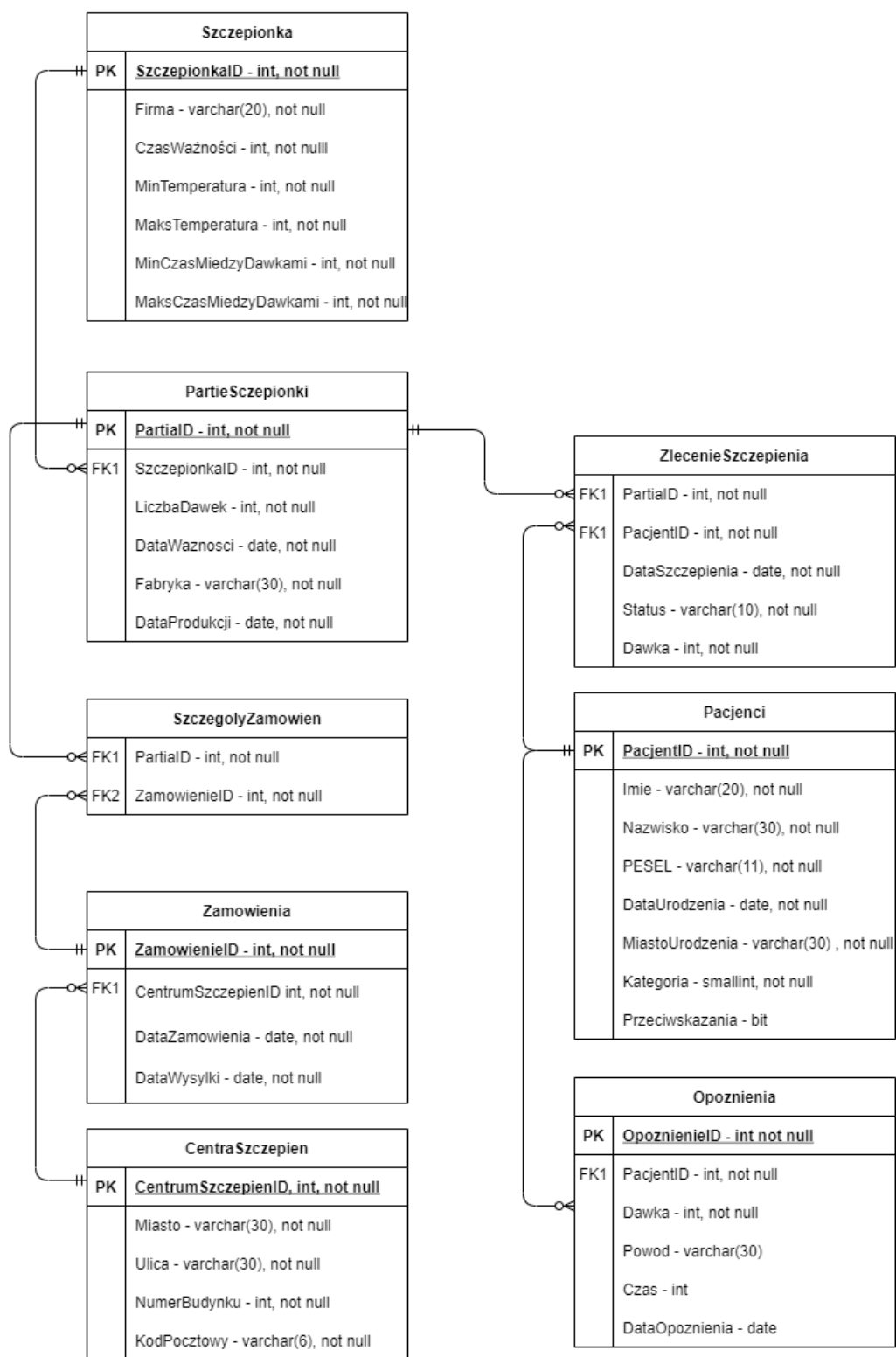


Bazy Danych, Projekt 2

Przemysław Olender

31 maja 2021

1 Projektowanie bazy danych



Rysunek 1: Diagram ER

Baza danych składa się z 8 powiązanych ze sobą tabel, są to:

- 'Pacjenci' - zawiera informacje o pacjentach szczepiących się przeciw wirusowi, imię, nazwisko, pesel, datę i miasto urodzenia, kategorię oraz informację o obecności lub braku przeciwwskazań.
- 'Opóźnienia' - zawiera informacje o opóźnieniu szczepienia pacjenta, np z powodu kwarantanny, informacje przy której dawce wystąpiło opóźnienie oraz liczbę dni, które trzeba odczekać.
- 'ZlecenieSzczepienia' - zawiera informacje o zapisach pacjentów na szczepienia, kto się zapisał, partia z której pochodzi szczepionka, kiedy odbywa się szczepienie, informacje o statusie - czy zostało zrealizowane, odwołane lub dopiero będzie zrealizowane. Zawiera dwa klucze obce, ID pacjenta oraz ID Partii szczepionki.
- 'PartieSzczepionki' - Zawiera informacje o danej partii sprowadzonej przez Centrum Szczepień, jaka jest liczba dawek, data ważności, skąd pochodzi partia i kiedy została wyprodukowana. Zawiera klucz obcy, ID szczepionki i klucz główny, ID Partii.
- 'Szczepionki' - Zawiera informacje o rodzajach produkowanych szczepionek, firmie, czasie ważności, zakresie temperatury przechowywania, oraz zakresie czasu między dwiema dawkami szczepionki. Zawiera klucz główny, ID szczepionki.
- 'SzczegolyZamowien' - zawiera informacje, jakie partie zostały zamówione w danych zamówieniu, można ją rozszerzyć na przykład o cenę danej partii. Zawiera dwa klucze obce, ID partii oraz ID zamówienia.
- 'Zamowienia' - zawiera konkretne informacje o zamówieniu, Centrum Szczepień które go dokonało, datę zamówienia, datę wysyłki. Zawiera klucz obcy, ID centrum szczepień oraz klucz główny, ID zamówienia. Chcąc bardziej uszczegółowić bazę danych można dodać kolumny z informacjami o tym, jaki pracownik był odpowiedzialny za zamówienie, datę doręczenia, informacje o transporcie.
- 'CentraSzczepien' - zawiera szczegółowe informacje o adresie centrum. Ma klucz główny, ID centrum.

2 Stworzenia bazy danych

2.1 Tworzenie tabel

Tabla 'Pacjenci' zawiera ograniczenia odnoszące się do długości Peselu (11 znaków), daty urodzenia, która nie może być późniejsza niż obecna oraz kategorii, która musi należeć do zbioru 0, 1, 2, 3, 4. Kod potrzebny do stworzenia tabeli:

```
CREATE TABLE Pacjenci(  
    "PacjentID" INT NOT NULL PRIMARY KEY CLUSTERED IDENTITY (1,1),  
    "Imie" VARCHAR(20) NOT NULL,  
    "Nazwisko" VARCHAR(30) NOT NULL,  
    "Pesel" VARCHAR(11) NOT NULL,  
    "DataUrodzenia" DATE NOT NULL,
```

```

    "MiastoUrodzenia" VARCHAR(30) NOT NULL,
    "Przeciwskazania" BIT,
    "Kategoria" SMALLINT NOT NULL,
    CONSTRAINT "Pesel" CHECK ( LEN("Pesel") = 11 ),
    CONSTRAINT "DataUrodzenia" CHECK( "DataUrodzenia" <= GETDATE() ),
    CONSTRAINT "Kategoria" CHECK( "Kategoria" >= 0 AND "Kategoria" <= 4 )
)

```

Tabela 'Opóźnienia' zawiera ograniczenia dotyczące klucza obcego, dawki szczepionki, może ona wynosić tylko 1 lub 2 oraz daty od której liczone jest opóźnienie, nie może ona być późniejsza niż dzisiejsza. Kod potrzebny do stworzenia tabeli:

```

CREATE TABLE Opóźnienia(
    "OpóźnienieID" INT NOT NULL PRIMARY KEY CLUSTERED IDENTITY (1, 1),
    "PacjentID" INT NOT NULL,
    "Dawka" INT NOT NULL,
    "Powod" VARCHAR(30),
    "Czas" INT NOT NULL,
    "DataOpóźnienia" DATE NOT NULL,
    CONSTRAINT "FK_Opóźnienie_PacjentID" FOREIGN KEY ("PacjentID") REFERENCES Pacjenci("PacjentID"),
    CONSTRAINT "OpóźnionaDawka" CHECK( Dawka=1 OR Dawka=2 ),
    CONSTRAINT "DataOpóźnienia" CHECK( "DataOpóźnienia" <= GETDATE() )
)

```

Tabela 'ZleceniaSzczepienia' zawiera ograniczenia dotyczące kluczy obcych, a także informacji o statusie szczepienia, może on przyjmować 3 wartości: 'zrealizowane', 'odwołane' i 'oczekujące'. Tabela zawiera też ograniczenia dotyczące dawki szczepionki, może ona wynosić tylko 1 lub 2. Kod potrzebny do stworzenia tabeli:

```

CREATE TABLE ZlecenieSzczepienia(
    "PartiaID" INT NOT NULL,
    "PacjentID" INT NOT NULL,
    "DataSzczepienia" DATETIME NOT NULL,
    "Status" VARCHAR(12) NOT NULL,
    "Dawka" INT NOT NULL,
    CONSTRAINT "FK_Partie_Zlecenie_PartiaID" FOREIGN KEY ("PartiaID")
REFERENCES PartieSzczepionki("PartiaID"),
    CONSTRAINT "FK_Pacjenci_PacjentID" FOREIGN KEY ("PacjentID")
REFERENCES Pacjenci("PacjentID"),
    CONSTRAINT "Status" CHECK( "Status" = 'zrealizowane'
OR "Status" = 'oczekujące' OR "Status" = 'odwołane' ),
    CONSTRAINT "Dawka" CHECK( "Dawka" >= 1 OR "Dawka" <= 2 )
)

```

Tabela 'PartieSzczepionki' zawiera ograniczenia dotyczące kluczy obcych oraz daty produkcji, która nie może być późniejsza od obecnej. Po stworzeniu tabeli jest ona modyfikowana, dodana jest kolumna 'DataWażności' oraz ustawiona zostaje jej domyślna wartość jako data produkcji szczepionki + czas jej ważności. Kod potrzebny do stworzenia tabeli:

```

CREATE TABLE PartieSzczepionki(
    "PartiaID" INT NOT NULL PRIMARY KEY CLUSTERED IDENTITY (1,1),
    "SzczepionkaID" INT NOT NULL,
    "LiczbaDawek" INT NOT NULL,
    "Fabryka" VARCHAR(30),
    "DataProdukcji" DATE NOT NULL,
    CONSTRAINT "FK_Szczepionki_SzczepionkaID" FOREIGN KEY ("SzczepionkaID")
    REFERENCES Szczepionki("SzczepionkaID"),
    CONSTRAINT "DataProdukcji" CHECK( "DataProdukcji" <= GETDATE() )

```

```

ALTER TABLE PartieSzczepionki ADD DataWaznosci DATE NULL

```

```

UPDATE PartieSzczepionki SET DataWaznosci=(
    SELECT DATEADD(DAY, CzasWaznosci, DataProdukcji)
    AS DataWaznosci FROM PartieSzczepionki ps
    JOIN Szczepionki s ON s.SzczepionkaID=ps.SzczepionkaID
    WHERE PartieSzczepionki.PartiaID = ps.PartiaID
)
)

```

Tabela 'Szczepionki' ma jedno ograniczenie, temperatura podana w stopniach Celsjusza nie może być mniejsza od zera bezwzględnego. Kod potrzebny do stworzenia tabeli:

```

CREATE TABLE Szczepionki(
    "SzczepionkaID" INT NOT NULL PRIMARY KEY CLUSTERED IDENTITY (1, 1),
    "Firma" VARCHAR(30) NOT NULL,
    "CzasWaznosci" INT NOT NULL,
    "MinTemperatura" FLOAT NOT NULL,
    "MaxTemperatura" FLOAT NOT NULL,
    "MinCzasMiedzyDawkami" INT NOT NULL,
    "MaxCzasMiedzyDawkami" INT NOT NULL,
    CONSTRAINT "MaxTemperatura" CHECK( "MaxTemperatura" > -273 ),
    CONSTRAINT "MinTemperatura" CHECK( "MinTemperatura" > -273 )
)

```

Tablela 'SzczegolyZamowien' zawiera ograniczenia dotyczące kluczy obcych. Kod potrzebny do stworzenia tabeli:

```

CREATE TABLE SzczegolyZamowien(
    "ZamowienieID" INT NOT NULL,
    "PartiaID" INT NOT NULL,
    CONSTRAINT "FK_Partie_Zamowienie_PartiaID" FOREIGN KEY ("PartiaID")
    REFERENCES PartieSzczepionki("PartiaID"),
    CONSTRAINT "FK_Zamowienia_ZamowienieID" FOREIGN KEY ("ZamowienieID")
    REFERENCES Zamowienia("ZamowienieID")
)

```

Tabela 'Zamowienia' zawiera ograniczenia dotyczące kluczy obcych i dat, które muszą być późniejsze niż dzisiejsza. Kod potrzebny do stworzenia tabeli:

```

CREATE TABLE Zamowienia(
    "ZamowienieID" INT NOT NULL PRIMARY KEY CLUSTERED IDENTITY(1, 1),
    "CentrumSzczepienID" INT NULL,
    "DataZamowienia" DATE NOT NULL,
    "DataWysylki" DATE NULL,
    CONSTRAINT "DataZamowienia" CHECK( "DataZamowienia" <= GETDATE() ),
    CONSTRAINT "DataWysylki" CHECK( "DataWysylki" <= GETDATE() ),
    CONSTRAINT FK_CentraSzczepien_CentrumID FOREIGN KEY ("CentrumSzczepienID")
    REFERENCES CentraSzczepien("CentrumSzczepienID")
)

```

Tabela 'CentraSzczepien' zawiera ograniczenie dotyczące długości kodu pocztowego. Kod potrzebny do stworzenia tabeli:

```

CREATE TABLE CentraSzczepien(
    "CentrumSzczepienID" INT NOT NULL PRIMARY KEY CLUSTERED IDENTITY (1, 1),
    "Miasto" VARCHAR(30) NOT NULL,
    "Ulica" VARCHAR(30) NOT NULL,
    "NumerBudynku" INT NOT NULL,
    "KodPocztowy" VARCHAR(6) NOT NULL,
    CONSTRAINT "KodPocztowy" CHECK ( LEN("KodPocztowy") = 6 )
)

```

2.2 Wypełnienie tabel danymi

```

INSERT INTO Pacjenci VALUES
('Jan', 'Nowak', '71032250188', '1971-03-22', 'Radom', 0, 2),
('Robert', 'Lewandowski', '83051355456', '1983-05-13', 'Karczew', 0, 4),
('Robert', 'Kubica', '83060150442', '1983-06-01', 'Sosnowiec', 1, 3),
('Laura', 'Bobas', '77060315460', '1977-06-03', 'Warszawa', 0, 0),
('Zbyszek', 'Leszek', '62070539045', '1962-07-05', 'Kraków', 0, 1),
('Ola', 'Kowalska', '82090144822', '1982-09-01', 'Warszawa', 0, 3),
('Marcin', 'Prokop', '93011817708', '1993-01-18', 'Warszawa', 1, 4),
('Wojciech', 'Kowalczyk', '95010111111', '1995-01-01', 'Warszawa', 0, 4)

```

```

UPDATE Pacjenci SET Imie = 'Jakub', DataUrodzenia = '1999-10-19'
WHERE PacjentID = 1
UPDATE Pacjenci SET Pesel = '84070120444', MiastoUrodzenia = 'Bytom'
WHERE PacjentID = 3
UPDATE Pacjenci SET Nazwisko = 'Najman', DataUrodzenia = '1985-01-02',
Kategoria = 3 WHERE PacjentID = 7

```

	PacjentID	Imie	Nazwisko	Pesel	DataUrodzenia	MiastoUrodzenia	Przeciwwskazania	Kategoria
1	1	Jakub	Nowak	71032250188	1999-10-19	Radom	0	2
2	2	Robert	Lewandowski	83051355456	1983-05-13	Karczew	0	4
3	3	Robert	Kubica	84070120444	1983-06-01	Bytom	1	3
4	4	Laura	Bobas	77060315460	1977-06-03	Warszawa	0	0
5	5	Zbyszek	Leszek	62070539045	1962-07-05	Kraków	0	1
6	6	Ola	Kowalska	82090144822	1982-09-01	Warszawa	0	3
7	7	Marcin	Najman	93011817708	1985-01-02	Warszawa	1	3
8	8	Wojci...	Kowalczyk	95010111111	1995-01-01	Warszawa	0	4

Rysunek 2: Tabela 'Pacjenci'

```
INSERT INTO Opoznienia VALUES
(3, 1, 'wirus', 14, '2021-03-03'),
(6, 2, 'kwarantanna', 14, '2021-04-12')
```

	OpoznienieID	PacjentID	Dawka	Powod	Czas	DataOpoznienia
1	1	3	1	wirus	14	2021-03-03
2	2	6	2	kwarantanna	14	2021-04-12

Rysunek 3: Tabela 'Opoznienia'

```
INSERT INTO Szczepionki VALUES
('Pfizer', 80, -60, -30, 15, 60),
('Moderna', 90, 3, 23, 10, 55),
('AstraZeneca', 100, 10, 20, 20, 30),
('Johnson&Johnson', 45, 5, 30, 14, 21),
('Winiary', 29, -10, 50, 28, 35)
```

	SzczepionkaID	Firma	CzasWaznosci	MinTemperatura	MaxTemperatura	MinCzasMiedzyDawkami	MaxCzasMiedzyDawkami
1	1	Pfizer	80	-60	-30	15	60
2	2	Moderna	90	3	23	10	55
3	3	AstraZ...	100	10	20	20	30
4	4	Johnso...	45	5	30	14	21
5	5	Winiary	29	-10	50	28	35

Rysunek 4: Tabela 'Szczepionki'

```
INSERT INTO PartieSzczepionki (SzczepionkaID, LiczbaDawek,
Fabryka, DataProdukcji) VALUES
(1, 100, 'Pfizer Warszawa', '2021-01-14'),
(2, 50, 'Moderna Berlin', '2021-01-17'),
(3, 45, 'AstraZeneca Madryt', '2021-02-05'),
(4, 30, 'Johnson&Johnson Londyn', '2021-02-28'),
```

```
(5, 100, 'Wininary Kielce', '2021-03-04'),
(1, 200, 'Pfizer Warszawa', '2021-03-07'),
(2, 60, 'Moderna Lublin', '2021-03-26'),
(3, 75, 'AstraZeneca Łódź', '2021-04-01'),
(4, 85, 'Johnson&Johnson Lublin', '2021-04-14'),
(5, 40, 'Winiary Kielce', '2021-04-17'),
(1, 400, 'Pfizer Warszawa', '2021-04-19'),
(1, 50, 'Pfizer Gdańsk', '2021-04-28'),
(2, 200, 'Modrna Berlin', '2021-05-12'),
(1, 200, 'Pfizer Gdańsk', '2021-05-18'),
(2, 40, 'Moderna Lublin', '2021-05-20'),
(3, 50, 'AstraZeneca Łódź', '2021-05-25')
```

	PartialID	SzczepionkaID	LiczbaDawek	Fabryka	DataProdukcji	DataWaznosci
1	1	1	100	Pfizer Warszawa	2021-01-14	2021-04-04
2	2	2	50	Moderna Berlin	2021-01-17	2021-04-17
3	3	3	45	AstraZeneca ...	2021-02-05	2021-05-16
4	4	4	30	Johnson&John...	2021-02-28	2021-04-14
5	5	5	100	Wininary Kielce	2021-03-04	2021-04-02
6	6	1	200	Pfizer Warszawa	2021-03-07	2021-05-26
7	7	2	60	Moderna Lublin	2021-03-26	2021-06-24
8	8	3	75	AstraZeneca Ł...	2021-04-01	2021-07-10
9	9	4	85	Johnson&John...	2021-04-14	2021-05-29
10	10	5	40	Winiary Kielce	2021-04-17	2021-05-16
11	11	1	400	Pfizer Warszawa	2021-04-19	2021-07-08
12	12	1	50	Pfizer Gdańsk	2021-04-28	2021-07-17
13	13	2	200	Modma Berlin	2021-05-12	2021-08-10
14	14	1	200	Pfizer Gdańsk	2021-05-18	2021-08-06
15	15	2	40	Moderna Lublin	2021-05-20	2021-08-18
16	16	3	50	AstraZeneca Ł...	2021-05-25	2021-09-02

Rysunek 5: Tabela 'PartieSzczepionki'

```
INSERT INTO CentraSzczepien VALUES
('Warszawa', 'Wawelska', 44, '03-420'),
('Radom', 'Majowa', 53, '87-648'),
('Gliwice', 'Długa', 2, '12-234'),
('Warszawa', 'Piękna', 29, '01-001'),
('Kraków', 'Mokra', 9, '32-543')
```


	CentrumSzczepienID	Miasto	Ulica	NumerBudynku	KodPocztowy
1	1	Warszawa	Wawelska	44	03-420
2	2	Radom	Majowa	53	87-648
3	3	Gliwice	Długa	2	12-234
4	4	Warszawa	Piękna	29	01-001
5	5	Kraków	Mokra	9	32-543

Rysunek 6: Tabela 'CentraSzczepien'

```
INSERT INTO Zamowienia VALUES
(1, '2021-01-13', '2021-01-18'),
(2, '2021-02-02', '2021-02-05'),
(3, '2021-02-25', '2021-03-07'),
(4, '2021-03-21', '2021-04-01'),
(5, '2021-04-14', '2021-04-19'),
(1, '2021-04-27', '2021-04-29'),
(2, '2021-05-11', '2021-05-12'),
(1, '2021-05-17', '2021-05-19')
```

	ZamowienieID	CentrumSzczepienID	DataZamowienia	DataWysylki
1	1	1	2021-01-13	2021-01-18
2	2	2	2021-02-02	2021-02-05
3	3	3	2021-02-25	2021-03-07
4	4	4	2021-03-21	2021-04-01
5	5	5	2021-04-14	2021-04-19
6	6	1	2021-04-27	2021-04-29
7	7	2	2021-05-11	2021-05-12
8	8	1	2021-05-17	2021-05-19

Rysunek 7: Tabela 'Zamowienia'

```
INSERT INTO SzczegolyZamowien VALUES
(1, 1),
(1, 2),
(2, 3),
(3, 4),
(3, 5),
(3, 6),
(4, 7),
(4, 8),
(5, 9),
(5, 10),
(5, 11),
(6, 12),
(7, 13),
(8, 14)
```

	ZamowienieID	PartialID
1	1	1
2	1	2
3	2	3
4	3	4
5	3	5
6	3	6
7	4	7
8	4	8
9	5	9
10	5	10
11	5	11
12	6	12
13	7	13
14	8	14

Rysunek 8: Tabela 'SzczegolyZamowien'

```

INSERT INTO ZlecenieSzczepienia VALUES
(1, 4, '2021-01-21 08:15:00', 'zrealizowane', 1),
(1, 4, '2021-02-20 10:30:00', 'zrealizowane', 2),
(1, 5, '2021-01-29 09:45:00', 'zrealizowane', 1),
(2, 5, '2021-03-02 08:00:00', 'zrealizowane', 2),
(1, 1, '2021-02-10 09:00:00', 'zrealizowane', 1),
(6, 1, '2021-03-15 09:00:00', 'zrealizowane', 2),
(2, 3, '2021-03-05 21:00:00', 'odwolane', 1),
(2, 3, '2021-03-22 12:30:00', 'zrealizowane', 1),
(13, 3, '2021-05-13 14:55:00', 'oczekujace', 2),
(6, 6, '2021-03-15 18:00:00', 'zrealizowane', 1),
(6, 6, '2021-04-15 20:00:00', 'odwolane', 2),
(11, 6, '2021-05-15 13:00:00', 'oczekujace', 2),
(6, 2, '2021-04-18 11:55:00', 'zrealizowane', 1),
(12, 2, '2021-05-20 12:40:00', 'oczekujace', 2),
(11, 7, '2021-05-03 16:00:00', 'zrealizowane', 1)

```

	PartialID	PacjentID	DataSzczepienia	Status	Dawka
1	1	4	2021-01-21 08:15:00.000	zrealizowane	1
2	1	4	2021-02-20 10:30:00.000	zrealizowane	2
3	1	5	2021-01-29 09:45:00.000	zrealizowane	1
4	2	5	2021-03-02 08:00:00.000	zrealizowane	2
5	1	1	2021-02-10 09:00:00.000	zrealizowane	1
6	6	1	2021-03-15 09:00:00.000	zrealizowane	2
7	2	3	2021-03-05 21:00:00.000	odwolane	1
8	2	3	2021-03-22 12:30:00.000	zrealizowane	1
9	13	3	2021-05-13 14:55:00.000	oczekujace	2
10	6	6	2021-03-15 18:00:00.000	zrealizowane	1
11	6	6	2021-04-15 20:00:00.000	odwolane	2
12	11	6	2021-05-15 13:00:00.000	oczekujace	2
13	6	2	2021-04-18 11:55:00.000	zrealizowane	1
14	12	2	2021-05-20 12:40:00.000	oczekujace	2
15	11	7	2021-05-03 16:00:00.000	zrealizowane	1

Rysunek 9: Tabela 'ZlecenieSzczepienia'

3 Indeksy

Indeksy sklasteryzowane są ustawione przy kluczach głównych wszystkich tabel, zostały automatycznie stworzone podczas powoływania tabel. Poniżej znajduje się kod, który mógłby posłużyć do stworzenia indeksów sklasteryzowanych, jednak przez automatyzację tego procesu nie był potrzebny.

```
CREATE CLUSTERED INDEX IND_PK_Pacjenci_PacjentID
ON Pacjenci("PacjentID" ASC);
CREATE UNIQUE CLUSTERED INDEX IND_PK_Opoznienia_OpoznienieID
ON Opoznienia("OpoznienieID" ASC);
CREATE UNIQUE CLUSTERED INDEX IND_PK_PartieSzczepionki_PartialID
ON PartieSzczepionki("PartialID" ASC);
CREATE UNIQUE CLUSTERED INDEX IND_PK_Szczepionki_SzczepionkaID
ON Szczepionki("SzczepionkaID" ASC);
CREATE UNIQUE CLUSTERED INDEX IND_PK_Zamowienia_ZamowienieID
ON Zamowienia("ZamowienieID" ASC);
CREATE UNIQUE CLUSTERED INDEX IND_PK_CentraSzczepien_CentrumSzczepienID
ON CentraSzczepien("CentrumSzczepienID" ASC);
```

Indeksy niesklasteryzowane są ustawione w kluczach obcych wszystkich tabel, dodatkowo jeden indeks ustawiony jest na kolumnie 'Firma' w tabeli 'Szczepionki', umożliwi to szybsze wyszukiwanie szczepionek pochodzących od konkretnego producenta.

Kod tworzenia indeksów niesklasteryzowanych:

```
CREATE NONCLUSTERED INDEX IND_FK_Opoznienia_PacjentID
ON Opoznienia("PacjentID");
```

```

CREATE NONCLUSTERED INDEX IND_FK_ZlecenieSzczepienia_PacjentID
ON ZlecenieSzczepienia("PacjentID");
CREATE NONCLUSTERED INDEX IND_FK_ZlecenieSzczepienia_PartiaID
ON ZlecenieSzczepienia("PartiaID");
CREATE NONCLUSTERED INDEX IND_FK_PartieSzczepionki_SzczepionkaID
ON PartieSzczepionki("SzczepionkaID");
CREATE NONCLUSTERED INDEX IND_FK_SzczegolyZamowien_PartiaID
ON SzczegolyZamowien("PartiaID");
CREATE NONCLUSTERED INDEX IND_FK_SzczegolyZamowien_ZamowienieID
ON SzczegolyZamowien("ZamowienieID");
CREATE NONCLUSTERED INDEX IND_FK_Zamowienia_CentrumSzczepienID
ON Zamowienia("CentrumSzczepienID");
CREATE NONCLUSTERED INDEX IND_FK_Szczepionki_Firma
ON Szczepionki("Firma");

```

4 Zapytania SQL

4.1 Zapytanie 1

Liczba zleconych szczepień na najbliższy tydzień.

Zapytanie korzysta z tabeli 'ZlecenieSzczepienia', wybiera i zlicza te rekordy, które mają status 'oczekujace' i datę większą niż obecna, ale mniejszą niż obecna plus siedem dni.

Kod zapytania:

```

SELECT COUNT(*) AS LiczbaSzczepienWPrzyszlymTygodniu
FROM ZlecenieSzczepienia
WHERE DataSzczepienia >= GETDATE()
AND DataSzczepienia <= DATEADD(DAY, 7, DataSzczepienia)
AND Status = 'oczekujace'

```

	LiczbaSzczepienWPrzyszlymTygodniu
1	0

Rysunek 10: Wynik zapytania 1

4.2 Zapytanie 2

Id Centrum Szczepień, które najwięcej procentowo nie wykorzystało szczepionek (skończyła się data ważności). Zapytanie łączy tabele 'CentraSzczepien', 'Zamowienia', 'Szczegolyzamowien', 'PartieSzczepionki', a następnie wykorzystując left join dołącza tabele 'ZlecenieSzczepienia', wybiera przeterminowane szczepionki i oblicza ile z nich zostało podanych. Grupuje rekordy po ID centrum Szczepien i ID Partii a następnie oblicza ile procent szczepionek nie zostało wykorzystanych.

Kod zapytania:

```

SELECT TOP 1 CentrumSzczepienID, ROUND(CAST(SUM(ZakupioneDawki) -
SUM(PodaneDawki) AS FLOAT) /
CAST(SUM(ZakupioneDawki) AS FLOAT), 4) * 100
AS ProcentNiewykorzytsanych FROM (
    SELECT cs.CentrumSzczepienID, MAX(LiczbaDawek) AS ZakupioneDawki,
    COUNT(PacjentID) AS PodaneDawki
    FROM CentraSzczepien cs
    JOIN Zamowienia z ON cs.CentrumSzczepienID=z.CentrumSzczepienID
    JOIN SzczegolyZamowien sz ON sz.ZamowienieID=z.ZamowienieID
    JOIN PartieSzczepionki ps ON ps.PartiaID=sz.PartiaID
    LEFT JOIN ZlecenieSzczepienia zs ON zs.PartiaID=ps.PartiaID
    WHERE DataWaznosci < GETDATE()
    AND (Status IS NULL OR Status = 'zrealizowane')
    GROUP BY cs.CentrumSzczepienID, ps.PartiaID
) AS x
GROUP BY CentrumSzczepienID
ORDER BY ProcentNiewykorzytsanych DESC

```

	CentrumSzczepienID	ProcentNiewykorzytsanych
1	2	100

Rysunek 11: Wynik zapytania 2

4.3 Zapytanie 3

Łączna liczba wszystkich szczepionek osobno dla każdej z firm.

Zapytanie łączy tabele 'Szczepionki' i 'PartieSzczepionek' za pomocą kolumny 'SzczepionkaID', grupuje rekordy po nazwie firmy szczepionki i sumuje liczbę dawek.

Kod zapytania:

```

SELECT Firma, SUM(LiczbaDawek) AS LacznaLiczbaDawek FROM Szczepionki s JOIN
PartieSzczepionki ps ON s.SzczepionkaID=ps.SzczepionkaID
GROUP BY Firma

```

	Firma	LacznaLiczbaDawek
1	AstraZeneca	170
2	Johnson&Johnson	115
3	Moderna	350
4	Pfizer	950
5	Winiary	140

Rysunek 12: Wynik zapytania 3

4.4 Zapytanie 4

Lista pacjentów będących po pierwszej dawce, którzy drugą dawkę będą mieć opóźnioną ze względu na kwarantannę, sortowaną po kategorii pacjenta (od 0 do 4).

Zapytanie łączy tabele 'Pacjenci', 'ZlecenieSzczepienia' i 'Opóźnienia' za pomocą kolumny 'PacjentID' a następnie wybiera rekordy gdzie pierwsza dawka została zrealizowana a druga jest opóźniona.

Kod zapytania:

```
SELECT Imie, Nazwisko, o.Dawka, DataSzczepienia, Powod, Kategoria FROM Pacjenci p
JOIN ZlecenieSzczepienia zs ON p.PacjentID=zs.PacjentID
JOIN Opóźnienia o ON o.PacjentID=p.PacjentID
WHERE zs.Dawka = 1 AND Status = 'zrealizowane' AND DataSzczepienia < GETDATE()
AND o.Dawka=2 AND Powod='kwarantanna'
ORDER BY Kategoria
```

	Imie	Nazwisko	Dawka	DataSzczepienia	Powod
1	Ola	Kowalska	2	2021-03-15 18:00:00.000	kwarantanna

Rysunek 13: Wynik zapytania 4

4.5 Zapytanie 5

Lista fabryk szczepionek posortowana malejąco względem liczby wykorzystanych szczepionek danej firmy.

Zapytanie łączy tabele 'Szczepionki', 'PartieSzczepionki' i 'ZlecenieSzczepienia', wybiera rekordy o statusie 'zrealizowane' i grupuje tabele po zmiennych 'Fabryka' i 'Firma', w końcu sortuje rekordy względem wykorzystanych szczepionek.

Kod zapytania:

```
SELECT Fabryka, Firma, COUNT(*) AS LiczbaWykorzystanychSzczepionek
FROM Szczepionki s
JOIN PartieSzczepionki ps ON s.SzczepionkaID=ps.SzczepionkaID
JOIN ZlecenieSzczepienia zs ON zs.PartiaID=ps.PartiaID
WHERE Status='zrealizowane'
GROUP BY Fabryka, Firma
ORDER BY LiczbaWykorzystanychSzczepionek DESC
```

	Fabryka	Firma	LiczbaWykorzystanychSzczepionek
1	Pfizer Warszawa	Pfizer	8
2	Moderna Berlin	Moderna	2

Rysunek 14: Wynik zapytania 5

5 Procedura składowana

5.1 Wyjaśnienie logiki

Procedura składowana przyjmuje cztery argumenty: @PacjentID - ID pacjenta, @RodzajSzczepionki - nazwę firmy szczepionki, którą ma się zaszczepić (zakładam, że rejestrując się na drugą dawkę szczepienia wyszukiwana jest ta sama szczepionka co przy 1 dawce), @CentrumSzczepienID - ID Centrum Szczepien w którym zaszczepi się pacjent i @Data - preferowany termin szczepienia.

Pseudokod pozwalający zrozumieć logikę procedury:

p1 = zapytanie zwracające partie dostępnej szczepionki w centrum

```
if ( len(PartiaDostepnaWPrzychodni ) is null )
{
    p2 = zapytanie zwracające partie dostępnej szczepionki w magazynie
    if ( len(p2) IS NOT NULL)
    {
        dodanie zamówienia
    }
}
else
{
    d = Dawka szczepionki którą ostatnio przyjął pacjent
    if ( d = 0 ) AND (exists ( zapytanie o czas ))
    {
        dodanie zlecenia 1 dawki szczepienia
    }
    if ( d = 1 ) AND (exists ( zapytanie o czas ))
    {
        dodanie zlecenia 2 dawki szczepienia
    }
}
```

Na początku konieczne jest danego dnia będą dostępne szczepionki z danej firmy, wykorzystuję do tego zapytanie, które zwraca liczbę dostępnych szczepionek danego producenta w danym Centrum i dniu. Jego kod to:

```
SELECT TOP 1 ps.PartiaID FROM CentraSzczepien cs
JOIN Zamowienia z ON cs.CentrumSzczepienID=z.CentrumSzczepienID
JOIN SzczegolyZamowien sz ON sz.ZamowienieID=z.ZamowienieID
JOIN PartieSzczepionki ps ON ps.PartiaID=sz.PartiaID
LEFT JOIN ZlecenieSzczepienia zs ON zs.PartiaID=ps.PartiaID
WHERE DataWaznosci > @Data
AND ISNULL(DataSzczepienia, 0) < @Data
AND cs.CentrumSzczepienID = @CentrumSzczepienID
AND SzczepionkaID = (
    SELECT SzczepionkaID FROM Szczepionki WHERE Firma = @RodzajSzczepionki
```

```
) AND (Status IS NULL OR Status = 'zrealizowane' OR Status = 'oczekujace')
GROUP BY cs.CentrumSzczepienID, ps.PartiaID, SzczepionkaID
HAVING MAX(LiczbaDawek) - COUNT(PacjentID) > 0
```

Jeśli nie ma takiej szczepionki w Centrum, sprawdzane jest, czy znajduje się w magazynie, jeśli tak, dana partia jest rezerwowana, czyli zostają dodane nowe rekordy do tabeli 'Zamowienia' i 'SzczegolyZamowien'. Zapytanie sprawdzające czy w magazynie jest dana szczepionka:

```
SELECT TOP 1 PartiaID FROM PartieSzczepionki
WHERE PartiaID NOT IN (SELECT DISTINCT PartiaID FROM SzczegolyZamowien)
AND SzczepionkaID = (
    SELECT SzczepionkaID FROM Szczepionki WHERE Firma = @RodzajSzczepionki
) AND DataWaznosci > @Data
```

Jeśli wybrana szczepionka jest dostępna w podanym dniu to sprawdzane jest, czy upłynął minimalny zalecany czas od 1 dawki lub opóźnienia (opóźnienie traktuje jako liczbę dni dodaną do minimalnego czasu między dwoma dawkami). Jeśli minął odpowiedni czas pacjent jest zapisywany na szczepienie, dodawany jest zatem rekord do tabeli 'ZlecenieSzczepienia'.

Te procedurę rozbiłem na 3 zapytania, pierwsze sprawdza ile dawek szczepionki przyjął pacjent, zwraca ono liczbę 0, 1 lub 2:

```
SELECT MAX(ISNULL(Dawka, 0)) FROM Pacjenci p
LEFT JOIN ZlecenieSzczepienia zs ON zs.PacjentID = p.PacjentID
WHERE (Status != 'odwolane' OR Status IS NULL) AND p.PacjentID= @PacjentID
GROUP BY p.PacjentID
```

Jeśli pacjent nie przyjął jeszcze żadnej dawki to sprawdzane jest, czy nie jest akurat w trakcie opóźnienia, jeśli nie, jest zapisywany na szczepienie:

```
SELECT * FROM Pacjenci p
LEFT JOIN ZlecenieSzczepienia zs ON zs.PacjentID = p.PacjentID
LEFT JOIN Opoznienia o ON o.PacjentID=p.PacjentID
WHERE p.PacjentID=8
AND DATEADD(DAY, ISNULL(Czas, 0), ISNULL(o.DataOpoznienia, '1900-01-01')) < @Data
```

Jeśli pacjent przyjął już pierwszą dawkę to sprawdzane jest, czy minął wystarczający czas od niej i od ewentualnego opóźnienia, jeśli tak pacjent jest zapisywany na drugą dawkę:

```
SELECT * FROM Pacjenci p
LEFT JOIN ZlecenieSzczepienia zs ON zs.PacjentID = p.PacjentID
LEFT JOIN Opoznienia o ON o.PacjentID=p.PacjentID
JOIN PartieSzczepionki ps ON ps.PartiaID=zs.PartiaID
JOIN Szczepionki s ON s.SzczepionkaID = ps.SzczepionkaID
WHERE p.PacjentID=7 AND Status = 'zrealizowane'
AND DATEADD(DAY, ISNULL(Czas, 0), ISNULL(o.DataOpoznienia, '1900-01-01')) < @Data
AND DATEADD(DAY, MinCzasMiedzyDawkami, DataSzczepienia) < @Data
```


5.2 Kod całej procedury

```
CREATE PROCEDURE NoweZlecenie @PacjentID INT, @RodzajSzczepionki varchar(30),
@CentrumSzczepienID INT, @Data DATE
AS
BEGIN
    --sprawdzenie czy jest ta szczepionka
    DECLARE @DostepnaPartia INT = (
        SELECT TOP 1 ps.PartiaID FROM CentraSzczepien cs
        JOIN Zamowienia z ON cs.CentrumSzczepienID=z.CentrumSzczepienID
        JOIN SzczegolyZamowien sz ON sz.ZamowienieID=z.ZamowienieID
        JOIN PartieSzczepionki ps ON ps.PartiaID=sz.PartiaID
        LEFT JOIN ZlecenieSzczepienia zs ON zs.PartiaID=ps.PartiaID
        WHERE DataWaznosci > @Data
        AND ISNULL(DataSzczepienia, 0) < @Data
        AND cs.CentrumSzczepienID = @CentrumSzczepienID
        AND SzczepionkaID = (
            SELECT SzczepionkaID FROM Szczepionki WHERE Firma = @RodzajSzczepionki
        ) AND (Status IS NULL OR Status = 'zrealizowane' OR Status = 'oczekujace')
        GROUP BY cs.CentrumSzczepienID, ps.PartiaID, SzczepionkaID
        HAVING MAX(LiczbaDawek) - COUNT(PacjentID) > 0
    )
    -- jesli nie ma
    IF LEN(@DostepnaPartia) IS NULL
    BEGIN
        DECLARE @PartaMag INT = (
            SELECT TOP 1 PartiaID FROM PartieSzczepionki
            WHERE PartiaID NOT IN (SELECT DISTINCT PartiaID FROM SzczegolyZamowien)
            AND SzczepionkaID = (
                SELECT SzczepionkaID FROM Szczepionki WHERE Firma = @RodzajSzczepionki
            )
            AND DataWaznosci > @Data
        )
        IF LEN(@PartaMag) > 0
        BEGIN
            INSERT INTO Zamowienia (CentrumSzczepienID, DataZamowienia) VALUES
            (@CentrumSzczepienID, GETDATE())
            INSERT INTO SzczegolyZamowien (ZamowienieID, PartiaID) VALUES
            ((SELECT MAX(ZamowienieID) FROM Zamowienia), @PartaMag)
        END
    END
    --jesli szczepionka jest dostepna w wybranym centrum
    ELSE
    BEGIN
        --sprawdzenie ile dawek przyjal pacjent
        DECLARE @Dawka INT =
        (
```

```

SELECT MAX(ISNULL(Dawka, 0)) FROM Pacjenci p
LEFT JOIN ZlecenieSzczepienia zs ON zs.PacjentID = p.PacjentID
WHERE (Status != 'odwolane' OR Status IS NULL) AND p.PacjentID= @PacjentID
GROUP BY p.PacjentID
)
--jesli 0 to zapis na 1
IF @Dawka = 0 AND EXISTS (
SELECT * FROM Pacjenci p
LEFT JOIN ZlecenieSzczepienia zs ON zs.PacjentID = p.PacjentID
LEFT JOIN Opoznienia o ON o.PacjentID=p.PacjentID
WHERE p.PacjentID=8
AND DATEADD(
    DAY, ISNULL(Czas, 0), ISNULL(o.DataOpoznienia, '1900-01-01')
) < @Data
)
BEGIN
INSERT INTO ZlecenieSzczepienia VALUES
(@DostepnaPartia, @PacjentID, @Data, 'oczekujace', 1)
END
--jesli 1 to zapis na 2
IF @Dawka = 1 AND EXISTS (
SELECT * FROM Pacjenci p
LEFT JOIN ZlecenieSzczepienia zs ON zs.PacjentID = p.PacjentID
LEFT JOIN Opoznienia o ON o.PacjentID=p.PacjentID
JOIN PartieSzczepionki ps ON ps.PartiaID=zs.PartiaID
JOIN Szczepionki s ON s.SzczepionkaID = ps.SzczepionkaID
WHERE p.PacjentID=7 AND Status = 'zrealizowane'
AND DATEADD(
    DAY, ISNULL(Czas, 0), ISNULL(o.DataOpoznienia, '1900-01-01')
) < @Data
AND DATEADD(DAY, MinCzasMiedzyDawkami, DataSzczepienia) < @Data
)
BEGIN
INSERT INTO ZlecenieSzczepienia VALUES
(@DostepnaPartia, @PacjentID, @Data, 'oczekujace', 2)
END
END
END

```

5.3 Test działania

Aby przetestować działanie procedury powstały 3 testy: zapis pacjenta na 1 dawkę szczepienia, zapis na drugą dawkę i wyszukanie szczepionki niedostępnej w wybranym Centrum.

Zapis na 1 dawkę:

```

BEGIN TRANSACTION
SELECT * FROM ZlecenieSzczepienia WHERE PacjentID = 8

```

```
EXEC NoweZlecenie @PacjentID = 8, @RodzajSzczepionki = 'Pfizer',
@CentrumSzczepienID = 1, @Data = '2021-06-01';
SELECT * FROM ZlecenieSzczepienia WHERE PacjentID = 8
ROLLBACK
```

	PartialID	PacjentID	DataSzczepienia	Status	Dawka
1	12	8	2021-06-01 00:00:00.000	oczekujace	1

Zapis na 2 dawkę:

```
BEGIN TRANSACTION
SELECT * FROM ZlecenieSzczepienia WHERE PacjentID = 7
EXEC NoweZlecenie @PacjentID = 7, @RodzajSzczepionki = 'Pfizer',
@CentrumSzczepienID = 1, @Data = '2021-06-01';
SELECT * FROM ZlecenieSzczepienia WHERE PacjentID = 7
ROLLBACK
```

	PartialID	PacjentID	DataSzczepienia	Status	Dawka
1	11	7	2021-05-03 16:00:00.000	zrealizowane	1
1	11	7	2021-05-03 16:00:00.000	zrealizowane	1
2	12	7	2021-06-01 00:00:00.000	oczekujace	2

Wybór niedostępnej szczepionki:

```
BEGIN TRANSACTION
SELECT * FROM Zamowienia WHERE CentrumSzczepienID = 3
SELECT * FROM SzczegolyZamowien
EXEC NoweZlecenie @PacjentID = 1, @RodzajSzczepionki = 'Moderna',
@CentrumSzczepienID = 3, @Data = '2021-06-01';
SELECT * FROM Zamowienia WHERE CentrumSzczepienID = 3
SELECT * FROM SzczegolyZamowien
ROLLBACK
```

	ZamowienieID	CentrumSzczepienID	DataZamowienia	DataWysylki
1	3	3	2021-02-25	2021-03-07

	ZamowienieID	PartialID
12	6	12
13	7	13
14	8	14

	ZamowienieID	CentrumSzczepienID	DataZamowienia	DataWysylki
1	3	3	2021-02-25	2021-03-07
2	10	3	2021-06-03	NULL

	ZamowienieID	PartialID
13	7	13
14	8	14
15	10	15

6 Oświadczenie

Oświadczam, że wszystkie prace stanowiące podstawę do uznania osiągnięcia efektów uczenia się z przedmiotu Bazy Danych zostały wykonane przeze mnie samodzielnie.

Przemysław Olender
305761