**Wojskowa Akademia Techniczna**

**Bazy Danych**

**Sprawozdanie z Projektu**

*„Sklep rowerowy”*

***Dane logowania Oracle:***

*Username: I5Y6\_03*

*Password: I5Y6\_03*

*Hostname: city.wsisiz.edu.pl*

*Port: 1521*

*SID: orcl*

***Dane logowania Sybase:***

*User Id: I5Y6\_03*

*Password: KUMPLE22a*

*Host: city.wsisiz.edu.pl*

*Port: 2639*

*Server name: asa12\_city*

*Database name: asa12\_city*

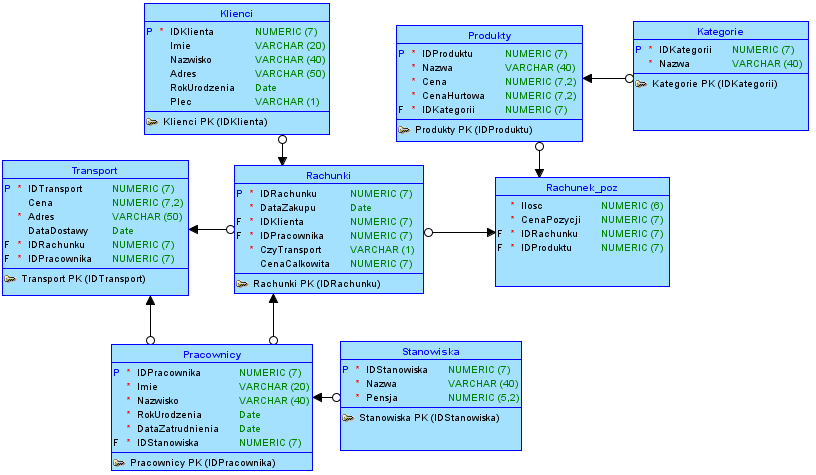
*Autor: Paweł Skrzypkowski*

*Grupa: I5Y6S1*

1. **Analiza biznesowa**

Przedstawiany projekt prezentuje implementację sklepu rowerowego. Produkty w sklepie rowerowym podzielone są na 8 kategorii. W sklepie pracuje 5 osób, 2 sprzedawców, jeden kierownik, jedna księgowa i jeden kierowca/dostawca. Zaprezentowany projekt sprawdziłby się w stacjonarnym sklepie rowerowym w celu skomputeryzowania oraz usprawnienia sklepu.

1. **Model logiczny**

****

Wyjaśnienie mniej oczywistych encji i atrybutów:

- Rachunek\_poz – encja, w której przechowywane są poszczególne pozycje rachunków

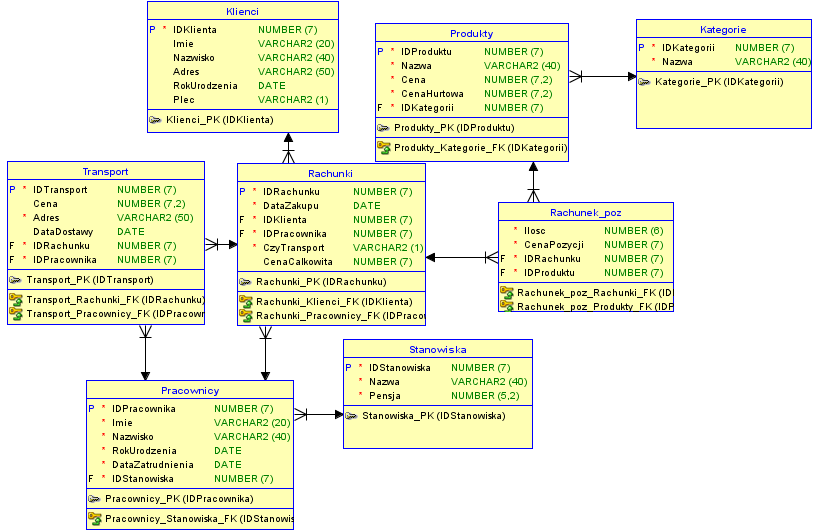
- Stanowiska – encja, w której przechowywane są rodzaje stanowisk(np. Kierownik, Sprzedawca)

- Cena w encji Produkty - cena za którą produkt sprzedawany jest klientowi

- DataDostawy w encji Transport – dzien, w którym został zrealizowany transport

- CzyTransport w Rachunki – jeśli ma wartość „T” to jest zamówienie z transportem, a jeśli „N” to bez transportu

1. **Model relacyjny**

****

Wyjaśnienie mniej oczywistych encji i atrybutów w poprzednim punkcie.

1. **Oprogramowanie bazy danych z komentarzami**

* **Baza Oracle:**
  + **Tworzenie tabel**

CREATE TABLE kategorie (

idkategorii NUMBER(7) NOT NULL,

nazwa VARCHAR2(40) NOT NULL

);

ALTER TABLE kategorie ADD CONSTRAINT kategorie\_pk PRIMARY KEY ( idkategorii );

CREATE TABLE klienci (

idklienta NUMBER(7) NOT NULL,

imie VARCHAR2(20),

nazwisko VARCHAR2(40),

adres VARCHAR2(50),

rokurodzenia DATE,

plec VARCHAR2(1)

);

ALTER TABLE klienci ADD CONSTRAINT klienci\_pk PRIMARY KEY ( idklienta );

CREATE TABLE pracownicy (

idpracownika NUMBER(7) NOT NULL,

imie VARCHAR2(20) NOT NULL,

nazwisko VARCHAR2(40) NOT NULL,

rokurodzenia DATE NOT NULL,

datazatrudnienia DATE NOT NULL,

idstanowiska NUMBER(7) NOT NULL

);

ALTER TABLE pracownicy ADD CONSTRAINT pracownicy\_pk PRIMARY KEY ( idpracownika );

CREATE TABLE produkty (

idproduktu NUMBER(7) NOT NULL,

nazwa VARCHAR2(100) NOT NULL,

cena NUMBER(7,2) NOT NULL,

cenahurtowa NUMBER(7,2) NOT NULL,

idkategorii NUMBER(7) NOT NULL

);

ALTER TABLE produkty ADD CONSTRAINT produkty\_pk PRIMARY KEY ( idproduktu );

CREATE TABLE rachunek\_poz (

ilosc NUMBER(6) NOT NULL,

cenapozycji NUMBER(7) NOT NULL,

idrachunku NUMBER(7) NOT NULL,

idproduktu NUMBER(7) NOT NULL

);

CREATE TABLE rachunki (

idrachunku NUMBER(7) NOT NULL,

datazakupu DATE NOT NULL,

idklienta NUMBER(7) NOT NULL,

idpracownika NUMBER(7) NOT NULL,

czytransport VARCHAR2(1) NOT NULL,

cenacalkowita NUMBER(7)

);

ALTER TABLE rachunki ADD CONSTRAINT rachunki\_pk PRIMARY KEY ( idrachunku );

CREATE TABLE stanowiska (

idstanowiska NUMBER(7) NOT NULL,

nazwa VARCHAR2(40) NOT NULL,

pensja NUMBER(7,2) NOT NULL

);

ALTER TABLE stanowiska ADD CONSTRAINT stanowiska\_pk PRIMARY KEY ( idstanowiska );

CREATE TABLE transport (

idtransport NUMBER(7) NOT NULL,

cena NUMBER(7,2),

adres VARCHAR2(50) NOT NULL,

datadostawy DATE,

idrachunku NUMBER(7) NOT NULL,

idpracownika NUMBER(7) NOT NULL

);

ALTER TABLE transport ADD CONSTRAINT transport\_pk PRIMARY KEY ( idtransport );

ALTER TABLE pracownicy ADD CONSTRAINT pracownicy\_stanowiska\_fk FOREIGN KEY ( idstanowiska )

REFERENCES stanowiska ( idstanowiska );

ALTER TABLE produkty ADD CONSTRAINT produkty\_kategorie\_fk FOREIGN KEY ( idkategorii )

REFERENCES kategorie ( idkategorii );

ALTER TABLE rachunek\_poz ADD CONSTRAINT rachunek\_poz\_produkty\_fk FOREIGN KEY ( idproduktu )

REFERENCES produkty ( idproduktu );

ALTER TABLE rachunek\_poz ADD CONSTRAINT rachunek\_poz\_rachunki\_fk FOREIGN KEY ( idrachunku )

REFERENCES rachunki ( idrachunku );

ALTER TABLE rachunki ADD CONSTRAINT rachunki\_klienci\_fk FOREIGN KEY ( idklienta )

REFERENCES klienci ( idklienta );

ALTER TABLE rachunki ADD CONSTRAINT rachunki\_pracownicy\_fk FOREIGN KEY ( idpracownika )

REFERENCES pracownicy ( idpracownika );

ALTER TABLE transport ADD CONSTRAINT transport\_pracownicy\_fk FOREIGN KEY ( idpracownika )

REFERENCES pracownicy ( idpracownika );

ALTER TABLE transport ADD CONSTRAINT transport\_rachunki\_fk FOREIGN KEY ( idrachunku )

REFERENCES rachunki ( idrachunku );

* + **Dodawanie danych(pracownicy, stanowiska, produkty, kategorie)**
    - **Tabele z transakcjami są za duże(zajmują ponad 90 stron)**

Insert into KATEGORIE (IDKATEGORII,NAZWA) values ('1','Buty rowerowe');

Insert into KATEGORIE (IDKATEGORII,NAZWA) values ('2','Akcesoria');

Insert into KATEGORIE (IDKATEGORII,NAZWA) values ('3','Kaski i ochraniacze');

Insert into KATEGORIE (IDKATEGORII,NAZWA) values ('4','Narzedzia');

Insert into KATEGORIE (IDKATEGORII,NAZWA) values ('5','Odziez');

Insert into KATEGORIE (IDKATEGORII,NAZWA) values ('6','Odzywki');

Insert into KATEGORIE (IDKATEGORII,NAZWA) values ('7','Ramy rowerowe');

Insert into KATEGORIE (IDKATEGORII,NAZWA) values ('8','Rowery');

Insert into STANOWISKA (IDSTANOWISKA,NAZWA,PENSJA) values ('1','Kierownik','3700');

Insert into STANOWISKA (IDSTANOWISKA,NAZWA,PENSJA) values ('2','Sprzedawca','2000');

Insert into STANOWISKA (IDSTANOWISKA,NAZWA,PENSJA) values ('3','Kurier','2500');

Insert into STANOWISKA (IDSTANOWISKA,NAZWA,PENSJA) values ('4','Ksiegowy','2500');

Insert into PRACOWNICY (IDPRACOWNIKA,IMIE,NAZWISKO,ROKURODZENIA,DATAZATRUDNIENIA,IDSTANOWISKA) values ('5','Karolina','Wolna',to\_date('97/11/13','RR/MM/DD'),to\_date('15/08/31','RR/MM/DD'),'4');

Insert into PRACOWNICY (IDPRACOWNIKA,IMIE,NAZWISKO,ROKURODZENIA,DATAZATRUDNIENIA,IDSTANOWISKA) values ('1','Jaroslaw','Kowalski',to\_date('89/08/27','RR/MM/DD'),to\_date('14/04/18','RR/MM/DD'),'2');

Insert into PRACOWNICY (IDPRACOWNIKA,IMIE,NAZWISKO,ROKURODZENIA,DATAZATRUDNIENIA,IDSTANOWISKA) values ('2','Jan','Apacki',to\_date('92/05/23','RR/MM/DD'),to\_date('16/10/04','RR/MM/DD'),'2');

Insert into PRACOWNICY (IDPRACOWNIKA,IMIE,NAZWISKO,ROKURODZENIA,DATAZATRUDNIENIA,IDSTANOWISKA) values ('3','Pawel','Skrzypkowski',to\_date('95/06/01','RR/MM/DD'),to\_date('14/04/18','RR/MM/DD'),'1');

Insert into PRACOWNICY (IDPRACOWNIKA,IMIE,NAZWISKO,ROKURODZENIA,DATAZATRUDNIENIA,IDSTANOWISKA) values ('4','Wojciech','Gębocki',to\_date('75/12/19','RR/MM/DD'),to\_date('15/08/31','RR/MM/DD'),'3');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('1','Buty SPD SH-M089','339,39','300','1');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('2','Buty SH-RP2','293','270','1');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('3','Buty SH-XC51','400,9','350','1');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('4','Buty MTB Cape żółto-czarne','699,99','610','1');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('5','Buty SPD SH-MT34','228','205','1');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('6','GPS rowerowy Edge Touring Plus + mapa Polski TOPO','1050','900','2');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('7','Zegarek Forerunner 920 XT','1482','1350','2');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('8','Bagażnik samochodowy na hak Como 3','1100','990','2');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('9','Licznik bezprzewodowy KRC 315W czarny','76,99','69','2');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('10','Wieszak Endo','199','179','2');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('11','Kask Slant czarny mat','134,9','115','3');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('12','Kask dziecięcy MV5-2','25,9','19','3');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('13','Kask MB-01','99,99','88','3');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('14','Kask MB-03','110','99','3');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('15','Kask Mirage','67,9','55','3');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('16','Preparat WD-40 aerozol 250ml','13','9,90','4');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('17','Stojak montaż. PRS-25 składany','1399','1250','4');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('18','Środek do czyszcznia łańcucha Drivetrain Cleaner 500ml','64','57','4');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('19','Stojak montażowy Biketrim','329,99','289','4');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('20','Preparat ochronny Bike Spray','35','29','4');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('21','Bluza męska Thermo','134,9','120','5');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('22','Skarpetki termoaktywne WARM','40','33','5');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('23','Spodenki długie Fobello','200','188','5');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('24','Bluza damska Extreme Wool','210','195','5');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('25','Rękawiczki Road Windstopper Thermo Split','270','245','5');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('26','Organiczny żel energetyczny','6,9','5','6');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('27','Krem Chamois zapobiegający obtarciom 125ml','55','46','6');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('28','Koncentrat napoju Isotonic Sports Drink','55','48','6');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('29','Rozgrzewający żel SOS','67','58','6');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('30','Gumy energetyczne 50g','10','8','6');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('31','Rama Volt','499','410','7');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('32','Rama damska Mitori','369','301','7');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('33','Rama Acid 27,5 niebiesko-czerwona','629','529','7');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('34','Rama Aim Pro 29','825','710','7');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('35','Rama Vision 29','1150','999','7');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('36','Rower Rambler Kid 20"','719,99','630','8');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('37','Rower EVADO 3.0 Dama','1499,99','1200','8');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('38','Kellys Rower Phanatic 90','5499,9','4100','8');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('39','Kellys Rower Phutura 10','2500','1999','8');

Insert into PRODUKTY (IDPRODUKTU,NAZWA,CENA,CENAHURTOWA,IDKATEGORII) values ('40','Rower Wasper','559,36','499','8');

* + **Procedury i sekwencje**

**Procedury umożliwiają:**

- DODAJ\_KLIENTA – umożliwia łatwe dodanie klienta na podstawie podanych danych

- DODAJ\_PRODUKT – umożliwia dodanie produktu

-DODAJ\_PRACOWNIKA – umożliwia dodanie pracownika

-GENERATOR\_TRANSAKCJI – umożliwia dodawanie x losowych rachunków, po y produktów w każdym, z możliwością wyboru czy generować nowych klientów czy rachunki dodawać istniejącym już w bazie klientom

CREATE SEQUENCE "SEQ\_KATEGORIE" MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 9 NOCACHE NOORDER NOCYCLE NOPARTITION ;

CREATE SEQUENCE "SEQ\_KLIENCI" MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 388 NOCACHE NOORDER NOCYCLE NOPARTITION ;

CREATE SEQUENCE "SEQ\_PRACOWNICY" MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 6 NOCACHE NOORDER NOCYCLE NOPARTITION ;

CREATE SEQUENCE "SEQ\_PRODUKTY" MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 41 NOCACHE NOORDER NOCYCLE NOPARTITION ;

CREATE SEQUENCE "SEQ\_RACHUNKI" MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 488 NOCACHE NOORDER NOCYCLE NOPARTITION ;

CREATE SEQUENCE "SEQ\_STANOWISKA" MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 5 NOCACHE NOORDER NOCYCLE NOPARTITION ;

CREATE SEQUENCE "SEQ\_TRANSPORT" MINVALUE 1 MAXVALUE 9999999999999999999999999999 INCREMENT BY 1 START WITH 247 NOCACHE NOORDER NOCYCLE NOPARTITION ;

CREATE OR REPLACE EDITIONABLE PROCEDURE "DODAJ\_KLIENTA"

(

imie IN klienci.imie%type default NULL,

nazwisko in klienci.nazwisko%type DEFAULT NULL,

adres in klienci.adres%type default NULL,

rokurodzenia in KLIENCI.ROKURODZENIA%type default NULL,

plec in KLIENCI.PLEC%type default NULL

) AS

BEGIN

insert into klienci values(seq\_klienci.nextval, imie, nazwisko, adres, rokurodzenia, plec);

END DODAJ\_KLIENTA;

/

CREATE OR REPLACE EDITIONABLE PROCEDURE "DODAJ\_PRACOWNIKA"

(

imie IN pracownicy.imie%type,

nazwisko in pracownicy.nazwisko%type,

rokurodzenia in pracownicy.rokurodzenia%type,

datazatrudnienia in pracownicy.datazatrudnienia%type,

stanowisko in PRACOWNICY.IDSTANOWISKA%type

) AS

BEGIN

insert into pracownicy(IDPRACOWNIKA, imie, nazwisko, rokurodzenia, datazatrudnienia, idstanowiska) values (seq\_pracownicy.nextval, imie, nazwisko, rokurodzenia, datazatrudnienia, stanowisko);

END DODAJ\_PRACOWNIKA;

/

CREATE OR REPLACE EDITIONABLE PROCEDURE "DODAJ\_PRODUKT"

(

nazwa IN produkty.nazwa%type

, cena IN produkty.cena%type,

cena\_hurtowa in produkty.cenahurtowa%type,

kategoria in PRODUKTY.IDKATEGORII%type

) AS

BEGIN

insert into produkty(idproduktu, nazwa, cena, cenahurtowa, idkategorii) values (seq\_produkty.nextval, nazwa, cena, cena\_hurtowa, kategoria);

END DODAJ\_PRODUKT;

/

create or replace PROCEDURE "GENERATOR\_TRANSAKCJI"

(

ilosc\_rachunkow IN NUMBER,

ilosc\_produktow IN NUMBER,

czy\_generowac\_klientow in boolean

) AS

liczba number;

los number;

los2 number;

losowy\_klient number;

losowy\_sprzedawca number;

nr\_rachunku number;

losowy\_produkt number;

cena\_produktu number;

suma\_cen number;

ilosc\_jednostek number;

imie klienci.imie%type;

nazwisko klienci.nazwisko%type;

adres klienci.adres%type;

data\_zak rachunki.datazakupu%type;

kurier transport.idpracownika%type;

BEGIN

if czy\_generowac\_klientow = false then

SELECT count(\*) into liczba from klienci;

if liczba = 0 then

dbms\_output.put\_line('Brak klientów w bazie');

else

for los in 1..ilosc\_rachunkow

loop

suma\_cen := 0;

Select idklienta, adres into losowy\_klient, adres from ( SELECT idklienta, adres from klienci order by DBMS\_RANDOM.value ) where rownum <= 1;

Select idpracownika into losowy\_sprzedawca from ( SELECT idpracownika from pracownicy where pracownicy.IDSTANOWISKA = 2 order by DBMS\_RANDOM.value ) where rownum < 2;

if(round(dbms\_random.value(0,1)) = 0) then

data\_zak := sysdate;

insert into rachunki values(seq\_rachunki.nextval, data\_zak, losowy\_klient, losowy\_sprzedawca, 'T', NULL);

select max(idrachunku) into nr\_rachunku from rachunki;

Select idpracownika into kurier from ( SELECT idpracownika from pracownicy where pracownicy.IDSTANOWISKA = 3 order by DBMS\_RANDOM.value ) where rownum < 2;

insert into transport values(seq\_transport.nextval, round(dbms\_random.value(20,300)), adres, to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(data\_zak, 'J')), to\_number(to\_char(data\_zak + 30, 'J'))))), 'J'), nr\_rachunku, kurier);

else

insert into rachunki values(seq\_rachunki.nextval, sysdate, losowy\_klient, losowy\_sprzedawca, 'N', NULL);

select max(idrachunku) into nr\_rachunku from rachunki;

end if;

for los2 in 1..ilosc\_produktow

loop

Select idproduktu, cena into losowy\_produkt, cena\_produktu from ( SELECT idproduktu, cena from produkty order by DBMS\_RANDOM.value ) where rownum <= 1;

ilosc\_jednostek := round(dbms\_random.value(1, 10));

insert into rachunek\_poz values(ilosc\_jednostek, cena\_produktu, nr\_rachunku, losowy\_produkt);

suma\_cen := suma\_cen+cena\_produktu\*ilosc\_jednostek;

end loop;

update rachunki set cenacalkowita=suma\_cen where idrachunku=nr\_rachunku;

end loop;

end if;

else

for los in 1..ilosc\_rachunkow

loop

suma\_cen := 0;

losowy\_klient := round(dbms\_random.value(0,9));

case losowy\_klient

when 0 then imie := 'Jakub';

when 1 then imie := 'Antoni';

when 2 then imie := 'Szymon';

when 3 then imie := 'Jan';

when 4 then imie := 'Filip';

when 5 then imie := 'Kacper';

when 6 then imie := 'Aleksander';

when 7 then imie := 'Franciszek';

when 8 then imie := 'Mikolaj';

when 9 then imie := 'Wojciech';

end case;

losowy\_klient := round(dbms\_random.value(0,9));

case losowy\_klient

when 0 then nazwisko := 'Suska';

when 1 then nazwisko := 'Lojek';

when 2 then nazwisko := 'Kondrat';

when 3 then nazwisko := 'Obloch';

when 4 then nazwisko := 'Zloto';

when 5 then nazwisko := 'Konsat';

when 6 then nazwisko := 'Kaszuba';

when 7 then nazwisko := 'Goniec';

when 8 then nazwisko := 'Alerw';

when 9 then nazwisko := 'Samos';

end case;

losowy\_klient := round(dbms\_random.value(0,9));

case losowy\_klient

when 0 then adres := 'Suska'; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 1 then adres := 'ul. Polna '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 2 then adres := 'ul. Lesna '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 3 then adres := 'ul. Sloneczna '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 4 then adres := 'ul. Krotka '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 5 then adres := 'ul. Szkolna '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 6 then adres := 'ul. Ogrodowa '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 7 then adres := 'ul. Lipowa '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 8 then adres := 'ul. Brzozowa '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 9 then adres := 'ul. Kwiatowa '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

end case;

losowy\_klient := round(dbms\_random.value(0,9));

case losowy\_klient

when 0 then adres :=concat(adres, ' Elk');

when 1 then adres :=concat(adres, ' Warszawa');

when 2 then adres :=concat(adres, ' Bialystok');

when 3 then adres :=concat(adres, ' Gdansk');

when 4 then adres :=concat(adres, ' Poznan');

when 5 then adres :=concat(adres, ' Wroclaw');

when 6 then adres :=concat(adres, ' Krakow');

when 7 then adres :=concat(adres, ' Katowice');

when 8 then adres :=concat(adres, ' Olsztyn');

when 9 then adres :=concat(adres, ' Rzeszow');

end case;

dodaj\_klienta(imie, nazwisko, adres, to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(sysdate - 20000, 'J')), to\_number(to\_char(sysdate - 7000, 'J'))))), 'J'), 'M');

Select max(idklienta) into losowy\_klient from klienci;

Select idpracownika into losowy\_sprzedawca from ( SELECT idpracownika from pracownicy where pracownicy.IDSTANOWISKA = 2 order by DBMS\_RANDOM.value ) where rownum < 2;

if(round(dbms\_random.value(0,1)) = 0) then

data\_zak := sysdate;

insert into rachunki values(seq\_rachunki.nextval, data\_zak, losowy\_klient, losowy\_sprzedawca, 'T', NULL);

select max(idrachunku) into nr\_rachunku from rachunki;

Select idpracownika into kurier from ( SELECT idpracownika from pracownicy where pracownicy.IDSTANOWISKA = 3 order by DBMS\_RANDOM.value ) where rownum < 2;

insert into transport values(seq\_transport.nextval, round(dbms\_random.value(20,300)), adres, to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(data\_zak, 'J')), to\_number(to\_char(data\_zak + 30, 'J'))))), 'J'), nr\_rachunku, kurier);

else

insert into rachunki values(seq\_rachunki.nextval, sysdate, losowy\_klient, losowy\_sprzedawca, 'N', NULL);

select max(idrachunku) into nr\_rachunku from rachunki;

end if;

for los2 in 1..ilosc\_produktow

loop

Select idproduktu, cena into losowy\_produkt, cena\_produktu from ( SELECT idproduktu, cena from produkty order by DBMS\_RANDOM.value ) where rownum <= 1;

ilosc\_jednostek := round(dbms\_random.value(1, 10));

insert into rachunek\_poz values(ilosc\_jednostek, cena\_produktu, nr\_rachunku, losowy\_produkt);

suma\_cen := suma\_cen+cena\_produktu\*ilosc\_jednostek;

end loop;

update rachunki set cenacalkowita=suma\_cen where idrachunku=nr\_rachunku;

end loop;

end if;

END GENERATOR\_TRANSAKCJI;

* + **Generator z zakresem dat (dzisiaj-rok; dzisiaj)**

Generator umożliwia to samo co GENERATOR\_TRANSAKCJI + generuje rachunki losowo w okresie od (dzisiaj-rok; dzisiaj). Przydał mi się do generowania analiz całorocznych.

CREATE OR REPLACE EDITIONABLE PROCEDURE "GENERATOR\_TRANSAKCJI\_NA\_ROK"

(

ilosc\_rachunkow IN NUMBER,

ilosc\_produktow IN NUMBER,

czy\_generowac\_klientow in boolean

) AS

liczba number;

los number;

los2 number;

losowy\_klient number;

losowy\_sprzedawca number;

nr\_rachunku number;

losowy\_produkt number;

cena\_produktu number;

suma\_cen number;

ilosc\_jednostek number;

imie klienci.imie%type;

nazwisko klienci.nazwisko%type;

adres klienci.adres%type;

data\_zak rachunki.datazakupu%type;

kurier transport.idpracownika%type;

BEGIN

if czy\_generowac\_klientow = false then

SELECT count(\*) into liczba from klienci;

if liczba = 0 then

dbms\_output.put\_line('Brak klientów w bazie');

else

for los in 1..ilosc\_rachunkow

loop

suma\_cen := 0;

Select idklienta, adres into losowy\_klient, adres from ( SELECT idklienta, adres from klienci order by DBMS\_RANDOM.value ) where rownum <= 1;

Select idpracownika into losowy\_sprzedawca from ( SELECT idpracownika from pracownicy where pracownicy.IDSTANOWISKA = 2 order by DBMS\_RANDOM.value ) where rownum < 2;

if(round(dbms\_random.value(0,1)) = 0) then

data\_zak := to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(sysdate - 365, 'J')), to\_number(to\_char(sysdate, 'J'))))), 'J');

insert into rachunki values(seq\_rachunki.nextval, data\_zak, losowy\_klient, losowy\_sprzedawca, 'T', NULL);

select max(idrachunku) into nr\_rachunku from rachunki;

Select idpracownika into kurier from ( SELECT idpracownika from pracownicy where pracownicy.IDSTANOWISKA = 3 order by DBMS\_RANDOM.value ) where rownum < 2;

insert into transport values(seq\_transport.nextval, round(dbms\_random.value(20,300)), adres, to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(data\_zak, 'J')), to\_number(to\_char(data\_zak + 30, 'J'))))), 'J'), nr\_rachunku, kurier);

else

insert into rachunki values(seq\_rachunki.nextval, to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(sysdate - 365, 'J')), to\_number(to\_char(sysdate, 'J'))))), 'J'), losowy\_klient, losowy\_sprzedawca, 'N', NULL);

select max(idrachunku) into nr\_rachunku from rachunki;

end if;

for los2 in 1..ilosc\_produktow

loop

Select idproduktu, cena into losowy\_produkt, cena\_produktu from ( SELECT idproduktu, cena from produkty order by DBMS\_RANDOM.value ) where rownum <= 1;

ilosc\_jednostek := round(dbms\_random.value(1, 10));

insert into rachunek\_poz values(ilosc\_jednostek, cena\_produktu, nr\_rachunku, losowy\_produkt);

suma\_cen := suma\_cen+cena\_produktu\*ilosc\_jednostek;

end loop;

update rachunki set cenacalkowita=suma\_cen where idrachunku=nr\_rachunku;

end loop;

end if;

else

for los in 1..ilosc\_rachunkow

loop

suma\_cen := 0;

losowy\_klient := round(dbms\_random.value(0,9));

case losowy\_klient

when 0 then imie := 'Jakub';

when 1 then imie := 'Antoni';

when 2 then imie := 'Szymon';

when 3 then imie := 'Jan';

when 4 then imie := 'Filip';

when 5 then imie := 'Kacper';

when 6 then imie := 'Aleksander';

when 7 then imie := 'Franciszek';

when 8 then imie := 'Mikolaj';

when 9 then imie := 'Wojciech';

end case;

losowy\_klient := round(dbms\_random.value(0,9));

case losowy\_klient

when 0 then nazwisko := 'Suska';

when 1 then nazwisko := 'Lojek';

when 2 then nazwisko := 'Kondrat';

when 3 then nazwisko := 'Obloch';

when 4 then nazwisko := 'Zloto';

when 5 then nazwisko := 'Konsat';

when 6 then nazwisko := 'Kaszuba';

when 7 then nazwisko := 'Goniec';

when 8 then nazwisko := 'Alerw';

when 9 then nazwisko := 'Samos';

end case;

losowy\_klient := round(dbms\_random.value(0,9));

case losowy\_klient

when 0 then adres := 'Suska'; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 1 then adres := 'ul. Polna '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 2 then adres := 'ul. Lesna '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 3 then adres := 'ul. Sloneczna '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 4 then adres := 'ul. Krotka '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 5 then adres := 'ul. Szkolna '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 6 then adres := 'ul. Ogrodowa '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 7 then adres := 'ul. Lipowa '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 8 then adres := 'ul. Brzozowa '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

when 9 then adres := 'ul. Kwiatowa '; adres :=concat(adres, to\_char(round(dbms\_random.value(1,150))));

end case;

losowy\_klient := round(dbms\_random.value(0,9));

case losowy\_klient

when 0 then adres :=concat(adres, ' Elk');

when 1 then adres :=concat(adres, ' Warszawa');

when 2 then adres :=concat(adres, ' Bialystok');

when 3 then adres :=concat(adres, ' Gdansk');

when 4 then adres :=concat(adres, ' Poznan');

when 5 then adres :=concat(adres, ' Wroclaw');

when 6 then adres :=concat(adres, ' Krakow');

when 7 then adres :=concat(adres, ' Katowice');

when 8 then adres :=concat(adres, ' Olsztyn');

when 9 then adres :=concat(adres, ' Rzeszow');

end case;

dodaj\_klienta(imie, nazwisko, adres, to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(sysdate - 20000, 'J')), to\_number(to\_char(sysdate - 7000, 'J'))))), 'J'), 'M');

Select max(idklienta) into losowy\_klient from klienci;

Select idpracownika into losowy\_sprzedawca from ( SELECT idpracownika from pracownicy where pracownicy.IDSTANOWISKA = 2 order by DBMS\_RANDOM.value ) where rownum < 2;

if(round(dbms\_random.value(0,1)) = 0) then

data\_zak := to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(sysdate - 365, 'J')), to\_number(to\_char(sysdate, 'J'))))), 'J');

insert into rachunki values(seq\_rachunki.nextval, data\_zak, losowy\_klient, losowy\_sprzedawca, 'T', NULL);

select max(idrachunku) into nr\_rachunku from rachunki;

Select idpracownika into kurier from ( SELECT idpracownika from pracownicy where pracownicy.IDSTANOWISKA = 3 order by DBMS\_RANDOM.value ) where rownum < 2;

insert into transport values(seq\_transport.nextval, round(dbms\_random.value(20,300)), adres, to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(data\_zak, 'J')), to\_number(to\_char(data\_zak + 30, 'J'))))), 'J'), nr\_rachunku, kurier);

else

insert into rachunki values(seq\_rachunki.nextval, to\_date(to\_char(round(dbms\_random.value(to\_number(to\_char(sysdate - 365, 'J')), to\_number(to\_char(sysdate, 'J'))))), 'J'), losowy\_klient, losowy\_sprzedawca, 'N', NULL);

select max(idrachunku) into nr\_rachunku from rachunki;

end if;

for los2 in 1..ilosc\_produktow

loop

Select idproduktu, cena into losowy\_produkt, cena\_produktu from ( SELECT idproduktu, cena from produkty order by DBMS\_RANDOM.value ) where rownum <= 1;

ilosc\_jednostek := round(dbms\_random.value(1, 10));

insert into rachunek\_poz values(ilosc\_jednostek, cena\_produktu, nr\_rachunku, losowy\_produkt);

suma\_cen := suma\_cen+cena\_produktu\*ilosc\_jednostek;

end loop;

update rachunki set cenacalkowita=suma\_cen where idrachunku=nr\_rachunku;

end loop;

end if;

END GENERATOR\_TRANSAKCJI\_NA\_ROK;

* + **Perspektywy**

Perspektywy umożliwiają:

- Roczna/Polroczna/Kwartalna/Miesieczna\_sprzedaz – obliczają sumę zarobków za sprzedane produktu w wybranym okresie

- Roczny/Polroczny/Kwartalny/Miesieczny\_zysk – obliczają zysk w wybranym okresie czasu

- Przeglad\_produktow – umożliwia przegląd produktów w kolejności od gwarantującego największy zysk

- Analiza\_klientów – umożliwa przegląd klientów zostawiających najwięcej pieniędzy w kasie sklepu

- Analiza\_biznesowa\_irmy – przedstawia wydatki na pensje, lokal(10000 zł/miesiąc), zysk ze sprzedaży, zysk obniżony o wydatki w skali roku

- faktura – umożliwia przedstawienie dowolnego rachunku

create or replace view Roczna\_sprzedaz as

select sum(cenacalkowita) as "Roczna Sprzedaż" from rachunki where DATAZAKUPU > sysdate - 366;

create or replace view Polroczna\_sprzedaz as

select sum(cenacalkowita) as "Polroczna Sprzedaż" from rachunki where DATAZAKUPU > sysdate - 183;

create or replace view Kwartalna\_sprzedaz as

select sum(cenacalkowita) as "Kwartalna Sprzedaż" from rachunki where DATAZAKUPU > sysdate - 92;

create or replace view Miesieczna\_sprzedaz as

select sum(cenacalkowita) as "Miesieczna Sprzedaż" from rachunki where DATAZAKUPU > sysdate - 31;

create or replace view Roczny\_zysk as

select sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as "Roczny zysk"

from rachunek\_poz, produkty, rachunki where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU and rachunek\_poz.IDRACHUNKU = rachunki.IDRACHUNKU

and rachunki.DATAZAKUPU > sysdate-366;

create or replace view Polroczny\_zysk as

select sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as "Polroczny zysk"

from rachunek\_poz, produkty, rachunki where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU and rachunek\_poz.IDRACHUNKU = rachunki.IDRACHUNKU

and rachunki.DATAZAKUPU > sysdate-183;

create or replace view Kwartalny\_zysk as

select sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as "Kwartalny zysk"

from rachunek\_poz, produkty, rachunki where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU and rachunek\_poz.IDRACHUNKU = rachunki.IDRACHUNKU

and rachunki.DATAZAKUPU > sysdate-92;

create or replace view Miesieczny\_zysk as

select sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as "Miesieczny zysk"

from rachunek\_poz, produkty, rachunki where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU and rachunek\_poz.IDRACHUNKU = rachunki.IDRACHUNKU

and rachunki.DATAZAKUPU > sysdate-31;

create or replace view Przeglad\_produktow as

select produkty.NAZWA, sum(ilosc) as "Ilosc sprzedanych produktow", sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc) as Sprzedaz,

sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as Zysk

from rachunek\_poz, produkty

where rachunek\_poz.IDPRODUKTU=produkty.IDPRODUKTU group by produkty.nazwa

order by sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) desc;

create or replace view Analiza\_klientow as

select klienci.imie as "Imie", klienci.NAZWISKO as "Nazwisko", count(rachunki.idrachunku) as " Ilosc rachunkow",

sum(rachunki.cenacalkowita) as "Suma rachunków", sum(rachunki.cenacalkowita)/count(rachunki.idrachunku) as "Srednia kwota na rachunku"

from klienci, rachunki where rachunki.IDKLIENTA=klienci.IDKLIENTA group by klienci.IDKLIENTA, klienci.IMIE, klienci.NAZWISKO

order by sum(rachunki.cenacalkowita)/count(rachunki.idrachunku) desc;

create or replace view Analiza\_biznesowa\_firmy as

select (select \* from Roczny\_zysk) as "Roczny zysk",

(select sum(pensja) from stanowiska, pracownicy where pracownicy.idstanowiska = stanowiska.idstanowiska)\*12 as "Roczne wydatki na pensje",

10000\*12 as "Roczne wydatki na sklep",

(select \* from Roczny\_zysk) - ((select sum(pensja) from stanowiska, pracownicy where pracownicy.idstanowiska = stanowiska.idstanowiska)\*12 + 10000\*12)

as "Roczny zysk z kosztami"

from dual;

create or replace view faktura as

select rachunki.idrachunku, klienci.IMIE as "Imie klienta", klienci.NAZWISKO as "Nazwisko klienta", klienci.ADRES as "Adres klienta", pracownicy.imie as "Imie pracownika",

pracownicy.nazwisko as "Nazwisko pracownika", rachunki.DATAZAKUPU as "Data zakupu", rachunki.CENACALKOWITA as "Cena calkowita",

rachunek\_poz.ILOSC as "Ilosc", rachunek\_poz.CENAPOZYCJI "Cena pozycji", produkty.NAZWA as "Nazwa produktu"

from rachunki, rachunek\_poz, klienci,pracownicy, produkty where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU

and rachunki.IDRACHUNKU = rachunek\_poz.IDRACHUNKU and rachunki.IDPRACOWNIKA=pracownicy.IDPRACOWNIKA and rachunki.IDKLIENTA=klienci.IDKLIENTA;

* + **Joby**

Joby wykonują:

-JOB\_MALI – wykonuje 3 transakcje po 1 produkcie na rachunku, co godzine w godzinach 8-20.

-JOB\_DUZI – wykonuje 1 transakcje po 5 produktów na rachunku o godzinie 10.

BEGIN

DBMS\_SCHEDULER.CREATE\_JOB (

job\_name => 'JOB\_MALI',

job\_type => 'PLSQL\_BLOCK',

job\_action => 'begin

GENERATOR\_TRANSAKCJI(3,1,TRUE);

END;',

number\_of\_arguments => 0,

start\_date => NULL,

repeat\_interval => 'FREQ=DAILY; BYHOUR=8,9,10,11,12,13,14,15,16,17,18,19,20;',

end\_date => NULL,

enabled => FALSE,

auto\_drop => FALSE,

comments => 'cogodzinne zakupy zwyklych klientow');

DBMS\_SCHEDULER.SET\_ATTRIBUTE(

name => 'JOB\_MALI',

attribute => 'store\_output', value => TRUE);

DBMS\_SCHEDULER.SET\_ATTRIBUTE(

name => 'JOB\_MALI',

attribute => 'logging\_level', value => DBMS\_SCHEDULER.LOGGING\_OFF);

DBMS\_SCHEDULER.enable(

name => 'JOB\_MALI');

END;

/

BEGIN

DBMS\_SCHEDULER.CREATE\_JOB (

job\_name => 'JOB\_DUZI',

job\_type => 'PLSQL\_BLOCK',

job\_action => 'begin

GENERATOR\_TRANSAKCJI(1,6,FALSE);

END;',

number\_of\_arguments => 0,

start\_date => NULL,

repeat\_interval => 'FREQ=DAILY; BYHOUR=10',

end\_date => NULL,

enabled => FALSE,

auto\_drop => FALSE,

comments => 'codzienne zakupy waznych klientow');

DBMS\_SCHEDULER.SET\_ATTRIBUTE(

name => 'JOB\_DUZI',

attribute => 'store\_output', value => TRUE);

DBMS\_SCHEDULER.SET\_ATTRIBUTE(

name => 'JOB\_DUZI',

attribute => 'logging\_level', value => DBMS\_SCHEDULER.LOGGING\_OFF);

DBMS\_SCHEDULER.enable(

name => 'JOB\_DUZI');

END;

* + **Usuwanie projektu**

Umożliwia usunięcie całego projektu

drop table transport;

drop table rachunek\_poz;

drop table produkty;

drop table rachunki;

drop table pracownicy;

drop table stanowiska;

drop table klienci;

drop table kategorie;

drop procedure DODAJ\_KLIENTA;

drop procedure DODAJ\_PRACOWNIKA;

drop procedure DODAJ\_PRODUKT;

drop procedure GENERATOR\_TRANSAKCJI;

drop sequence SEQ\_KATEGORIE;

drop sequence SEQ\_KLIENCI;

drop sequence SEQ\_PRACOWNICY;

drop sequence SEQ\_PRODUKTY;

drop sequence SEQ\_RACHUNKI;

drop sequence SEQ\_TRANSPORT;

drop sequence SEQ\_STANOWISKA;

drop view Roczna\_sprzedaz;

drop view Polroczna\_sprzedaz;

drop view Kwartalna\_sprzedaz;

drop view Miesieczna\_sprzedaz;

drop view Roczny\_zysk;

drop view Polroczny\_zysk;

drop view Kwartalny\_zysk;

drop view Miesieczny\_zysk;

drop view Przeglad\_produktow;

drop view Analiza\_klientow;

drop view Analiza\_biznesowa\_firmy;

drop view faktura;

begin

DBMS\_SCHEDULER.drop\_job (job\_name => 'JOB\_MALI');

DBMS\_SCHEDULER.drop\_job (job\_name => 'JOB\_DUZI');

end;

* **Baza Sybase(działanie opisane przy okazji bazy Oracle)**
  + **Tworzenie tabel**

create table kategorie(

idkategorii numeric(7) primary key,

nazwa varchar(40) not null

);

create table klienci (

idklienta numeric(7) primary key,

imie varchar(20),

nazwisko varchar(40),

adres varchar(50),

rokurodzenia date,

plec varchar(1)

);

create table stanowiska (

idstanowiska numeric(7) primary key,

nazwa varchar(40) not null,

pensja numeric(7,2) not null

);

create table pracownicy (

idpracownika numeric(7) primary key,

imie varchar(20) not null,

nazwisko varchar(40) not null,

rokurodzenia date not null,

datazatrudnienia date not null,

idstanowiska numeric(7) references stanowiska(idstanowiska)

);

create table rachunki (

idrachunku numeric(7) primary key,

datazakupu date not null,

idklienta numeric(7) references klienci(idklienta),

idpracownika numeric(7) references pracownicy(idpracownika),

czytransport varchar(1) not null,

cenacalkowita numeric(7)

);

create table produkty (

idproduktu numeric(7) primary key,

nazwa varchar(100) not null,

cena numeric(7,2) not null,

cenahurtowa numeric(7,2) not null,

idkategorii numeric(7) references kategorie(idkategorii)

);

create table rachunek\_poz (

ilosc numeric(6) not null,

cenapozycji numeric(7) not null,

idrachunku numeric(7) references rachunki(idrachunku),

idproduktu numeric(7) references produkty(idproduktu)

);

create table transport (

idtransport numeric(7) primary key,

cena numeric(7,2),

adres varchar(50) not null,

datadostawy date,

idrachunku numeric(7) references rachunki(idrachunku),

idpracownika numeric(7) references pracownicy(idpracownika)

);

* + **Dodawanie danych(pracownicy, stanowiska, produkty, kategorie)**

Insert into kategorie (idkategorii,nazwa) values ('1','Buty rowerowe');

Insert into kategorie (idkategorii,nazwa) values ('2','Akcesoria');

Insert into kategorie (idkategorii,nazwa) values ('3','Kaski i ochraniacze');

Insert into kategorie (idkategorii,nazwa) values ('4','Narzedzia');

Insert into kategorie (idkategorii,nazwa) values ('5','Odziez');

Insert into kategorie (idkategorii,nazwa) values ('6','Odzywki');

Insert into kategorie (idkategorii,nazwa) values ('7','Ramy rowerowe');

Insert into kategorie (idkategorii,nazwa) values ('8','Rowery');

Insert into stanowiska (idstanowiska,nazwa,pensja) values ('1','Kierownik','3700');

Insert into stanowiska (idstanowiska,nazwa,pensja) values ('2','Sprzedawca','2000');

Insert into stanowiska (idstanowiska,nazwa,pensja) values ('3','Kurier','2500');

Insert into stanowiska (idstanowiska,nazwa,pensja) values ('4','Ksiegowy','2500');

Insert into pracownicy (idpracownika,imie,nazwisko,rokurodzenia,datazatrudnienia,idstanowiska) values ('5','Karolina','Wolna',date('97/11/13'),date('15/08/31'),'4');

Insert into pracownicy (idpracownika,imie,nazwisko,rokurodzenia,datazatrudnienia,idstanowiska) values ('1','Jaroslaw','Kowalski',date('89/08/27'),date('14/04/18'),'2');

Insert into pracownicy (idpracownika,imie,nazwisko,rokurodzenia,datazatrudnienia,idstanowiska) values ('2','Jan','Apacki',date('92/05/23'),date('16/10/04'),'2');

Insert into pracownicy (idpracownika,imie,nazwisko,rokurodzenia,datazatrudnienia,idstanowiska) values ('3','Pawel','Skrzypkowski',date('95/06/01'),date('14/04/18'),'1');

Insert into pracownicy (idpracownika,imie,nazwisko,rokurodzenia,datazatrudnienia,idstanowiska) values ('4','Wojciech','Gębocki',date('75/12/19'),date('15/08/31'),'3');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('1','Buty SPD SH-M089','339.39','300','1');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('2','Buty SH-RP2','293','270','1');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('3','Buty SH-XC51','400.9','350','1');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('4','Buty MTB Cape żółto-czarne','699.99','610','1');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('5','Buty SPD SH-MT34','228','205','1');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('6','GPS rowerowy Edge Touring Plus + mapa Polski TOPO','1050','900','2');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('7','Zegarek Forerunner 920 XT','1482','1350','2');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('8','Bagażnik samochodowy na hak Como 3','1100','990','2');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('9','Licznik bezprzewodowy KRC 315W czarny','76.99','69','2');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('10','Wieszak Endo','199','179','2');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('11','Kask Slant czarny mat','134.9','115','3');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('12','Kask dziecięcy MV5-2','25.9','19','3');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('13','Kask MB-01','99.99','88','3');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('14','Kask MB-03','110','99','3');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('15','Kask Mirage','67.9','55','3');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('16','Preparat WD-40 aerozol 250ml','13','9.90','4');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('17','Stojak montaż. PRS-25 składany','1399','1250','4');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('18','Środek do czyszcznia łańcucha Drivetrain Cleaner 500ml','64','57','4');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('19','Stojak montażowy Biketrim','329.99','289','4');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('20','Preparat ochronny Bike Spray','35','29','4');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('21','Bluza męska Thermo','134.9','120','5');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('22','Skarpetki termoaktywne WARM','40','33','5');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('23','Spodenki długie Fobello','200','188','5');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('24','Bluza damska Extreme Wool','210','195','5');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('25','Rękawiczki Road Windstopper Thermo Split','270','245','5');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('26','Organiczny żel energetyczny','6.9','5','6');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('27','Krem Chamois zapobiegający obtarciom 125ml','55','46','6');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('28','Koncentrat napoju Isotonic Sports Drink','55','48','6');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('29','Rozgrzewający żel SOS','67','58','6');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('30','Gumy energetyczne 50g','10','8','6');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('31','Rama Volt','499','410','7');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('32','Rama damska Mitori','369','301','7');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('33','Rama Acid 27,5 niebiesko-czerwona','629','529','7');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('34','Rama Aim Pro 29','825','710','7');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('35','Rama Vision 29','1150','999','7');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('36','Rower Rambler Kid 20"','719.99','630','8');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('37','Rower EVADO 3.0 Dama','1499.99','1200','8');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('38','Kellys Rower Phanatic 90','5499.9','4100','8');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('39','Kellys Rower Phutura 10','2500','1999','8');

Insert into produkty (idproduktu,nazwa,cena,cenahurtowa,idkategorii) values ('40','Rower Wasper','559.36','499','8');

* + **Sekwencje**

CREATE SEQUENCE "SEQ\_KATEGORIE" MINVALUE 1 MAXVALUE 99999999999 INCREMENT BY 1 START WITH 9;

CREATE SEQUENCE "SEQ\_KLIENCI" MINVALUE 1 MAXVALUE 99999999999 INCREMENT BY 1 START WITH 369;

CREATE SEQUENCE "SEQ\_PRACOWNICY" MINVALUE 1 MAXVALUE 99999999999 INCREMENT BY 1 START WITH 6;

CREATE SEQUENCE "SEQ\_PRODUKTY" MINVALUE 1 MAXVALUE 99999999999 INCREMENT BY 1 START WITH 41;

CREATE SEQUENCE "SEQ\_RACHUNKI" MINVALUE 1 MAXVALUE 99999999999 INCREMENT BY 1 START WITH 469;

CREATE SEQUENCE "SEQ\_STANOWISKA" MINVALUE 1 MAXVALUE 99999999999 INCREMENT BY 1 START WITH 5;

CREATE SEQUENCE "SEQ\_TRANSPORT" MINVALUE 1 MAXVALUE 99999999999 INCREMENT BY 1 START WITH 236;

* + **Perspektywy**

create or replace view Roczna\_sprzedaz as

select sum(cenacalkowita) as "Roczna Sprzedaż" from rachunki where DATAZAKUPU > getdate() - 366;

create or replace view Polroczna\_sprzedaz as

select sum(cenacalkowita) as "Polroczna Sprzedaż" from rachunki where DATAZAKUPU > getdate() - 183;

create or replace view Kwartalna\_sprzedaz as

select sum(cenacalkowita) as "Kwartalna Sprzedaż" from rachunki where DATAZAKUPU > getdate() - 92;

create or replace view Miesieczna\_sprzedaz as

select sum(cenacalkowita) as "Miesieczna Sprzedaż" from rachunki where DATAZAKUPU > getdate() - 31;

create or replace view Roczny\_zysk as

select sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as "Roczny zysk"

from rachunek\_poz, produkty, rachunki where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU and rachunek\_poz.IDRACHUNKU = rachunki.IDRACHUNKU

and rachunki.DATAZAKUPU > getdate()-366;

create or replace view Polroczny\_zysk as

select sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as "Polroczny zysk"

from rachunek\_poz, produkty, rachunki where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU and rachunek\_poz.IDRACHUNKU = rachunki.IDRACHUNKU

and rachunki.DATAZAKUPU > getdate()-183;

create or replace view Kwartalny\_zysk as

select sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as "Kwartalny zysk"

from rachunek\_poz, produkty, rachunki where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU and rachunek\_poz.IDRACHUNKU = rachunki.IDRACHUNKU

and rachunki.DATAZAKUPU > getdate()-92;

create or replace view Miesieczny\_zysk as

select sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as "Miesieczny zysk"

from rachunek\_poz, produkty, rachunki where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU and rachunek\_poz.IDRACHUNKU = rachunki.IDRACHUNKU

and rachunki.DATAZAKUPU > getdate()-31;

create or replace view Przeglad\_produktow as

select produkty.NAZWA, sum(ilosc) as "Ilosc sprzedanych produktow", sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc) as Sprzedaz,

sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) as Zysk

from rachunek\_poz, produkty

where rachunek\_poz.IDPRODUKTU=produkty.IDPRODUKTU group by produkty.nazwa

order by sum(rachunek\_poz.CENAPOZYCJI\*rachunek\_poz.ilosc - produkty.cenahurtowa\*rachunek\_poz.ilosc) desc;

create or replace view Analiza\_klientow as

select klienci.imie as "Imie", klienci.NAZWISKO as "Nazwisko", count(rachunki.idrachunku) as " Ilosc rachunkow",

sum(rachunki.cenacalkowita) as "Suma rachunków", sum(rachunki.cenacalkowita)/count(rachunki.idrachunku) as "Srednia kwota na rachunku"

from klienci, rachunki where rachunki.IDKLIENTA=klienci.IDKLIENTA group by klienci.IDKLIENTA, klienci.IMIE, klienci.NAZWISKO

order by sum(rachunki.cenacalkowita)/count(rachunki.idrachunku) desc;

create or replace view Analiza\_biznesowa\_firmy as

select (select \* from Roczny\_zysk) as "Roczny zysk",

(select sum(pensja) from stanowiska, pracownicy where pracownicy.idstanowiska = stanowiska.idstanowiska)\*12 as "Roczne wydatki na pensje",

10000\*12 as "Roczne wydatki na sklep",

(select \* from Roczny\_zysk) - ((select sum(pensja) from stanowiska, pracownicy where pracownicy.idstanowiska = stanowiska.idstanowiska)\*12 + 10000\*12)

as "Roczny zysk z kosztami";

create or replace view faktura as

select rachunki.idrachunku, klienci.IMIE as "Imie klienta", klienci.NAZWISKO as "Nazwisko klienta", klienci.ADRES as "Adres klienta", pracownicy.imie as "Imie pracownika",

pracownicy.nazwisko as "Nazwisko pracownika", rachunki.DATAZAKUPU as "Data zakupu", rachunki.CENACALKOWITA as "Cena calkowita",

rachunek\_poz.ILOSC as "Ilosc", rachunek\_poz.CENAPOZYCJI "Cena pozycji", produkty.NAZWA as "Nazwa produktu"

from rachunki, rachunek\_poz, klienci,pracownicy, produkty where produkty.IDPRODUKTU=rachunek\_poz.IDPRODUKTU

and rachunki.IDRACHUNKU = rachunek\_poz.IDRACHUNKU and rachunki.IDPRACOWNIKA=pracownicy.IDPRACOWNIKA and rachunki.IDKLIENTA=klienci.IDKLIENTA;

* + **Usuwanie projektu**

drop table transport;

drop table rachunek\_poz;

drop table produkty;

drop table rachunki;

drop table pracownicy;

drop table stanowiska;

drop table klienci;

drop table kategorie;

drop procedure DODAJ\_KLIENTA;

drop procedure DODAJ\_PRACOWNIKA;

drop procedure DODAJ\_PRODUKT;

drop procedure GENERATOR\_TRANSAKCJI;

drop sequence SEQ\_KATEGORIE;

drop sequence SEQ\_KLIENCI;

drop sequence SEQ\_PRACOWNICY;

drop sequence SEQ\_PRODUKTY;

drop sequence SEQ\_RACHUNKI;

drop sequence SEQ\_TRANSPORT;

drop sequence SEQ\_STANOWISKA;

drop view Roczna\_sprzedaz;

drop view Polroczna\_sprzedaz;

drop view Kwartalna\_sprzedaz;

drop view Miesieczna\_sprzedaz;

drop view Roczny\_zysk;

drop view Polroczny\_zysk;

drop view Kwartalny\_zysk;

drop view Miesieczny\_zysk;

drop view Przeglad\_produktow;

drop view Analiza\_klientow;

drop view Analiza\_biznesowa\_firmy;

drop view faktura;

1. **Instrukcje instalacji projektu**

* **Instalacja bazy Oracle**

Bazę można wgrać za pomocą skryptu wdrożeniowego:

/oracle/skrypt\_wdrozeniowy.sql

Lub za pomocą pojedynczych plików w kolejności:

/oracle/pojedyncze skrypty/tabele.sql

/oracle/pojedyncze skrypty/dodawanie\_danych.sql

/oracle/pojedyncze skrypty/transakcje.sql

/oracle/pojedyncze skrypty/procedury\_sekwencje.sql

/oracle/pojedyncze skrypty/generator\_na\_rok.sql

/oracle/pojedyncze skrypty/perspektywy.sql

/oracle/pojedyncze skrypty/Schedule.sql

* **Instalacja bazy Sybase**

Bazę można wgrać za pomocą skryptu wdrożeniowego:

/SAP/skrypt\_wdrozeniowy.sql

Lub za pomocą pojedynczych plików w kolejności:

/SAP/pojedyncze skrypty/tabele.sql

/SAP/pojedyncze skrypty/dodawanie\_danych.sql

/SAP/pojedyncze skrypty/transakcje.sql

/SAP/pojedyncze skrypty/sekwencje.sql

/SAP/pojedyncze skrypty/perspektywy.sql

* **Deinstalacja bazy Oracle**

Deinstalacja odbywa się za pomocą skryptu:

/oracle/drop.sql

* **Deinstalacja bazy Sybase**

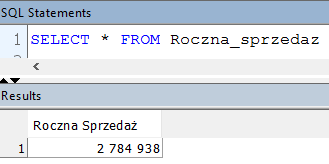
Deinstalacja odbywa się za pomocą skryptu:

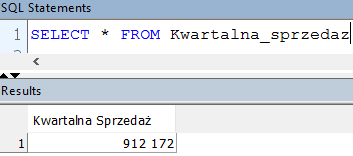
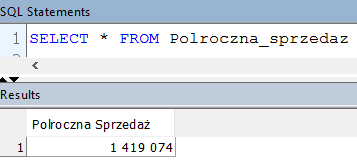
/SAP/drop.sql

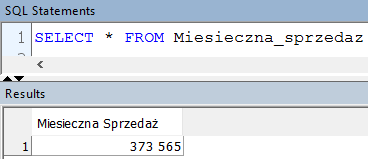
1. **Działanie perspektyw oraz raporty w iReport Designer**

* **Działanie perspektyw w Sybase**

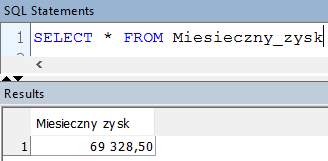
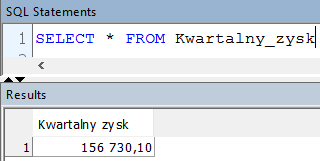
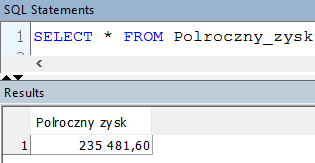
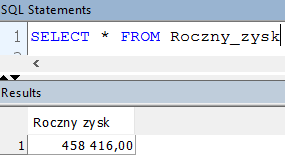
Sprzedaż:

****

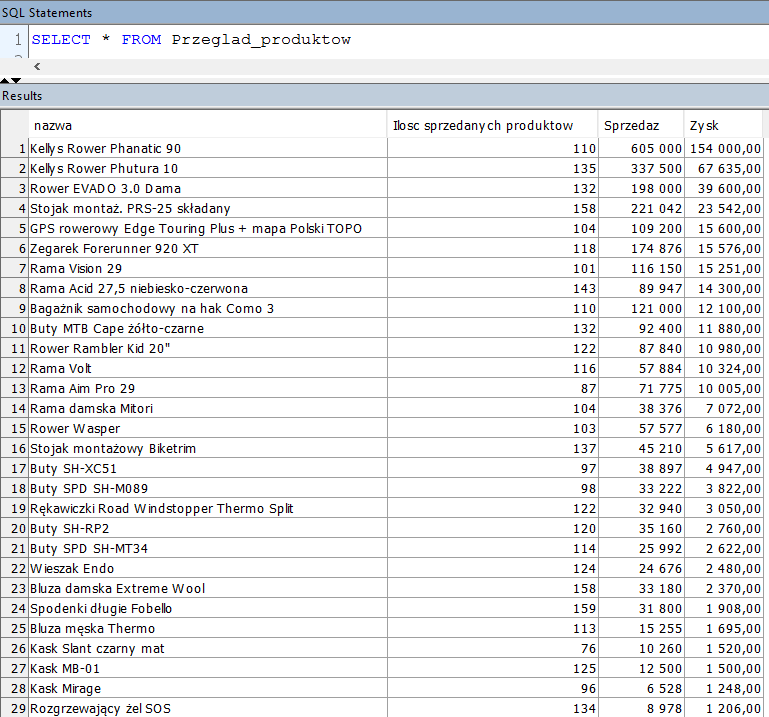
****

****

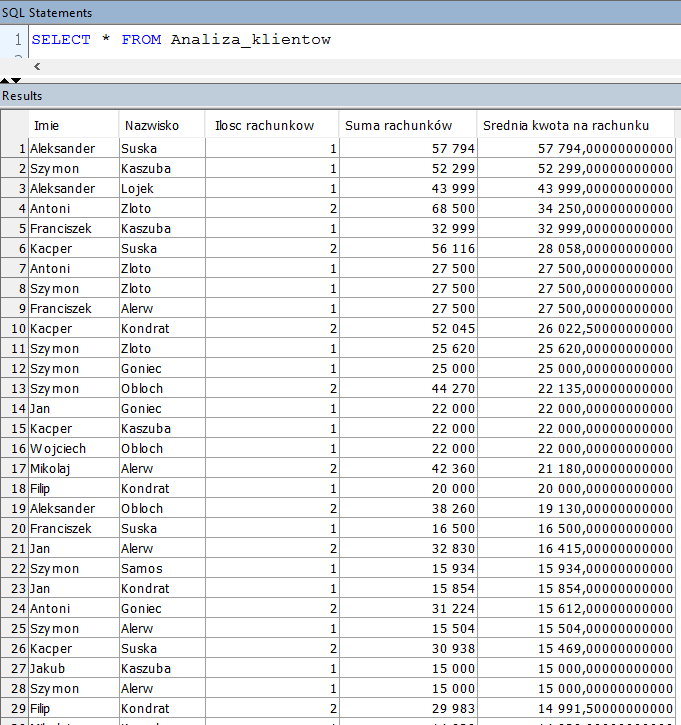
Zysk:



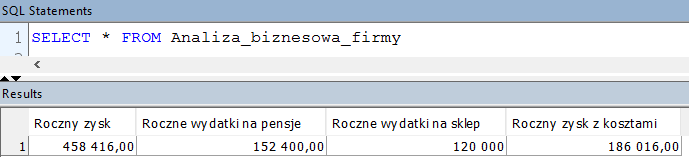
Przegląd produktów:



Analiza klientów:



Analiza biznesowa firmy:



Faktura nr 431:



* **Raporty na podstawie bazy ORACLE**

- Przegląd produktów na podstawie perspektywy Przeglad\_produktow

- Przegląd klientów na podstawie perspektywy Analiza\_klientow

- Pojedynczy rachunek na podstawie perspektywy faktura

- Analiza biznesowa firmy na podstawie perspektywy Analiza\_biznesowa\_firmy