

BADA - PROJEKT część 1.

-

Projekt bazy danych klubu piłkarskiego

Mateusz Plichta 324939, Karol Żelazowski 324953

Grudzień 2023

Spis treści

1	Zakres i cel projektu (opis założeń funkcjonalnych projektowanej bazy danych)	3
1.1	Założenia funkcjonalne	3
2	Definicja systemu	3
2.1	Perspektywy użytkowników	3
2.2	Wyróżnione funkcjonalności systemu	4
3	Model konceptualny	5
3.1	Definicja zbiorów encji określonych w projekcie (decyzje projektowe)	5
3.2	Ustalenie związków między encjami i ich typów	7
3.3	Określenie atrybutów i ich dziedzin	9
3.4	Dodatkowe reguły integralnościowe (reguły biznesowe)	9
3.5	Klucze kandydujące i główne (decyzje projektowe)	9
3.6	Schemat ER na poziomie konceptualnym	10
3.7	Problem pułapek szczelinowych i wachlarzowych – analiza i przykłady	11
4	Model logiczny	11
4.1	Charakterystyka modelu relacyjnego	11
4.2	Usunięcie właściwości niekompatybilnych z modelem relacyjnym - przykłady	11
4.3	Proces normalizacji – analiza i przykłady	13
4.4	Schemat ER na poziomie modelu logicznego	15
4.5	Więzy integralności	16
4.6	Proces denormalizacji – analiza i przykłady	16
5	Faza fizyczna	16
5.1	Projekt transakcji i weryfikacja ich wykonalności	16
5.2	Strojenie bazy danych – dobór indeksów	17
5.3	Skrypt SQL zakładający bazę danych	18
5.4	Skrypt SQL uzupełniający bazę danych	29
5.5	Przykłady zapytań i poleceń SQL odnoszących się do bazy danych	33

1 Zakres i cel projektu (opis założeń funkcjonalnych projektowanej bazy danych)

Naszym celem jest zaprojektowanie bazy danych dla klubu piłkarskiego. Na początku na poziomie konceptualnym, który reprezentuje istotę wymagań projektu oraz następnie na poziomie logicznym. Po to aby potem projekt zaimplementować fizycznie.

Przy wykonywaniu projektu korzystaliśmy z poniższych narzędzi :

- Toad Data Modeler
- Oracle Database 21c XE
- SQL Developer

1.1 Założenia funkcjonalne

Klub piłkarski ma drużyny damskie i męskie, do których zatrudnia piłkarzy oraz trenerów. Drużyny biorą udział w meczach w różnych ligach. Klub jest w posiadaniu stadionu, na którym rozgrywane są mecze z przeciwnymi drużynami, zbiera informacje odnośnie przeciwnika, wyniku i innych ważnych dla analityka danych. Dodatkowo klub śledzi statystyki piłkarzy oraz gromadzi informacje o członkach drużyny.

2 Definicja systemu

2.1 Perspektywy użytkowników

Baza danych klubu piłkarskiego jest oparta na zasadzie role-based access control. To znaczy, że działa w sposób oparty na ograniczonej kontroli dostępu. Z tego wynika podział ról, które następnie przypisaliśmy użytkownikom, korzystających z bazy danych w ramach klubu:

- *Administrator* - ma dostęp do wszystkich danych w systemie, może je dowolnie modyfikować. Ma także dostęp do wszystkich funkcjonalności systemu
- *Właściciel klubu* - ma dostęp do całej bazy danych. Nie może nadawać uprawnień do modyfikowania danych.
- *Piłkarz* - ma wgląd w swoje dane. Może również przeglądać nadchodzące i rozegrane mecze.
- *Trener* - ma wgląd w dane piłkarzy jak i drużyn. Ma dostęp do meczy.
- *Menedżer drużyny* - ma dostęp do danych piłkarzy i drużyny może je modyfikować. Ma dostęp do rozegranych i nadchodzących meczy może uzupełniać te dane.

2.2 Wyróżnione funkcjonalności systemu

Charakterystyką systemu jest to, że każda encja może być poddana modyfikacji na przykład dodawanie lub usuwanie. W dodatku możemy mieć po prostu podgląd do różnych danych tj.: nadchodzące mecze, statystyki, itd.

Nazwa funkcjonalności	A	W	P	T	M
Nadawanie uprawnień do modyfikowania baz danych	TAK	NIE	NIE	NIE	NIE
Dodawanie, usuwanie i modyfikacja danych klubu	TAK	TAK	NIE	NIE	NIE
Podgląd danych dotyczących klubu	TAK	TAK	TAK	TAK	TAK
Dodawanie, usuwanie i modyfikacja danych dotyczących członków drużyny	TAK	TAK	NIE	NIE	TAK
Podgląd danych dotyczących drużyny	TAK	TAK	NIE	TAK	TAK
Dodawanie usuwanie i modyfikacja danych dotyczących piłkarza	TAK	TAK	TAK	NIE	TAK
Pogląd danych dotyczących piłkarza	TAK	TAK	TAK	TAK	TAK
Dodawanie usuwanie i modyfikacja danych dotyczących trenera	TAK	TAK	NIE	TAK	TAK
Podgląd danych dotyczących trenera	TAK	TAK	NIE	TAK	TAK
Dodawanie usuwanie i modyfikacja danych dotyczących meczów	TAK	TAK	NIE	NIE	TAK
Podgląd danych dotyczących meczów	TAK	TAK	TAK	TAK	TAK
Dodawanie usuwanie i modyfikacja danych dotyczących stadionu	TAK	TAK	NIE	NIE	NIE
Podgląd danych dotyczących stadionu	TAK	TAK	TAK	TAK	TAK

3 Model konceptualny

3.1 Definicja zbiorów encji określonych w projekcie (decyzje projektowe)

Klub Piłkarski - encja reprezentująca klub piłkarski

Nazwa atrybutu	Typ i dziedzina	Wymagania dostępności	Opis
Nr_Klubu	Integer	Klucz główny, obowiązkowy	Unikalny identyfikator klubu piłkarskiego
Nazwa_Klubu	VarChar(50)	Obowiązkowy	Pełna nazwa klubu piłkarskiego
Adres_Klubu	VarChar(400)	Obowiązkowy	Adres przy, którym mieści się klub, pole wielosegmentowe
Numer_telefonu	VarChar(12)	Obowiązkowy	Kontaktowy numer telefonu do klubu
Adres_email	VarChar(40)	Obowiązkowy	Kontaktowy adres email do klubu

Stadion - encja reprezentująca stadion klubu piłkarskiego

Nazwa atrybutu	Typ i dziedzina	Wymagania dostępności	Opis
Nr_Stadionu	Integer	Klucz główny, obowiązkowy	Unikalny identyfikator stadionu
Adres_Stadionu	VarChar(400)	Obowiązkowy	Adres przy, którym mieści się stadion, pole wielosegmentowe
Pojemnosc_stadionu	Integer	Obowiązkowy	Liczba miejsc na stadionie
Nazwa_Stadionu	VarChar(50)	Obowiązkowy	Nazwa stadionu

Mecz - encja reprezentująca mecze, które rozgrywa klub piłkarski

Nazwa Atrybutu	Typ i dziedzina	Wymagania dostępności	Opis
Nr_meczu	Integer	Klucz główny, obowiązkowy	Unikalny identyfikator meczu
Data_meczu	Date	Obowiązkowy	Data kiedy jest rozgrywany mecz
Liga_rozgrywek	VarChar(30)	Obowiązkowy	Liga w ramach, której rozgrywany jest mecz
Druzyna_przeciwna	VarChar(30)	Obowiązkowy	Drużyna przeciwna, z którą rozgrywany jest mecz
Wynik_meczu	VarChar(10)	Nieobowiązkowy	Wynik, którym zakończył się mecz
Sedzia	VarChar(30)	Nieobowiązkowy	Sędzia, który sędziował mecz
Liczba_widzow	Integer	Nieobowiązkowy	Frekwencja na meczu

Drużyna - encja reprezentująca drużynę wchodzącą w skład klubu piłkarskiego

Nazwa atrybutu	Typ i dziedzina	Wymagania dostępności	Opis
Nr_drużyny	Integer	Klucz główny, obowiązkowy	Unikalny identyfikator drużyny
Nazwa_drużyny	VarChar(30)	Obowiązkowy	Nazwa danej drużyny
Max_liczba_zawodnikow	SmallInt	Obowiązkowy	Liczba zawodników wchodzących w skład drużyny
Liga	VarChar(30)	Obowiązkowy	Nazwa ligi, w której występuje dana drużyna
Trener_drużyny	VarChar(30)	Obowiązkowy	Trener, który prowadzi daną drużynę
Plec	Plec(domain)	Obowiązkowy	Płeć drużyny jest wartością z zakresu (męska, żeńska)
Wiek_drużyny	wiek_drużyny(domain)	Obowiązkowy	wiek drużyny jest wartością z zakresu (U17, U21, 'S'(Seniorzy))

Pracownik - encja reprezentująca pracownika

Nazwa Atrybutu	Typ i dziedzina	Wymagania dostępności	Opis
Nr_pracownika	Integer	Klucz główny, obowiązkowy	Unikalny identyfikator pracownika
Imie	VarChar(20)	Obowiązkowy	Imię pracownika
Nazwisko	VarChar(30)	Obowiązkowy	Nazwisko pracownika
Pesel	Character(11)	Nieobowiązkowy	Numer PESEL pracownika
Adres	VarChar(400)	Obowiązkowy	Adres Pracownika, pole wielosegmentowe
Data_urodzenia	Date	Obowiązkowy	Data urodzenia pracownika
Data_zatrudnienia	Date	Obowiązkowy	Data zatrudnienia pracownika
Stanowisko	stanowisko (domain)	Obowiązkowy	Stanowisko na jakim pracuje pracownik
Nr_telefonu	VarChar(12)	Obowiązkowy	Numer telefonu pracownika
Nr_konta	VarChar(30)	Obowiązkowy	Numer konta bankowego pracownika
Adres_email	VarChar(40)	Obowiązkowy	Adres email pracownika
Wynagrodzenie	Money	Obowiązkowy	Wynagrodzenie pracownika
Plec	Plec(domain)	Obowiązkowy	Płeć pracownika

Piłkarz - encja reprezentująca piłkarza

Nazwa Atrybutu	Typ i dziedzina	Wymagania dostępności	Opis
Data_wygasniecia_kontraktu	Date	Nieobowiązkowy	Data wygaśnięcia kontraktu z piłkarzem
Nr_koszulki	SmallInt	Obowiązkowy	Numer na koszulce piłkarza
Liczba_wystepow	Integer	Obowiązkowy	Liczba występów piłkarza dla klubu
Liczba_goli	Integer	Obowiązkowy	Liczba goli strzelonych przez piłkarza dla klubu
Liczba_asyst	Integer	Obowiązkowy	Liczba asyst zdobytych przez piłkarza
Liczba_czystych_kont	Integer	Nieobowiązkowy	Liczba zachowanych czystych kont przez piłkarza
Pozycja	pozycja(domain)	Nieobowiązkowy	Pozycja, na której gra piłkarz

Trener - encja reprezentująca trenera

Nazwa Atrybutu	Typ i dziedzina	Wymagania dostępności	Opis
Licencja	VarChar(30)	Obowiązkowy	Licencja trenerska
Specjalizacja	VarChar(30)	Obowiązkowy	Specjalizacja trenera

3.2 Ustalenie związków między encjami i ich typów

Drużyna - Klub_Piłkarski Drużyna gra dla Klubu Piłkarskiego.

Związek 0..m - 1..1

Klub piłkarski początkowo nie posiada żadnych drużyn jednak może ją założyć, każda drużyna należy do klubu piłkarskiego.

Klub_Piłkarski - Pracownik Klub piłkarski zatrudnia pracownika.

Związek 1..1 - 0..m

Klub piłkarski początkowo może nie posiadać żadnego pracownika, ale może ich również mieć więcej. Każdy pracownik należy do klubu piłkarskiego.

Stadion - Klub_Piłkarski Stadion należy do Klubu Piłkarskiego

Związek 0..m - 1..1

Klub piłkarski początkowo może nie być w posiadaniu stadionu, ale może takowy wynajmować bądź wybudować. Stadion, który jest w bazie danych z założenie jest wykorzystywany przez dany klub, nie bierze pod uwagę innych klubów.

Mecz - Stadion Mecz jest rozgrywany na stadionie

Związek 0..m - 1..1

Jeżeli mecz jest rozgrywany to musi być on na stadionie, jednak jest taka możliwość, że na stadionie nie są rozgrywane żadne mecze z powodu np. remontu.

Drużyna - Mecz Drużyna rozgrywa mecz.

Związek 1..m - 0..n

Drużyna może być wykluczona z rozgrywek bądź nie należeć do żadnej ligi wtedy nie rozgrywa meczu, jednak mecze, które są w bazie danych muszą być rozgrywane przez co najmniej jedną z drużyn.

Klub_Pilkarski - Mecz Klub piłkarski bierze udział w meczu

Związek 1..1 - 0..m

Klub w momencie założenia może nie brać udziału w żadnych meczach, ale w każdym meczu w bazie danych bierze udział klub piłkarski, gdyż baza danych dotyczy naszego klubu

Pracownik - Druzyna Pracownik wchodzi w skład drużyny

Związek 0..m - 0..n

Drużyna może zostać założona i nie posiadać jeszcze żadnych zawodników. Równie dobrze piłkarz może być zawieszony bądź kontuzjowany i odsunięty od drużyny. Dodatkowo pracownik może brać udział w obu drużynach, np. fizjoterapeuta może udzielać się w obu drużynach.

Pracownik - Mecz Pracownik bierze udział w meczu

Związek 0..n - 0..m

W meczu bierze udział więcej niż jeden pracownik. Poza tym każdy pracownik może rozgrywać wiele meczów, a w najgorszym przypadku nie rozgrywa żadnego meczu.

Relacja		Typ Związku	Typ uczestnictwa		Stopień
Druzyna	Klub_Pilkarski	n:1	Opcjonalny	Obowiązkowy	Binarny
Klub_Pilkarski	Pracownik	1:n	Obowiązkowy	Opcjonalny	Binarny
Stadion	Klub_Pilkarski	n:1	Opcjonalny	Obowiązkowy	Binarny
Mecz	Stadion	n:m	Opcjonalny	Obowiązkowy	Binarny
Druzyna	Mecz	n:m	Obowiązkowy	Opcjonalny	Binarny
Pracownik	druzyna	n:m	Opcjonalny	Opcjonalny	Binarny
Klub_Pilkarski	Mecz	1:m	Obowiązkowy	Opcjonalny	Binarny
Pracownik	Mecz	n:m	Opcjonalny	Opcjonalny	Binarny

Dodatkowo w naszym modelu konceptualnym posiadamy specjalizacje dla pracownika. Pracownik ma specjalizacje Piłkarz oraz specjalizacje Trener.

3.3 Określenie atrybutów i ich dziedzin

Uznaliśmy, że dla części atrybutów warto stworzyć własne dziedziny :

- PlecD - dziedzina określająca płeć członka klubu . Dopuszczalne opcje to K (kobieta) bądź M (Mężczyzna).
- StanowiskoD - dziedzina określająca stanowisko. Dopuszczalne opcje to (T) trener bądź (P) piłkarz, (I) inne
- PozycjaD - dziedzina określająca pozycje piłkarza, tutaj wybór jest znacznie szerszy :
BR (bramkarz)
SO (środkowy obrońca)
LO (lewy obrońca)
PO (prawy obrońca)
CLS (cofnięty lewoskrzydłowy)
CPS (cofnięty prawoskrzydłowy)
SP (środkowy pomocnik)
LP (lewy pomocnik)
PP (prawy pomocnik)
SPD (środkowy pomocnik defensywny)
SPO (środkowy pomocnik ofensywny)
LS (lewoskrzydłowy)
PS (prawoskrzydłowy)
SN (środkowy napastnik)
N (napastnik)
- Wiek_drużynyD - podzieliśmy również drużyny na różne grupy wiekowe, wstępnie umieściliśmy grupy wiekowe 'U17'(do 17 roku życia), 'U21'(do 21 roku życia) i 'S'(Seniorzy)
- PlecD - drużyna może być damska (K), bądź męska (M).

3.4 Dodatkowe reguły integralnościowe (reguły biznesowe)

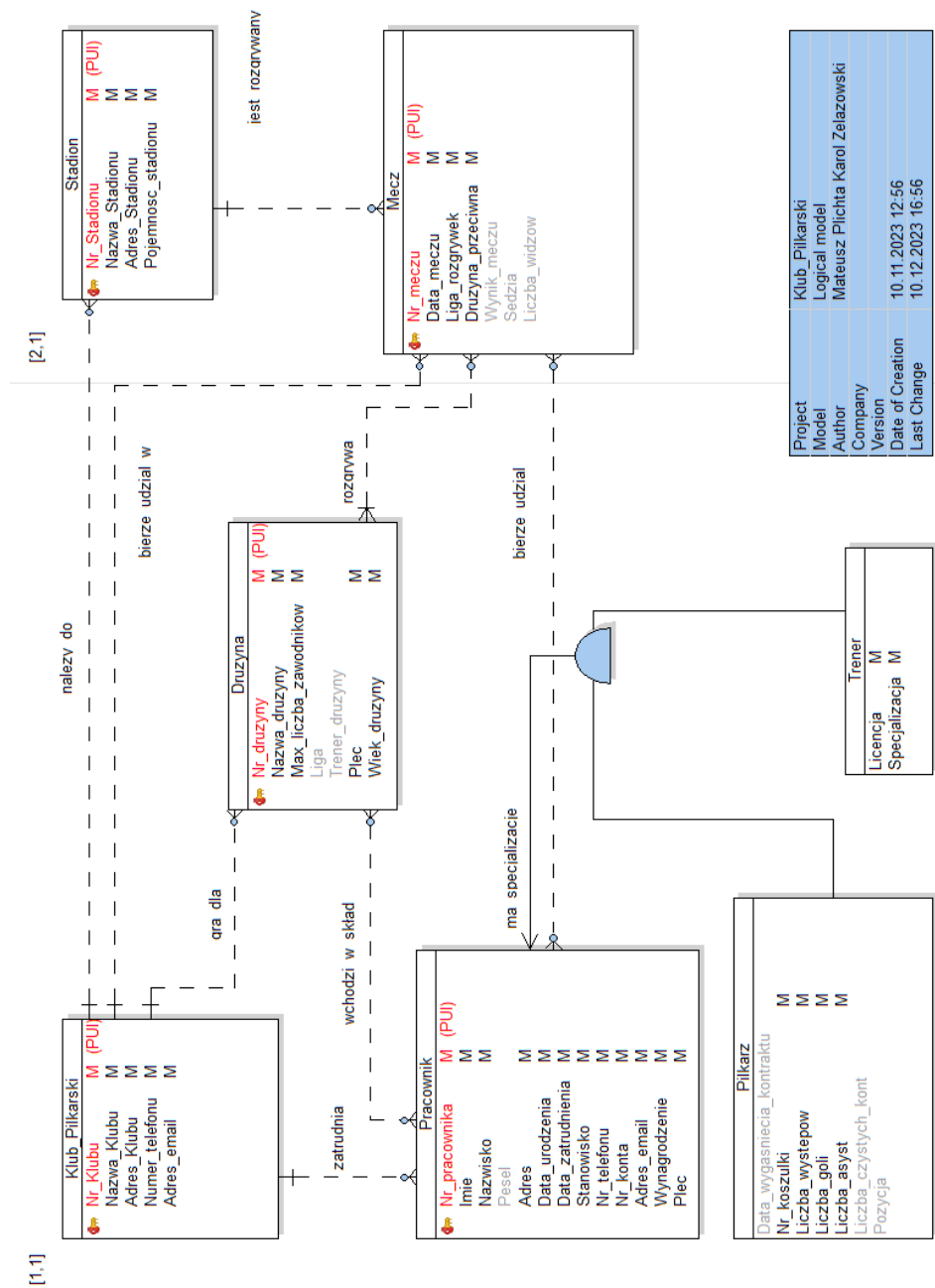
Integralność danych odnosi się do stanu, w którym dane w systemie są zgodne z oczekiwaniami co do ich dokładności, spójności, rzetelności i pełności. W kontekście baz danych integralność danych jest istotnym aspektem, który gwarantuje, że przechowywane informacje są poprawne i zgodne z określonymi regułami biznesowymi oraz zasadami strukturalnymi bazy danych. W kontekście naszej bazy danych pamiętaliśmy o regułach dziedziny, unikalności, autoryzacji itp.

3.5 Klucze kandydujące i główne (decyzje projektowe)

W naszej bazie danych zdecydowaliśmy się na wybór kluczy sztucznych, czyli generowanie kolejnych numerów - identyfikatorów. Stwierdziliśmy, że będzie to najlepsze, ponieważ jest najefektywniejsze i pod względem uniwersalności identyfikowania rekordów możliwie jak najkrótszym kluczem. Przedstawiliśmy także klucze kandydujące.

ENCJA	KLUCZ GŁÓWNY	KLUCZ KANDYDUJĄCY
Klub Piłkarski	Nr_Klubu	Nazwa_Klubu
Stadion	Nr_Stadionu	Nazwa_stadionu
Drużyna	Nr_drużyny	Nazwa_drużyny
Mecz	Nr_meczu	Data_meczu, Liga_rozgrywek
Pracownik	Nr_pracownika	pesel

3.6 Schemat ER na poziomie konceptualnym



Rysunek 1: Schemat ER na poziomie konceptualnym

3.7 Problem pułapek szczelinowych i wachlarzowych – analiza i przykłady

Przed poprawą projektu na konsultacjach w trakcie analizy zauważyliśmy pułapkę szczelinową w naszym projekcie. Z założeń projektowych wynika, że członek drużyny (teraz encja Pracownik) nie musi być powiązany z żadną z drużyn, gdyż może być wykluczony bądź kontuzjowany, w takim jednak przypadku nie jestem w stanie dodać zawodnika do bazy danych i mieć powiązanie między nim a klubem nie wiążąc go z drużyną. Zatem rozwiązaniem tego problemu będzie stworzenie związku między członkiem drużyny (teraz Pracownik) a klubem piłkarskim.

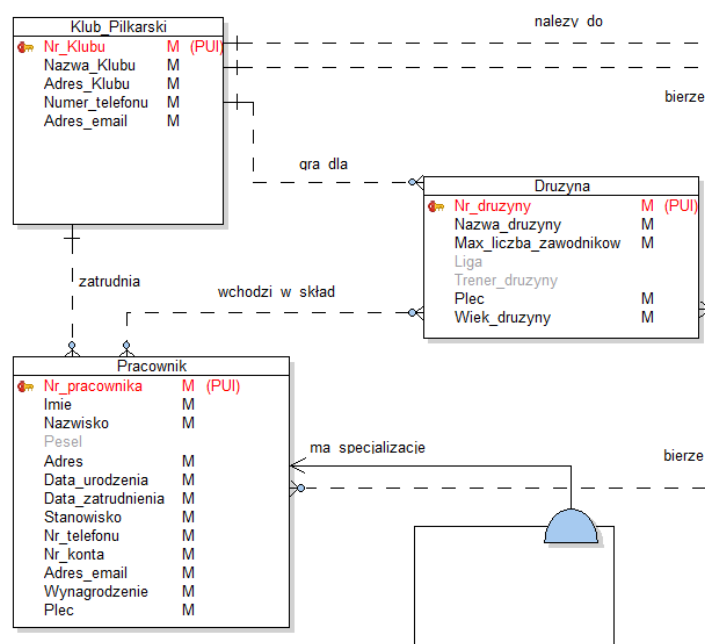
Związek który utworzyliśmy. Klub Piłkarski - Członek drużyny

Klub piłkarski zatrudnia członka drużyny

Związek 1..1 - 0..m

Klub piłkarski w najgorszym przypadku nie ma żadnych członków (Pracowników). Każdy członek zespołu przynależy do klubu.

Związek jest opcjonalny dla klubu i obowiązkowy dla piłkarza. Stopień związku jest binarny.



Rysunek 2: Po poprawie

4 Model logiczny

4.1 Charakterystyka modelu relacyjnego

Toad Data Modeler, w którym pracujemy umożliwia nam konwersję modelu konceptualnego na model logiczny. Natomiast bezpośrednio po konwersji nie jest on w pełni zgodny z modelem relacyjnym, dlatego niektóre elementy musieliśmy poprawić.

4.2 Usunięcie właściwości niekompatybilnych z modelem relacyjnym - przykłady

W ramach dostosowywania naszego modelu logicznego do kompatybilności z modelem relacyjnym, zmieniliśmy nazwę wszystkich relacji z liczby pojedynczej na liczbę mnogą: *Pilkarz* → *Pilkarze* itd. Następnie klucze główne tabel powstały na podstawie atrybutów identyfikujących encje. Z tego wynika, że wszystkie inne atrybuty tabel stały się niegłównymi atrybutami tabel. W ramach dalej opisanych procesach normalizacji powstały takie tabele jak: *Adresy*, *Pozycje*, *Typy_Licencji*, *Specjalizacje*, *Wynagrodzenia*

Dalej w związkach jeden do wielu po stronie krótszych łapek wstawiliśmy klucz obcy, a po drugiej stronie klucz główny. Następnie dla wszystkich związków wielu do wielu stworzyliśmy tablice pośrednie. W ramach

tego procesu powstały tabele:

Pierwsza relacja	Druga relacja	Nazwa tabeli pośredniej
Drużyny	Pracownicy	Drużyny_Pracownicy
Drużyny	Mecze	Drużyny_Mecze
Mecze	Pracownicy	Mecze_Pracownicy
Trenerzy	Specjalizacje	Trenerzy_Specjalizacje

Drużyny_Pracownicy

Nazwa atrybutu	Typ	Obowiązkowość
Nr_drużyny	Integer	Obowiązkowy
Nr_pracownika	Integer	Obowiązkowy

Drużyny_Mecze

Nazwa atrybutu	Typ	Obowiązkowość
Nr_drużyny	Integer	Obowiązkowy
Nr_meczu	Integer	Obowiązkowy

Mecze_Pracownicy

Nazwa atrybutu	Typ	Obowiązkowość
Nr_meczu	Integer	Obowiązkowy
Nr_pracownika	Integer	Obowiązkowy

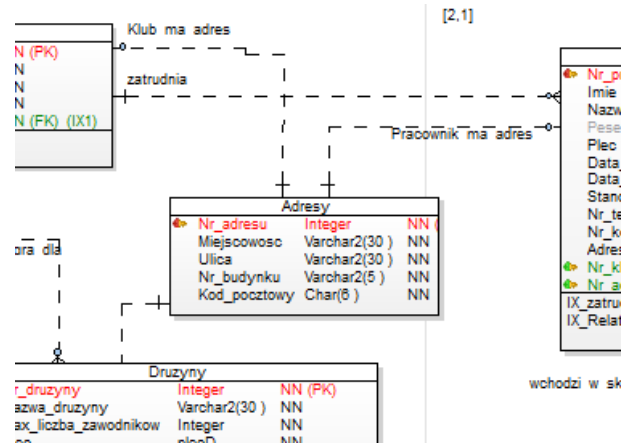
Trenerzy_Specjalizacje

Nazwa atrybutu	Typ	Obowiązkowość
Nr_specjalizacji	Integer	Obowiązkowy
Nr_pracownika	Integer	Obowiązkowy

4.3 Proces normalizacji – analiza i przykłady

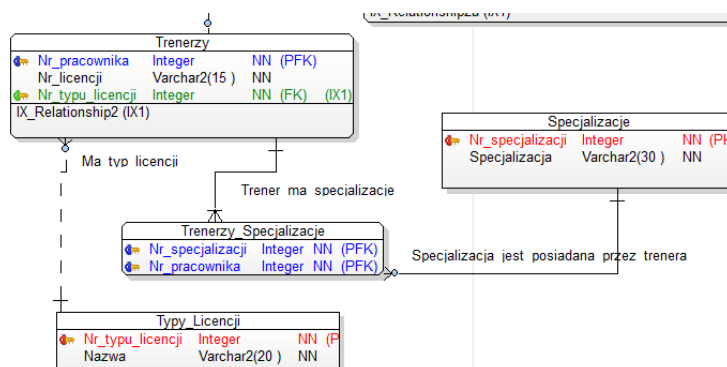
Pierwsza Postać Normalna

W celach otrzymania pierwszej postaci normalnej musieliśmy ustanowić wszystkie pola atomowymi, poza tym pozbyć się powtarzających się komórek w kolumnach. Dlatego też ustanowiliśmy relację *Adresy*, gdyż pole to nie było atomowe.



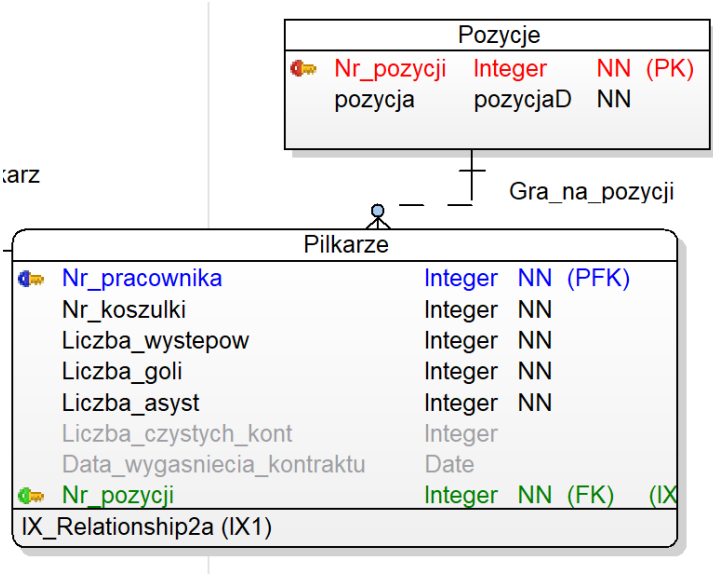
Rysunek 3: Relacje Adresy

W relacji *Trenerzy* zauważyliśmy, że trener może mieć kilka specjalizacji, dlatego została stworzona relacja *Specjalizacje*, która została połączona z relacją *Trenerzy* tablicą bridgującą, ponieważ jest to relacja wielu do wielu. Powstała także relacja *Typy_Licencji*, ponieważ kilku trenerów może mieć tę samą licencję.



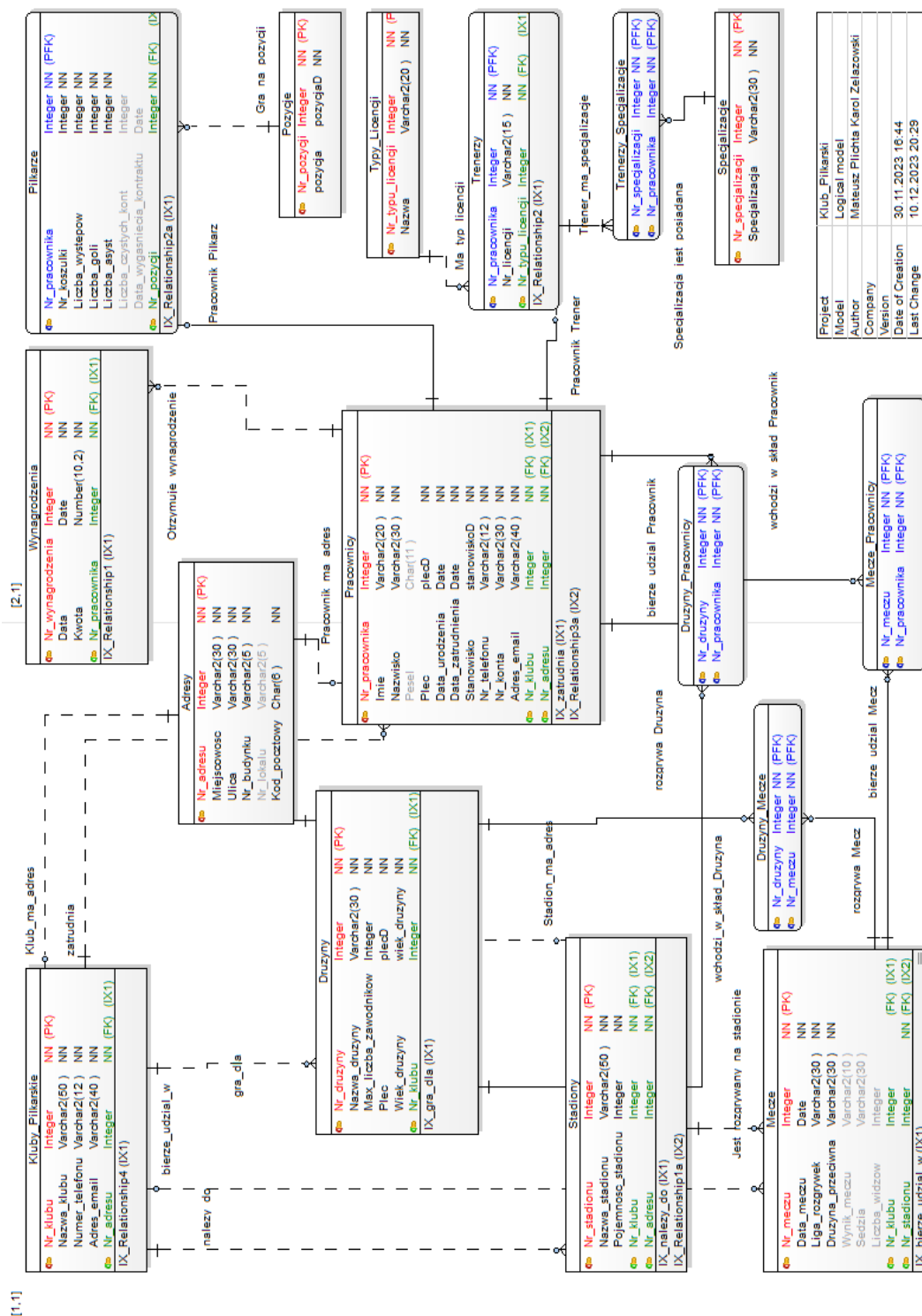
Rysunek 4: Relacje Typy_Licencji i Specjalizacje

Kolejnym przykładem powtarzających się wartości są pozycje zawodników. Założyliśmy że zawodnik gra na jednej pozycji. Stąd nowa relacja Pozycja. Warto zwrócić na związek, każdy piłkarz musi mieć swoją pozycję, jednak z pewnością może być pozycja na której nie ma tymczasowo zawodnika stąd opcjonalność od strony Piłkarza.



Rysunek 5: relacja pozycje

4.4 Schemat ER na poziomie modelu logicznego



Rysunek 6: Model Logiczny

4.5 Więzy integralności

Wszystkie stworzone przez nas pola w tablicach bazy są wartościami atomowymi oraz klucze główne. zostały przez nas oznaczone jako *PrimaryKey*. Zadbaliśmy również o to aby żadne nazwy atrybutów i relacji nie powielały się. Nie powielają się także żadne nazwy związków pomiędzy relacjami.

4.6 Proces denormalizacji – analiza i przykłady

Proces denormalizacji polega na łączeniu wybranych znormalizowanych tabel w celu przyspieszenia dostępu do niektórych danych. W naszym modelu zdecydowaliśmy na użycie jednego atrybutu, który będzie określał czy dana osoba jest piłkarzem czy też trenerem na poziomie relacji Pracownicy, dzięki temu jeżeli potrzebujemy danych o wszystkich piłkarzach, bez ich statystyk to wystarczy nam proste zapytanie o pracowników, których wartość atrybutów Stanowisko to "P" -piłkarz lub "T" - trener. Uznaliśmy że może być to często wykorzystywane zapytanie dlatego też warto usprawnić ten proces.

5 Faza fizyczna

5.1 Projekt transakcji i weryfikacja ich wykonalności

Transakcja	Wykorzystywane tabele	Czy do wykonania?
Dodawanie, modyfikowanie i usuwanie drużyn	Kluby_Pilkarskie, Drużyny	TAK
Dodawanie, modyfikowanie i usuwanie stadionów	Adresy, Kluby_Pilkarskie, Stadiony	TAK
Dodawanie, modyfikowanie i usuwanie meczów	Kluby_pilkarskie, Mecze, Stadiony	TAK
Dodawanie, modyfikowanie i usuwanie trenerów	Pracownicy, Kluby_Pilkarskie, Adresy, Trenerzy, Typy_Licencji	TAK
Dodawanie, modyfikowanie i usuwanie piłkarzy	Pracownicy, Kluby_Pilkarskie, Adresy, Pilkarze, Pozycje	TAK
Dodawanie, modyfikowanie i usuwanie adresów	Adresy	TAK

5.2 Strojanie bazy danych – dobór indeksów

dobór indeksów

```
-- Create indexes for table Kluby_Pilkarskie

CREATE INDEX IX_Relationship4 ON Kluby_Pilkarskie (Nr_adresu)
/

-- Create indexes for table Stadiony

CREATE INDEX IX_nalezy_do ON Stadiony (Nr_Klubu)
/

CREATE INDEX IX_Relationship1a ON Stadiony (Nr_adresu)
/

-- Create indexes for table Pracownicy

CREATE INDEX IX_zatrudnia ON Pracownicy (Nr_Klubu)
/

CREATE INDEX IX_Relationship3a ON Pracownicy (Nr_adresu)
/

-- Create indexes for table Druzyny

CREATE INDEX IX_gra_dla ON Druzyny (Nr_Klubu)
/

-- Create indexes for table Pilkarze

CREATE INDEX IX_Relationship2a ON Pilkarze (Nr_pozycji)
/

-- Create indexes for table Trenerzy

CREATE INDEX IX_Relationship2 ON Trenerzy (Nr_typu_licencji)
/

-- Create indexes for table Mecze

CREATE INDEX IX_bierze_udzial_w ON Mecze (Nr_Klubu)
/

CREATE INDEX IX_Relationship3 ON Mecze (Nr_stadionu)
/

-- Create indexes for table Wynagrodzenia

CREATE INDEX IX_Relationship1 ON Wynagrodzenia (Nr_pracownika)
/
```

5.3 Skrypt SQL zakładający bazę danych

```
/*
Created: 30.11.2023
Modified: 06.12.2023
Project: Klub_Pilkarski
Model: Logical model
Author: Mateusz Plichta Karol Zelazowski
Database: Oracle 18c
*/

-- Create sequences section -----

CREATE SEQUENCE Kluby_Pilkarskie_S
  INCREMENT BY 1
  START WITH 1
  NOMAXVALUE
  NOMINVALUE
  CACHE 20
/

CREATE SEQUENCE Adresy_S
  INCREMENT BY 1
  START WITH 1
  NOMAXVALUE
  NOMINVALUE
  CACHE 20
/

CREATE SEQUENCE Pracownicy_S
  INCREMENT BY 1
  START WITH 1
  NOMAXVALUE
  NOMINVALUE
  CACHE 20
/

CREATE SEQUENCE Wynagrodzenia_S
  INCREMENT BY 1
  START WITH 1
  NOMAXVALUE
  NOMINVALUE
  CACHE 20
/

CREATE SEQUENCE Pozycje_S
  INCREMENT BY 1
  START WITH 1
  NOMAXVALUE
  NOMINVALUE
  CACHE 20
/

CREATE SEQUENCE Stadiony_S
  INCREMENT BY 1
```

```

START WITH 1
NOMAXVALUE
NOMINVALUE
CACHE 20
/

CREATE SEQUENCE Mecze_S
INCREMENT BY 1
START WITH 1
NOMAXVALUE
NOMINVALUE
CACHE 20
/

CREATE SEQUENCE Druzyzny_S
INCREMENT BY 1
START WITH 1
NOMAXVALUE
NOMINVALUE
CACHE 20
/

CREATE SEQUENCE Typy_Licencji_S
INCREMENT BY 1
START WITH 1
NOMAXVALUE
NOMINVALUE
CACHE 20
/

CREATE SEQUENCE Specjalizacje_S
INCREMENT BY 1
START WITH 1
NOMAXVALUE
NOMINVALUE
CACHE 20
/

-- Create tables section -----

-- Table Kluby_Pilkarskie

CREATE TABLE Kluby_Pilkarskie(
  Nr_klubu Integer NOT NULL,
  Nazwa_klubu Varchar2(50 ) NOT NULL,
  Numer_telefonu Varchar2(12 ) NOT NULL,
  Adres_email Varchar2(40 ) NOT NULL,
  Nr_adresu Integer NOT NULL
)
/

-- Create indexes for table Kluby_Pilkarskie

CREATE INDEX IX_Relationship4 ON Kluby_Pilkarskie (Nr_adresu)
/

```

```

-- Add keys for table Kluby_Pilkarskie

ALTER TABLE Kluby_Pilkarskie ADD CONSTRAINT Unique_Identifier1 PRIMARY KEY (Nr_klubu)
/

-- Table Stadiony

CREATE TABLE Stadiony(
    Nr_stadionu Integer NOT NULL,
    Nazwa_stadionu Varchar2(50 ) NOT NULL,
    Pojemnosc_stadionu Integer NOT NULL,
    Nr_Klubu Integer NOT NULL,
    Nr_adresu Integer NOT NULL
)
/

-- Create indexes for table Stadiony

CREATE INDEX IX_nalezy_do ON Stadiony (Nr_Klubu)
/

CREATE INDEX IX_Relationship1a ON Stadiony (Nr_adresu)
/

-- Add keys for table Stadiony

ALTER TABLE Stadiony ADD CONSTRAINT Unique_Identifier2 PRIMARY KEY (Nr_stadionu)
/

-- Table Pracownicy

CREATE TABLE Pracownicy(
    Nr_pracownika Integer NOT NULL,
    Imie Varchar2(20 ) NOT NULL,
    Nazwisko Varchar2(30 ) NOT NULL,
    Pesel Char(11 ),
    Plec Varchar2(1 ) NOT NULL
        CHECK (plec IN ('K', 'M')),
    Data_urodzenia Date NOT NULL,
    Data_zatrudnienia Date NOT NULL,
    Stanowisko Varchar2(1 ) NOT NULL
        CHECK (stanowisko IN ('T', 'P', 'I')),
    Nr_telefonu Varchar2(12 ) NOT NULL,
    Nr_konta Varchar2(30 ) NOT NULL,
    Adres_email Varchar2(40 ) NOT NULL,
    Nr_Klubu Integer NOT NULL,
    Nr_adresu Integer NOT NULL
)
/

-- Create indexes for table Pracownicy

CREATE INDEX IX_zatrudnia ON Pracownicy (Nr_Klubu)
/

CREATE INDEX IX_Relationship3a ON Pracownicy (Nr_adresu)

```

```

/

-- Add keys for table Pracownicy

ALTER TABLE Pracownicy ADD CONSTRAINT Unique_Identifier3 PRIMARY KEY (Nr_pracownika)
/

-- Table Druzyzny

CREATE TABLE Druzyzny(
    Nr_druzyzny Integer NOT NULL,
    Nazwa_druzyzny Varchar2(30 ) NOT NULL,
    Max_liczba_zawodnikow Integer NOT NULL,
    Liga Varchar2(30 ),
    Plec Varchar2(1 ) NOT NULL
        CHECK (plec IN ('K', 'M')),
    Wiek_druzyzny Varchar2(3 ) NOT NULL
        CHECK (wiek_druzyzny IN ('S', 'U17', 'U21')),
    Nr_Klubu Integer NOT NULL
)
/

-- Create indexes for table Druzyzny

CREATE INDEX IX_gra_dla ON Druzyzny (Nr_Klubu)
/

-- Add keys for table Druzyzny

ALTER TABLE Druzyzny ADD CONSTRAINT Unique_Identifier4 PRIMARY KEY (Nr_druzyzny)
/

-- Table Pilkarze

CREATE TABLE Pilkarze(
    Nr_pracownika Integer NOT NULL,
    Nr_koszulki Integer NOT NULL,
    Liczba_wystepow Integer NOT NULL,
    Liczba_goli Integer NOT NULL,
    Liczba_asyst Integer NOT NULL,
    Liczba_czystych_kont Integer,
    Data_wygasniecia_kontraktu Date,
    Nr_pozycji Integer NOT NULL
)
/

-- Create indexes for table Pilkarze

CREATE INDEX IX_Relationship2a ON Pilkarze (Nr_pozycji)
/

-- Add keys for table Pilkarze

ALTER TABLE Pilkarze ADD CONSTRAINT Unique_Identifier5 PRIMARY KEY (Nr_pracownika)
/

```

```

-- Table Trenerzy

CREATE TABLE Trenerzy(
    Nr_pracownika Integer NOT NULL,
    Nr_licencji Varchar2(15 ) NOT NULL,
    Nr_typu_licencji Integer NOT NULL
)
/

-- Create indexes for table Trenerzy

CREATE INDEX IX_Relationship2 ON Trenerzy (Nr_typu_licencji)
/

-- Add keys for table Trenerzy

ALTER TABLE Trenerzy ADD CONSTRAINT Unique_Identifier6 PRIMARY KEY (Nr_pracownika)
/

-- Table Mecze

CREATE TABLE Mecze(
    Nr_meczu Integer NOT NULL,
    Data_meczu Date NOT NULL,
    Liga_rozgrywek Varchar2(30 ) NOT NULL,
    Druzyna_przeciwna Varchar2(30 ) NOT NULL,
    Wynik_meczu Varchar2(10 ),
    Sedzia Varchar2(30 ),
    Liczba_widzow Integer,
    Nr_Klubu Integer,
    Nr_stadionu Integer NOT NULL
)
/

-- Create indexes for table Mecze

CREATE INDEX IX_bierze_udzial_w ON Mecze (Nr_Klubu)
/

CREATE INDEX IX_Relationship3 ON Mecze (Nr_stadionu)
/

-- Add keys for table Mecze

ALTER TABLE Mecze ADD CONSTRAINT Unique_Identifier7 PRIMARY KEY (Nr_meczu)
/

-- Table Druzyzny_Mecze

CREATE TABLE Druzyzny_Mecze(
    Nr_druzyzny Integer NOT NULL,
    Nr_meczu Integer NOT NULL
)
/

-- Table Druzyzny_Pracownicy

```

```

CREATE TABLE Druzynty_Pracownicy(
    Nr_druzynty Integer NOT NULL,
    Nr_pracownika Integer NOT NULL
)
/

-- Table Mecze_Pracownicy

CREATE TABLE Mecze_Pracownicy(
    Nr_meczu Integer NOT NULL,
    Nr_pracownika Integer NOT NULL
)
/

-- Table Adresy

CREATE TABLE Adresy(
    Nr_adresu Integer NOT NULL,
    Miejscowosc Varchar2(30 ) NOT NULL,
    Ulica Varchar2(30 ) NOT NULL,
    Nr_budynku Varchar2(5 ) NOT NULL,
    Kod_pocztowy Char(6 ) NOT NULL
)
/

-- Add keys for table Adresy

ALTER TABLE Adresy ADD CONSTRAINT PK_Adresy PRIMARY KEY (Nr_adresu)
/

-- Table Specjalizacje

CREATE TABLE Specjalizacje(
    Nr_specjalizacji Integer NOT NULL,
    Specjalizacja Varchar2(30 ) NOT NULL
)
/

-- Add keys for table Specjalizacje

ALTER TABLE Specjalizacje ADD CONSTRAINT PK_Specjalizacje PRIMARY KEY (Nr_specjalizacji)
/

ALTER TABLE Specjalizacje ADD CONSTRAINT Specjalizacja UNIQUE (Specjalizacja)
/

-- Table Trenerzy_Specjalizacje

CREATE TABLE Trenerzy_Specjalizacje(
    Nr_specjalizacji Integer NOT NULL,
    Nr_pracownika Integer NOT NULL
)
/

-- Add keys for table Trenerzy_Specjalizacje

```

```

ALTER TABLE Trenerzy_Specjalizacje ADD CONSTRAINT PK_Trenerzy_Specjalizacje PRIMARY KEY
↳ (Nr_specjalizacji,Nr_pracownika)
/

-- Table Pozycje

CREATE TABLE Pozycje(
  Nr_pozycji Integer NOT NULL,
  pozycja Varchar2(3 ) NOT NULL
    CHECK (pozycja IN ('BR', 'SO', 'LO', 'PO', 'CLS', 'CPS', 'SP', 'LP', 'PP', 'SPD',
↳ 'SPO', 'LS', 'PS', 'SN', 'N'))
)
/

-- Add keys for table Pozycje

ALTER TABLE Pozycje ADD CONSTRAINT PK_Pozycje PRIMARY KEY (Nr_pozycji)
/

-- Table Typy_Licencji

CREATE TABLE Typy_Licencji(
  Nr_typu_licencji Integer NOT NULL,
  Nazwa Varchar2(20 ) NOT NULL
)
/

-- Add keys for table Typy_Licencji

ALTER TABLE Typy_Licencji ADD CONSTRAINT PK_Typy_Licencji PRIMARY KEY (Nr_typu_licencji)
/

ALTER TABLE Typy_Licencji ADD CONSTRAINT Nazwa UNIQUE (Nazwa)
/

-- Table Wynagrodzenia

CREATE TABLE Wynagrodzenia(
  Nr_wynagrodzenia Integer NOT NULL,
  Data Date NOT NULL,
  Kwota Number(10,2) NOT NULL,
  Nr_pracownika Integer NOT NULL
)
/

-- Create indexes for table Wynagrodzenia

CREATE INDEX IX_Relationship1 ON Wynagrodzenia (Nr_pracownika)
/

-- Add keys for table Wynagrodzenia

ALTER TABLE Wynagrodzenia ADD CONSTRAINT PK_Wynagrodzenia PRIMARY KEY (Nr_wynagrodzenia)

```



```

/

-- Trigger for sequence Kluby_Pilkarskie_S for column Nr_klubu in table Kluby_Pilkarskie
-- -----
CREATE OR REPLACE TRIGGER ts_Kluby_Pilkarskie_Kluby_Pilkarskie_S BEFORE INSERT
ON Kluby_Pilkarskie FOR EACH ROW
BEGIN
    :new.Nr_klubu := Kluby_Pilkarskie_S.nextval;
END;
/

CREATE OR REPLACE TRIGGER tsu_Kluby_Pilkarskie_Kluby_Pilkarskie_S AFTER UPDATE OF Nr_klubu
ON Kluby_Pilkarskie FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010, 'Cannot update column Nr_klubu in table Kluby_Pilkarskie as
    ↪ it uses sequence. ');
END;
/

-- Trigger for sequence Stadiony_S for column Nr_stadionu in table Stadiony -----
CREATE OR REPLACE TRIGGER ts_Stadiony_Stadiony_S BEFORE INSERT
ON Stadiony FOR EACH ROW
BEGIN
    :new.Nr_stadionu := Stadiony_S.nextval;
END;
/

CREATE OR REPLACE TRIGGER tsu_Stadiony_Stadiony_S AFTER UPDATE OF Nr_stadionu
ON Stadiony FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010, 'Cannot update column Nr_stadionu in table Stadiony as it
    ↪ uses sequence. ');
END;
/

-- Trigger for sequence Pracownicy_S for column Nr_pracownika in table Pracownicy -----
CREATE OR REPLACE TRIGGER ts_Pracownicy_Pracownicy_S BEFORE INSERT
ON Pracownicy FOR EACH ROW
BEGIN
    :new.Nr_pracownika := Pracownicy_S.nextval;
END;
/

CREATE OR REPLACE TRIGGER tsu_Pracownicy_Pracownicy_S AFTER UPDATE OF Nr_pracownika
ON Pracownicy FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010, 'Cannot update column Nr_pracownika in table Pracownicy as
    ↪ it uses sequence. ');
END;
/

-- Trigger for sequence Druzyzny_S for column Nr_druzyzny in table Druzyzny -----
CREATE OR REPLACE TRIGGER ts_Druzyzny_Druzyzny_S BEFORE INSERT
ON Druzyzny FOR EACH ROW
BEGIN
    :new.Nr_druzyzny := Druzyzny_S.nextval;
END;
/

CREATE OR REPLACE TRIGGER tsu_Druzyzny_Druzyzny_S AFTER UPDATE OF Nr_druzyzny

```

```

ON Druzyzny FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010,'Cannot update column Nr_druzyzny in table Druzyzny as it
    ↪ uses sequence.');
```

END;

/

-- Trigger for sequence Mecze_S for column Nr_meczu in table Mecze -----

```

CREATE OR REPLACE TRIGGER ts_Mecze_Mecze_S BEFORE INSERT
ON Mecze FOR EACH ROW
BEGIN
    :new.Nr_meczu := Mecze_S.nextval;
END;
```

/

```

CREATE OR REPLACE TRIGGER tsu_Mecze_Mecze_S AFTER UPDATE OF Nr_meczu
ON Mecze FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010,'Cannot update column Nr_meczu in table Mecze as it uses
    ↪ sequence.');
```

END;

/

-- Trigger for sequence Adresy_S for column Nr_adresu in table Adresy -----

```

CREATE OR REPLACE TRIGGER ts_Adresy_Adresy_S BEFORE INSERT
ON Adresy FOR EACH ROW
BEGIN
    :new.Nr_adresu := Adresy_S.nextval;
END;
```

/

```

CREATE OR REPLACE TRIGGER tsu_Adresy_Adresy_S AFTER UPDATE OF Nr_adresu
ON Adresy FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010,'Cannot update column Nr_adresu in table Adresy as it uses
    ↪ sequence.');
```

END;

/

-- Trigger for sequence Specjalizacje_S for column Nr_specjalizacji in table Specjalizacje
 ↪ *-----*

```

CREATE OR REPLACE TRIGGER ts_Specjalizacje_Specjalizacje_S BEFORE INSERT
ON Specjalizacje FOR EACH ROW
BEGIN
    :new.Nr_specjalizacji := Specjalizacje_S.nextval;
END;
```

/

```

CREATE OR REPLACE TRIGGER tsu_Specjalizacje_Specjalizacje_S AFTER UPDATE OF Nr_specjalizacji
ON Specjalizacje FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010,'Cannot update column Nr_specjalizacji in table
    ↪ Specjalizacje as it uses sequence.');
```

END;

/

-- Trigger for sequence Pozycje_S for column Nr_pozycji in table Pozycje -----

```

CREATE OR REPLACE TRIGGER ts_Pozycje_Pozycje_S BEFORE INSERT
ON Pozycje FOR EACH ROW
```

```

BEGIN
    :new.Nr_pozycji := Pozycje_S.nextval;
END;
/
CREATE OR REPLACE TRIGGER tsu_Pozycje_Pozycje_S AFTER UPDATE OF Nr_pozycji
ON Pozycje FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010,'Cannot update column Nr_pozycji in table Pozycje as it
    ↪ uses sequence.');
```

END;

/

-- Trigger for sequence Typy_Licencji_S for column Nr_typu_licencji in table Typy_Licencji

↪ -----

```

CREATE OR REPLACE TRIGGER ts_Typy_Licencji_Typy_Licencji_S BEFORE INSERT
ON Typy_Licencji FOR EACH ROW
BEGIN
    :new.Nr_typu_licencji := Typy_Licencji_S.nextval;
END;
/
CREATE OR REPLACE TRIGGER tsu_Typy_Licencji_Typy_Licencji_S AFTER UPDATE OF Nr_typu_licencji
ON Typy_Licencji FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010,'Cannot update column Nr_typu_licencji in table
    ↪ Typy_Licencji as it uses sequence.');
```

END;

/

-- Trigger for sequence Wynagrodzenia_S for column Nr_wynagrodzenia in table Wynagrodzenia

↪ -----

```

CREATE OR REPLACE TRIGGER ts_Wynagrodzenia_Wynagrodzenia_S BEFORE INSERT
ON Wynagrodzenia FOR EACH ROW
BEGIN
    :new.Nr_wynagrodzenia := Wynagrodzenia_S.nextval;
END;
/
CREATE OR REPLACE TRIGGER tsu_Wynagrodzenia_Wynagrodzenia_S AFTER UPDATE OF Nr_wynagrodzenia
ON Wynagrodzenia FOR EACH ROW
BEGIN
    RAISE_APPLICATION_ERROR(-20010,'Cannot update column Nr_wynagrodzenia in table
    ↪ Wynagrodzenia as it uses sequence.');
```

END;

/

-- Create foreign keys (relationships) section

↪ -----

```

ALTER TABLE Stadiony ADD CONSTRAINT nalezy_do FOREIGN KEY (Nr_Klubu) REFERENCES
    ↪ Kluby_Pilkarskie (Nr_klubu)
/
```

ALTER TABLE Druzyzny ADD CONSTRAINT gra_dla FOREIGN KEY (Nr_Klubu) REFERENCES

↪ Kluby_Pilkarskie (Nr_klubu)

/

```
ALTER TABLE Mecze ADD CONSTRAINT bierze_udzial_w FOREIGN KEY (Nr_Klubu) REFERENCES  
↳ Kluby_Pilkarskie (Nr_klubu)
```

/

```
ALTER TABLE Pracownicy ADD CONSTRAINT zatrudnia FOREIGN KEY (Nr_Klubu) REFERENCES  
↳ Kluby_Pilkarskie (Nr_klubu)
```

/

```
ALTER TABLE Stadiony ADD CONSTRAINT Stadion_ma_adres FOREIGN KEY (Nr_adresu) REFERENCES  
↳ Adresy (Nr_adresu)
```

/

```
ALTER TABLE Pracownicy ADD CONSTRAINT Pracownik_ma_adres FOREIGN KEY (Nr_adresu) REFERENCES  
↳ Adresy (Nr_adresu)
```

/

```
ALTER TABLE Kluby_Pilkarskie ADD CONSTRAINT Klub_ma_adres FOREIGN KEY (Nr_adresu) REFERENCES  
↳ Adresy (Nr_adresu)
```

/

```
ALTER TABLE Trenerzy_Specjalizacje ADD CONSTRAINT Specjalizacja_jest_posiadana_przez_trenera  
↳ FOREIGN KEY (Nr_specjalizacji) REFERENCES Specjalizacje (Nr_specjalizacji)
```

/

```
ALTER TABLE Trenerzy_Specjalizacje ADD CONSTRAINT Trener_ma_specjalizacje FOREIGN KEY  
↳ (Nr_pracownika) REFERENCES Trenerzy (Nr_pracownika)
```

/

```
ALTER TABLE Pilkarze ADD CONSTRAINT Gra_na_pozycji FOREIGN KEY (Nr_pozycji) REFERENCES  
↳ Pozycje (Nr_pozycji)
```

/

```
ALTER TABLE Wynagrodzenia ADD CONSTRAINT Otrzymuje_wynagrodzenie FOREIGN KEY (Nr_pracownika)  
↳ REFERENCES Pracownicy (Nr_pracownika)
```

/

```
ALTER TABLE Trenerzy ADD CONSTRAINT Ma_typ_licencji FOREIGN KEY (Nr_typu_licencji)
↳ REFERENCES Typy_Licencji (Nr_typu_licencji)
/
```

```
ALTER TABLE Mecze ADD CONSTRAINT Jest_rozgrywany_na_stadionie FOREIGN KEY (Nr_stadionu)
↳ REFERENCES Stadiony (Nr_stadionu)
/
```

5.4 Skrypt SQL uzupełniający bazę danych

```
INSERT INTO Adresy VALUES (1,'Warszawa', 'Ulica Marymoncka', '10', '01-869');
INSERT INTO Adresy VALUES (2,'Kielce', 'Ulica Marca', '11', '02-469');
INSERT INTO Adresy VALUES (3,'Gdańsk', 'Ulica Kizo', '69', '69-420');
INSERT INTO Adresy VALUES (4,'Warszawa', 'Ulica Falkowska', '15', '01-869');
INSERT INTO Adresy VALUES (5,'Warszawa', 'Ulica Mokotowska', '415', '02-333');
INSERT INTO Adresy VALUES (6,'Warszawa', 'Ulica Praska', '42', '04-333');
INSERT INTO Adresy VALUES (7,'Warszawa', 'Ulica francuska', '52', '05-234');
INSERT INTO Adresy VALUES (8,'Warszawa', 'Ulica narutowicza', '12', '09-324');
```

```
INSERT INTO Kluby_Pilkarskie VALUES (1,'NRG Warszawa', '123-456-789',
↳ 'nrg_warszawa@poczta.pl', 1);
```

```
INSERT INTO Stadiony
VALUES (1, 'Stadion miejski im. Michała Kępi', 50000, 1, 4);
```

```
INSERT INTO Pozycje VALUES (1,'SPO');
INSERT INTO Pozycje VALUES (2,'N');
INSERT INTO Pozycje VALUES (3,'SO');
INSERT INTO Pozycje VALUES (4,'BR');
INSERT INTO Pozycje VALUES (5,'SP');
INSERT INTO Pozycje VALUES (6,'SN');
INSERT INTO Pozycje VALUES (7,'LS');
```

```
INSERT INTO Pracownicy
VALUES (1, 'Jan', 'Kowalski', '12345678901', 'M', TO_DATE('1990-05-15', 'YYYY-MM-DD'),
↳ TO_DATE('2020-01-01', 'YYYY-MM-DD'), 'P', '+48321654987', 'PL12345678901234567890123456',
↳ 'jan.kowalski@example.com', 1, 2);
```

```
INSERT INTO Pilkarze
↳ ("NR_PRACOWNIKA","NR_KOSZULKI","LICZBA_WYSTEPOW","LICZBA_GOLI","LICZBA_ASYST","NR_POZYCJI")
↳ VALUES (1,10,15,3,1,1);
```

```
INSERT INTO Pracownicy
VALUES (2, 'Filip', 'Rakoczy', '32456879115', 'M', TO_DATE('1993-04-25', 'YYYY-MM-DD'),
↳ TO_DATE('2021-08-01', 'YYYY-MM-DD'), 'P', '+48333333987', 'PL12345679999934567890123456',
↳ 'filip.rakoczy@example.com', 1, 3);
```

```
INSERT INTO Pilkarze
↳ ("NR_PRACOWNIKA","NR_KOSZULKI","LICZBA_WYSTEPOW","LICZBA_GOLI","LICZBA_ASYST","NR_POZYCJI")
↳ VALUES (2,9,10,0,1,5);
```

```
INSERT INTO Pracownicy
VALUES (3, 'Igor', 'Strzalek', '99563256974', 'M', TO_DATE('1996-06-25', 'YYYY-MM-DD'),
↳ TO_DATE('2022-08-01', 'YYYY-MM-DD'), 'P', '+48111133987', 'PL12345671119934567890123456',
↳ 'igor.strzalek@example.com', 1, 5);
```

```
INSERT INTO Pilkarze
↳ ("NR_PRACOWNIKA","NR_KOSZULKI","LICZBA_WYSTEPOW","LICZBA_GOLI","LICZBA_ASYST","NR_POZYCJI")
↳ VALUES (3,44,8,4,3,6);
```

```
INSERT INTO Pracownicy
VALUES (4, 'Dominik', 'Haldun', '98565512351', 'M', TO_DATE('1998-06-25', 'YYYY-MM-DD'),
↳ TO_DATE('2020-08-01', 'YYYY-MM-DD'), 'P', '+48568933987', 'PL12345672219934567890123456',
↳ 'dominik.haldun@example.com', 1, 6);
```

```
INSERT INTO Pilkarze
↳ ("NR_PRACOWNIKA","NR_KOSZULKI","LICZBA_WYSTEPOW","LICZBA_GOLI","LICZBA_ASYST","LICZBA_CZYSTYCH_KONT")
↳ VALUES (4,1,20,0,0,10,4);
```

```
INSERT INTO Specjalizacje VALUES (1, 'Fizjoterapeuta')
INSERT INTO Specjalizacje VALUES (2, 'Trener Dynamiki')
INSERT INTO Specjalizacje VALUES (3, 'Szkoleniowiec')
```

```
INSERT INTO Typy_Licencji VALUES (1, 'UEFA A')
INSERT INTO Typy_Licencji VALUES (2, 'UEFA B')
INSERT INTO Typy_Licencji VALUES (3, 'PRO A EUFA')
```

```
INSERT INTO Trenerzy_Specjalizacje VALUES (1, 5)
INSERT INTO Trenerzy_Specjalizacje VALUES (3, 6)
```

```
INSERT INTO Pracownicy
VALUES (5, 'Maurycy', 'Fabijański', '324593652187', 'M', TO_DATE('1991-03-23',
↳ 'YYYY-MM-DD'), TO_DATE('2022-07-04', 'YYYY-MM-DD'), 'T',
↳ '+48333398337', 'PL12399993245674567890193456', 'Maurycy.fabijanski@example.com', 1, 7);
```

```
INSERT INTO Trenerzy VALUES (5, '135134562', 2)
```

```
INSERT INTO Pracownicy
VALUES (6, 'Konrad', 'Faworyt', '294768791131', 'M', TO_DATE('1993-04-25', 'YYYY-MM-DD'),
↳ TO_DATE('2021-08-01', 'YYYY-MM-DD'), 'T', '+48233543987', 'PL12345634999912345678907956',
↳ 'Konrad.Faworyt@example.com', 1, 8);
```

```
INSERT INTO Trenerzy VALUES (6, '186733652', 1)
```

```
INSERT INTO Wynagrodzenia VALUES (1, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 9800.00, 1)
```

```
INSERT INTO Wynagrodzenia VALUES (2, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 5256.00, 2)
```

```

INSERT INTO Wynagrodzenia VALUES (3, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 7632.00, 3)
INSERT INTO Wynagrodzenia VALUES (4, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 5416.00, 4)
INSERT INTO Wynagrodzenia VALUES (5, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 9855.00, 5)
INSERT INTO Wynagrodzenia VALUES (6, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 8478.00, 6)

INSERT INTO druzyzny VALUES (1, 'Druzyna A', 23, 'M', 'S', 1);
INSERT INTO druzyzny VALUES (2, 'Druzyna B', 23, 'M', 'S', 1);

INSERT INTO druzyzny_pracownicy VALUES (1, 1);
INSERT INTO druzyzny_pracownicy VALUES (1, 2);
INSERT INTO druzyzny_pracownicy VALUES (1, 3);
INSERT INTO druzyzny_pracownicy VALUES (1, 4);

INSERT INTO Wynagrodzenia VALUES (1, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 9800.00, 1)
INSERT INTO Wynagrodzenia VALUES (2, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 5256.00, 2)
INSERT INTO Wynagrodzenia VALUES (3, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 7632.00, 3)
INSERT INTO Wynagrodzenia VALUES (4, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 5416.00, 4)
INSERT INTO Wynagrodzenia VALUES (5, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 9855.00, 5)
INSERT INTO Wynagrodzenia VALUES (6, TO_DATE('2023-08-25', 'YYYY-MM-DD'), 8478.00, 6)

INSERT INTO Mecze VALUES (1, TO_DATE('2023-11-11', 'YYYY-MM-DD'), '8 liga', 'fc rzodkiewki',
↳ '3:1', 'Mateusz Frankowski', '1300', 1, 1)

INSERT INTO Mecze VALUES (2, TO_DATE('2023-11-14', 'YYYY-MM-DD'), '8 liga', 'fc kibice',
↳ '2:0', 'Kamil Jakubowski', '1230', 1, 1)

INSERT INTO Mecze VALUES (3, TO_DATE('2023-11-21', 'YYYY-MM-DD'), '8 liga', 'gwardia
↳ warszawa', '1:0', 'Mateusz Frankowski', '1534', 1, 1)

INSERT INTO Mecze VALUES (4, TO_DATE('2023-11-28', 'YYYY-MM-DD'), '8 liga', 'sokoły
↳ podlasia', '3:2', 'Kamil Jakubowski', '7632', 1, 1)

INSERT INTO Mecze VALUES (5, TO_DATE('2023-12-01', 'YYYY-MM-DD'), '8 liga', 'fc kalafiory',
↳ '0:0', 'Mateusz Frankowski', '8621', 1, 1)

INSERT INTO Mecze VALUES (6, TO_DATE('2023-12-15', 'YYYY-MM-DD'), '8 liga', 'ekipa walusia',
↳ '', 'Kamil Jakubowski', '', 1, 1)

INSERT INTO Mecze VALUES (7, TO_DATE('2023-12-20', 'YYYY-MM-DD'), '8 liga', 'obroncy
↳ tobruku', '', 'Mateusz Frankowski', '', 1, 1)

INSERT INTO Mecze VALUES (8, TO_DATE('2023-12-31', 'YYYY-MM-DD'), '8 liga', 'kryształ
↳ targowek', '', 'Kamil Jakubowski', '', 1, 1)

```

```

INSERT INTO druzyny_mecze VALUES (1,1);
INSERT INTO druzyny_mecze VALUES (1,2);
INSERT INTO druzyny_mecze VALUES (1,3);
INSERT INTO druzyny_mecze VALUES (1,4);
INSERT INTO druzyny_mecze VALUES (1,5);
INSERT INTO druzyny_mecze VALUES (1,6);
INSERT INTO druzyny_mecze VALUES (1,7);
INSERT INTO druzyny_mecze VALUES (1,8);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (1, 1);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (1, 2);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (1, 3);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (2, 1);
INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (2, 2);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (2, 3);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (2, 4);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (3, 3);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (4, 1);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (4, 2);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (4, 3);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (5, 2);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (6, 4);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (7, 2);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (7, 3);

INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)

```



```
VALUES (7, 4);
```

```
INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (8, 1);
```

```
INSERT INTO Mecze_Pracownicy (Nr_meczu, Nr_pracownika)
VALUES (8, 3);
```

5.5 Przykłady zapytań i poleceń SQL odnoszących się do bazy danych

```
SELECT * FROM kluby_pilkarskie WHERE nr_adresu in
(SELECT adresy.nr_adresu FROM adresy WHERE nr_budynku = 10)
```

	NR_KLUBU	NAZWA_KLUBU	NUMER_TELEFONU	ADRES_EMAIL	NR_ADRESU
1	1	NRG Warszawa	123-456-789	nrg_warszawa@poczta.pl	1

Rysunek 7: wynik zapytania zwraca informacje o klubie piłkarskim

```
SELECT * FROM pracownicy WHERE Nr_pracownika IN (SELECT Nr_pracownika FROM
Drużyny_Pracownicy WHERE Nr_drużyny IN (SELECT Nr_drużyny FROM drużyny WHERE
Nazwa_drużyny = 'Drużyna A'))
```

	NR_PRACOWNIKA	IMIE	NAZWISKO	PESEL	PLEC	DATA_URODZENIA	DATA_ZATRUDNIENIA	STANOWISKO	NR_TELEFONU	NR_KONTA	ADRES_EMAIL	NR_KLUBU	NR_ADRESU
1	1	Jan	Kowalski	12345678901	M	90/05/15	20/01/01	P	+48321654987	PL12345678901234567890123456	jan.kowalski@example.com	1	2
2	2	Filip	Rakoczy	32456789115	M	93/04/25	21/08/01	P	+48333333987	PL12345679999934567890123456	filip.rakoczy@example.com	1	3
3	3	Igor	Strzalek	99563256974	M	96/06/25	22/08/01	P	+48111133987	PL12345671119934567890123456	igor.strzalek@example.com	1	5
4	4	Dominik	Haldun	98565512351	M	98/06/25	20/08/01	P	+48568933987	PL12345672219934567890123456	dominik.haldun@example.com	1	6

Rysunek 8: wynika zapytania zwraca informacje o pracownikach z drużyny A

```
SELECT *
FROM pracownicy
JOIN pilkarze ON pracownicy.Nr_pracownika = pilkarze.nr_pracownika
```

	NR_PRACOWNIKA	IMIE	NAZWISKO	PESEL	PLEC	DATA_URODZENIA	DATA_ZATRUDNIENIA	STANOWISKO	NR_TELEFONU	NR_KONTA
1	1	Jan	Kowalski	12345678901	M	90/05/15	20/01/01	P	+48321654987	PL12345678901234567890123456
2	2	Filip	Rakoczy	32456789115	M	93/04/25	21/08/01	P	+48333333987	PL12345679999934567890123456
3	3	Igor	Strzalek	99563256974	M	96/06/25	22/08/01	P	+48111133987	PL12345671119934567890123456
4	4	Dominik	Haldun	98565512351	M	98/06/25	20/08/01	P	+48568933987	PL12345672219934567890123456

Rysunek 9: Wynik zapytania zwraca informacje o wszystkich pracownikach, którzy są piłkarzami, razem z informacjami o piłkarzach

	ADRES_EMAIL	NR_KLUBU	NR_ADRESU	NR_PRACOWNIKA_1	NR_KOSZULKI	LICZBA_WYSTĘPOW	LICZBA_GOLI	LICZBA_ASYST	LICZBA_CZYSTYCH_KONT	DATA_WYGASNIENIA_KONTRAKTU	NR_POZYCII
	jan.kowalski@example.com	1	2	1	10	15	3	1	(null)	(null)	1
	filip.rakoczy@example.com	1	3	2	9	10	0	1	(null)	(null)	5
	igor.strzalek@example.com	1	5	3	44	8	4	3	(null)	(null)	6
	dominik.haldun@example.com	1	6	4	1	20	0	0	10	(null)	4

Rysunek 10: część druga zapytania

```
SELECT * FROM Pracownicy WHERE stanowisko = 'P'
```

1	NR_PRAWNOWNIKA	1	IMIE	1	NAZWISKO	1	PESEL	1	PLEC	1	DATA_URODZENIA	1	DATA_ZATRUJNIENIA	1	STANOWISKO	1	NR_TELFONU	1	NR_KONTA	1	ADRES_EMAIL	1	NR_KLUBU	1	NR_ADRESU
	1	1	tan		Kowalski		3245678901	M		90/05/15	20/01/01		P		+48321654907		PL12345678901234567890123456				tan.kowalski@example.com		1		2
	2	2	Filip		Rakoczy		3245678915	M		90/04/25	21/08/01		P		+48333333997		PL123456789999934567890123456				filip.rakoczy@example.com		1		3
	3	3	igor		Strzalek		99563256974	M		96/06/25	22/08/01		P		+48111133997		PL12345671119934567890123456				igor.strzalek@example.com		1		5
	4	4	Dominik		Haldun		9856512351	M		98/06/25	20/08/01		P		+48565693997		PL12345672219934567890123456				dominik.haldun@example.com		1		4

Rysunek 11: Zwraca informacje o pracownikach, którzy są piłkarzami

```
SELECT * FROM Pracownicy WHERE stanowisko = 'T'
```

	NR_PRACOWNIKA	IMIE	NAZWISKO	PESEL	PLEC	DATA_URODZENIA	DATA_ZATRUDNIENIA	STANOWISKO	NR_TELEFONU	NR_KONTA	ADRES_EMAIL	NR_KLUBU	NR_ADRESU
1	5	Murczyk	Fabijanski	32593652187	M	91/03/23	22/07/04	T	+48333396337	PL12399993245674567890193456	Murczyk.Fabijanski@example.com	1	7
2	6	Konrad	Faworyt	29467891111	M	93/04/25	21/08/01	T	+483233543987	PL12345678901234567890123456789056	Konrad.Faworyt@example.com	1	8

Rysunek 12: Zwraca informacje o pracownikach, którzy są trenerami

```
SELECT * FROM Pracownicy WHERE Nr_pracownika IN (SELECT Nr_pracownika FROM Mecze_Pracownicy
↳ WHERE Nr_meczu IN (SELECT Nr_meczu FROM Mecze WHERE Wynik_meczu = '3:2'))
```

NR_PRACOWNIKA	IMIE	NAZWISKO	PESEL	PLEC	DATA_URODZENIA	DATA_ZATRUCIENIA	STANOWISKO	NR_TELEFONU	NR_KONTA	ADRES_EMAIL	NR_KLUBU	NR_ADRESU
1	Jan	Kowalski	32456789101	M	90/05/15	20/01/01	P	+48321654587	PL12345678901234567890123456	jan.kowalski@example.com	1	2
2	Filip	Rozalany	32456789115	M	93/04/25	21/08/01	P	+48333333987	PL12345679999994567890123456	filip.rozalany@example.com	1	3
3	Tygor	Straczek	99663256974	M	96/06/25	22/08/01	P	+48111333987	PL1234567113994567890123456	tygor.straczek@example.com	1	3

Rysunek 13: Zwraca informacje o pracownikach, którzy brali udział w meczu, który zakończył się wynikiem 3:2

```
SELECT * FROM pracownicy WHERE Nr_pracownika in (IN SELECT Nr_pracownika FROM
↳ Trenerzy_Specjalizacje WHERE Nr_specjalizacji IN (SELECT Nr_specjalizacji FROM
↳ Specjalizacje WHERE specjalizacja = 'Fizjoterapeuta'))
```

	NR_PRACOWNIKA	IMIE	NAZWISKO	PESEL	PLEC	DATA_URODZENIA	DATA_ZATRUDNIENIA	STANOWISKO	NR_TELEFONU	NR_KONTA	ADRES_EMAIL	NR_KLUBU	NR_ADRESU
1	5	Maurycy	Fabijański	32593652187	M	91/03/23	22/07/04	T	+48333398337	PL12399993245674567890193456	Maurycy.fabjanski@example.com	1	7

Rysunek 14: Wynik zapytania zwraca informacje o pracowniku, który ma specjalizację fizjoterapeuta

```
SELECT *
FROM pracownicy
JOIN pilkarze ON pracownicy.Nr_pracownika = pilkarze.nr_pracownika WHERE Nr_pozycji IN
↳ (SELECT Nr_pozycji From pozycje WHERE pozycja = 'BR')
```

NR_PRACOWNIKA	IMIE	NAZWISKO	PESEL	PLEC	DATA_URODZENIA	DATA_ZATRUDNIENIA	STANOWISKO	NR_TELEFONU	NR_KONTA
1	2	Filip Rakoczy	32456879115	M	93/04/25	21/08/01	P	+48333333987	PL12345679999934567890123456

Rysunek 15: Wynik zapytania Zwraca informacje o pracowniku piłkarzu na pozycji bramkarza

ADRES_EMAIL	NR_KLUBU	NR_ADRESU	NR_PRACOWNIKA_1	NR_KOSZULKI	LICZBA_WYSTEPOW	LICZBA_GOLI	LICZBA_ASYST	LICZBA_CZYSTYCH_KONT	DATA_WYGASNIENIA_KONTRAKTU	NR_POZYCJI
filip.rakoczy@example.com	1	3	2	9	10	0	1	(null)	(null)	5

Rysunek 16: część druga zapytania