

# Politechnika Częstochowska

Katedra Informatyki Teoretycznej i Stosowanej

Bazy Danych

Opis pracy projektowej.

Temat projektu: przedszkole.

Pracę zrobił: Valerii Verevkin,

WIMil, Informatyka

2 rok, IV semestr, Gr. Lab. 9, Nr. Indeksu: 130919

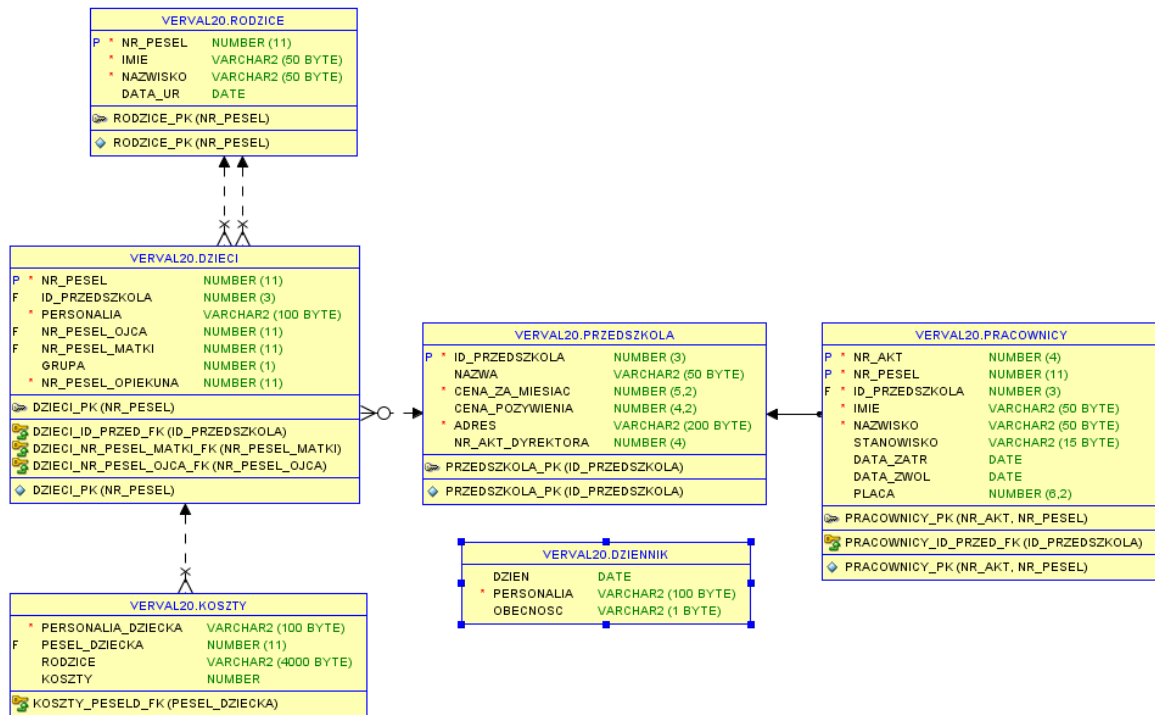
15.08.2020

## WSTĘP.

Dana praca projektowa jest przykładowym działaniem bazy danych, która została zrobiona dla wymiany informacją między przedszkolami. W tabelach zgromadzona jest informacja o ludziach i organizacjach w różnych miastach. **Wszystkie zbieżności (numery PESEL, personalia, nazwy przedszkoli i tp.) są przypadkowe i nie mają nic wspólnego z rzeczywistością.**

## TREŚĆ PROJEKTU.

Baza Przedszkole - dana baza składa się z 6 tabel (Przedszkola, Pracownicy, Rodzice, Dzieci, Dziennik i Koszty), 3 perspektyw (V\_dzieci\_pracownikow, V\_single\_fathers i V\_single\_mothers) i 2 sekwencji dla tworzenia wartości unikatowych (seq\_id\_przedszkola, seq\_nr\_akt).



(Rys. 1 – Model relacyjny w bazie **Przedszkole**)

## TABELE.

**Tabela Przedszkola** utworzona jest z następujących pól:

- 1) `id_przedszkola` - klucz główny tabeli, jest trzeczcyfrową liczbą całkowitą unikatową
- 2) `nazwa` - wartość typu `varchar2`, domyślnie przyjmuje znaczenie ''
- 3) `cena_za_miesiac` - liczba zmiennoprzecinkowa z dokładnością do .01, nie przyjmuje wartości pustych, oznacza cenę miesięczną miejsca w przedszkolu.
- 4) `cena_pozywienia` – liczba zmiennoprzecinkowa z dokładnością do .01, domyślnie przyjmuje wartość 0, oznacza dzienną cenę pożywienia.
- 5) `adres` – pole typu `varchar2`, zawiera adres przedszkola, nie przyjmuje wartości pustych.
- 6) `nr_akt_dyrektora` - wartość typu `number`, zawiera informację o dyrektorze.

```
create table Przedszkola
(
    id_przedszkola number(3) constraint Przedszkola_pk primary key,
    nazwa varchar2(50) default ' ',
    cena_za_miesiac number(5,2) constraint przedszkola_cena_nn not null,
    cena_pozywienia number(4,2) default 0,
    adres varchar2(200) constraint Przedszkola_adres_nn not null,
    nr_akt_dyrektora number(4)
);
```

(Rys. 2 – Inicjalizacja tabeli **Przedszkola**)

**Tabela Pracownicy** utworzona jest z następujących pól:

- 1) `nr_akt` - wartość typu `number`, klucz główny tabeli (wyjątek zostanie zdefiniowany podczas modyfikacji tabel)
- 2) `nr_pesel` - wartość typu `number`, drugi główny klucz tabeli (wyjątek zostanie zdefiniowany podczas modyfikacji tabel)
- 3) `id_przedszkola` - liczba całkowita, jest kluczem obcym (łączy z tabelą **Przedszkola**), zawiera numer przedszkola, przy usunięciu wpisu z tabeli nadrzędnej wartość związana z nim będzie zamieniona na **null**
- 4) `imie` – typ `varchar2`, nie przyjmuje wartości pustych
- 5) `nazwisko` – typ `varchar2`, nie przyjmuje wartości pustych.
- 6) `stanowisko` - wartość typu `varchar2`, przyjmuje tylko określone wartości
- 7) `data_zatr` - wartość typu `date`, wartość domyślna jest ustalona na 1 stycznia 2000r., zawiera datę zatrudnienia pracownika
- 8) `data_zwol` – typ `date`, zawiera datę zwolnienia

9) placa - wartość zmiennoprzecinkowa 4-cyfrowa z dokładnością do .01, nie przyjmuje wartości pustych

10) nip - wartość *varchar2*, zostanie usunięta podczas modyfikacji tabeli

```
create table pracownicy
(
    nr_akt number(4),
    nr_pesel number(11),
    id_przedszkola number(3) constraint Pracownicy_przedszkole_nn not null,
    imie varchar2(50) constraint Pracownicy_imie_nn not null,
    nazwisko varchar2(50) constraint Pracownicy_nazwisko_nn not null,
    stanowisko varchar2(15) constraint Pracownicy_stanowisko_ch check(stanowisko in
                                                                    ('dyrektor','sprzatacz','kucharz','opiekun')),
    data_zatr date default to_date('01.01.2000', 'dd.mm.yyyy'),
    data_zwol date,
    placa number(6,2) default null,
    nip varchar2(12),
    constraints Pracownicy_id_przed_fk foreign key (id_przedszkola) references przedszkola(id_przedszkola) on delete set null
);
```

(Rys. 3 – Inicjalizacja tabeli **Pracownicy**)

**Tabela Rodzice** utworzona jest z następujących pól:

- 1) nr\_pesel - wartość typu *number*, klucz główny tabeli
- 2) imie - typ *varchar2*, nie przyjmuje wartości pustych
- 3) nazwisko – typ *varchar2*, nie przyjmuje wartości pustych
- 4) data\_ur – typ *date*, nie przyjmuje wartości pustych, zawiera datę urodzenia rodzica

```
create table rodzice
(
    nr_pesel number(11) constraint Rodzice_pk primary key,
    imie varchar2(50) constraint Rodzice_imie_nn not null,
    nazwisko varchar2(50) constraint Rodzice_nazwisko_nn not null,
    data_ur date constraint Rodzice_urodzenie_nn not null
);
```

(Rys. 4 – Inicjalizacja tabeli **Rodzice**)

**Tabela Dzieci** utworzona jest z następujących pól:

- 1) nr\_pesel - wartość typu *number*, klucz główny tabeli
- 2) id\_przedszkola - wartość zawiera liczbę całkowitą, jest kluczem obcym (łączy z tabelą **Przedszkole**), usunięcie informacji z tabeli nadrzędnej powoduje ustawianie wpisów w tabeli **Dzieci** na NULL
- 3) personalia– typ *varchar2*, nie przyjmuje wartości pustych, zawiera imię i nazwisko dziecka
- 4) nr\_pesel\_ojca – 11-cyfrowa liczba całkowita, pusta wartość domyślna, zawiera numer pesel ojca, jest kluczem obcym tabeli (łączy z tabelą **Rodzice**), usunięcie wpisu z tabeli nadrzędnej powoduje usunięcie wpisu w tabeli **Dzieci**
- 5) nr\_pesel\_matki– 11-cyfrowa liczba całkowita, pusta wartość domyślna, zawiera numer pesel mamy, kluczem obcym tabeli (łączy z tabelą **Rodzice**)

6) grupa – *cyfra*, wartość musi być w przedziale od 1 do 4

7) nr\_pesel\_opiekuna - wartość 11-cyfrowa, zawiera numer pesel opiekuna dziecka, nie przyjmuje wartości pustych

```
create table dzieci
(
    nr_pesel number(11) constraint Dzieci_pk primary key,
    id_przedszkola number(3),
    personalia varchar2(100) constraint Dzieci_pers_nn not null,
    nr_pesel_ojca number(11) default null,
    nr_pesel_matki number(11) default null,
    grupa number(1) constraints dzieci_gr_ch check(grupa between 1 and 4),
    nr_pesel_opiekuna number(11) constraint dzieci_opiekun_nn not null,
    constraint Dzieci_id_przed_fk foreign key (id_przedszkola) references przedszkola(id_przedszkola) on delete set null,
    constraint dzieci_nr_pesel_ojca_fk foreign key(nr_pesel_ojca) references rodzice(nr_pesel) on delete cascade,
    constraint dzieci_nr_pesel_matki_fk foreign key(nr_pesel_matki) references rodzice(nr_pesel) on delete cascade
);
```

(Rys. 5 – Inicjalizacja tabeli **Dzieci**)

**Tabele Dziennik oraz Koszty są wirtualne.** Sposób inicjalizacji danych tabel oraz ich modyfikacja zostanie opisana później (po dodaniu wpisów do tabel).

## MODYFIKACJA TABEL.

Modyfikacji w danym kroku podległy tylko tabele **Rodzice** oraz **Pracownicy**.

```
--PRACOWNICY
alter table pracownicy add constraint Pracownicy_pk primary key (nr_akt, nr_pesel);

alter table pracownicy add constraint Pracownicy_data_zwol_ch check(data_zwol>data_zatr);

alter table pracownicy drop constraint Pracownicy_stanowisko_ch;

alter table pracownicy drop column nip;

alter table pracownicy modify(placa default 2000);

--RODZICE

alter table rodzice drop constraint Rodzice_urodzenie_nn;

alter table rodzice modify data_ur default to_date('01.01.1960','dd.mm.yyyy');
```

(Rys. 6 – Modyfikacja tabel **Pracownicy** oraz **Rodzice**)

**W tabeli Pracownicy:** klucz główny na kolumny nr\_akt oraz nr\_pesel; sprawdzanie daty zwolnienia pracownika (czy jest większa od daty zatrudnienia); usunęliśmy ograniczenie, nałożone na kolumnę stanowisko; została usunięta kolumna nip; dodana wartość domyślna dla kolumny placa.

**W tabeli Rodzice:** usunęliśmy ograniczenie z kolumny data\_ur oraz dodaliśmy wartość domyślną na tę kolumnę.

## TWORZENIE SEKWENCJI DLA WARTOŚCI UNIKATOWYCH.

Dla tworzenia wartości dla kolumn **nr\_akt** i **id\_przedszkola** zostaną wprowadzone sekwencje.

```
create sequence seq_id_przedszkola
start with 100
minvalue 100
maxvalue 999
increment by 1;

create sequence seq_nr_akt
start with 1000
minvalue 1000
maxvalue 9999
increment by 1
cycle;
```

(Rys. 7 – Sekwencje *seq\_id\_przedszkola* i *seq\_nr\_akt*)

Pierwsza sekwencja zaczyna się od 100 i z krokiem równym 1 zwiększa się do 999. Dana sekwencja jest skończona, czyli wartość maksymalna jest wartością ostatnią danej sekwencji.

Druga sekwencja startuje od liczby 1000 i z każdym krokiem zwiększa się o 1 aż do 9999. Ta sekwencja jest nieskończona (cycle), czyli po 9999 wartości zaczynają się powtarzać.

## WPROWADZENIE DANYCH.

Poleceniem insert dodajemy dane:

```
insert into przedszkola(id_przedszkola, nazwa, adres, cena_za_miesiac) values(seq_id_przedszkola.nextval, 'Przedszkole sw. Jana Pawła', 'Czestochowa 42-202 al. Jana Pawła II 26/30', 100.00);
```

```
insert into przedszkola values(seq_id_przedszkola.nextval, 'Teczowe', 125.50, 0, 'Czestochowa 42-218 ul. Dekabrystow 15', null);
```

```
insert into przedszkola values(seq_id_przedszkola.nextval, 'Przedszkole prywatne "Katarzyna Pogodna"', 95.20, 0, 'Radomsko 97-504 ul. Rolna 48', null);
```

```
insert into przedszkola values(seq_id_przedszkola.nextval, 'Przedszkole samorządowe nr.15', 0, 0, 'Kielce 25-018 ul. Leszczynska 40b', null);
```

```
insert into przedszkola values(seq_id_przedszkola.nextval, 'Przedszkole specjalne Jas i Mąlgosia', 155.80, 0, 'Wrocław 50-015 ul. Sudecka 2a', null);
```

```
insert into przedszkola values(seq_id_przedszkola.nextval, 'Przedszkole Puchatkowa Zaloga', 115.50, 0, 'Czestochowa 42-222 ul. Lwowska 1', null);
```

```
insert into przedszkola values(seq_id_przedszkola.nextval, 'Przedszkole publiczne nr.18', 0, 0, 'Czestochowa 42-205 ul. Wreczycka 68', null);
```

```
insert into przedszkola values(seq_id_przedszkola.nextval, 'Male skarby', 120.00, 0, 'Jelenia Gora 58-507 ul. Krasickiego 15/28', null);
```

```
insert into przedszkola values(seq_id_przedszkola.nextval, 'Przedszkole miejskie nr.1 im. Misia Uszatka', 0, 0, 'Czestochowa 42-211 ul. Szymanowskiego 9', null);
```

```
insert into przedszkola values(seq_id_przedszkola.nextval, 'Przedszkole nr.108', 0, 0, 'Wrocław 53-312 ul. Drukarska 8a', null);
```

```
insert into pracownicy values(seq_nr_akt.nextval, 68012412345, 100, 'Justyna', 'Bawor', 'Dyrektor', to_date('02.05.1988', 'dd.mm.yyyy'), null, 4500);
```

```
insert into pracownicy values(seq_nr_akt.nextval, 72120308054, 101, 'Kazimierz', 'Stala', 'Dyrektor', to_date('12.03.1995', 'dd.mm.yyyy'), null, 3800.50);
```

```
insert into pracownicy values(seq_nr_akt.nextval, 69091587654, 102, 'Mateusz', 'Praski', 'Dyrektor', to_date('08.03.1989', 'dd.mm.yyyy'), null, 4300);
```

```
insert into pracownicy values(seq_nr_akt.nextval, 65082365473, 103, 'Jozefa', 'Witkowska', 'Dyrektor', to_date('01.01.1987', 'dd.mm.yyyy'), null, 4700);
```

```

insert into pracownicy values(seq_nr_akt.nextval, 71121311325, 104, 'Otylia', 'Wojciechowska', 'Dyrektor', to_date('02.10.1992', 'dd.mm.yyyy'), null, 3750);

insert into pracownicy values(seq_nr_akt.nextval, 68122690876, 105, 'Stefania', 'Gajewska', 'Dyrektor', to_date('18.11.1990', 'dd.mm.yyyy'), null, 3980);

insert into pracownicy values(seq_nr_akt.nextval, 57091584857, 106, 'Daniel', 'Rajewski', 'Dyrektor', to_date('13.09.1978', 'dd.mm.yyyy'), null, 4500);

insert into pracownicy values(seq_nr_akt.nextval, 61121237645, 107, 'Amalia', 'Lis', 'Dyrektor', to_date('03.12.1991', 'dd.mm.yyyy'), null, 3900);

insert into pracownicy values(seq_nr_akt.nextval, 70011788223, 108, 'Mateusz', 'Wypych', 'Dyrektor', to_date('19.05.1995', 'dd.mm.yyyy'), null, 3600);

insert into pracownicy values(seq_nr_akt.nextval, 79031584756, 109, 'Krzysztof', 'Maj', 'Dyrektor', to_date('19.03.2002', 'dd.mm.yyyy'), null, 3400);


insert into pracownicy values(seq_nr_akt.nextval, 77021586961, 100, 'Anita', 'Szymanska', 'Opiekun', to_date('15.02.2003', 'dd.mm.yyyy'), null, 1800.2);

insert into pracownicy values(seq_nr_akt.nextval, 74061842341, 109, 'Aniela', 'Kowalska', 'Opiekun', sysdate-5894, null, 1900.3);

insert into pracownicy values(seq_nr_akt.nextval, 84080636623, 107, 'Jowita', 'Kucharska', 'Opiekun', sysdate-3801, null, 1750.35);

insert into pracownicy values(seq_nr_akt.nextval, 77010721146, 102, 'Adriana', 'Kamienska', 'Opiekun', sysdate-7826, null, 2100);

insert into pracownicy values(seq_nr_akt.nextval, 82010719534, 108, 'Urszula', 'Urbanska', 'Opiekun', sysdate-15, sysdate+124, 1600.5);

insert into pracownicy values(seq_nr_akt.nextval, 98042169294, 105, 'Alisa', 'Walczak', 'Opiekun', sysdate-1200, sysdate-290, 1500);

insert into pracownicy values(seq_nr_akt.nextval, 81042795767, 102, 'Natalia', 'Mroz', 'Opiekun', sysdate-985, null, 1900.4);

insert into pracownicy values(seq_nr_akt.nextval, 90062011979, 107, 'Aneta', 'Lewandowska', 'Opiekun', sysdate-2400, sysdate-1400, 1950.3);

insert into pracownicy values(seq_nr_akt.nextval, 74062531253, 101, 'Urszula', 'Rajewska', 'Opiekun', sysdate-5376, null, 2090);

insert into pracownicy values(seq_nr_akt.nextval, 96092862594, 105, 'Dominika', 'Dziwinska', 'Opiekun', sysdate-1234, null, 1750);

insert into pracownicy values(seq_nr_akt.nextval, 94021542474, 103, 'Karol', 'Sikora', 'Opiekun', sysdate-265, sysdate+74, 1800.55);

insert into pracownicy values(seq_nr_akt.nextval, 98122518518, 104, 'Justyna', 'Paweloszek', 'Opiekun', sysdate-365, sysdate+100, 1800);

insert into pracownicy values(seq_nr_akt.nextval, 79120822845, 100, 'Katarzyna', 'Kubiak', 'Opiekun', sysdate-6587, null, 2250);

insert into pracownicy values(seq_nr_akt.nextval, 86031721316, 106, 'Karolina', 'Sadowska', 'Opiekun', sysdate-4365, null, 2000.05);

insert into pracownicy values(seq_nr_akt.nextval, 89031741566, 101, 'Patrik', 'Samus', 'Opiekun', sysdate-875, null, 1780.9);

insert into pracownicy values(seq_nr_akt.nextval, 76042837757, 109, 'Aleksandra', 'Piekarska', 'Opiekun', sysdate-9999, null, 2400.4);

insert into pracownicy values(seq_nr_akt.nextval, 92061761824, 106, 'Ida', 'Zurek', 'Opiekun', sysdate-165, sysdate+100, 1800);

insert into pracownicy values(seq_nr_akt.nextval, 71111696954, 104, 'Natalia', 'Balcewicz', 'Opiekun', sysdate-7824, null, 2100);

insert into pracownicy values(seq_nr_akt.nextval, 98021979544, 108, 'Katarzyna', 'Chodkiewicz', 'Opiekun', sysdate-366, null, 1600);

insert into pracownicy values(seq_nr_akt.nextval, 89051562141, 103, 'Urszula', 'Kubiak', 'Opiekun', sysdate-5, null, 1750.2);


insert into pracownicy values(seq_nr_akt.nextval, 83110183568, 104, 'Natalia', 'Lis', 'Kucharz', sysdate-5486, null, 1700);

insert into pracownicy values(seq_nr_akt.nextval, 72052588793, 109, 'Katarzyna', 'Mroz', 'Kucharz', sysdate-7654, null, 1840);

insert into pracownicy values(seq_nr_akt.nextval, 64031597812, 101, 'Ida', 'Kubiak', 'Kucharz', sysdate-3547, sysdate-376, 1700);

insert into pracownicy values(seq_nr_akt.nextval, 83040252312, 107, 'Dominika', 'Pawlak', 'Kucharz', sysdate-300, null, 1680);

insert into pracownicy values(seq_nr_akt.nextval, 63030587929, 100, 'Marzanna', 'Czarnecka', 'Kucharz', sysdate-7387, sysdate-90, 1900);

insert into pracownicy values(seq_nr_akt.nextval, 69020441345, 102, 'Idalia', 'Kowalska', 'Kucharz', sysdate-5876, null, 2000.2);

insert into pracownicy values(seq_nr_akt.nextval, 60032563438, 103, 'Matylda', 'Wysocka', 'Kucharz', sysdate-6987, sysdate-1000, 1900);

insert into pracownicy values(seq_nr_akt.nextval, 84112943963, 106, 'Karolina', 'Kowalska', 'Kucharz', sysdate-3000, null, 1800.5);

insert into pracownicy values(seq_nr_akt.nextval, 96102837471, 108, 'Kamila', 'Krzyzanowska', 'Kucharz', sysdate-340, null, 1600);

insert into pracownicy values(seq_nr_akt.nextval, 90082072871, 105, 'Joanna', 'Laskowska', 'Kucharz', sysdate-1500, sysdate-34, 1790.6);

```

```
insert into rodzice values(96060937688, 'Angelika', 'Adamska', to_date('09.06.1996', 'dd.mm.yyyy'));

insert into rodzice values(80071698494, 'Jola', 'Rutkowska', to_date('16.07.1980', 'dd.mm.yyyy'));

insert into rodzice values(98021979544, 'Katarzyna', 'Chodkiewicz', to_date('19.02.1998', 'dd.mm.yyyy'));--BEZ PARY

insert into rodzice values(94070821216, 'Pawel', 'Adamski', to_date('08.07.1994', 'dd.mm.yyyy'));

insert into rodzice values(81021513693, 'Adam', 'Rutkowski', to_date('15.02.1981', 'dd.mm.yyyy'));

insert into rodzice values(94092718211, 'Aisha', 'Malinowska', to_date('27.09.1994', 'dd.mm.yyyy'));

insert into rodzice values(90052089196, 'Daniel', 'Malinowski', to_date('20.05.1990', 'dd.mm.yyyy'));

insert into rodzice values(93010267392, 'Amanda', 'Chmielewska', to_date('02.01.1993', 'dd.mm.yyyy'));

insert into rodzice values(93032844289, 'Mateusz', 'Chmielewski', to_date('28.03.1993', 'dd.mm.yyyy'));

insert into rodzice values(98042169294, 'Alisa', 'Walczak', to_date('21.04.1998', 'dd.mm.yyyy'));

insert into rodzice values(92113061579, 'Damian', 'Walczak', to_date('30.11.1992', 'dd.mm.yyyy'));

insert into rodzice values(89031741566, 'Patryk', 'Samus', to_date('17.03.1989', 'dd.mm.yyyy'));

insert into rodzice values(86032235232, 'Magda', 'Samus', to_date('22.03.1986', 'dd.mm.yyyy'));

insert into rodzice values(96062078969, 'Marysia', 'Wojcik', to_date('20.06.1996', 'dd.mm.yyyy'));

insert into rodzice values(94040948426, 'Kamil', 'Wojcik', to_date('09.04.1994', 'dd.mm.yyyy'));

insert into rodzice values(77042814397, 'Alicja', 'Rutkowska', to_date('28.04.1977', 'dd.mm.yyyy'));

insert into rodzice values(76081159645, 'Pawel', 'Rutkowski', to_date('11.08.1976', 'dd.mm.yyyy'));

insert into rodzice values(93032253197, 'Michal', 'Grabowski', to_date('22.03.1993', 'dd.mm.yyyy'));


insert into dzieci values(16232598567, 100, 'Marcin Grabowski', 93032253197, null, 1, 79120822845);

insert into dzieci values(16311576452, 108, 'Kamila Chodkiewicz', null, 98021979544, 1, 98021979544);

insert into dzieci values(14240641774, 101, 'Iga Adamska', 94070821216, 96060937688, 3, 74062531253);

insert into dzieci values(15231418551, 101, 'Amelia Adamska', 94070821216, 96060937688, 2, 89031741566);

insert into dzieci values(14231488661, 100, 'Mateusz Rutkowski', 81021513693, 80071698494, 3, 77021586961);

insert into dzieci values(16322689814, 100, 'Jolanta Rutkowska', 81021513693, 80071698494, 1, 79120822845);

insert into dzieci values(14231456286, 100, 'Pawel Rutkowski', 81021513693, 80071698494, 3, 77021586961);

insert into dzieci values(14310138162, 102, 'Bartosz Malinowski', 90052089196, 94092718211, 3, 81042795767);

insert into dzieci values(16220848319, 103, 'Klaudia Malinowska', 90052089196, 94092718211, 1, 89051562141);

insert into dzieci values(15310949862, 104, 'Martyna Chmielewska', 93032844289, 93010267392, 2, 98122518518);

insert into dzieci values(13312764795, 106, 'Krzysztof Chmielewski', 93032844289, 93010267392, 4, 86031721316);

insert into dzieci values(16291398339, 105, 'Pawel Walczak', 92113061579, 98042169294, 1, 96092862594);

insert into dzieci values(14242817278, 101, 'Dawid Samus', 89031741566, 86032235232, 3, 74062531253);

insert into dzieci values(15310592374, 101, 'Gabriel Samus', 89031741566, 86032235232, 2, 89031741566);

insert into dzieci values(16302967624, 101, 'Justyna Samus', 89031741566, 86032235232, 1, 89031741566);

insert into dzieci values(14221699965, 107, 'Mariusz Wojcik', 94040948426, 96062078969, 3, 84080636623);

insert into dzieci values(16251076947, 108, 'Katarzyna Wojcik', 94040948426, 96062078969, 1, 98021979544);

insert into dzieci values(13282644888, 109, 'Damian Rutkowski', 76081159645, 77042814397, 4, 76042837757);
```



```
insert into dzieci values(15322749614, 100, 'Matylda Rutkowska', 76081159645, 77042814397, 2, 79120822845);
```

## MODYFIKACJA DANYCH W TABELACH.

W tabeli **Przedszkola** musimy ustawić dyrektorów na podstawie wprowadzonych danych. Robimy to za pomocą następnego polecenia:

```
update przedszkola pr1 set nr_akt_dyrektora=
(
    select nr_akt from pracownicy where lower(stanowisko) like 'dyrektor' and id_przedszkola=pr1.id_przedszkola
);
```

(Rys. 8 – Aktualizacja dyrektorów w tabeli Przedszkola)

W tabeli **Pracownicy**, dla aktualnych pracowników, którzy dostają wynagrodzenie mniejsze od minimalnego, musimy go podnieść:

```
update pracownicy set placa=2250
where (data_zwol is null or data_zwol>sysdate) and placa<2250;
```

(Rys. 9 – Aktualizacja płacy w tabeli Pracownicy)

Dla utworzenia tabeli **Koszty**, musimy znać cenę pożywienia w każdym przedszkolu:

```
update przedszkola set cena_pozywienia=10 where id_przedszkola=100;
update przedszkola set cena_pozywienia=10.5 where id_przedszkola=101;
update przedszkola set cena_pozywienia=11.2 where id_przedszkola=102;
update przedszkola set cena_pozywienia=9.5 where id_przedszkola=103;
update przedszkola set cena_pozywienia=12 where id_przedszkola=104;
update przedszkola set cena_pozywienia=11.5 where id_przedszkola=105;
update przedszkola set cena_pozywienia=9 where id_przedszkola=106;
update przedszkola set cena_pozywienia=10.2 where id_przedszkola=107;
update przedszkola set cena_pozywienia=10 where id_przedszkola=108;
update przedszkola set cena_pozywienia=11 where id_przedszkola=109;
```

(Rys. 10 – Ustalenie ceny pożywienia w każdym przedszkolu)

## TABELE DZIENNIK I KOSZTY.

(UWAGA: Dla utworzenia tabeli **Dziennik** skorzystałem z tabeli **Rejestry**, ponieważ dana tabela zawiera bardzo dużo wpisów.

**Autor tabeli:** dr. Inż. Krzysztof Kaczmarek, Katedra Informatyki Teoretycznej I Stosowanej, Wydział Informatyki I Inżynierii Mechanicznej Politechniki Częstochowskiej)

Tabela **Dziennik** została zdefiniowana na podstawie tabel **Rejestry** oraz **Dzieci**. Tabela **Koszty** – na podstawie tabel **Rodzice**, **Dzieci**, **Dziennik I Przedszkola**.

```

create table dziennik as
(
    select cast(czas as date) dzien, personalia
    from ds.rejestry cross join
    (
        select personalia from dzieci
    )
);

alter table dziennik add obecosc varchar2(1) constraint Dziennik_obecnosc_ch check(obecnosc in ('X', '-'));
delete from dziennik where extract(year from dzien)<2018;

```

(Rys. 11 – Inicjalizacja i modyfikacja tabeli Dziennik)

Tabela została zdefiniowana poprzez złączenie typu “każdy z każdym” kolumny czas z tabeli **ds.rejestry** (użyte jawne rzutowanie z typu timestamp na typ date) z kolumną personalia z tabeli **Dzieci**.

Modyfikacja tabeli polegała na dodaniu kolumny “**Obecność**”, dodaniu dla niej ograniczenia oraz usuwanie wpisów, starszych od 2018 roku.

Dla większego prawdopodobieństwa musimy zmienić kolumnę “**Obecność**” dla niektórych dzieci. Aby to zrobić, wpisujemy losowe parametry dla operatora update:

```

update dziennik set obecnosc='X';

--DLA ZMIANY OBECNOSCI WPISUJEMY LOSOWE PARAMETRY

update dziennik set obecnosc='-' where (extract(day from dzien) between 16 and 17 or extract(day from dzien) between 1 and 2)
and personalia like '%i_';
update dziennik set obecnosc='-' where (extract(month from dzien)=12 or extract(month from dzien)=5)
and personalia like '%a_';
update dziennik set obecnosc='-' where (extract(month from dzien)=1 and extract(month from dzien)=1)
and personalia like '%ows%' and personalia not like '____ %';

```

(Rys. 12 – Wystawienie “nieobecności” w dzienniku)

```

create table koszty as
(
    select dz.personalia, dz.nr_pesel,
    listagg(ro.imie||' '||ro.nazwisko||' Nr. PESEL: '||ro.nr_pesel, ' | ')
    within group (order by ro.imie, ro.nazwisko, ro.nr_pesel) RODZICE,
    ((liczba*prz.cena_pozywienia)+(grupa*12*prz.cena_za_miesiac)) koszty
    from rodzice ro join dzieci dz on(ro.nr_pesel=nr_pesel_ojca or ro.nr_pesel=nr_pesel_matki)
    join
    (
        select count(*) liczba, personalia from dziennik
        where obecnosc='X'
        group by personalia
    ) tl on(dz.personalia=tl.personalia)
    join przedszkola prz on(dz.id_przedszkola=prz.id_przedszkola)
    group by dz.personalia, dz.nr_pesel, ((liczba*prz.cena_pozywienia)+(grupa*12*prz.cena_za_miesiac))
);

alter table koszty add constraint koszty_peselR_fk foreign key (pesel_rodzica) references rodzice(nr_pesel) on delete cascade;
alter table koszty add constraint koszty_peselD_fk foreign key (pesel_dziecka) references dzieci(nr_pesel) on delete cascade;

```

(Rys. 13 – Inicjalizacja i modyfikacja tabeli Koszty)

Tabela **Koszty** jest utworzona z pól tabel **Rodzice**, **Dzieci**, **Dziennik** oraz **Przedszkola**. Dana tabela zawiera informację o kosztach dla rodziców na każde dziecko. Koszty są liczone od pierwszego roku bycia dziecka

w przedszkolu. Koszty żywienia są uwzględniane (w przypadku, jeśli dziecko było obecne w dany dzień, wtedy jest doliczana cena jedzenia w zależności od przedszkola)

Tabela zawiera kolumnę, która jest listą argumentów. Dana kolumna wypisuje informację o rodzicach dziecka (imię, nazwisko i numer PESEL).

Modyfikacja tabeli polega na dodaniu kluczy obcych.

Po wprowadzeniu i modyfikacji danych nasze tabele wyglądają następująco:

ID_PRZEDSZKOLA	NAZWA	CENA_ZA_MIESIAC	CENA_POZYWIENIA	ADRES	NR_AKT_DYREKTORA
1	100 Przedszkole sw. Jana Pawła	100	10	Częstochowa 42-202 al. Jana Pawła II 26/30	1000
2	101 Teczowe	125,5	10,5	Częstochowa 42-218 ul. Dekabrystów 15	1001
3	102 Przedszkole prywatne "Katarzyna Pogodna"	95,2	11,2	Radomsko 97-504 ul. Rolna 48	1002
4	103 Przedszkole samorządowe nr.15	0	9,5	Kielce 25-018 ul. Leszczyńska 40b	1003
5	104 Przedszkole specjalne Jas i Malgosia	155,8	12	Wrocław 50-015 ul. Sudecka 2a	1004
6	105 Przedszkole Puchatkowa Załoga	115,5	11,5	Częstochowa 42-222 ul. Lwowska 1	1005
7	106 Przedszkole publiczne nr.18	0	9	Częstochowa 42-205 ul. Wroczycka 68	1006
8	107 Małe skarby	120	10,2	Jelenia Góra 58-507 ul. Krasickiego 15/28	1007
9	108 Przedszkole miejskie nr.1 im. Misia Uszatka	0	10	Częstochowa 42-211 ul. Szymanowskiego 9	1008
10	109 Przedszkole nr.108	0	11	Wrocław 53-312 ul. Drukarska 8a	1009

(Rys. 14 – Tabela Przedszkola)

NR_AKT	NR_PESEL	ID_PRZEDSZKOLA	IMIE	NAZWISKO	STANOWISKO	DATA_ZATR	DATA_ZW...	PLACA	
1	1000	68012412345	100	Justyna	Bawor	Dyrektor	02.05.88	(null)	4500
2	1001	72120308054	101	Kazimierz	Stala	Dyrektor	12.03.95	(null)	3800,5
3	1002	69091587654	102	Mateusz	Praski	Dyrektor	08.03.89	(null)	4300
4	1003	65082365473	103	Jozefa	Witkowska	Dyrektor	01.01.87	(null)	4700
5	1004	71121311325	104	Otylia	Wojciechowska	Dyrektor	02.10.92	(null)	3750
6	1005	68122690876	105	Stefania	Gajewska	Dyrektor	18.11.90	(null)	3980
7	1006	57091584857	106	Daniel	Rajewski	Dyrektor	13.09.78	(null)	4500
8	1007	61121237645	107	Amalia	Lis	Dyrektor	03.12.91	(null)	3900
9	1008	70011788223	108	Mateusz	Wypych	Dyrektor	19.05.95	(null)	3600
10	1009	79031584756	109	Krzysztof	Maj	Dyrektor	19.03.02	(null)	3400
11	1010	77021586961	100	Anita	Szymanska	Opiekun	15.02.03	(null)	2250
12	1011	74061842341	109	Aniela	Kowalska	Opiekun	09.07.04	(null)	2250
13	1012	84080636623	107	Jowita	Kucharska	Opiekun	02.04.10	(null)	2250
14	1013	77010721146	102	Adriana	Kamienska	Opiekun	26.03.99	(null)	2250
15	1014	82010719534	108	Urszula	Urbanska	Opiekun	13.08.20	30.12.20	2250
16	1015	98042169294	105	Alisa	Walczak	Opiekun	16.05.17	12.11.19	1500
17	1016	81042795767	102	Natalia	Mroz	Opiekun	17.12.17	(null)	2250
18	1017	90062011979	107	Aneta	Lewandowska	Opiekun	01.02.14	28.10.16	1950,3

(Rys. 15 – Tabela Pracownicy)

NR_PESEL	IMIE	NAZWISKO	DATA_UR	
1	96060937688	Angelika	Adamska	09.06.96
2	80071698494	Jola	Rutkowska	16.07.80
3	98021979544	Katarzyna	Chodkiewicz	19.02.98
4	94070821216	Pawel	Adamski	08.07.94
5	81021513693	Adam	Rutkowski	15.02.81
6	94092718211	Aisha	Malinowska	27.09.94
7	90052089196	Daniel	Malinowski	20.05.90
8	93010267392	Amanda	Chmielewska	02.01.93
9	93032844289	Mateusz	Chmielewski	28.03.93
10	98042169294	Alisa	Walczak	21.04.98
11	92113061579	Damian	Walczak	30.11.92
12	89031741566	Patryk	Samus	17.03.89
13	86032235232	Magda	Samus	22.03.86
14	96062078969	Marysia	Wojcik	20.06.96
15	94040948426	Kamil	Wojcik	09.04.94
16	77042814397	Alicja	Rutkowska	28.04.77
17	76081159645	Pawel	Rutkowski	11.08.76
18	93032253197	Michal	Grabowski	22.03.93

(Rys.16 - Tabela Rodzice)

NR_PESEL	ID_PRZEDSZKOLA	PERSONALIA	NR_PESEL_OJCA	NR_PESEL_MATKI	GRUPA	NR_PESEL_OPIEKUNA
1	16232598567	100 Marcin Grabowski	93032253197	(null)	1	79120822845
2	16311576452	108 Kamila Chodkiewicz	(null)	98021979544	1	98021979544
3	14240641774	101 Iga Adamska	94070821216	96060937688	3	74062531253
4	15231418551	101 Amelia Adamska	94070821216	96060937688	2	89031741566
5	14231488661	100 Mateusz Rutkowski	81021513693	80071698494	3	77021586961
6	16322689814	100 Jolanta Rutkowska	81021513693	80071698494	1	79120822845
7	14231456286	100 Pawel Rutkowski	81021513693	80071698494	3	77021586961
8	14310138162	102 Bartosz Malinowski	90052089196	94092718211	3	81042795767
9	16220848319	103 Klaudia Malinowska	90052089196	94092718211	1	89051562141
10	15310949862	104 Martyna Chmielewska	93032844289	93010267392	2	98122518518
11	13312764795	106 Krzysztof Chmielewski	93032844289	93010267392	4	86031721316
12	16291398339	105 Pawel Walczak	92113061579	98042169294	1	96092862594
13	14242817278	101 Dawid Samus	89031741566	86032235232	3	74062531253
14	15310592374	101 Gabriel Samus	89031741566	86032235232	2	89031741566
15	16302967624	101 Justyna Samus	89031741566	86032235232	1	89031741566
16	14221699965	107 Mariusz Wojcik	94040948426	96062078969	3	84080636623
17	16251076947	108 Katarzyna Wojcik	94040948426	96062078969	1	98021979544
18	13282644888	109 Damian Rutkowski	76081159645	77042814397	4	76042837757

(Rys. 17 – Tabela Dzieci)

	DZIEŃ	PERSONALIA	OBECNOŚĆ
1	24.01.18	Mateusz Rutkowski	-
2	24.01.18	Marcin Grabowski	-
3	24.01.18	Paweł Walczak	X
4	24.01.18	Iga Adamska	X
5	24.01.18	Martyna Chmielewska	X
6	24.01.18	Bartosz Malinowski	-
7	24.01.18	Damian Rutkowski	-
8	24.01.18	Gabriel Samus	X
9	24.01.18	Klaudia Malinowska	-
10	24.01.18	Justyna Samus	X
11	24.01.18	Kamila Chodkiewicz	X
12	24.01.18	Amelia Adamska	X
13	24.01.18	Mariusz Wojcik	X
14	24.01.18	Katarzyna Wojcik	X
15	24.01.18	Paweł Rutkowski	X
16	24.01.18	Matylda Rutkowska	-
17	24.01.18	Jolanta Rutkowska	-
18	24.01.18	Dawid Samus	-

(Rys. 18 – Tabela Dziennik)

PERSONALIA_DZIECKA	PESEL_DZIECKA	RODZICE	KOSZTY
1 Dawid Samus	14242817278	Magda Samus Nr. PESEL: 86032235232   Patryk Samus Nr. PESEL: 89031741566	7174,5
2 Iga Adamska	14240641774	Angelika Adamska Nr. PESEL: 96060937688   Paweł Adamski Nr. PESEL: 94070821216	7174,5
3 Gabriel Samus	15310592374	Magda Samus Nr. PESEL: 86032235232   Patryk Samus Nr. PESEL: 89031741566	5668,5
4 Justyna Samus	16302967624	Magda Samus Nr. PESEL: 86032235232   Patryk Samus Nr. PESEL: 89031741566	4162,5
5 Paweł Walczak	16291398339	Alisa Walczak Nr. PESEL: 98042169294   Damian Walczak Nr. PESEL: 92113061579	3950,5
6 Amelia Adamska	15231418551	Angelika Adamska Nr. PESEL: 96060937688   Paweł Adamski Nr. PESEL: 94070821216	5668,5
7 Mariusz Wojcik	14221699965	Kamil Wojcik Nr. PESEL: 94040948426   Marysia Wojcik Nr. PESEL: 96062078969	6400,8
8 Paweł Rutkowski	14231456286	Adam Rutkowski Nr. PESEL: 81021513693   Jola Rutkowska Nr. PESEL: 80071698494	6130
9 Damian Rutkowski	13282644888	Alicja Rutkowska Nr. PESEL: 77042814397   Paweł Rutkowski Nr. PESEL: 76081159645	2563
10 Katarzyna Wojcik	16251076947	Kamil Wojcik Nr. PESEL: 94040948426   Marysia Wojcik Nr. PESEL: 96062078969	2040
11 Marcin Grabowski	16232598567	Michał Grabowski Nr. PESEL: 93032253197	3530
12 Jolanta Rutkowska	16322689814	Adam Rutkowski Nr. PESEL: 81021513693   Jola Rutkowska Nr. PESEL: 80071698494	3530
13 Mateusz Rutkowski	14231488661	Adam Rutkowski Nr. PESEL: 81021513693   Jola Rutkowska Nr. PESEL: 80071698494	5930
14 Matylda Rutkowska	15322749614	Alicja Rutkowska Nr. PESEL: 77042814397   Paweł Rutkowski Nr. PESEL: 76081159645	4730
15 Bartosz Malinowski	14310138162	Aisha Malinowska Nr. PESEL: 94092718211   Daniel Malinowski Nr. PESEL: 90052089196	6036,8
16 Kamila Chodkiewicz	16311576452	Katarzyna Chodkiewicz Nr. PESEL: 98021979544	2530
17 Klaudia Malinowska	16220848319	Aisha Malinowska Nr. PESEL: 94092718211   Daniel Malinowski Nr. PESEL: 90052089196	2213,5
18 Martyna Chmielewska	15310949862	Amanda Chmielewska Nr. PESEL: 93010267392   Mateusz Chmielewski Nr. PESEL: 93032844289	6775,2

(Rys. 19 – Tabela Koszty)

### DODANIE PERSPEKTYW.

Następnym krokiem będzie dodanie perspektyw do naszego projektu. Na podstawie perspektyw w przyszłości będziemy mieli możliwość modyfikowania danych w tabeli.

Dla utworzenia pierwszej perspektywy weźmiemy informację o dzieciach, rodzice których pracują w przedszkolach (osoby, wpisy o których znajdują się w tabelach **Rodzice** i **Pracownicy**)

```
--DZIECI, RODZICE KTORYCH PRACUJA W PRZEDSZKOLACH
create or replace view V_dzieci_pracownikow as
select personalia personalia_dziecka, dz.nr_pesel pesel_dziecka, grupa, dz.id_przedszkola
from pracownicy pr join rodzice ro on(pr.nr_pesel=ro.nr_pesel)
join dzieci dz on(dz.nr_pesel_ojca=ro.nr_pesel or dz.nr_pesel_matki=ro.nr_pesel)
where pr.data_zwol is null or pr.data_zwol>sysdate;
```

(Rys. 20 – Tworzenie perspektywy V\_rodzice\_pracownicy)

	PERSONALIA_DZIECKA	PESEL_DZIECKA	GRUPA	ID_PRZEDSZKOLA
1	Kamila Chodkiewicz	16311576452	1	108
2	Justyna Samus	16302967624	1	101
3	Gabriel Samus	15310592374	2	101
4	Dawid Samus	14242817278	3	101

(Rys. 21 - Zawartość perspektywy V\_rodzice\_pracownicy)

Druga perspektywa będzie zawierała informację o mamach bez pary i je dzieciach.

```
--MAMY BEZ MEZA
create or replace view V_single_mothers as
select ro.nr_pesel pesel, ro.imie||' '||ro.nazwisko personalia_rodzica,
dz.nr_pesel pesel_dziecka, dz.personalia personalia_dziecka
from rodzice ro join dzieci dz on(ro.nr_pesel=dz.nr_pesel_matki)
where dz.nr_pesel_ojca is null
order by 2;
```

(Rys. 22 – Tworzenie perspektywy V\_single\_mothers)

	PESEL	PERSONALIA_RODZICA	PESEL_DZIECKA	PERSONALIA_DZIECKA
1	98021979544	Katarzyna Chodkiewicz	16311576452	Kamila Chodkiewicz

(Rys. 23 - Zawartość perspektywy V\_single\_mothers)

Trzecia perspektywa będzie podobna do drugiej, ale będzie zawierała ojców bez pary.

```
--TATY BEZ ZONY
create or replace view V_single_fathers as
select ro.nr_pesel pesel, ro.imie||' '||ro.nazwisko personalia_rodzica,
dz.nr_pesel pesel_dziecka, dz.personalia personalia_dziecka
from rodzice ro join dzieci dz on(ro.nr_pesel=dz.nr_pesel_ojca)
where dz.nr_pesel_matki is null
order by 2;
```

(Rys. 23 – Tworzenie perspektywy V\_single\_fathers)

	PESEL	PERSONALIA_RODZICA	PESEL_DZIECKA	PERSONALIA_DZIECKA
1	93032253197	Michał Grabowski	16232598567	Marcin Grabowski

(Rys. 24 - Zawartość perspektywy V\_single\_fathers)

## MODYFIKACJA WPISÓW W TABELACH NA PODSTAWIE PERSPEKTYW.

Pierwsza modyfikacja będzie polegała na zmianie wynagrodzenia dla rodziców bez pary (zrobimy to na podstawie wpisów w perspektywach 2 i 3).

```
--PODWYZKA DO PENSJI DLA MAM I TAT BEZ PARY

update pracownicy pr set placa=placa*1.5
where pr.nr_pesel in
(
    select pesel from v_single_mothers
)
or pr.nr_pesel in
(
    select pesel from v_single_fathers
);
```

(Rys. 25 - Podwyżka do wynagrodzenia dla rodziców bez pary)

Drugim krokiem zmniejszymy koszty o 30 procent dla rodziców bez pary.

```
--OBNIZENIE KOSZTOW O 30 PROCENT DLA RODZICOW BEZ PAR

update koszty ko set koszty=0.7*koszty
where ko.pesel_dziecka in
(
    select pesel_dziecka from v_single_mothers
) or ko.pesel_dziecka in
(
    select pesel_dziecka from v_single_fathers
);
```

(Rys. 26 - Obniżenie kosztów dla rodziców bez pary)

Uwzględnimy też rodziców, którzy pracują w przedszkolach i zmniejszymy dla nich koszty o 10 procent.

```
--OBNIZENIE KOSZTOW O 10 PROCENT DLA PRACOWNIKOW

update koszty ko set koszty=0.9*koszty
where ko.pesel_dziecka in
(
    select pesel_dziecka from v_dzieci_pracownikow
);
```

(Rys. 27 - Obniżenie kosztów dla pracowników)



## ZAPYTANIA NA PODSTAWIE BAZY DANYCH.

Pierwsze zapytanie będzie zwracało nam informację o opiekunach, przez których opiekuje się dane dziecko. Zapytanie wygląda następująco:

```
--ZAPYTANIE, ZWRACAJĄCE INFORMACJE O OPIEKUNACH(zwykłe zapytanie)
select personalia dziecko, ' opiekuje się przez ' komentarz, imie||' '||nazwisko opiekun,
nr_akt, stanowisko, data_zatr, data_zwol
from dzieci dz join pracownicy pr on(dz.nr_pesel_opiekuna=pr.nr_pesel);
```

(Rys. 28 - Zwykłe zapytanie, zwracające informację o opiekunach)

DZIECKO	KOMENTARZ	OPIEKUN	NR_AKT	STANOWISKO	DATA_ZATR	DATA_ZWOL
1 Paweł Rutkowski	opiekuje się przez	Anita Szymanska	1010	Opiekun	15.02.03	(null)
2 Mateusz Rutkowski	opiekuje się przez	Anita Szymanska	1010	Opiekun	15.02.03	(null)
3 Mariusz Wojcik	opiekuje się przez	Jowita Kucharska	1012	Opiekun	02.04.10	(null)
4 Bartosz Malinowski	opiekuje się przez	Natalia Mroz	1016	Opiekun	17.12.17	(null)
5 Dawid Samus	opiekuje się przez	Urszula Rajewska	1018	Opiekun	09.12.05	(null)
6 Iga Adamska	opiekuje się przez	Urszula Rajewska	1018	Opiekun	09.12.05	(null)
7 Paweł Walczak	opiekuje się przez	Dominika Dziwinska	1019	Opiekun	12.04.17	(null)
8 Martyna Chmielewska	opiekuje się przez	Justyna Paweloszek	1021	Opiekun	29.08.19	06.12.20
9 Matylda Rutkowska	opiekuje się przez	Katarzyna Kubiak	1022	Opiekun	16.08.02	(null)
10 Jolanta Rutkowska	opiekuje się przez	Katarzyna Kubiak	1022	Opiekun	16.08.02	(null)
11 Marcin Grabowski	opiekuje się przez	Katarzyna Kubiak	1022	Opiekun	16.08.02	(null)
12 Krzysztof Chmielewski	opiekuje się przez	Karolina Sadowska	1023	Opiekun	15.09.08	(null)
13 Justyna Samus	opiekuje się przez	Patryk Samus	1024	Opiekun	06.04.18	(null)
14 Gabriel Samus	opiekuje się przez	Patryk Samus	1024	Opiekun	06.04.18	(null)
15 Amelia Adamska	opiekuje się przez	Patryk Samus	1024	Opiekun	06.04.18	(null)
16 Damian Rutkowski	opiekuje się przez	Aleksandra Piekarska	1025	Opiekun	13.04.93	(null)
17 Katarzyna Wojcik	opiekuje się przez	Katarzyna Chodkiewicz	1028	Opiekun	28.08.19	(null)
18 Kamila Chodkiewicz	opiekuje sie przez	Katarzyna Chodkiewicz	1028	Opiekun	28.08.19	(null)

(Rys. 29 – Wynik pierwszego zapytania)

Kolejne zapytanie będzie miało w sobie złączenie kilku tabel. Zwraca ono informację o rodzicach i listę ich dzieci.

```
--ZAPYTANIE ZWRACA LISTE RODZICOW I ICH DZIECI(zlaczanie)
select imie||' '||nazwisko rodzic, rod.nr_pesel pesel_rodzica, data_ur,
listagg(personalia||' PESEL: '||dz.nr_pesel, ', ') within group(order by personalia) dzieci
from rodzice rod join dzieci dz on(dz.nr_pesel_ojca=rod.nr_pesel or dz.nr_pesel_matki=rod.nr_pesel)
group by imie||' '||nazwisko, rod.nr_pesel, data_ur
order by 3;
```

(Rys. 30 – Zapytanie ze złączeniem tabel)



RODZIC	PESEL_RODZICA	DATA_UR	DZIECI
1 Paweł Rutkowski	76081159645	11.08.76	Damian Rutkowski PESEL: 13282644888, Matylda Rutkowska PESEL: 15322749614
2 Alicja Rutkowska	77042814397	28.04.77	Damian Rutkowski PESEL: 13282644888, Matylda Rutkowska PESEL: 15322749614
3 Jola Rutkowska	80071698494	16.07.80	Jolanta Rutkowska PESEL: 16322689814, Mateusz Rutkowski PESEL: 14231488661, Paweł Rutkowski PESEL: 14231456286
4 Adam Rutkowski	81021513693	15.02.81	Jolanta Rutkowska PESEL: 16322689814, Mateusz Rutkowski PESEL: 14231488661, Paweł Rutkowski PESEL: 14231456286
5 Magda Samus	86032235232	22.03.86	Dawid Samus PESEL: 14242817278, Gabriel Samus PESEL: 15310592374, Justyna Samus PESEL: 16302967624
6 Patryk Samus	89031741566	17.03.89	Dawid Samus PESEL: 14242817278, Gabriel Samus PESEL: 15310592374, Justyna Samus PESEL: 16302967624
7 Daniel Malinowski	90052089196	20.05.90	Bartosz Malinowski PESEL: 14310138162, Klaudia Malinowska PESEL: 16220848319
8 Damian Walczak	92113061579	30.11.92	Paweł Walczak PESEL: 16291398339
9 Amanda Chmielewska	93010267392	02.01.93	Krzysztof Chmielewski PESEL: 13312764795, Martyna Chmielewska PESEL: 15310949862
10 Michał Grabowski	93032253197	22.03.93	Marcin Grabowski PESEL: 16232598567
11 Mateusz Chmielewski	93032844289	28.03.93	Krzysztof Chmielewski PESEL: 13312764795, Martyna Chmielewska PESEL: 15310949862
12 Kamil Wojcik	94040948426	09.04.94	Katarzyna Wojcik PESEL: 16251076947, Mariusz Wojcik PESEL: 14221699965
13 Paweł Adamski	94070821216	08.07.94	Amelia Adamska PESEL: 15231418551, Iga Adamska PESEL: 14240641774
14 Aisha Malinowska	94092718211	27.09.94	Bartosz Malinowski PESEL: 14310138162, Klaudia Malinowska PESEL: 16220848319
15 Angelika Adamska	96060937688	09.06.96	Amelia Adamska PESEL: 15231418551, Iga Adamska PESEL: 14240641774
16 Marysia Wojcik	96062078969	20.06.96	Katarzyna Wojcik PESEL: 16251076947, Mariusz Wojcik PESEL: 14221699965
17 Katarzyna Chodkiewicz	98021979544	19.02.98	Kamila Chodkiewicz PESEL: 16311576452
18 Alisa Walczak	98042169294	21.04.98	Paweł Walczak PESEL: 16291398339

(Rys. 31 – Wynik zapytania ze złączeniem)

W trzecim zapytaniu zostało użyte podzapytanie. Zwraca natomiast ono liczbę imion w tabelach **Dzieci**, **Pracownicy** i **Rodzice**, rozpoczynających się na daną literę.

```
--LICZBA IMION ROZPOCZYNAJĄCYCH SIĘ NA DANA LITERĘ W TABELACH DZIECI, RODZICE I PRACOWNICY(podzapytanie)
select litera, sum(l_i) liczba_imion from
(
  select distinct(trim(substr(personalia, 1, 1))) litera, count(personalia) l_i from dzieci
  group by trim(substr(personalia, 1, 1))
  union all
  select distinct(trim(substr(imie, 1, 1))), count(imie) from pracownicy
  group by trim(substr(imie, 1, 1))
  union all
  select distinct(trim(substr(imie, 1, 1))), count(imie) from rodzice
  group by trim(substr(imie, 1, 1))
)
group by litera
order by 2 desc;
```

(Rys. 32 – Zapytanie z podzapytaniem)

LITERA	LICZBA_IMION
1 K	15
2 A	14
3 M	13
4 J	8
5 D	7
6 P	6
7 I	4
8 U	3
9 N	3
10 G	1
11 O	1
12 S	1
13 B	1

(Rys. 33 – Wynik trzeciego zapytania)

Następne zapytanie będzie zwracało nam imiona I PESEL dzieci i ich rodziców oraz liczbę nieobecności w okresie prowadzenia dziennika (z podziałem na lata I miesiące). Czwarte zapytanie polega na grupowaniu.

```
--LICZBA NIEOBECNOSCI ZA CALY OKRES DZIENNIKA(operator GROUP BY)
select tl.pers personalia_dziecka, rok, miesiac, liczba
listagg('NR. PESEL: '||rod.nr_pesel ||', PERSONALIA: '||rod.imie || ' ' || rod.nazwisko, ' | ')
within group(order by rod.nr_pesel, rod.imie, rod.nazwisko) rodzice
from
(
select personalia pers, extract(year from dzien) rok, to_char(dzien, 'month') miesiac, count(*) liczba from dziennik
where obecnosci='-'
group by personalia, extract(year from dzien), to_char(dzien, 'month')
) tl join dzieci dz on(tl.pers=dz.personalia)
join rodzice rod on(dz.nr_pesel_ojca=rod.nr_pesel or dz.nr_pesel_matki=rod.nr_pesel)
group by tl.pers, rok, miesiac, liczba
order by 2, 4 desc;
```

(Rys. 34 – Zapytanie z grupowaniem)

PERSONALIA_DZIECKA	ROK	MIESIAC	LICZBA_NIEOBECNOSCI	RODZICE
1 Pawel Walczak	2018	май	5 NR. PESEL: 92113061579,	PERSONALIA: Damian Walczak   NR. PESEL: 98042169294, PERSONALIA: Alisa Walczak
2 Bartosz Malinowski	2018	январь	3 NR. PESEL: 90052089196,	PERSONALIA: Daniel Malinowski   NR. PESEL: 94092718211, PERSONALIA: Aisha Malinowska
3 Damian Rutkowski	2018	январь	3 NR. PESEL: 76081159645,	PERSONALIA: Pawel Rutkowski   NR. PESEL: 77042814397, PERSONALIA: Alicja Rutkowska
4 Jolanta Rutkowska	2018	январь	3 NR. PESEL: 80071698494,	PERSONALIA: Jola Rutkowska   NR. PESEL: 81021513693, PERSONALIA: Adam Rutkowski
5 Klaudia Malinowska	2018	январь	3 NR. PESEL: 90052089196,	PERSONALIA: Daniel Malinowski   NR. PESEL: 94092718211, PERSONALIA: Aisha Malinowska
6 Marcin Grabowski	2018	январь	3 NR. PESEL: 93032253197,	PERSONALIA: Michal Grabowski
7 Mateusz Rutkowski	2018	январь	3 NR. PESEL: 80071698494,	PERSONALIA: Jola Rutkowska   NR. PESEL: 81021513693, PERSONALIA: Adam Rutkowski
8 Matylda Rutkowska	2018	январь	3 NR. PESEL: 76081159645,	PERSONALIA: Pawel Rutkowski   NR. PESEL: 77042814397, PERSONALIA: Alicja Rutkowska
9 Katarzyna Wojcik	2018	май	2 NR. PESEL: 94040948426,	PERSONALIA: Kamil Wojcik   NR. PESEL: 96062078969, PERSONALIA: Marysia Wojcik
10 Mariusz Wojcik	2018	май	2 NR. PESEL: 94040948426,	PERSONALIA: Kamil Wojcik   NR. PESEL: 96062078969, PERSONALIA: Marysia Wojcik
11 Pawel Walczak	2018	декабрь	1 NR. PESEL: 92113061579,	PERSONALIA: Damian Walczak   NR. PESEL: 98042169294, PERSONALIA: Alisa Walczak
12 Katarzyna Wojcik	2018	июль	1 NR. PESEL: 94040948426,	PERSONALIA: Kamil Wojcik   NR. PESEL: 96062078969, PERSONALIA: Marysia Wojcik
13 Mariusz Wojcik	2018	июль	1 NR. PESEL: 94040948426,	PERSONALIA: Kamil Wojcik   NR. PESEL: 96062078969, PERSONALIA: Marysia Wojcik
14 Pawel Walczak	2019	май	24 NR. PESEL: 92113061579,	PERSONALIA: Damian Walczak   NR. PESEL: 98042169294, PERSONALIA: Alisa Walczak
15 Katarzyna Wojcik	2019	июнь	15 NR. PESEL: 94040948426,	PERSONALIA: Kamil Wojcik   NR. PESEL: 96062078969, PERSONALIA: Marysia Wojcik
16 Mariusz Wojcik	2019	июнь	15 NR. PESEL: 94040948426,	PERSONALIA: Kamil Wojcik   NR. PESEL: 96062078969, PERSONALIA: Marysia Wojcik
17 Katarzyna Wojcik	2019	май	8 NR. PESEL: 94040948426,	PERSONALIA: Kamil Wojcik   NR. PESEL: 96062078969, PERSONALIA: Marysia Wojcik
18 Mariusz Wojcik	2019	май	8 NR. PESEL: 94040948426,	PERSONALIA: Kamil Wojcik   NR. PESEL: 96062078969, PERSONALIA: Marysia Wojcik

(Rys. 35 – Wynik czwartego zapytania)

Piąte zapytanie będzie miało klauzulę HAVING. Zwraca ono informację o przedszkolach, do którego chodzą nie mniej, niż 2 dziecka.

```
--ZAPYTANIE WYPISUJE PRZEDSZKOLA DO KTOREGO CHODZA NIE MNIEJ, NIZ 2 DZIECKA(klauzula HAVING)
select prz.id_przedszkola, nazwa, adres, imie||' '||nazwisko personalia_dyrektora, count(dz.nr_pesel) liczba_dzieci
from przedszkola prz join pracownicy pr on(nr_akt_dyrektora=nr_akt)
join dzieci dz on(prz.id_przedszkola=dz.id_przedszkola)
group by prz.id_przedszkola, nazwa, adres, imie||' '||nazwisko having count(dz.nr_pesel)>=2;
```

(Rys. 36 – Zapytanie z klauzulą HAVING)

ID_PRZEDSZKOLA	NAZWA	ADRES	PERSONALIA_DYREKTORA	LICZBA_DZIECI
1	101 Teczowe	Czestochowa 42-218 ul. Dekabrystow 15	Kazimierz Stala	5
2	100 Przedszkole sw. Jana Pawla	Czestochowa 42-202 al. Jana Pawla II 26/30	Justyna Bawor	5
3	108 Przedszkole miejskie nr.1 im. Misia Uszatka	Czestochowa 42-211 ul. Szymanowskiego 9	Mateusz Wypych	2

(Rys. 37 – Wynik piątego zapytania)

## USUNIĘCIE DANYCH.

Na końcu naszej pracy trzeba nie zapomnieć usunąć wszystkie tabele, sekwencje oraz perspektywy. Robimy to za pomocą polecenia drop.

```
drop table rodzice cascade constraints;  
drop table pracownicy cascade constraints;  
drop table przedszkola cascade constraints;  
drop table dziennik cascade constraints;  
drop table dzieci cascade constraints;  
drop table koszty cascade constraints;  
drop sequence seq_nr_akt;  
drop sequence seq_id_przedszkola;  
drop view V_dzieci_pracownikow cascade constraints;  
drop view v_single_mothers cascade constraints;  
drop view v_single_fathers cascade constraints;
```

## WNIOSKI.

Podczas robienia swojej bazy danych nauczyłem się podstaw programowania w programie Oracle SQL Developer. Dzięki tej pracy teraz mam doświadczenie w tworzeniu baz danych oraz we wpisywaniu i modyfikowaniu informacji w tabelach i perspektywach.