

Catering

Zespół 2 w składzie:

- Michał Kindeusz
- Hubert Soroka
- Richard Staszkievicz
- Gabriela Topczewska
- Hubert Truszeński
- Maciej Tymoftejewicz

Organizacja pracy

Michał Kindeusz – Programista PL/SQL

Hubert Soroka – Kierownik / Właściciel Zadania

Richard Staszekiewicz – Projektant Raportowania i Analizy Danych

Gabriela Topczewska – Projektant Struktury Danych

Hubert Truszeński – Projektant Wymiany Danych

Maciej Tymoftejewicz – Projektant Aplikacji CRUD w Apex

Opis zadania

- Firma oferująca usługi cateringowe potrzebuje systemu wspomagającego działanie przedsiębiorstwa.
- Klienci, którzy są osobami prywatnymi lub firmami, składają zamówienia na dania z oferty.
- Dania wymagają określonej liczby różnych składników do wytworzenia.
- Składniki są składowane w firmowym magazynie i uzupełniane dostawami przez zewnętrznych dostawców towarów spożywczych.
- System przechowuje dane o fakturach wystawianych przez dostawców towarów oraz wystawianych klientom w związku z zamówieniami.

Opis zadania

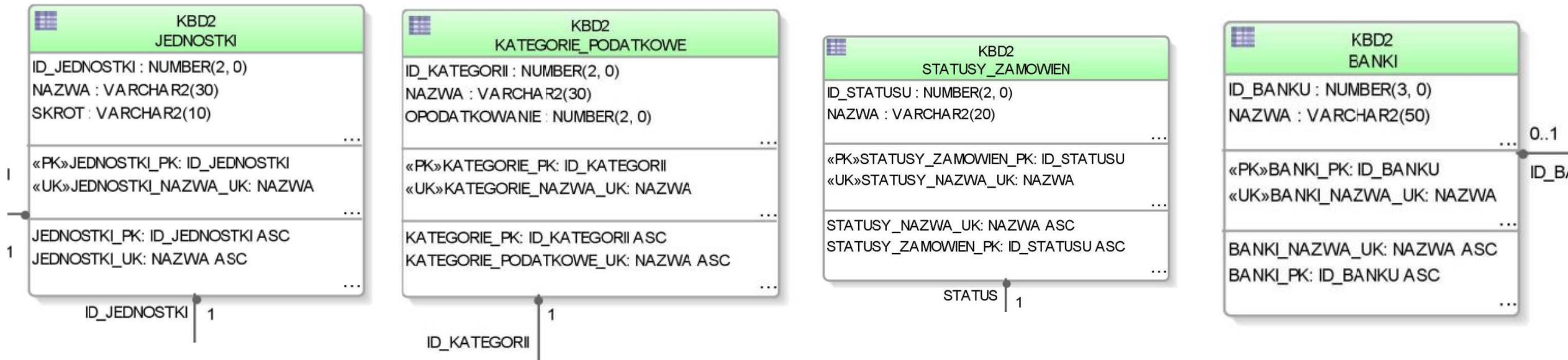
- Klienci, którzy są **osobami prywatnymi** lub **firmami**, składają **zamówienia** na **dania** z oferty.
- Dania wymagają określonej liczby różnych **składników** do wytworzenia.
- Składniki są składowane w firmowym magazynie i uzupełniane **dostawami** przez zewnętrznych **dostawców** towarów spożywczych.
- System przechowuje dane o **fakturach** wystawianych przez dostawców towarów oraz wystawianych klientom w związku z zamówieniami.

Struktura Danych

- Baza typu OLTP
- Wykorzystane zaawansowane narzędzia:
 - Index Organized Tables (IOT)
 - Partycje
 - Indeks funkcyjny



IO Ty – Słowniki



Ludzie



KBD2 KONTRAHENCI

ID_KONTRAHENTA : NUMBER(5, 0)
IMIE : VARCHAR2(20)
NAZWISKO : VARCHAR2(40)
NAZWA : VARCHAR2(50)
NIP : CHAR(10)
MAIL : VARCHAR2(40)
TELEFON : VARCHAR2(15)
ID_BANKU : NUMBER(3, 0)
NR_KONTA : VARCHAR2(26)
CZY_FIRMA : CHAR(1)
MIEJSCOWOSC : VARCHAR2(40)
KOD_POCZTOWY : VARCHAR2(6)
NAZWA_ULICY : VARCHAR2(40)
NR_BUDYNKU : VARCHAR2(10)
NR_LOKALU : VARCHAR2(10)

«PK»KONTRAHENCI_PK: ID_KONTRAHENTA

«UK»KONTRAHENCI_MAIL_UK: MAIL

«UK»KONTRAHENCI_NIP_UK: NIP

«FK»KONTRAHENCI_BANK_FK: ID_BANKU

«Check»KONTRAHENCI_CZY_FIRMA_CHK: CZY_FIRMA IN ('0', '1')

«Check»KONTRAHENCI_WYMAGANE_DANE: 1 = (CASE WHEN CZY_FIRMA = 1 AND NIP IS NOT NULL AND NAZWA IS NOT NULL THEN 1 WHEN CZY_FIRMA = 0 AND IMIE IS NOT NULL AND NAZWISKO IS NOT NULL THEN 1 ELSE 0 END)

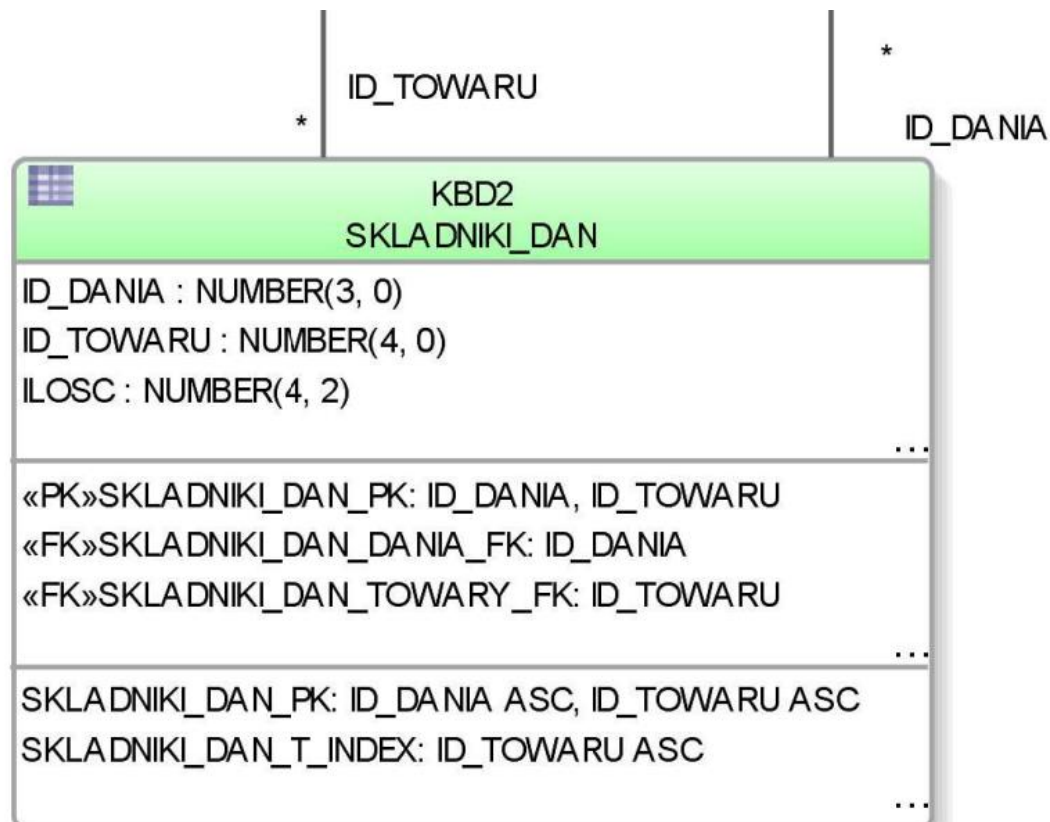
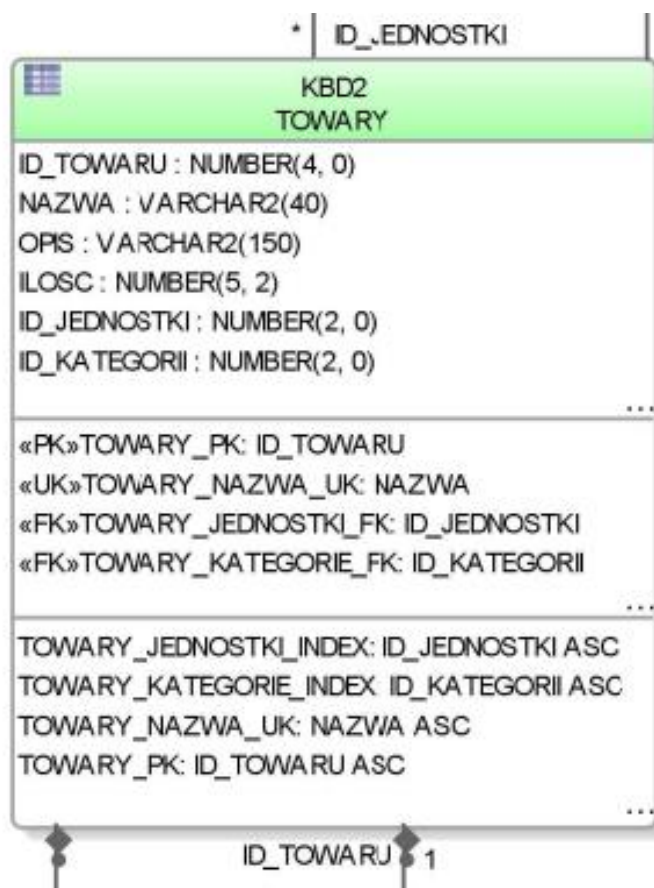
* KONTRAHENCI_BANKI_INDEX: ID_BANKU ASC

KONTRAHENCI_MAIL_UK: MAIL ASC

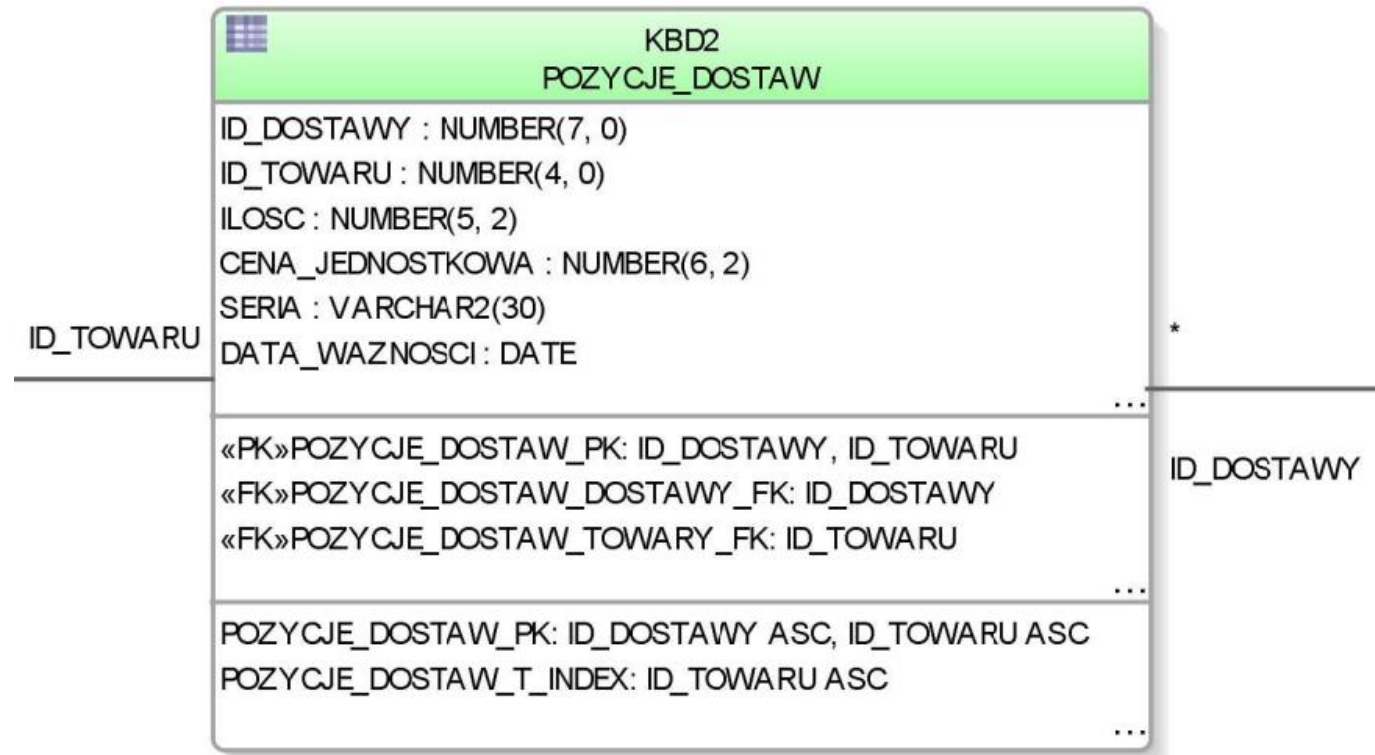
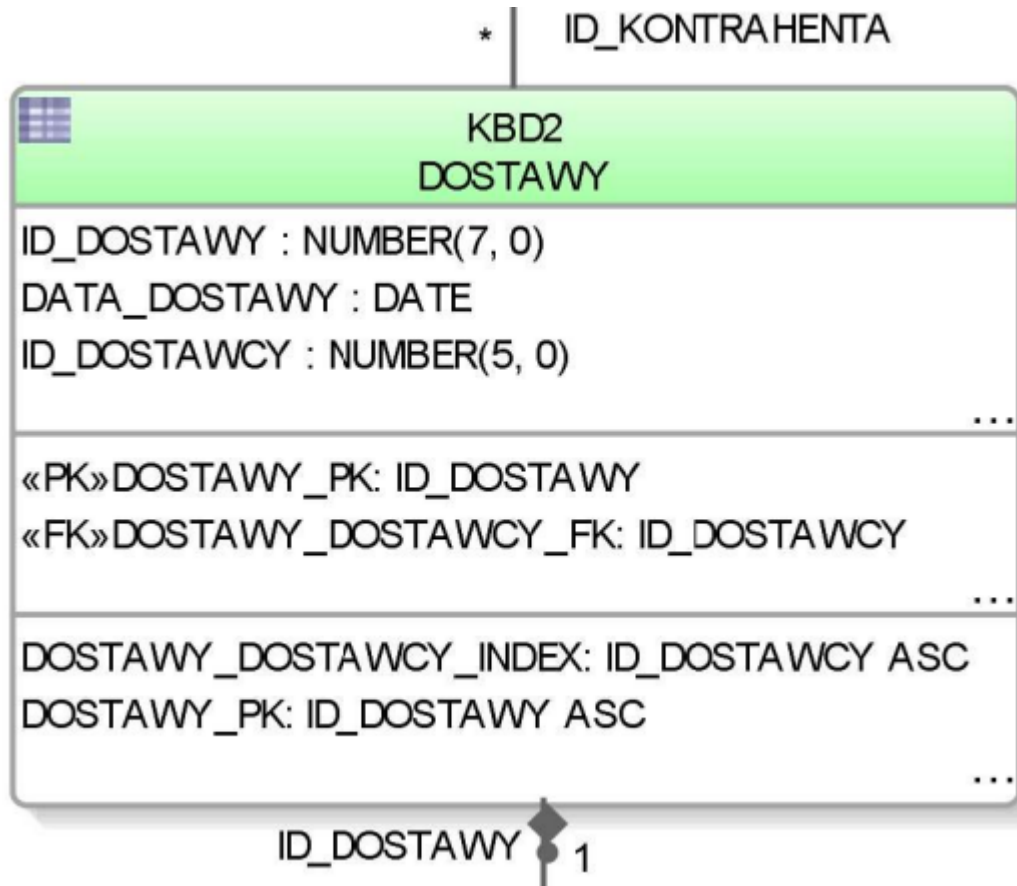
KONTRAHENCI_NIP_UK: NIP ASC

KONTRAHENCI_PK: ID_KONTRAHENTA ASC

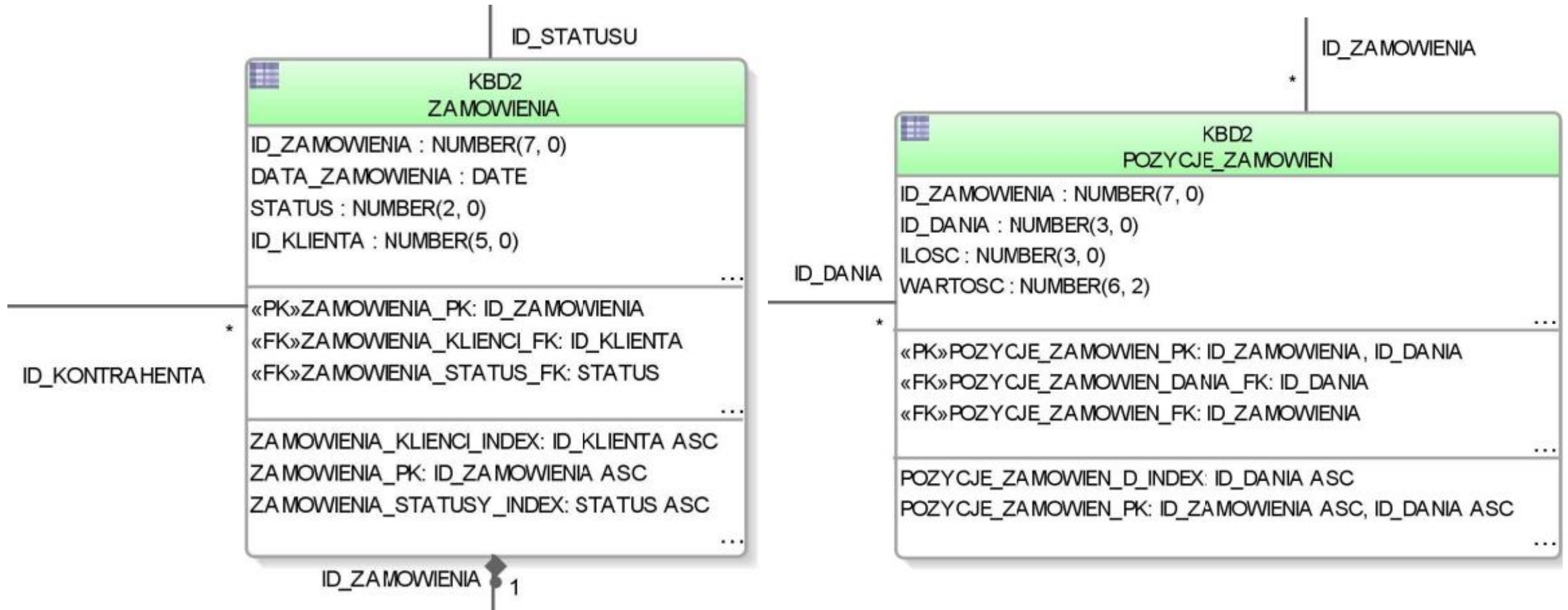
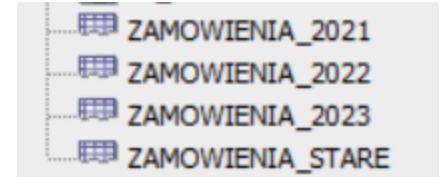
Jedzenie



Dostawy



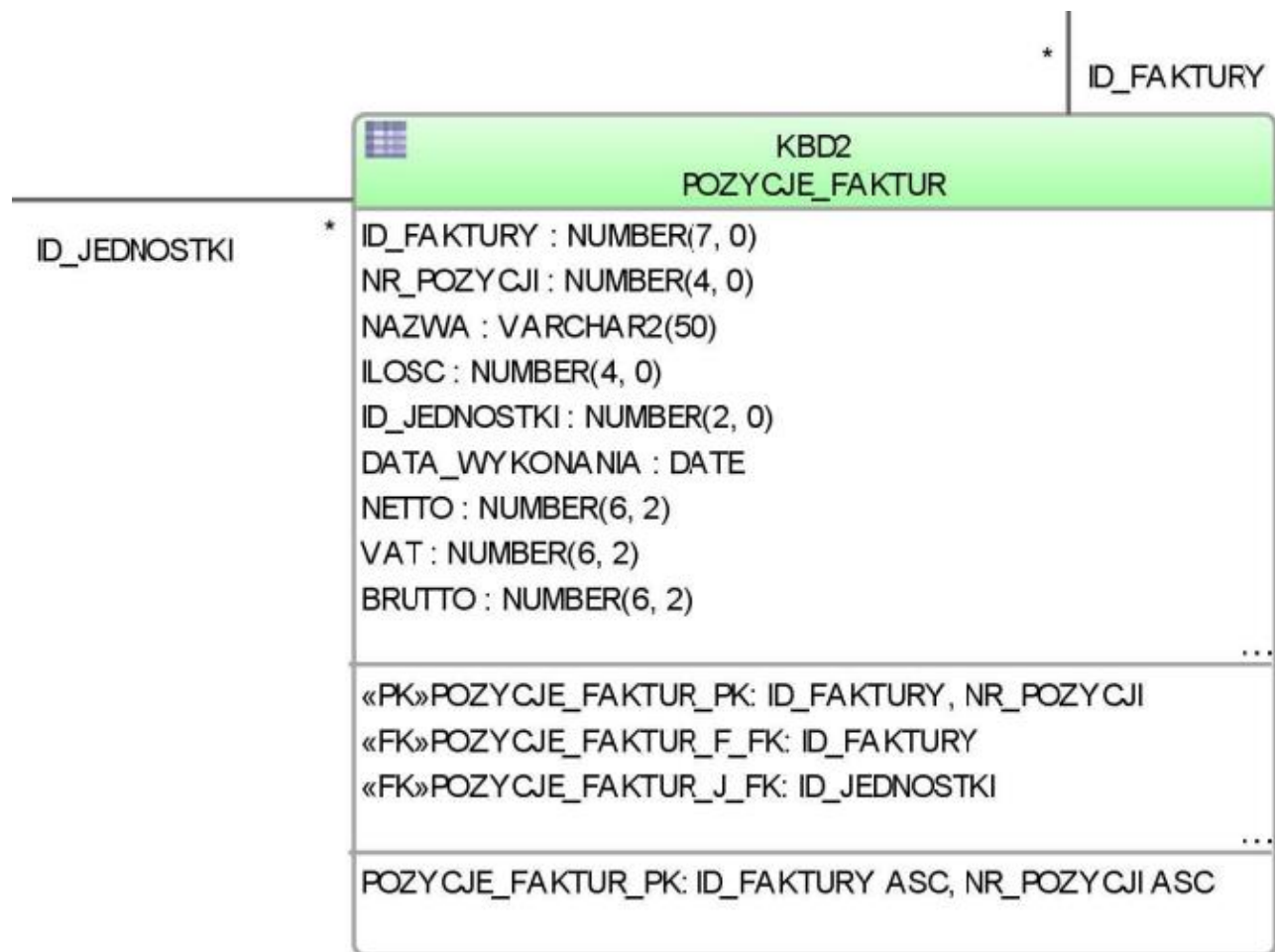
Zamówienia



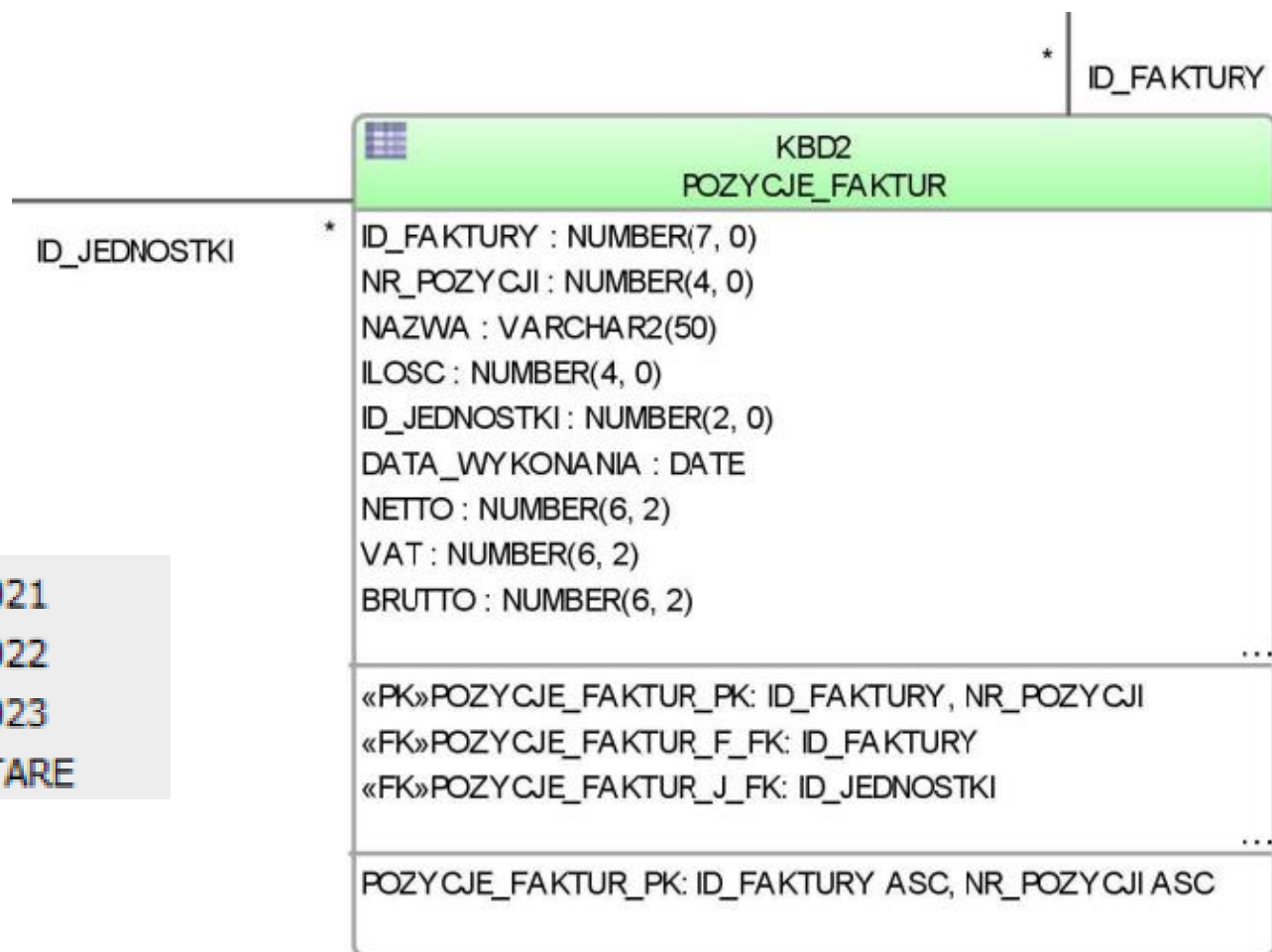
Faktury...

		* ID_KONTRAHENTA	* ID_KONTRAHENTA
		KBD2 FAKTURY	
		ID_FAKTURY : NUMBER(7, 0) NR_FAKTURY : VARCHAR2(20) ID_NABYWCY : NUMBER(5, 0) ID_SPRZEDAWCY : NUMBER(5, 0) DATA_WYSTAWIENIA : DATE DATA_ZAPLATY : DATE TERMIN_ZAPLATY : DATE STAWKA_VAT : NUMBER(3, 0) SPOSOB_ZAPLATY : CHAR(1) NETTO : NUMBER(7, 2) VAT : NUMBER(7, 2) BRUTTO : NUMBER(7, 2) WARTOSC_KWOTY_SLOWNIE : VARCHAR2(100) CZY_WYCHODZACA : CHAR(1) ...	
		«PK»FAKTURY_PK: ID_FAKTURY «FK»FAKTURY_NABYWCY_FK: ID_NABYWCY «FK»FAKTURY_SPRZEDAWCY_FK: ID_SPRZEDAWCY 1 «Check»FAKTURY_PARTIE_CHK: ID_NABYWCY = ID_SPRZEDAWCY ...	
		FAKTURY_NABYWCY_INDEX: ID_NABYWCY ASC FAKTURY_PK: ID_FAKTURY ASC FAKTURY_ROK_MIESIAC_INDEX: (EXTRACT(YEAR FROM "DATA_WYSTAWIENIA") DESC, EXTRACT(MONTH FROM "DATA_WYSTAWIENIA") DESC FAKTURY_SPRZEDAWCY_INDEX: ID_SPRZEDAWCY ASC ...	

...i ich pozycje



...i ich pozycje

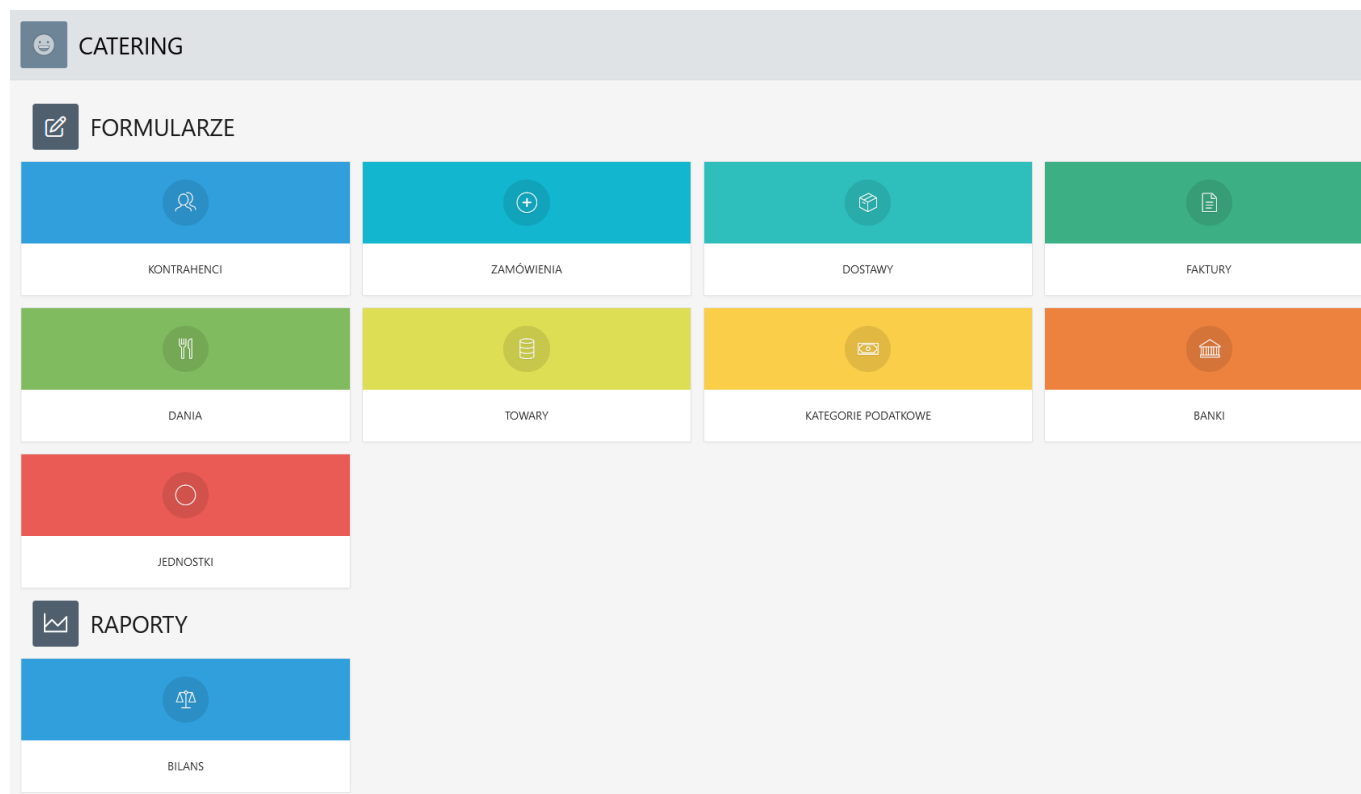


Aplikacja APEX

W skrócie:

- 9 formularzy
- 5 raportów

MENU GŁÓWNE





CATERING



FORMULARZE



KONTRAHENCI



ZAMÓWIENIA



DOSTAWY



FAKTURY



DANIA



TOWARY



KATEGORIE PODATKOWE



BANKI



JEDNOSTKI



RAPORTY



BILANS

KONTRAHENCI

Reset

Create



Search...

Adam Nowak
hubisoroka1@gmail.com

Jan Kowalski
JKOWALSKI@wp.pl

Sławomir Test
hubisoroka2@gmail.com

Stefan Dostawca
SDOSTAWCA@gmail.com

Dostawy Towarow 365
kdb3@pw.edu.pl

Transport Mariusz
Pudzianowski
tanio_skory-
nie_sprzedam@gmail.com

KBD2 Katering Bazodanowy
kdb2@pw.edu.pl

Edit

Kontrahenci



Status
Osoba Prywatna

Imię

Nazwisko

Mail

Telefon

Bank

Nr Konta

Nazwa Ulicy

Nr Budynku

Nr Lokalu

Kod Pocztowy

Miejscowosc

Cancel

Create

2023-05-14

2023-05-10

1 - 1

1 - 2

ZAMÓWIENIA

Reset

+ Create

Search...

Zamówienie

Edit

2023-05-15
Sławomir Test

2023-05-14
Adam Nowak

2023-04-27
Dostawy Towarow 365

2023-04-07
Jan Kowalski

2023-03-18
Stefan Dostawca

2023-02-26
Jan Kowalski

Pozycje Zamowien

Danie

Kruche ciastka

Ilosc

3

Cancel

Delete

Apply Changes

Cancel

Create



1 x Sernik z rodzynekami

20,99zł



1 x Kremówka

20zł

1 - 6

DOSTAWY

Search...

2023-05-14

Dostawy Towarow 365

2023-05-12

Adam Nowak

2023-05-11

KBD2 Katering Bazodanowy

2023-05-10

Dostawy Towarow 365

2023-04-17

Jan Kowalski

2023-03-08

Stefan Dostawca

Pozycje Dostaw

Towar
boczek

Ilość
15

JEDNOSTKA
kilogram

Cena Jednostkowa (zł)
37

Seria
1000

Data Waznosci
2023-07-14

Cancel

Delete

Apply Changes

Reset

Create

Edit

+

Waznosci

05-28

07-14

1 - 2

FAKTURY

Search...

07/2023
2023-05-12

06/2023
2023-02-10

05/2023
2023-05-05

04/2023
2023-04-01

03/2023
2023-05-11

02/2023
2023-03-08

01/2023
2023-02-26

Faktury

Nr Faktury

Reset

Create

Pozycje Faktur

Nr Pozycji
1

Nazwa
Dostawa boczku

Ilosc
20

Jednostka
sztuka

Data Wykonania
2023-03-08

Netto (zł)
60

Vat (zł)
20

Brutto (zł)
100

Cancel

Delete

Apply Changes

Cancel

Create

Edit

roszy

+			
Netto	Vat	Brutto	
60zł	20zł	100zł	
65zł	15zł	80zł	

1 - 2

DANIA

Reset

Create

Search...

Boczek w cieście
35,50zł

Kremówka
21,37zł

Kruche ciastka
15,99zł

Sernik "po Bożemu"
18,75zł

Sernik z rodzynkami
20,99zł

Szejk czekoladowy
15,99zł

Winny boczek
31,21zł

Dania

Składniki Dan



Towar
cukier



Ilość
1

Jednostka

Cancel

Delete

Apply Changes

Cancel

Create

Edit

+

ość

kg

kg

1 - 2

TOWARY

Q v	
	Nazwa ↑
✎	boczek
✎	cukier
✎	czekolada
✎	jajka
✎	jogurt
✎	maka
✎	mleko
✎	wino

FORMULARZ

✎ Nazwa

Opis

📦 Ilość

○ Jednostka

% Kategoria

Cancel

Create

↶ Reset

✎ te

Kategoria

Domyslny VAT

Slodycze

Slodycze

Domyslny VAT

Nabial

Domyslny VAT

Nabial

Domyslny VAT

1 - 8

KATEGORIE POD

FORMULARZ

KATEGORIE PODATKOWE \ FORMULARZ



Nazwa



% Opodatkowanie

Cancel

Create

Create

Podatek

8%

8%

23%

1 - 3

BANKI



Go

Actions ▾

Create



Nazwa banku

FORMULARZ



BANKI \ FORMULARZ



Nazwa banku

Cancel

Create



Skok Na Kasę



mBank

1 - 9

JEDNOSTKI

FORMULARZ



JEDNOSTKI \ FORMULARZ



Nazwa



Skrot

Cancel

Create

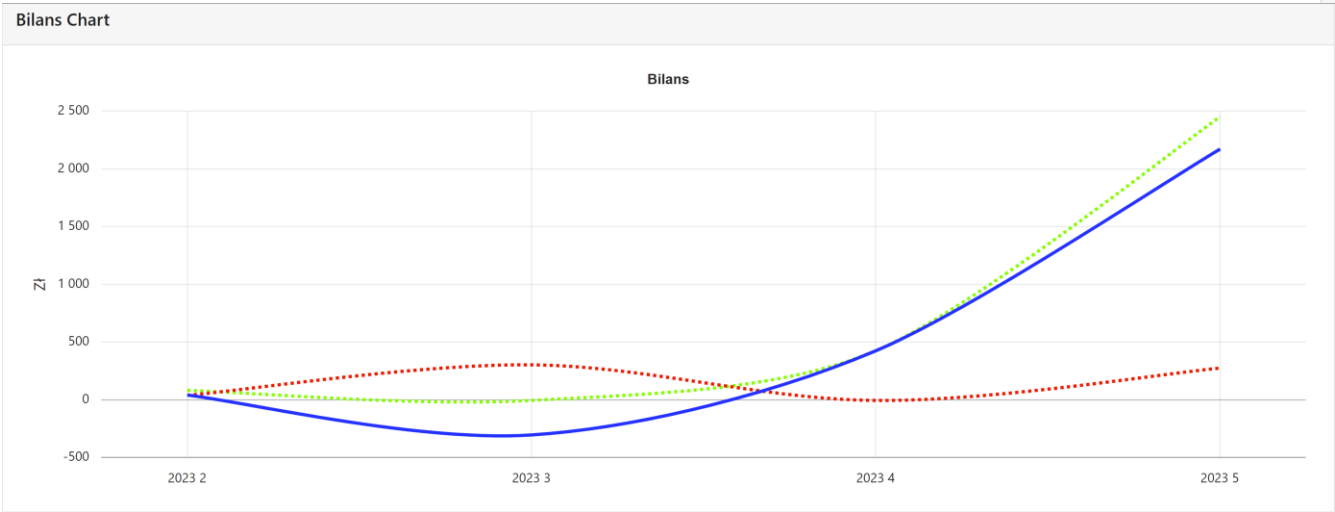
1 - 4

Bilans Zamówione Dania Zamówienia Kontrahentów Użyte Towary Faktury Kontrahentów

Bilans Interactive Bilans Chart

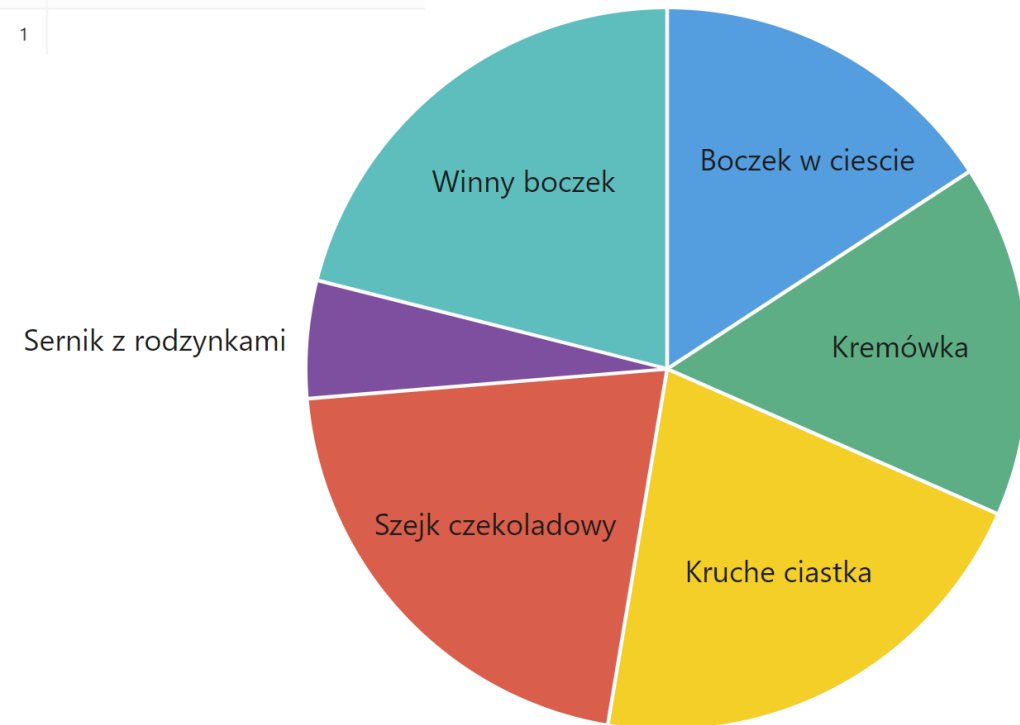
Miesiac	Suma Netto Przychodow	Suma Netto Rozchodow	Bilans
LUTY 2023	82,35	40,5	41,85
MARZEC 2023	0	305	-305
KWIECIEŃ 2023	423,35	0	423,35
MAJ 2023	2451,23	275	2176,23
			2336,43

Bilans



Zamówione Dania Interactive Zamówione Dania Chart				
<div> <div>Q</div> <div>Go</div> <div>2. Break Miesiące</div> <div> <div></div> <div></div> </div> <div>Actions</div> </div>				
<div> <div>Alternative Default: Break Miesiące</div> <div>Edit Pivot</div> </div>				
	KWIECIEŃ 2023	LUTY 2023	MAJ 2023	MARZEC 2023
Danie	Sumarycznych Zamówień	Sumarycznych Zamówień	Sumarycznych Zamówień	Sumarycznych Zamówień
Kruche ciastka	2		1	1
Szejk czekoladowy	1	1	1	1
Sernik z rodzynkami			1	
Boczek w cieście		1	2	
Kremówka	1		1	1
Winy boczek	2	1	1	

Zamawiane Dania



Kontrahenci

Miesiac : KWIECIEŃ 2023				
Id Kontrahenta	Cena Netto Po Daniach	Data	Status	Ranking Miesieczny ↑☰
2	243,84	2023-04-27	1	1
3	47,2	2023-04-07	2	2
Miesiac : LUTY 2023				
Id Kontrahenta	Cena Netto Po Daniach	Data	Status	Ranking Miesieczny
3	149,41	2023-02-26	2	1
Miesiac : MAJ 2023				
Id Kontrahenta	Cena Netto Po Daniach	Data	Status	Ranking Miesieczny
29	378,22	2023-05-14	1	1
31	355	2023-05-15	1	2

Id Kontrahenta : 2		
Cena Netto Po Daniach	Data	Status
243,84	2023-04-27	1
243,84		
Id Kontrahenta : 3		
Cena Netto Po Daniach	Data	Status
149,41	2023-02-26	2
47,2	2023-04-07	2
196,61		
Id Kontrahenta : 4		
Cena Netto Po Daniach	Data	Status
266,6	2023-03-18	2
266,6		

		KWIECIEŃ 2023	LUTY 2023	MAJ 2023	MARZEC 2023
Nazwa Towaru	Jednostka	Ilość	Ilość	Ilość	Ilość
czekolada	kilogram	0	0	0	0
jajka	sztuka	6	3	10	4
cukier	kilogram	2	1	2	1
wino	litr	2	1	1	
mleko	litr	0	0	0	0
maka	kilogram	2	1	2	1
jogurt	litr	0	0	0	0
boczek	kilogram	2	2	2	

Towary

	0		1		null	
Nazwa	Zakupił Brutto	Sprzedał Brutto	Zakupił Brutto	Sprzedał Brutto	Zakupił Brutto	Sprzedał Brutto
Jan Kowalski	3 623	0				
Dostawy Towarow 365						
Stefan Dostawca	0	391	0	401		
KBD2 Katering Bazodanowy	391	3 623	401	0		
Transport Mariusz Pudzianowski						
Adam Nowak						
Sławomir Test						

	Id Kontrahenta	Nazwa	Bank	Id Faktury	Sposob Zapłaty	Zakupił Netto	Zakupił Brutto	Sprzedał Netto	Sprzedał Brutto	Netto	Brutto
	31	Sławomir Test	PKO Bank Polski			0	0	0	0		
	1	KBD2 Katering Bazodanowy	Krzak Bank	1	1	305	401,4	0	0	305	401,4
	1	KBD2 Katering Bazodanowy	Krzak Bank	2	0	0	0	82,35	101	82,35	101
	1	KBD2 Katering Bazodanowy	Krzak Bank	31	0	0	0	1220	1390	1220	1390
	1	KBD2 Katering Bazodanowy	Krzak Bank	4	0	275	341	0	0	275	341
	1	KBD2 Katering Bazodanowy	Krzak Bank	5	0	0	0	423,35	533,2	423,35	533,2
	1	KBD2 Katering Bazodanowy	Krzak Bank	7	0	40,5	50	0	0	40,5	50
	1	KBD2 Katering Bazodanowy	Krzak Bank	6	0	0	0	1231,23	1599	1231,23	1599
	2	Dostawy Towarow 365	Bank Spółka Zło			0	0	0	0		
	3	Jan Kowalski	Skok Na Kasę	2	0	82,35	101	0	0	82,35	101

Faktury

Wymiana danych

- Formaty danych do eksportu/importu:
 - XML
 - CSV
- Bezpośrednie użycie danych: Arkusz Microsoft Excel + ODBC

Wymiana danych w CSV

- Eksport:
 - Plik .sql z dyrektywami, uruchamiany przez SQL*Plus
 - Najważniejsza dyrektywa: **SET MARK CSV ON QUOTE OFF**
- Import:
 - Przy użyciu SQL Loadera
 - Wymagane pliki:
 - Plik CSV z danymi
 - Plik .ctl określający strukturę pliku z danymi oraz ich przeznaczenie
 - Plik .par z parametrami takimi jak nazwy używanych plików czy sposób komunikacji

Wymiana danych w XML

- Eksport oraz import danych przez XML realizowany przy pomocy skryptów napisanych w PL/SQL, które należy uruchomić przy użyciu SQL*Plus.
- Każdy z używanych plików poddawany jest walidacji ze zdefiniowanym schematem XML.
- Dostępny jest eksport i import danych o fakturach.

Przykładowy plik wyjściowy eksportu

```
<FAKTURA>
...
<NR_FAKTURY>01/2023</NR_FAKTURY>
<DATA_WYSTAWIENIA>2023-02-26</DATA_WYSTAWIENIA>
<SPRZEDAWCA>
  <NAZWA>KBD2 Katering Bazodanowy</NAZWA>
  <NIP>3873379735</NIP>
  <ADRES>
    <MIEJSCOWOSC>Warszawa</MIEJSCOWOSC>
    <KOD_POCZTOWY>01-248</KOD_POCZTOWY>
    <NAZWA_ULICY>Plac Politechniki</NAZWA_ULICY>
    <NR_BUDYNKU>1</NR_BUDYNKU>
    <NR_LOKALU></NR_LOKALU>
  </ADRES>
</SPRZEDAWCA>
<NABYWCA>
  <IMIE>Jan</IMIE>
  <NAZWISKO>Kowalski</NAZWISKO>
  <ADRES>
    <MIEJSCOWOSC>Bytom</MIEJSCOWOSC>
    <KOD_POCZTOWY>02-252</KOD_POCZTOWY>
    <NAZWA_ULICY>Kowalskiego</NAZWA_ULICY>
    <NR_BUDYNKU>123</NR_BUDYNKU>
    <NR_LOKALU></NR_LOKALU>
  </ADRES>
</NABYWCA>
```

```
<POZYCJE_FAKTURY>
  <POZYCJA_FAKTURY>
    <NAZWA>Boczek w ciescie</NAZWA>
    <ILOSC>1</ILOSC>
    <NAZWA_JEDNOSTKI>sztuka</NAZWA_JEDNOSTKI>
    <DATA_WYKONANIA>2023-02-26</DATA_WYKONANIA>
    <NETTO>0</NETTO>
    <VAT>0</VAT>
    <BRUTTO>0</BRUTTO>
  </POZYCJA_FAKTURY>
  <POZYCJA_FAKTURY>
    <NAZWA>Dostarczenie katering</NAZWA>
    <ILOSC>1</ILOSC>
    <NAZWA_JEDNOSTKI>sztuka</NAZWA_JEDNOSTKI>
    <DATA_WYKONANIA>2023-02-26</DATA_WYKONANIA>
    <NETTO>10</NETTO>
    <VAT>2</VAT>
    <BRUTTO>12</BRUTTO>
  </POZYCJA_FAKTURY>
</POZYCJE_FAKTURY>
```

Przykładowy plik wyjściowy eksportu

```
<STAWKA_VAT>23</STAWKA_VAT>
<NETTO>82.35</NETTO>
<VAT>18.65</VAT>
<BRUTTO>101</BRUTTO>
<WARTOSC_KWOTY_SLOWNIE>osiemdziesiąt dziewięć złotych</WARTOSC_KWOTY_SLOWNIE>
<NR_KONTA>12345678901234567890234324</NR_KONTA>
<NAZWA_BANKU>Krzak Bank</NAZWA_BANKU>
<DATA_ZAPLATY>2023-03-08</DATA_ZAPLATY>
<TERMIN_ZAPLATY>2023-03-08</TERMIN_ZAPLATY>
<SPOSOB_ZAPLATY>0</SPOSOB_ZAPLATY>
</FAKTURA>
<!--Schemat
zweryfikowany pomysłnie-->
```

Plik schematu faktury

```
<xs:complexType name="POZYCJA_FAKTURY">
  <xs:sequence>
    <xs:element name="NAZWA" type="xs:string" />
    <xs:element name="ILOSC" type="xs:positiveInteger" />
    <xs:element name="NAZWA_JEDNOSTKI" type="xs:string" />
    <xs:element name="DATA_WYKONANIA" type="xs:date" />
    <xs:element name="NETTO" type="KWOTA" />
    <xs:element name="VAT" type="KWOTA" />
    <xs:element name="BRUTTO" type="KWOTA" />
  </xs:sequence>
</xs:complexType>
```

```
<xs:simpleType name="KWOTA">
  <xs:restriction base="xs:double">
    <xs:minInclusive value="0" />
  </xs:restriction>
</xs:simpleType>
```

Plik schematu faktury

```
<xs:complexType name="KONTRAHENT">
  <xs:sequence>
    <xs:choice>
      <xs:sequence>
        <xs:element name="IMIE" type="xs:string" />
        <xs:element name="NAZWISKO" type="xs:string" />
      </xs:sequence>
      <xs:sequence>
        <xs:element name="NAZWA" type="xs:string" />
        <xs:element name="NIP">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:length value="10" />
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:choice>
    <xs:element name="ADRES" type="ADRES" />
  </xs:sequence>
</xs:complexType>
```

Plik schematu faktury

```
<xs:complexType name="ADRES">
  <xs:sequence>
    <xs:element name="MIEJSCOWOSC" type="xs:string" />
    <xs:element name="KOD_POCZTOWY">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern value="[0-9]{2}-[0-9]{3}" />
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="NAZWA_ULICY" type="xs:string" />
    <xs:element name="NR_BUDYNKU" type="xs:string" />
    <xs:element name="NR_LOKALU" type="xs:string" minOccurs="0" />
  </xs:sequence>
</xs:complexType>
```

Plik schematu faktury

```
<xs:complexType name="ADRES">
  <xs:sequence>
    <xs:element name="MIEJSCOWOSC" type="xs:string" />
    <xs:element name="KOD_POCZTOWY">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern value="[0-9]{2}-[0-9]{3}" />
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="NAZWA_ULICY" type="xs:string" />
    <xs:element name="NR_BUDYNKU" type="xs:string" />
    <xs:element name="NR_LOKALU" type="xs:string" minOccurs="0" />
  </xs:sequence>
</xs:complexType>
```


Plik schematu faktury

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="FAKTURA">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="NR_FAKTURY" type="xs:string" />
        <xs:element name="DATA_WYSTAWIENIA" type="xs:date" />
        <xs:element name="SPRZEDAWCA" type="KONTRAHENT" />
        <xs:element name="NABYWCA" type="KONTRAHENT" />
        <xs:element name="POZYCJE_FAKTURY">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="POZYCJA_FAKTURY" type="POZYCJA_FAKTURY"
                maxOccurs="unbounded"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="STAWKA_VAT" type="xs:positiveInteger" />
        <xs:element name="NETTO" type="KWOTA" />
        <xs:element name="VAT" type="KWOTA" />
        <xs:element name="BRUTTO" type="KWOTA" />
        <xs:element name="WARTOSC_KWOTY_SLOWNIE" />
        <xs:element name="NR_KONTA">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:length value="26"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Plik schematu faktury

```
<xs:element name="STAWKA_VAT" type="xs:positiveInteger" />
<xs:element name="NETTO" type="KWOTA" />
<xs:element name="VAT" type="KWOTA" />
<xs:element name="BRUTTO" type="KWOTA" />
<xs:element name="WARTOSC_KWOTY_SLOWNIE" />
<xs:element name="NR_KONTA">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:length value="26"></xs:length>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="NAZWA_BANKU" type="xs:string" />
<xs:element name="DATA_ZAPLATY" type="xs:date" minOccurs="0" />
<xs:element name="TERMIN_ZAPLATY" type="xs:date" />
<xs:element name="SPOSOB_ZAPLATY" type="xs:string" />
</xs:sequence>
</xs:complexType>
</xs:element>
```

Użycie w Excelu

ROK	MIESIAC	ID	NAZWA	LICZBA_ZAMOWIEN	SUMA_NETTO
2023	5	3	Jan Kowalski	2	2462,46
2023	4	3	Jan Kowalski	2	846,7
2023	2	3	Jan Kowalski	2	164,7

ROK	MIESIAC	SUMA_NETTO_PRZYCHODOW	SUMA_NETTO_ROZCHODOW	BILANS
2023	5	1231,23	275	956,23
2023	4	423,35	0	423,35
2023	3	0	305	-305
2023	2	82,35	40,5	41,85

ROK	MIESIAC	ID	NAZWA	LICZBA_ZAMOWIEN	SUMA_NETTO
2023	5	3	Jan Kowalski	2	2462,46
2023	4	3	Jan Kowalski	2	846,7
2023	2	3	Jan Kowalski	2	164,7

Trzy arkusze zawierające dane
pobrane ze zdefiniowanych
widoków przez ODBC

PL/SQL

- 6 wyzwalaczy
- 3 procedury

TR_USUNIETE_DANIA

```
CREATE TRIGGER TR_USUNIETE_DANIA
  AFTER DELETE ON dania
  FOR EACH ROW
  -- wyzwalacz usuwajacy zamowienia usunietych dan
DECLARE
  PRAGMA AUTONOMOUS_TRANSACTION;
  -- kursor iterujacy po zamowieniach
  CURSOR c_zamowienia IS
  SELECT z.status
  FROM zamowienia z INNER JOIN POZYCJE_ZAMOWIEN zp
  ON z.id_zamowienia = zp.id_zamowienia
  WHERE zp.id_dania = :old.id_dania;

  row_zamowienia c_zamowienia%ROWTYPE;
BEGIN
  OPEN c_zamowienia;

  LOOP
    -- petla przechodzaca po kolejnych zamowieniach
    FETCH c_zamowienia INTO row_zamowienia;
    EXIT WHEN c_zamowienia%NOTFOUND;

    -- ustawianie statusu na 'odwoany automatycznie'
    row_zamowienia.status := 3;
  END LOOP;

  CLOSE c_zamowienia;
  COMMIT;
END;
```

TR_ZMIENIONE_TOWARY I TR_ZMIENIONY_SKLAD_DAN

```
CREATE TRIGGER TR_ZMIENIONE_TOWARY
  AFTER INSERT OR UPDATE OR DELETE ON towary
  -- wyzwalacz aktualizujący dostępność dan po zmianie ilości towarów
DECLARE
  -- kursor iterujący po daniach dotkniętych zmianą
  CURSOR c_dania IS
  SELECT d.id_dania, d.dostepnosc
  FROM dania d
  INNER JOIN skladniki_dan dk
  ON d.id_dania = dk.id_dania
  INNER JOIN towary t
  ON dk.id_towaru = t.id_towaru
  FOR UPDATE OF dostepnosc;

  row_dania c_dania%ROWTYPE;
BEGIN
  OPEN c_dania;
  LOOP
    FETCH c_dania INTO row_dania;
    EXIT WHEN c_dania%NOTFOUND;
    -- obliczamy maksymalną ilość dan jaką możemy wyprodukować oraz ustawiamy ją jako dostępność
    SELECT MIN(FLOOR(t.ilosc / dk.ilosc)) INTO row_dania.dostepnosc
    FROM towary t
    INNER JOIN SKLADNIKI_DAN dk
    ON dk.id_towaru = t.id_towaru
    WHERE dk.id_dania = row_dania.id_dania;
    UPDATE dania d SET d.dostepnosc = row_dania.dostepnosc WHERE d.id_dania = row_dania.id_dania;
  END LOOP;
  CLOSE c_dania;
END;
```

Procedura DODAJ_TOWARY(id_dostawy)

```
CREATE OR REPLACE PROCEDURE dodaj_towary(p_id_dostawy IN NUMBER) AS
    -- Procedura dodajaca wszystkie towary z danej dostawy do stanu magazynowego
    CURSOR c_pozycje_dostaw IS
    SELECT ilosc, id_towaru
    FROM pozycje_dostaw
    WHERE id_dostawy = p_id_dostawy;

    row_pozycje_dostaw c_pozycje_dostaw%ROWTYPE;
BEGIN
    OPEN c_pozycje_dostaw;
    LOOP
        FETCH c_pozycje_dostaw INTO row_pozycje_dostaw;
        EXIT WHEN c_pozycje_dostaw%NOTFOUND;
        -- zwiksz ilosc w magazynie o dostarczona ilosc
        UPDATE towary SET ilosc = ilosc + row_pozycje_dostaw.ilosc
        WHERE id_towaru = row_pozycje_dostaw.id_towaru;
    END LOOP;
    CLOSE c_pozycje_dostaw;
    COMMIT;
END;
```

Harmonogram REALIZACJA_ZADAN

```
BEGIN
  DBMS_SCHEDULER.CREATE_JOB (
    job_name => '"KBD2"."REALIZACJA_ZADAN"',
    job_type => 'PLSQL_BLOCK',
    job_action => 'EXEC niezrealizowane_zamowienia;',
    number_of_arguments => 0,
    start_date => NULL,
    repeat_interval => 'FREQ=DAILY;BYTIME=030000',
    end_date => NULL,
    enabled => FALSE,
    auto_drop => FALSE,
    comments => 'Harmonogram powiadamiajacy pracownikow o niezrealizowanych zamowieniach');

  DBMS_SCHEDULER.SET_ATTRIBUTE(
    name => '"KBD2"."REALIZACJA_ZADAN"',
    attribute => 'store_output', value => TRUE);

  DBMS_SCHEDULER.SET_ATTRIBUTE(
    name => '"KBD2"."REALIZACJA_ZADAN"',
    attribute => 'logging_level', value => DBMS_SCHEDULER.LOGGING_OFF);

  DBMS_SCHEDULER.enable(
    name => '"KBD2"."REALIZACJA_ZADAN"');
END;
```


Procedura NIEREALIZOWANE_ZAMOWIENIA

```
CREATE OR REPLACE PROCEDURE niezrealizowane_zamowienia AS
-- Procedura wysylajaca liste niezrealizowanych i przedawnionych zamowien do pracownika oraz
--zmieniajaca status zamowien ktorych status nie zmienil sie zbyt dlugo na przedawnione
-- kursor iterujacy po zamowieniach
CURSOR c_zamowienia IS
SELECT z.id_zamowienia, z.data_zamowienia, z.status, sz.nazwa, k.imie, k.nazwisko, k.nazwa knazwa
FROM zamowienia z
INNER JOIN statusy_zamowien sz
ON z.status = sz.id_statusu
INNER JOIN kontrahenci k
ON k.id_kontrahenta = z.id_klienta
FOR UPDATE OF z.status;

row_zamowienia c_zamowienia%ROWTYPE;
var_dzisiaj TIMESTAMP;
var_klient VARCHAR(61 BYTE);
var_wiadomosc VARCHAR2(10000 BYTE);
var_przedawnione VARCHAR2(10000 BYTE);
BEGIN
var_wiadomosc := 'Niezrealizowane zamowienia:' || chr(10);
var_wiadomosc := var_wiadomosc || 'ID | Data zamowienia | Klient';
var_przedawnione := chr(10) || 'Przedawnione zamowienia:' || chr(10);
var_przedawnione := var_przedawnione || 'ID | Data zamowienia | Klient';

-- pobieranie dzisiejszej daty
SELECT CURRENT_TIMESTAMP
INTO var_dzisiaj
FROM dual;

OPEN c_zamowienia;
LOOP
FETCH c_zamowienia INTO row_zamowienia;
EXIT WHEN c_zamowienia%NOTFOUND;
```

Procedura NIEREALIZOWANE_ZAMOWIENIA

```
OPEN c_zamowienia;
LOOP
FETCH c_zamowienia INTO row_zamowienia;
EXIT WHEN c_zamowienia%NOTFOUND;

IF row_zamowienia.nazwa = 'Przyjete' THEN
-- jeżeli zamówienie zostało przyjęte i nie jest zrealizowane pracownik zostanie powiadomiony
IF row_zamowienia.knazwa IS NOT NULL THEN
-- nazwa klientów firm
var_klient := row_zamowienia.knazwa;
ELSE
-- nazwa klientów osób prywatnych
var_klient := row_zamowienia.imie || ' ' || row_zamowienia.nazwisko;
END IF;

IF var_dzisiaj - row_zamowienia.data_zamowienia < 30 THEN
-- nieprzedawnione zamówienie
var_wiadomosc := var_wiadomosc || chr(10) || row_zamowienia.id_zamowienia || ' ' ||
row_zamowienia.data_zamowienia || ' ' || var_klient;
ELSE
-- zmiana statusu na 'auto odwołane'
var_przedawnione := var_przedawnione || chr(10) || row_zamowienia.id_zamowienia || ' ' ||
row_zamowienia.data_zamowienia || ' ' || var_klient;
UPDATE zamowienia z SET z.status = 3 WHERE z.id_zamowienia = row_zamowienia.id_zamowienia;
END IF;
END IF;
END LOOP;
-- wysłanie maila pracownikowi
var_wiadomosc := var_wiadomosc || var_przedawnione;
wyslij_maila(p_do => 'pracownik@catering.com',
p_od => 'admin@catering.com',
p_wiadomosc => var_wiadomosc,
p_smtp_host => 'smtp.catering.com');
-- dbms_output do testowanie
--dbms_output.put_line(var_wiadomosc);
CLOSE c_zamowienia;
COMMIT;
END;
```

Procedura NIEREAALIZOWANE_ZAMOWIENIA

Niezrealizowane zamówienia:

ID	Data zamówienia	Klient
----	-----------------	--------

6	15-MAY-23	Sławomir Test
---	-----------	---------------

4	27-APR-23	Dostawy Towarow 365
---	-----------	---------------------

5	14-MAY-23	Adam Nowak
---	-----------	------------

Przedawnione zamówienia:

ID	Data zamówienia	Klient
----	-----------------	--------

Procedura WYSLIJ_MAILA

[illegible]

Testowanie

- Testy Apex – komfort korzystania z systemu
- Testy raportów – jakość dostępnych analiz i podsumowań
- Testy bazy danych – integralność i spójność danych
- Testy eksportu danych – zachowanie kontekstu danych



Dziękujemy za uwagę!

