

Bazy danych - sprawozdanie z projektu

1. Podstawowe informacje

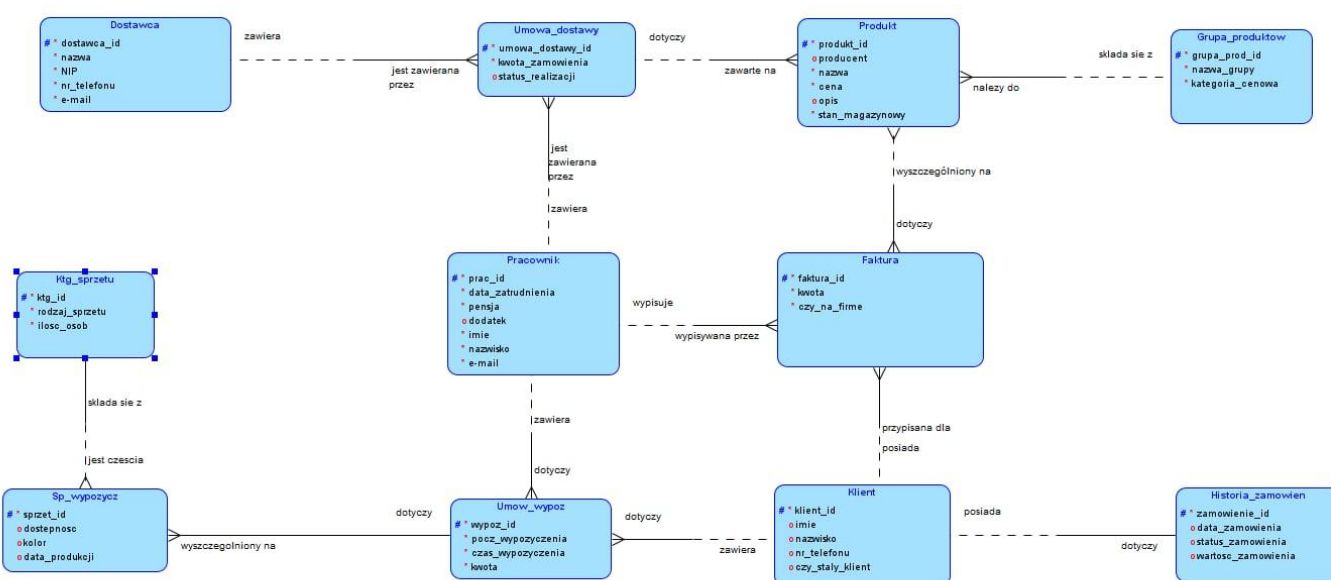
- **Autorzy projektu:** Hubert Dalmata 127752, Łukasz Kolanowski 127793
- **Nazwa grupy:** grupa 3
- **Temat:** Sklep wędkarski z wypożyczalnią sprzętu wodnego
- **Wybrany zakres (ocena):** 5
- **Baza danych: NAZWA:** prywatne2 **HASŁO:** LK88219LK

2. Wprowadzenie – opis studium przypadku

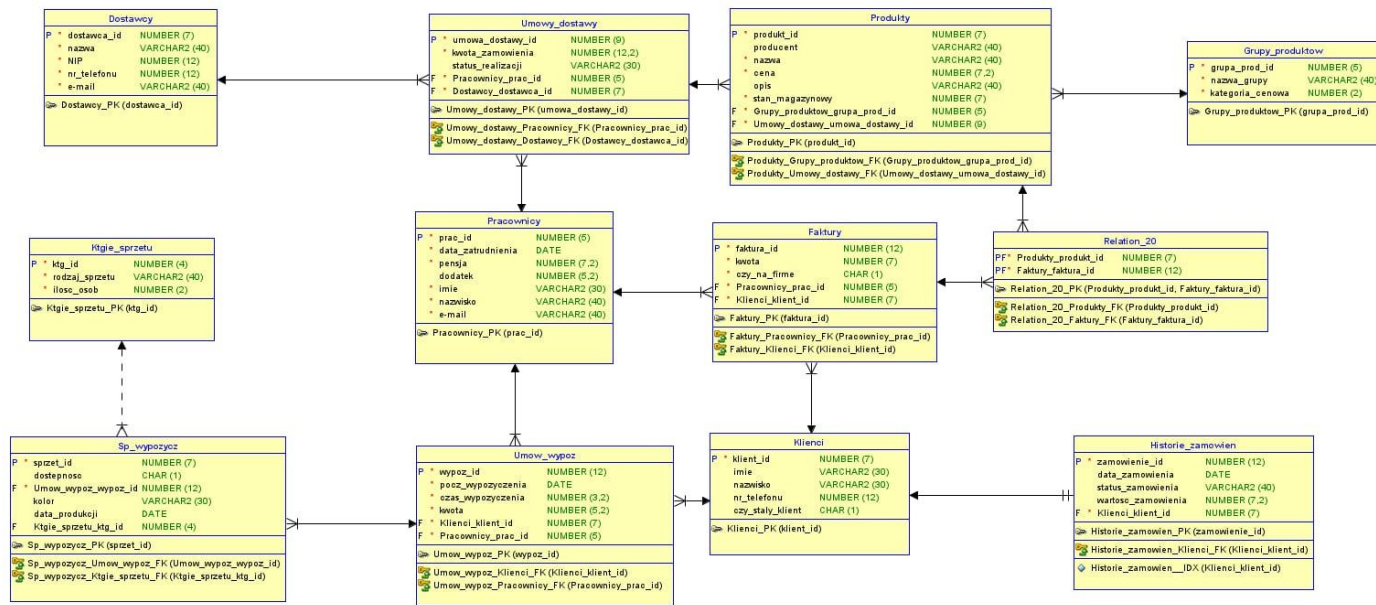
Krótkie rozwinięcie realizowanego tematu, przykładowe zastosowanie projektu itd.

Projekt obejmuje zaprojektowanie bazy danych, która obsługuje działalność fizycznego sklepu wędkarskiego z możliwością wypożyczenia sprzętu wodnego. W naszym projekcie zawarliśmy wszystkie kluczowe aspekty związane z prowadzeniem sklepu, takie jak: klienci, dostawcy, pracownicy sklepu, umowy i faktury, produkty oraz sprzęt do wypożyczenia. Naszą bazę danych można zastosować w obszarze analizy sprzedaży i funkcjonowania sklepu. Takie zagadnienia jak faktury zakupów i historia zakupowa klienta również są zawarte w projekcie. Informacje o pracownikach zawarte w bazie danych mogą nam posłużyć do mierzenia efektywności i wydajności zespołu. Na koniec warto wspomnieć, że nasze rozwiązanie pozwala sprawdzić dostępność sprzętu wodnego i monitorować czas wypożyczenia, oraz wyciągnąć informacje na temat umowy.

3. Model związków encji



4. Model relacyjnej bazy danych



5. Schemat bazy - tabele

Opisy tabel: nazwy, polecenia DDL i DML, klucze podstawowe i klucze obce.

```
-- Generated by Oracle SQL Developer Data Modeler 24.3.0.240.1210
-- at:      2025-01-22 16:28:45 CET
-- site:    Oracle Database 11g
-- type:    Oracle Database 11g
```

```
-- predefined type, no DDL - MDSYS.SDO_GEOMETRY
```

```
-- predefined type, no DDL - XMLTYPE
```

```
CREATE TABLE dostawcy (
  dostawca_id NUMBER(7) NOT NULL,
  nazwa      VARCHAR2(40) NOT NULL,
  nip        NUMBER(12) NOT NULL,
  nr_telefonu NUMBER(12) NOT NULL,
  "e-mail"   VARCHAR2(40) NOT NULL
);
```

```
ALTER TABLE dostawcy ADD CONSTRAINT dostawcy_pk PRIMARY KEY ( dostawca_id );
```

```
CREATE TABLE faktury (
  faktura_id      NUMBER(12) NOT NULL,
  kwota           NUMBER(7) NOT NULL,
  czy_na_firme    CHAR(1) NOT NULL,
  pracownicy_prac_id NUMBER(5) NOT NULL,
  klienci_klient_id NUMBER(7) NOT NULL
);
```

```
ALTER TABLE faktury ADD CONSTRAINT faktury_pk PRIMARY KEY ( faktura_id );
```

```
CREATE TABLE grupy_produktow (
  grupa_prod_id  NUMBER(5) NOT NULL,
  nazwa_grupy    VARCHAR2(40) NOT NULL,
  kategoria_cenowa NUMBER(2) NOT NULL
);
```

```
ALTER TABLE grupy_produktow ADD CONSTRAINT grupy_produktow_pk PRIMARY KEY (
grupa_prod_id );
```

```
CREATE TABLE historie_zamowien (
  zamowienie_id  NUMBER(12) NOT NULL,
  data_zamowienia DATE,
  status_zamowienia VARCHAR2(40),
  wartosc_zamowienia NUMBER(7, 2),
  klienci_klient_id NUMBER(7) NOT NULL
);
```

```
CREATE UNIQUE INDEX historie_zamowien__idx ON
  historie_zamowien (
    klienci_klient_id
  ASC );
```

```
ALTER TABLE historie_zamowien ADD CONSTRAINT historie_zamowien_pk PRIMARY KEY (
zamowienie_id );
```

```
CREATE TABLE klienci (
    klient_id    NUMBER(7) NOT NULL,
    imie         VARCHAR2(30),
    nazwisko     VARCHAR2(30),
    nr_telefonu  NUMBER(12),
    czy_staly_klient CHAR(1)
);
```

```
ALTER TABLE klienci ADD CONSTRAINT klienci_pk PRIMARY KEY ( klient_id );
```

```
CREATE TABLE ktgie_sprzetu (
    ktg_id      NUMBER(4) NOT NULL,
    rodzaj_sprzetu VARCHAR2(40) NOT NULL,
    ilosc_osob  NUMBER(2) NOT NULL
);
```

```
ALTER TABLE ktgie_sprzetu ADD CONSTRAINT ktgie_sprzetu_pk PRIMARY KEY ( ktg_id );
```

```
CREATE TABLE pracownicy (
    prac_id      NUMBER(5) NOT NULL,
    data_zatrudnienia DATE NOT NULL,
    pensja       NUMBER(7, 2) NOT NULL,
    dodatek      NUMBER(5, 2),
    imie         VARCHAR2(30) NOT NULL,
    nazwisko     VARCHAR2(40) NOT NULL,
    "e-mail"     VARCHAR2(40) NOT NULL
);
```

```
ALTER TABLE pracownicy ADD CONSTRAINT pracownicy_pk PRIMARY KEY ( prac_id );
```

```
CREATE TABLE produkty (
    produkt_id  NUMBER(7) NOT NULL,
    producent   VARCHAR2(40),
    nazwa       VARCHAR2(40) NOT NULL,
    cena        NUMBER(7, 2) NOT NULL,
    opis        VARCHAR2(40),
    stan_magazynowy NUMBER(7) NOT NULL,
    grupy_produkow_grupa_prod_id NUMBER(5) NOT NULL,
    umowy_dostawy_umowa_dostawy_id NUMBER(9) NOT NULL
);
```

```
ALTER TABLE produkty ADD CONSTRAINT produkty_pk PRIMARY KEY ( produkt_id );
```

```
CREATE TABLE relation_20 (
    produkty_produkt_id NUMBER(7) NOT NULL,
    faktury_faktura_id  NUMBER(12) NOT NULL
);
```

```
ALTER TABLE relation_20 ADD CONSTRAINT relation_20_pk PRIMARY KEY (
produkty_produkt_id,
faktury_faktura_id );
```

```
CREATE TABLE sp_wypożycz (
    sprzet_id      NUMBER(7) NOT NULL,
    dostepnosc     CHAR(1),
    kolor          VARCHAR2(30),
    data_produkcji DATE,
    umow_wypoz_wypoz_id NUMBER(12) NOT NULL,
    ktgie_sprzetu_ktg_id NUMBER(4)
);
```

```
ALTER TABLE sp_wypożycz ADD CONSTRAINT sp_wypożycz_pk PRIMARY KEY ( sprzet_id );
```

```
CREATE TABLE umow_wypoz (
    wypoz_id      NUMBER(12) NOT NULL,
    pocz_wypożyczenia DATE NOT NULL,
    czas_wypożyczenia NUMBER(3, 2) NOT NULL,
    kwota          NUMBER(5, 2) NOT NULL,
    klienci_klient_id NUMBER(7) NOT NULL,
    pracownicy_prac_id NUMBER(5) NOT NULL
);
```

```
ALTER TABLE umow_wypoz ADD CONSTRAINT umow_wypoz_pk PRIMARY KEY ( wypoz_id );
```

```
CREATE TABLE umowy_dostawy (
    umowa_dostawy_id NUMBER(9) NOT NULL,
    kwota_zamowienia NUMBER(12, 2) NOT NULL,
    status_realizacji VARCHAR2(30),
    pracownicy_prac_id NUMBER(5) NOT NULL,
    dostawcy_dostawca_id NUMBER(7) NOT NULL
);
```

```
ALTER TABLE umowy_dostawy ADD CONSTRAINT umowy_dostawy_pk PRIMARY KEY ( umowa_dostawy_id );
```

```
ALTER TABLE faktury
    ADD CONSTRAINT faktury_klienci_fk FOREIGN KEY ( klienci_klient_id )
    REFERENCES klienci ( klient_id );
```

```
ALTER TABLE faktury
    ADD CONSTRAINT faktury_pracownicy_fk FOREIGN KEY ( pracownicy_prac_id )
    REFERENCES pracownicy ( prac_id );
```

```
ALTER TABLE historie_zamowien
    ADD CONSTRAINT historie_zamowien_klienci_fk FOREIGN KEY ( klienci_klient_id )
    REFERENCES klienci ( klient_id );
```

```
ALTER TABLE produkty
    ADD CONSTRAINT produkty_grupy_produkow_fk FOREIGN KEY (
    grupy_produkow_grupa_prod_id )
    REFERENCES grupy_produkow ( grupa_prod_id );
```

```
ALTER TABLE produkty
    ADD CONSTRAINT produkty_umowy_dostawy_fk FOREIGN KEY (
    umowy_dostawy_umowa_dostawy_id )
    REFERENCES umowy_dostawy ( umowa_dostawy_id );
```

```

ALTER TABLE relation_20
  ADD CONSTRAINT relation_20_faktury_fk FOREIGN KEY ( faktury_faktura_id )
    REFERENCES faktury ( faktura_id );

ALTER TABLE relation_20
  ADD CONSTRAINT relation_20_produkty_fk FOREIGN KEY ( produkty_produk_t_id )
    REFERENCES produkty ( produkt_id );

ALTER TABLE sp_wypożycz
  ADD CONSTRAINT sp_wypożycz_ktgie_sprzetu_fk FOREIGN KEY ( ktgie_sprzetu_ktg_id )
    REFERENCES ktgie_sprzetu ( ktg_id );

ALTER TABLE sp_wypożycz
  ADD CONSTRAINT sp_wypożycz_umow_wypoz_fk FOREIGN KEY ( umow_wypoz_wypoz_id )
    REFERENCES umow_wypoz ( wypoz_id );

ALTER TABLE umow_wypoz
  ADD CONSTRAINT umow_wypoz_klienci_fk FOREIGN KEY ( klienci_klient_id )
    REFERENCES klienci ( klient_id );

ALTER TABLE umow_wypoz
  ADD CONSTRAINT umow_wypoz_pracownicy_fk FOREIGN KEY ( pracownicy_prac_id )
    REFERENCES pracownicy ( prac_id );

ALTER TABLE umowy_dostawy
  ADD CONSTRAINT umowy_dostawy_dostawcy_fk FOREIGN KEY ( dostawcy_dostawca_id )
    REFERENCES dostawcy ( dostawca_id );

ALTER TABLE umowy_dostawy
  ADD CONSTRAINT umowy_dostawy_pracownicy_fk FOREIGN KEY ( pracownicy_prac_id )
    REFERENCES pracownicy ( prac_id );

```

-- Oracle SQL Developer Data Modeler Summary Report:

```

--
-- CREATE TABLE                                12
-- CREATE INDEX                                1
-- ALTER TABLE                                25
-- CREATE VIEW                                  0
-- ALTER VIEW                                  0
-- CREATE PACKAGE                              0
-- CREATE PACKAGE BODY                         0
-- CREATE PROCEDURE                           0
-- CREATE FUNCTION                             0
-- CREATE TRIGGER                              0
-- ALTER TRIGGER                              0
-- CREATE COLLECTION TYPE                      0
-- CREATE STRUCTURED TYPE                      0
-- CREATE STRUCTURED TYPE BODY                 0
-- CREATE CLUSTER                             0
-- CREATE CONTEXT                             0
-- CREATE DATABASE                             0
-- CREATE DIMENSION                           0

```

```
-- CREATE DIRECTORY          0
-- CREATE DISK GROUP          0
-- CREATE ROLE                 0
-- CREATE ROLLBACK SEGMENT    0
-- CREATE SEQUENCE             0
-- CREATE MATERIALIZED VIEW    0
-- CREATE MATERIALIZED VIEW LOG 0
-- CREATE SYNONYM              0
-- CREATE TABLESPACE          0
-- CREATE USER                  0
-- DROP TABLESPACE            0
-- DROP DATABASE                0
-- REDACTION POLICY            0
-- ORDS DROP SCHEMA            0
-- ORDS ENABLE SCHEMA          0
-- ORDS ENABLE OBJECT          0
-- ERRORS                      0
-- WARNINGS                    0
```

--DOSTAWCY

```
INSERT INTO dostawcy (dostawca_id, nazwa, nip, nr_telefonu, "e-mail") VALUES
(15, 'RybyPol', 111111111111, 543987654, 'rybypol@rybypol.pl');
```

```
INSERT INTO dostawcy (dostawca_id, nazwa, nip, nr_telefonu, "e-mail") VALUES
(16, 'ZanetyPol', 222222222222, 685983853, 'kontakt@zanetypol.pl');
```

```
INSERT INTO dostawcy (dostawca_id, nazwa, nip, nr_telefonu, "e-mail") VALUES
(17, 'WszystkoDlaRybaka', 333333333333, 386947594, 'kontakt@wszystkodlarybaka.pl');
```

```
INSERT INTO dostawcy (dostawca_id, nazwa, nip, nr_telefonu, "e-mail") VALUES
(18, 'SupyPolskie', 444444444444, 490765239, 'kontakt@supypolskie.com');
```

```
INSERT INTO dostawcy (dostawca_id, nazwa, nip, nr_telefonu, "e-mail") VALUES
(19, 'WedkiDlaCiebie', 5555555555, 358940594, 'biuro@wedkidlaciebie.pl');
```

--KLIENCI

```
INSERT INTO klienci (klient_id, imie, nazwisko, nr_telefonu, czy_staly_klient) VALUES
(seq_10.NEXTVAL, 'Marek', 'Sandacz', 398458800, 'T');
```

```
INSERT INTO klienci (klient_id, imie, nazwisko, nr_telefonu, czy_staly_klient) VALUES
(seq_10.NEXTVAL, 'Radoslaw', 'Karp', 397307428, 'T');
```

```
INSERT INTO klienci (klient_id, imie, nazwisko, nr_telefonu, czy_staly_klient) VALUES
(seq_10.NEXTVAL, 'Euzebiusz', 'Miruna', 457790800, 'F');
```

```
INSERT INTO klienci (klient_id, imie, nazwisko, nr_telefonu, czy_staly_klient) VALUES
(seq_10.NEXTVAL, 'Miroslaw', 'Karasz', 118458840, 'T');
```

```
INSERT INTO klienci (klient_id, imie, nazwisko, nr_telefonu, czy_staly_klient) VALUES
(seq_10.NEXTVAL, 'Tomasz', 'Wegorz', 318118830, 'F');
```

--PRACOWNICY

```
INSERT INTO pracownicy (prac_id, data_zatrudnienia, pensja, dodatek, imie, nazwisko, "e-mail")
VALUES
```

(1, date '2010-04-08', 5000, 400, 'Andrzej', 'Sprzedawiecki', 'a.sprzedawiecki@wedkarztoty.pl');

INSERT INTO pracownicy (prac_id, data_zatrudnienia, pensja, dodatek, imie, nazwisko, "e-mail")
VALUES

(2, date '2013-10-09', 7000, 900, 'Radoslaw', 'Wedkarski', 'r.wedkarski@wedkarztoty.pl');

INSERT INTO pracownicy (prac_id, data_zatrudnienia, pensja, dodatek, imie, nazwisko, "e-mail")
VALUES

(3, date '2019-04-04', 12000, 250, 'Euzebiusz', 'Pracowniczy', 'e.pracowniczy@wedkarztoty.pl');

INSERT INTO pracownicy (prac_id, data_zatrudnienia, pensja, dodatek, imie, nazwisko, "e-mail")
VALUES

(4, date '2020-07-12', 5000, NULL, 'Arkadiusz', 'Lodzianin', 'a.lodzianin@wedkarztoty.pl');

INSERT INTO pracownicy (prac_id, data_zatrudnienia, pensja, dodatek, imie, nazwisko, "e-mail")
VALUES

(5, date '2021-03-11', 4600, NULL, 'Kuba', 'Jablkowski', 'k.jablkowski@wedkarztoty.pl');

--HISTORIE_ZAMOWIEN

INSERT INTO historie_zamowien (zamowienie_id, data_zamowienia, status_zamowienia,
wartosc_zamowienia, klienci_klient_id) VALUES

(1, date '2016-01-01', 'Zrealizowane', 432.45, 10);

INSERT INTO historie_zamowien (zamowienie_id, data_zamowienia, status_zamowienia,
wartosc_zamowienia, klienci_klient_id) VALUES

(2, date '2019-07-08', 'Zrealizowane', 190.50, 20);

INSERT INTO historie_zamowien (zamowienie_id, data_zamowienia, status_zamowienia,
wartosc_zamowienia, klienci_klient_id) VALUES

(3, date '2022-10-03', 'Zrealizowane', 234.98, 30);

INSERT INTO historie_zamowien (zamowienie_id, data_zamowienia, status_zamowienia,
wartosc_zamowienia, klienci_klient_id) VALUES

(4, date '2023-03-02', 'Anulowane', 89.99, 40);

INSERT INTO historie_zamowien (zamowienie_id, data_zamowienia, status_zamowienia,
wartosc_zamowienia, klienci_klient_id) VALUES

(5, date '2024-11-10', 'W trakcie realizacji', 139.99, 50);

--FAKTURY

INSERT INTO faktury (faktura_id, kwota, czy_na_firme, pracownicy_prac_id, klienci_klient_id)
VALUES

(1, 7482, 'T', 1, 20);

INSERT INTO faktury (faktura_id, kwota, czy_na_firme, pracownicy_prac_id, klienci_klient_id)
VALUES

(2, 412, 'F', 3, 30);

INSERT INTO faktury (faktura_id, kwota, czy_na_firme, pracownicy_prac_id, klienci_klient_id)
VALUES

(3, 51, 'T', 2, 10);


```
INSERT INTO faktury (faktura_id, kwota, czy_na_firme, pracownicy_prac_id, klienci_klient_id)
VALUES
(4, 421, 'F', 4, 50);
```

```
INSERT INTO faktury (faktura_id, kwota, czy_na_firme, pracownicy_prac_id, klienci_klient_id)
VALUES
(5, 5675, 'T', 1, 30);
```

--GRUPY_PRODUKTOW

```
INSERT INTO grupy_produktow (grupa_prod_id, nazwa_grupy, kategoria_cenowa) VALUES
(seq_1.NEXTVAL, 'wedki', 5);
```

```
INSERT INTO grupy_produktow (grupa_prod_id, nazwa_grupy, kategoria_cenowa) VALUES
(seq_1.NEXTVAL, 'zanety', 2);
```

```
INSERT INTO grupy_produktow (grupa_prod_id, nazwa_grupy, kategoria_cenowa) VALUES
(seq_1.NEXTVAL, 'haczyki', 1);
```

```
INSERT INTO grupy_produktow (grupa_prod_id, nazwa_grupy, kategoria_cenowa) VALUES
(seq_1.NEXTVAL, 'linki', 2);
```

```
INSERT INTO grupy_produktow (grupa_prod_id, nazwa_grupy, kategoria_cenowa) VALUES
(seq_1.NEXTVAL, 'podbieraki', 3);
```

```
INSERT INTO grupy_produktow (grupa_prod_id, nazwa_grupy, kategoria_cenowa) VALUES
(seq_1.NEXTVAL, 'kolowrotki', 4);
```

--KTGIE_SPRZETU

```
INSERT INTO ktgie_sprzetu (ktg_id, rodzaj_sprzetu, ilosc_osob) VALUES
(100, 'lodka', 4);
```

```
INSERT INTO ktgie_sprzetu (ktg_id, rodzaj_sprzetu, ilosc_osob) VALUES
(200, 'kajak', 2);
```

```
INSERT INTO ktgie_sprzetu (ktg_id, rodzaj_sprzetu, ilosc_osob) VALUES
(300, 'rower wodny', 4);
```

```
INSERT INTO ktgie_sprzetu (ktg_id, rodzaj_sprzetu, ilosc_osob) VALUES
(400, 'kapok', 1);
```

```
INSERT INTO ktgie_sprzetu (ktg_id, rodzaj_sprzetu, ilosc_osob) VALUES
(500, 'deska sup', 1);
```

--UMOW_WYPOZ

```
INSERT INTO umow_wypoz (wypoz_id, pocz_wypozyczenia, czas_wypozyczenia, kwota,
klienci_klient_id, pracownicy_prac_id) VALUES
(1, date '2025-01-01', 2.50, 50, 10, 1);
```

```
INSERT INTO umow_wypoz (wypoz_id, pocz_wypozyczenia, czas_wypozyczenia, kwota,
klienci_klient_id, pracownicy_prac_id) VALUES
(2, date '2025-01-04', 1, 20, 20, 2);
```

```
INSERT INTO umow_wypoz (wypoz_id, pocz_wypozyczenia, czas_wypozyczenia, kwota,
klienci_klient_id, pracownicy_prac_id) VALUES
(3, date '2025-01-07', 8, 100, 30, 1);
```

```
INSERT INTO umow_wypoz (wypoz_id, pocz_wypozyczenia, czas_wypozyczenia, kwota,
klienci_klient_id, pracownicy_prac_id) VALUES
(4, date '2025-01-09', 1.5, 60, 40, 5);
```

```
INSERT INTO umow_wypoz (wypoz_id, pocz_wypozyczenia, czas_wypozyczenia, kwota,
klienci_klient_id, pracownicy_prac_id) VALUES
(5, date '2024-12-12', 1.5, 30, 50, 2);
```

--UMOWY_DOSTAWY

```
INSERT INTO umowy_dostawy (umowa_dostawy_id, kwota_zamowienia, status_realizacji,
pracownicy_prac_id, dostawcy_dostawca_id) VALUES
(1, 400.45, 'Zrealizowane', 1, 15);
```

```
INSERT INTO umowy_dostawy (umowa_dostawy_id, kwota_zamowienia, status_realizacji,
pracownicy_prac_id, dostawcy_dostawca_id) VALUES
(2, 2000.50, 'Zrealizowane', 2, 16);
```

```
INSERT INTO umowy_dostawy (umowa_dostawy_id, kwota_zamowienia, status_realizacji,
pracownicy_prac_id, dostawcy_dostawca_id) VALUES
(3, 680.98, 'Zrealizowane', 3, 18);
```

```
INSERT INTO umowy_dostawy (umowa_dostawy_id, kwota_zamowienia, status_realizacji,
pracownicy_prac_id, dostawcy_dostawca_id) VALUES
(4, 500.99, 'W trakcie realizacji', 4, 17);
```

```
INSERT INTO umowy_dostawy (umowa_dostawy_id, kwota_zamowienia, status_realizacji,
pracownicy_prac_id, dostawcy_dostawca_id) VALUES
(5, 5900.99, 'W trakcie realizacji', 5, 19);
```

--PRODUKTY

```
INSERT INTO produkty (produkt_id, producent, nazwa, cena, opis, stan_magazynowy,
grupy_produkow_grupa_prod_id, umowy_dostawy_umowa_dostawy_id) VALUES
(1, 'Jaxon', 'wedka super ', 200, 'dobry wybor na karpia', 15, 1, 1);
```

```
INSERT INTO produkty (produkt_id, producent, nazwa, cena, opis, stan_magazynowy,
grupy_produkow_grupa_prod_id, umowy_dostawy_umowa_dostawy_id) VALUES
(2, 'Kamatsu', 'kolowrotek piorun', 250, 'wytrzymały i w uczciwej cenie', 21, 6, 2);
```

```
INSERT INTO produkty (produkt_id, producent, nazwa, cena, opis, stan_magazynowy,
grupy_produkow_grupa_prod_id, umowy_dostawy_umowa_dostawy_id) VALUES
(3, 'Konger', 'linka karpiowa 4 mm', 15, 'polecam - Hubert Dalmata', 33, 4, 3);
```

```
INSERT INTO produkty (produkt_id, producent, nazwa, cena, opis, stan_magazynowy,
grupy_produkow_grupa_prod_id, umowy_dostawy_umowa_dostawy_id) VALUES
(4, 'Fisker', 'podbierak medium', 95, 'na dorodne sztuki', 50, 5, 4);
```

```
INSERT INTO produkty (produkt_id, producent, nazwa, cena, opis, stan_magazynowy,
grupy_produkow_grupa_prod_id, umowy_dostawy_umowa_dostawy_id) VALUES
(5, 'Dam', 'podbierak mini', 45, 'poreczny', 15, 5, 5);
```

--SP_WYPOZYCZ

```
INSERT INTO sp_wyposzcz (sprzet_id, dostepnosc, kolor, data_produkcji, umow_wyposz_wyposz_id,
ktgie_sprzetu_ktg_id) VALUES
(12, 'T', 'czerwony', date '2009-08-01', 5, 400);
```

```
INSERT INTO sp_wyposzcz (sprzet_id, dostepnosc, kolor, data_produkcji, umow_wyposz_wyposz_id,
ktgie_sprzetu_ktg_id) VALUES
(13, 'T', 'srebrny', date '2010-04-06', 4, 300);
```

```
INSERT INTO sp_wyposzcz (sprzet_id, dostepnosc, kolor, data_produkcji, umow_wyposz_wyposz_id,
ktgie_sprzetu_ktg_id) VALUES
(14, 'T', 'czarny', date '2011-01-08', 3, 200);
```

```
INSERT INTO sp_wyposzcz (sprzet_id, dostepnosc, kolor, data_produkcji, umow_wyposz_wyposz_id,
ktgie_sprzetu_ktg_id) VALUES
(15, 'T', 'zielony', date '2017-12-12', 2, 100);
```

```
INSERT INTO sp_wyposzcz (sprzet_id, dostepnosc, kolor, data_produkcji, umow_wyposz_wyposz_id,
ktgie_sprzetu_ktg_id) VALUES
(16, 'T', 'morski', date '2018-04-08', 1, 500);
```

--Dane powyżej są fikcyjne

6. Przypadki użycia bazy danych

Opis przypadków użycia bazy danych wraz z poleceniami SELECT.

--1. Uzyskanie łącznej wartości zamówień dla danego dostawcy

```
SELECT
    dostawcy.nazwa AS nazwa_dostawcy,
    SUM(umowy_dostawy.kwota_zamowienia) AS laczna_kwota_zamowien
FROM
    umowy_dostawy
JOIN
    dostawcy ON umowy_dostawy.dostawcy_dostawca_id = dostawcy.dostawca_id
GROUP BY
    dostawcy.nazwa
ORDER BY
    laczna_kwota_zamowien DESC;
```

	NAZWA_DOSTAWCY	LACZNA_KWOTA_ZAMOWIEN
1	WedkiDlaCiebie	5900,99
2	ZanetyPol	2000,5
3	SupyPolskie	680,98
4	WszystkoDlaRybaka	500,99
5	RybyPol	400,45

--2. Uzyskanie danych pracowników, których średnia pensja jest wyższa niż średnia ogólna dla wszystkich pracowników.

```
SELECT *
from pracownicy
where pensja+NVL(dodatek,0) > (SELECT AVG(pensja+ NVL(dodatek,0)) from pracownicy);
```

	PRAC_ID	DATA_ZATRUDNIENIA	PENSJA	DODATEK	IMIE	NAZWISKO	e-mail
1	2	13/10/09	7000	900	Radosław	Wedkarski	r.wedkarski@wedkarztoty.pl
2	3	19/04/04	12000	250	Euzebiusz	Pracowniczy	e.pracowniczy@wedkarztoty.pl

--3. Uzyskanie informacji, który pracownik, jakiemu klientowi wypożyczył sprzęt w związku z daną umową, wraz z datą produkcji sprzętu i dla ilu osób jest przeznaczony, uwzględniając tylko klientów, którzy mają przynajmniej jedno zrealizowane zamówienie.

```
SELECT
    umow_wypoz.wypoz_id,
    umow_wypoz.pocz_wypozyczenia,
    klienci.imie || ' ' || klienci.nazwisko as klient,
    pracownicy.imie || ' ' || pracownicy.nazwisko as pracownik,
    TO_CHAR(sp_wypozycz.data_produkcji, 'DD-Mon-YYYY',
    'nls_date_language = Polish')as data_produkcji_sprzetu,
    ktgie_sprzetu.ilosc_osob as ilu_osobowy
FROM
    umow_wypoz
JOIN
    klienci ON umow_wypoz.klienci_klient_id = klienci.klient_id
JOIN
    pracownicy ON umow_wypoz.pracownicy_prac_id = pracownicy.prac_id
JOIN
    sp_wypozycz ON sp_wypozycz.umow_wypoz_wypoz_id = umow_wypoz.wypoz_id
JOIN
    ktgie_sprzetu ON ktgie_sprzetu.ktgie_id = sp_wypozycz.ktgie_sprzetu_ktgie_id
WHERE
    klienci.klient_id IN (
        SELECT
            klienci_klient_id
        FROM
            historie_zamowien
        WHERE
            historie_zamowien.status_zamowienia ='Zrealizowane');
```

	WYPOZ_ID	POCZ_WYPOZYCZENIA	KLIENT	PRACOWNIK	DATA_PRODUKCJI_SPRZETU	ILU_OSOBOWY
1	3	25/01/07	Euzebiusz Miruna	Andrzej Sprzedawiecki	08-Sty-2011	2
2	2	25/01/04	Radosław Karp	Radosław Wedkarski	12-Gru-2017	4
3	1	25/01/01	Marek Sandacz	Andrzej Sprzedawiecki	08-Kwi-2018	1

--4. Uzyskanie informacji ile czasu przepracował pracownik, w dniach, miesiącach i latach

```
SELECT
    imie,
    nazwisko,
    FLOOR(SYSDATE -data_zatrudnienia) AS przepracowane_dni,
    FLOOR(MONTHS_BETWEEN(SYSDATE, data_zatrudnienia)) AS przepracowane_miesiecy,
    FLOOR(MONTHS_BETWEEN(SYSDATE, data_zatrudnienia) / 12) AS przepracowane_lata
FROM pracownicy;
```

	IMIE	NAZWISKO	PRZEPACOWNE_DNI	PRZEPACOWANE_MIESIECY	PRZEPACOWANE_LATA
1	Andrzej	Sprzedawiecki	5403	177	14
2	Radosław	Wedkarski	4123	135	11
3	Euzebiusz	Pracowniczy	2120	69	5
4	Arkadiusz	Lodzianin	1655	54	4
5	Kuba	Jablkowski	1413	46	3

--5. Uzyskanie liczby zamówień i łącznej wartości zamówień dla każdego klienta

```
SELECT
    klienci.klient_id,
    SUBSTR(klienci.imie, 1,1) ||'. ' || klienci.nazwisko AS imie_i_nazwisko_klienta,
    COUNT(historie_zamowien.zamowienie_id) AS liczba_zamowien,
    SUM(historie_zamowien.wartosc_zamowienia) AS suma_zamowien
FROM
    klienci
JOIN
    historie_zamowien ON klienci.klient_id = historie_zamowien.klienci_klient_id
GROUP BY
    klienci.klient_id, klienci.imie, klienci.nazwisko
ORDER BY suma_zamowien DESC;
```

	KLIENT_ID	IMIE_I_NAZWISKO_KLIENTA	LICZBA_ZAMOWIEN	SUMA_ZAMOWIEN
1	10	M. Sandacz	1	432,45
2	30	E. Miruna	1	234,98
3	20	R. Karp	1	190,5
4	50	T. Węgorz	1	139,99
5	40	M. Karas	1	89,99

--6. Uzyskanie informacji o produktach droższych niż wedka super wraz z grupą, do jakiej należy dany produkt

```
SELECT produkt_id, producent, nazwa, grupy_produktow.nazwa_grupy
FROM produkty
JOIN grupy_produktow ON grupy_produktow.grupa_prod_id =
produkty.grupy_produktow_grupa_prod_id
WHERE cena > (SELECT cena
    FROM produkty
    WHERE nazwa = 'wedka super');
```

	PRODUKT_ID	PRODUCENT	NAZWA	NAZWA_GRUPY
1	2	Kamatsu	kolowrotek piorun	kolowrotki

--7. Uzyskanie informacji o sprzętach dedykowanych dla co najmniej dwóch osób

```
SELECT s.sprzet_id, s.kolor, k.ilosc_osob
FROM sp_wypozycz s
JOIN (SELECT ktg_id, rodzaj_sprzetu, ilosc_osob FROM ktgie_sprzetu WHERE ilosc_osob >= 2) k
ON k.ktg_id = s.ktgie_sprzetu_ktg_id;
```

	SPRZET_ID	KOLOR	ILOSC_OSOB
1	13	srebrny	4
2	14	czarny	2
3	15	zielony	4

--8. Uzyskanie informacji jacy dostawcy dostarczają wedki

```
SELECT d.nazwa as Kto_dostarcza_wedki from dostawcy d
JOIN umowy_dostawy um on um.dostawcy_dostawca_id=d.dostawca_id
JOIN produkty p on p.UMOWY_DOSTAWY_UMOWA_DOSTAWY_ID=um.umowa_dostawy_id
WHERE INSTR(p.nazwa,'wedka')>0;
```

	KTO_DOSTARCZA_WEDKI
1	RybyPol

7. Pozostałe obiekty bazy danych

SEKWENCERY :

Sekwencer poniżej służy do generowania id dla tabeli klienci (1,2,3...):

```
CREATE SEQUENCE seq_1
INCREMENT BY 1
START WITH 1
MAXVALUE 5000
NOCACHE
NOCYCLE;
```

Sekwencer poniżej służy do generowania id dla tabeli grupy_produkow(10,20,30...):

```
CREATE SEQUENCE seq_10
INCREMENT BY 10
START WITH 10
MAXVALUE 5000
NOCACHE
NOCYCLE;
```

PERSPEKTYWY:

--1. Uzyskanie łącznej wartości zamówień dla danego dostawcy

```
CREATE VIEW v1
AS SELECT
    dostawcy.nazwa AS nazwa_dostawcy,
    SUM(umowy_dostawy.kwota_zamowienia) AS laczna_kwota_zamowien
FROM
    umowy_dostawy
JOIN
    dostawcy ON umowy_dostawy.dostawcy_dostawca_id = dostawcy.dostawca_id
GROUP BY
    dostawcy.nazwa
ORDER BY
    laczna_kwota_zamowien DESC
WITH READ ONLY ;
```

--2. Uzyskanie danych pracowników, których średnia pensja jest wyższa niż średnia ogólna dla wszystkich pracowników.

```
CREATE VIEW v2
AS SELECT *
from pracownicy
where pensja+NVL(dodatek,0) > (SELECT AVG(pensja+ NVL(dodatek,0)) from pracownicy)
WITH READ ONLY;
```

--3. Uzyskanie informacji, który pracownik, jakiemu klientowi wypożyczył sprzęt w związku z daną umową, wraz z datą produkcji sprzętu i dla ilu osób jest przeznaczony, uwzględniając tylko klientów, którzy mają przynajmniej jedno zrealizowane zamówienie.

```
CREATE VIEW v3
```

```

AS SELECT
    umow_wypoz.wypoz_id,
    umow_wypoz.pocz_wypozyczenia,
    klienci.imie || ' ' || klienci.nazwisko as klient,
    pracownicy.imie || ' ' || pracownicy.nazwisko as pracownik,
    TO_CHAR(sp_wypozycz.data_produkcji, 'DD-Mon-YYYY',
    'nls_date_language = Polish')as data_produkcji_sprzetu,
    ktgie_sprzetu.ilosc_osob as ilu_osobowy
FROM
    umow_wypoz
JOIN
    klienci ON umow_wypoz.klienci_klient_id = klienci.klient_id
JOIN
    pracownicy ON umow_wypoz.pracownicy_prac_id = pracownicy.prac_id
JOIN
    sp_wypozycz ON sp_wypozycz.umow_wypoz_wypoz_id = umow_wypoz.wypoz_id
JOIN
    ktgie_sprzetu ON ktgie_sprzetu.ktg_id = sp_wypozycz.ktgie_sprzetu_ktg_id
WHERE
    klienci.klient_id IN (
        SELECT      klienci_klient_id      FROM      historie_zamowien      WHERE
historie_zamowien.status_zamowienia ='Zrealizowane'
    )
WITH READ ONLY;

```

--4. Uzyskanie informacji ile czasu przepracował pracownik, w dniach, miesiącach i latach

```

CREATE VIEW v4
AS SELECT
    imie,
    nazwisko,
    FLOOR(SYSDATE -data_zatrudnienia) AS przepracowane_dni,
    FLOOR(MONTHS_BETWEEN(SYSDATE, data_zatrudnienia)) AS przepracowane_miesiecy,
    FLOOR(MONTHS_BETWEEN(SYSDATE, data_zatrudnienia) / 12) AS przepracowane_lata
FROM pracownicy
WITH READ ONLY;

```

--5. Uzyskanie liczby zamówień i łącznej wartości zamówień dla każdego klienta

```

CREATE VIEW v5
AS SELECT
    klienci.klient_id,
    SUBSTR(klienci.imie, 1,1) ||'. ' || klienci.nazwisko AS imie_i_nazwisko_klienta,
    COUNT(historie_zamowien.zamowienie_id) AS liczba_zamowien,
    SUM(historie_zamowien.wartosc_zamowienia) AS suma_zamowien
FROM
    klienci
JOIN
    historie_zamowien ON klienci.klient_id = historie_zamowien.klienci_klient_id
GROUP BY

```

```

    klienci.klient_id, klienci.imie, klienci.nazwisko
ORDER BY
    suma_zamowien DESC
WITH READ ONLY;

```

--6. Uzyskanie informacji o produktach droższych niż wedka super wraz z grupa, do jakiej należy dany produkt

```

CREATE VIEW v6
AS SELECT produkt_id, producent, nazwa, grupy_produkow.nazwa_grupy
FROM produkty
JOIN      grupy_produkow      ON      grupy_produkow.grupa_prod_id      =
produkty.grupy_produkow_grupa_prod_id
WHERE cena > (SELECT cena
              FROM produkty
              WHERE nazwa = 'wedka super ')
WITH READ ONLY;

```

--7. Uzyskanie informacji o sprzętach dedykowanych dla co najmniej dwóch osób

```

CREATE VIEW v7
AS SELECT s.sprzet_id, s.kolor, k.ilosc_osob
FROM sp_wypozycz s
JOIN (SELECT ktg_id, rodzaj_sprzetu, ilosc_osob FROM ktgie_sprzetu WHERE ilosc_osob >= 2) k
ON k.ktg_id = s.ktgie_sprzetu_ktg_id
WITH READ ONLY;

```

--8. Uzyskanie informacji jacy dostawcy dostarczają wędkę

```

CREATE VIEW v8
AS SELECT d.nazwa as Kto_dostarcza_wedki from dostawcy d
JOIN umowy_dostawy um on um.dostawcy_dostawca_id=d.dostawca_id
JOIN produkty p on p.UMOWY_DOSTAWY_UMOWA_DOSTAWY_ID=um.umowa_dostawy_id
WHERE INSTR(p.nazwa,'wedka')>0
WITH READ ONLY;

```