

Sieć sklepów elektronicznych

Baza jest relacyjną bazą danych służącą do zarządzania siecią sklepów elektronicznych, w tym zamówieniami klientów, dostawami towarów do lokalów, zatrudnieniem pracowników, administracją lokalami i pojazdami. Podczas tworzenia bazy korzystaliśmy z programu SQL Developer.

Aktorzy: Klient, prezes, kierownik, kierowca

Baza danych będzie zawierać tabele: Produkty, Pracownicy, Zamówienia, Szczegóły zamówienia, Dostawy, Szczegóły dostawy, Pojazdy, Lokale, Adres, Pracownicy_Log

Tabela **Produkty**: ID_Projektu, Nazwa Produktu, Waga, Cena, Dostępność

Tabela **Pracownicy**: ID_Pracownika, Imię, Nazwisko, Stanowisko, ID_Lokalu, Pensja, ID_Kierownika(opcjonalne)

Tabela **Zamówienia**: Numer Zamówienia, ID_Adresu, Rodzaj_dostawy, Data_Zamówienia

Tabela **Szczegóły zamówienia**: Numer_zamówienia, ID_Projektu, Ilość_zam, Cena

Tabela **Dostawy**: Numer_Dostawy, ID_Lokalu, Data_dostawy, ID_Pojazdu

Tabela **Szczegóły dostawy**: Numer_Dostawy, ID_Projektu, Ilość_dost

Tabela **Pojazdy**: ID_Pojazdu, ID_Pracownika, Rejestracja

Tabela **Lokale**: ID_Lokalu, Adres, ID_Kierownika

Tabela **Adres**: ID_Adresu, Miasto, Kod_pocztowy, Ulica, Numer_lokalu

Tabela **Pracownicy_log**: ID_Pracownika, Stara_pensja, Data_Zmiany, Akcja

Klient będzie miał możliwość:

- Przeglądania towarów, sprawdzenia jego ceny oraz dostępności.
- Złożenia zamówienia na dane towary z wyborem adresu i rodzaju dostawy.

Pracownicy będą na następujących stanowiskach: prezes, kierownik_lokalu, kierowca i ich funkcje będą zależne od stanowiska

Prezes będzie najwyższym stanowiskiem, które będzie dawać mu możliwości:

- Podglądu aktualnych zamówień i szczegółów znajdujących się w nim.
- Dodawania lub usuwania danych produktów z bazy.
- Zwalniania lub zatrudniania nowych pracowników oraz zmianę ich pensji.
- Tworzenia nowych lokalów i przypisywania do nich kierownika.
- Dodawania lub usuwania pojazdów.

Kierownik lokalów będzie miał możliwość:

- Podglądu aktualnych zamówień i szczegółów w nim znajdujących się.
- Zwalniać lub zatrudniać nowych kierowców.
- Przypisywać kierowcę do pojazdu dostawczego.

Kierowca będzie miał możliwość:

- Sprawdzenia informacji na temat przypisanej mu dostawy.

Tabela **Produkty** połączona jest z tabelą **Szczegóły zamówienia** łączeniem jeden do wielu, ponieważ jeden produkt może znajdować się w kilku zamówieniach.

Tabela **Produkty** połączona jest z tabelą **Szczegóły dostawy** łączeniem jeden do wielu, ponieważ jeden produkt może znajdować się w kilku dostawach.

Tabela **Zamówienia** jest połączona z tabelą **Szczegóły zamówienia** łączeniem jeden do wielu, ponieważ każde zamówienie, może mieć kilka produktów, każdy posiadający Szczegóły zamówienia.

Tabela **Zamówienia** jest połączona z tabelą **Adres** łączeniem jeden do jednego, ponieważ jeden adres należy do jednego zamówienia

Tabela **Adres** jest połączona z tabelą **Lokale** łączeniem jeden do jednego, ponieważ każdy lokal posiada jeden adres.

Tabela **Pracownicy** jest połączona z tabelą **Pojazdy** łączeniem zero lub jeden do jednego, ponieważ jeden kierowca, może być przypisany do jednego pojazdu, ale nie każdy pracownik jest kierowcą.

Tabela **Pojazdy** jest połączona z tabelą **Dostawy** łączeniem jeden do jednego, ponieważ jeden samochód jest odpowiedzialny za dostarczenie jednego zamówienia.

Tabela **Dostawy** jest połączona z tabelą **Szczegóły dostawy** łączeniem jeden do wielu, ponieważ jedna dostawa może posiadać kilka produktów, każdy posiadający Szczegóły dostawy.

Tabela **Lokale** jest połączona z tabelą **Dostawy** łączeniem jeden do wielu, ponieważ do jednego lokalu może przychodzić kilka dostaw.

Tabela **Lokale** jest połączona z tabelą **Pracownicy** łączeniem jeden do wielu, ponieważ w jednym lokalu może pracować kilka pracowników.

Tabele „Szczegóły zamówienia” i „Szczegóły dostawy” są tabelami łącznikowymi, aby uzyskać połączenia wiele do wielu odpowiednio w tabelach „Zamówienia” ↔ „Produkty” i „Dostawy” ↔ „Produkty”

Trigger **trig_pracownicy** dodaje do tabeli **Pracownicy_log** informacje o każdej zmianie pensji pracownika naszego sklepu.

```
CREATE OR REPLACE TRIGGER TRIG_PRACOWNICY
AFTER UPDATE ON PRACOWNICY
FOR EACH ROW
WHEN(NEW.PENSJA>0)

BEGIN
  INSERT INTO PRACOWNICY_LOG(ID_LOG,ID_PRACOWNIKA,STARA_PENSJA,DATA_ZMIANY,AKCJA)
VALUES(SEQ_PRACOWNICY_LOG.NEXTVAL, :OLD.ID_PRACOWNIKA,:OLD.PENSJA,SYSDATE,'AFTER');
END;
```

Notacja kurzych łapek (crow's foot)

1 lub więcej do 1

1 do 1

0 lub więcej do 1

0 lub 1 do 1

