,
    citykey        INTEGER NOT NULL,
    administratorkey SMALLINT NOT NULL
);
```

```
CREATE UNIQUE INDEX salespoint_uidx ON
    salespoint (
        salespointkey
    ASC,
    address
```

ASC);

CREATE INDEX salespoint_idx ON

salespoint (
contactnumber
ASC,
modificationdate
DESC);

ALTER TABLE salespoint ADD CONSTRAINT salespoint_pk PRIMARY KEY (salespointkey);

ALTER TABLE salespoint ADD CONSTRAINT salespoint_address_un UNIQUE (address);

CREATE TABLE salesterritory (

salesterritorykey SMALLINT NOT NULL,
name VARCHAR2(50 CHAR) NOT NULL

);

ALTER TABLE salesterritory ADD CONSTRAINT salesterritory_pk PRIMARY KEY (salesterritorykey);

ALTER TABLE salesterritory ADD CONSTRAINT salesterritory_name_un UNIQUE (name);

ALTER TABLE customer

ADD CONSTRAINT administrator_fk FOREIGN KEY (administratorkey)
REFERENCES administrator (administratorkey);

ALTER TABLE salespoint

ADD CONSTRAINT administrator_fkv1 FOREIGN KEY (administratorkey)
REFERENCES administrator (administratorkey);

ALTER TABLE customer

ADD CONSTRAINT city_fk FOREIGN KEY (citykey)



Fundusze Europejskie
Wiedza Edukacja Rozwój



**Rzeczpospolita
Polska**

Unia Europejska
Europejski Fundusz Społeczny



```
REFERENCES city ( citykey );
```

```
ALTER TABLE salespoint
```

```
ADD CONSTRAINT city_fk1 FOREIGN KEY ( citykey )
```

```
REFERENCES city ( citykey );
```

```
ALTER TABLE city
```

```
ADD CONSTRAINT country_fk FOREIGN KEY ( countrykey )
```

```
REFERENCES country ( countrykey );
```

```
ALTER TABLE orderheader
```

```
ADD CONSTRAINT country_fk1 FOREIGN KEY ( countrykey )
```

```
REFERENCES country ( countrykey );
```

```
ALTER TABLE question
```

```
ADD CONSTRAINT customer_fk FOREIGN KEY ( customerkey )
```

```
REFERENCES customer ( customerkey );
```

```
ALTER TABLE orderheader
```

```
ADD CONSTRAINT customer_fk1 FOREIGN KEY ( customerkey )
```

```
REFERENCES customer ( customerkey );
```

```
ALTER TABLE orderheader
```

```
ADD CONSTRAINT deliverymethod_fk FOREIGN KEY ( deliverymethodkey )
```

```
REFERENCES deliverymethod ( deliverymethodkey );
```

```
ALTER TABLE answer
```

```
ADD CONSTRAINT employee_fk FOREIGN KEY ( employeekey )
```

```
REFERENCES employee ( employeekey );
```

```
ALTER TABLE orderdetail_product_fk
```

```
ADD CONSTRAINT od_product_fk_odl_fk FOREIGN KEY ( orderdetailkey,
```

```
                od_oh_orderkey )  
REFERENCES orderdetail ( orderdetailkey,  
                orderkey );
```

```
ALTER TABLE orderdetail_product_fk  
    ADD CONSTRAINT od_product_fk_product_fk FOREIGN KEY ( productkey )  
    REFERENCES product ( productkey );
```

```
ALTER TABLE orderdetail  
    ADD CONSTRAINT orderheader_fk FOREIGN KEY ( orderkey )  
    REFERENCES orderheader ( orderkey );
```

```
ALTER TABLE orderheader  
    ADD CONSTRAINT orderstatus_fk FOREIGN KEY ( orderstatuskey )  
    REFERENCES orderstatus ( orderstatuskey );
```

```
ALTER TABLE orderheader  
    ADD CONSTRAINT paymentmethod_fk FOREIGN KEY ( paymentmethodkey )  
    REFERENCES paymentmethod ( paymentmethodkey );
```

```
ALTER TABLE question  
    ADD CONSTRAINT product_fk FOREIGN KEY ( productkey )  
    REFERENCES product ( productkey );
```

```
ALTER TABLE productsubcategory  
    ADD CONSTRAINT productcategory_fk FOREIGN KEY ( productcategorykey )  
    REFERENCES productcategory ( productcategorykey );
```

```
ALTER TABLE product  
    ADD CONSTRAINT productsubcategory_fk FOREIGN KEY ( productsubcategorykey )  
    REFERENCES productsubcategory ( productsubcategorykey );
```

ALTER TABLE answer

```
ADD CONSTRAINT question_fk FOREIGN KEY ( questionkey )  
REFERENCES question ( questionkey );
```

ALTER TABLE orderdetail

```
ADD CONSTRAINT refund_fk FOREIGN KEY ( refundkey )  
REFERENCES refund ( refundkey );
```

ALTER TABLE answer

```
ADD CONSTRAINT salespoint_fk FOREIGN KEY ( salespointkey )  
REFERENCES salespoint ( salespointkey );
```

ALTER TABLE product

```
ADD CONSTRAINT salespoint_fkv1 FOREIGN KEY ( salespointkey )  
REFERENCES salespoint ( salespointkey );
```

ALTER TABLE employee

```
ADD CONSTRAINT salespoint_fkv2 FOREIGN KEY ( salespointkey )  
REFERENCES salespoint ( salespointkey );
```

ALTER TABLE salereport

```
ADD CONSTRAINT salespoint_fkv3 FOREIGN KEY ( salespointkey )  
REFERENCES salespoint ( salespointkey );
```

ALTER TABLE orderheader

```
ADD CONSTRAINT salespoint_fkv4 FOREIGN KEY ( salespointkey )  
REFERENCES salespoint ( salespointkey );
```

ALTER TABLE country

```
ADD CONSTRAINT salesterritory_fk FOREIGN KEY ( salesterritorykey )  
REFERENCES salesterritory ( salesterritorykey );
```




Raport powstał podczas zajęć laboratoryjnych z przedmiotu
prowadzonego w ramach projektu
*„Zintegrowany Program Rozwoju Politechniki Lubelskiej – część
druga”*,
umowa nr **POWR.03.05.00-00-Z060/18-00**
w ramach Programu Operacyjnego Wiedza Edukacja Rozwój
2014-2020
współfinansowanego ze środków Europejskiego Funduszu
Społecznego