Warszawa, 18 grudnia 2012r.

**Wstęp do Baz Danych**

**Projekt**

***„Salon motocyklowy”***

**Wykonali:**

*Mateusz Filipowicz  
Piotr Małecki*

Spis treści

[1. Zakres i cel projektu 4](#_Toc343681006)

[2. Definicja systemu 4](#_Toc343681007)

[2.1 Perspektywa sprzedawcy 4](#_Toc343681008)

[2.2 Perspektywa kierownika 4](#_Toc343681009)

[2.3 Perspektywa dyrektora 4](#_Toc343681010)

[3. Model konceptualny 5](#_Toc343681011)

[3.1 Definicja zbiorów encji 5](#_Toc343681012)

[3.2 Ustalenie związków między encjami 6](#_Toc343681013)

[3.3 Atrybuty i dziedziny atrybutów 7](#_Toc343681014)

[3.3.1 Atrybuty 7](#_Toc343681015)

[3.3.2 Dziedziny atrybutów 10](#_Toc343681016)

[3.4 Klucze kandydujące i główne 12](#_Toc343681017)

[3.5Weryfikacja pod kątem wystąpienia redundancji 12](#_Toc343681018)

[3.6 Schemat ER na poziomie konceptualnym 13](#_Toc343681019)

[4. Model logiczny 14](#_Toc343681020)

[4.1 Charakterystyka modelu relacyjnego 14](#_Toc343681021)

[4.2 Usunięcie właściwości niekompatybilnych z modelem relacyjnym - przykłady 14](#_Toc343681022)

[4.2.1 Usunięcie związków wielu do wielu. 14](#_Toc343681023)

[4.2.2 Usunięcie związków rekurencyjnych wielu do wielu oraz atrybutów złożonych. 15](#_Toc343681024)

[4.3 Usunięcie pułapek szczelinowych i wachlarzowych 15](#_Toc343681025)

[4.3.1 Pułapki wachlarzowe 15](#_Toc343681026)

[4.3.2 Pułapki szczelinowe 15](#_Toc343681027)

[4.4 Proces normalizacji – analiza i przykłady 15](#_Toc343681028)

[4.4.1Analiza anomalii wstawiania, modyfikacji i usuwania. 15](#_Toc343681029)

[4.4.2 Pierwsza postać normalna 16](#_Toc343681030)

[4.4.3 Druga postać normalna 16](#_Toc343681031)

[4.4.4 Trzecia postać normalna 16](#_Toc343681032)

[4.5 Schemat ER na poziomie modelu logicznego 19](#_Toc343681033)

[4.6 Więzy integralności 20](#_Toc343681034)

[4.6.1 Wymaganie obecności danych 20](#_Toc343681035)

[4.6.2 Integralność encji 20](#_Toc343681036)

[4.6.3 Integralność referencyjna 20](#_Toc343681037)

[4.7 Proces denormalizacji 20](#_Toc343681038)

[5. Model fizyczny 21](#_Toc343681039)

[5.1 Projekt transakcji 21](#_Toc343681040)

[5.1.1 Perspektywa sprzedawcy: 21](#_Toc343681041)

[5.1.2 Perspektywa kierownika 21](#_Toc343681042)

[5.1.3 Perspektywa dyrektora 21](#_Toc343681043)

[5.2 Dobór indeksów 22](#_Toc343681044)

[5.3 Skrypt SQL zakładający bazę danych 22](#_Toc343681045)

[5.4 Przykłady zapytań i poleceń SQL odnoszących się do bazy danych 31](#_Toc343681046)

[6. Bibliografia 32](#_Toc343681047)

# 1. Zakres i cel projektu

Celem naszego projektu, było stworzenie bazy danych, która ma uprościć ujednolicić zadania i pracę osób zatrudnionych w salonie motocyklowym, ułatwić zarządzanie takimi obiektami, czyli zoptymalizować pracę takiego obiektu.

Opracowana przez nas baza umożliwia założenie sieci salonów motocyklowych, informacje o każdym salonie będą przetrzymywana w bazie danych. Umożliwia ona podniesienie wydajności pracy.

# 2. Definicja systemu

## 2.1 Perspektywa sprzedawcy

Sprzedawca może wykonywać następujące czynności:

* tworzyć oraz uzupełniać dane dotyczące klientów,
* koordynować jazdy próbne,
* wyszukiwać motocykle spełniające oczekiwania klientów,
* obsługiwać zamówienia klientów.

2.2 Perspektywa kierownika  
Kierownik może wykonywać następujące czynności:

* nadzorować zamówienia klientów,
* kontrolować efektywność pracy sprzedawców,
* kontrolować dostępność motocykli do jazd próbnych.

2.3 Perspektywa dyrektora  
Dyrektor posiada całkowity dostęp do bazy danych, a więc może:

* kontrolować efektywność pracy pracowników,
* mieć dostęp do wszystkich informacji klientów,
* mieć dostęp do wszystkich zamówień klientów,
* nadzorować oddziały.

# 3. Model konceptualny

## 3.1 Definicja zbiorów encji

|  |  |  |
| --- | --- | --- |
| **Nazwa zbioru encji** | **Opis** | **Własności** |
| **Motocykle** | Opisuje model i markę motocykli | Określa, że każdy motocykl ma swój egzemplarz. |
| **Producenci** | Obrazuje producenta danego motoru | Łączy Kraje z Motocyklami. Użyta w celu eliminacji redundancji danych |
| **Kraje** | Encja obrazująca kraj pochodzenia motocykla | Łączy się z Encją Producenci, w celu uniknięcia redundancji. |
| **Egzemplarze** | Określa bardzo dokładnie wszystkie cechy danego motocyklu. | Encja Egzemplarze jest w relacji z encją Zamówienia w celu zamodelowania procesu zamówień i w relacji z DostępnoscEgzemplarza, aby sprawdzić czy dany motocykl może zostać sprzedany , zamówiony lub użyty do jazdy próbnej. |
| **TypyMotocykli** | Przedstawia jakiego typu jest dany motocykl, np. sportowy | Encja pomocnicza, łączy się z Egzemplarzami, w celu eliminacji redundancji danych |
| **Kolor** | Określenie koloru motocyklu | Encja pomocnicza, łączy się z Egzemplarzami, w celu eliminacji redundancji danych |
| **RodzajePaliwa** | Określa jakie paliwo może być stosowane w danym Egzemplarzu | Encja pomocnicza, łączy się z Egzemplarzami, w celu eliminacji redundancji danych |
| **Oddzialy** | Opisuje oddziały. | Sprawdzamy czy dany oddział posiada jakiś egzemplarz. Ponadto mamy związek z Zatrudnienia, modelujący zatrudnieniami. |
| **Klienci** | Opisuje klienta salonu motocyklowego. | Klienci mogą wykonywać różne czynności takie jak zamawianie egzemplarzy motocykli lub branie udziału wjazdach próbnych. |
| **Stanowiska** | Określa stanowisko pracownika. | Jeden pracownik może znajdować się wyłącznie na jednym stanowisku. |
| **Miasta** | Opisuje miasto w którym jest oddział, jak i miasto z którego pochodzi klient. | Encja pomocnicza do oddziałów i klientów. Określa kod pocztowy i przypisane mu miasto. |
| **Zatrudnienia** | Encja pośrednicząca pomiędzy Encjami Personel i Oddzialy opisująca zatrudnionych pracowników. | Encja modelująca zatrudnienia w celu uniknięcia związku wielu do wielu. Dany pracownik może tylko raz być zatrudniony w danym oddziale. |
| **Personel** | Encja opisująca personel | Mamy w niej imię i nazwisko personelu, datę urodzenia oraz początek zatrudnienia, i opcjonalnie datę zakończenia pracy. Pracownik może zamawiać Zamówienia (relacja z encją Zamówienia), oraz może organizować jazdy próbne. (relacja z encją Jazdy Próbne). |
| **JazdyProbne** | Encja opisująca jazdy próbne. | Określa kto bierze udział w jeździe, kto wypożyczył, jaki motor i na jaki czas. |
| **DostępnoscEgzemplarza** | Encja do opisu czy dany egzemplarz jest dostępny w danym oddziale. | Jest ona w związku z Egzemplarzami i jazdami próbnymi. Dostępny motocykl można jedynie wypożyczyć na jazdę próbną – nie można go kupić. |
| **Zamowienia** | Obrazuje stan zamówienia danego motocykla. Każde zamówienie obejmuje dokładnie 1 motocykl. | Z tej encji można iść do Encji sprzedane oraz rezygnacje. Jest ona w relacji z tymi dwiema encjami. |
| **Sprzedane** | Określa, które zamówienie zostały zrealizowane i jaka jest ich data odbioru. | Jest ono w relacji z encją zamówienia. |
| **Rezygnacje** | Określa które zamówienie zostało odrzucone. | Opcjonalnie jest podawany powód odrzucenia zamówienia |

## 3.2 Ustalenie związków między encjami

W projekcie wyróżniamy krotności:  
1..1 – jedno i tylko jedno wystąpienie encji  
0..\* - brak lub wiele wystąpień encji  
1..\* - jedno lub wiele wystąpień encji.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nazwa encji** | **Krotność** | **Związek** | | **Nazwa encji** | **Krotność** |
| Kraje | 1..1 | PochodziZ | R1 | Producenci | 1..\* |
| Producenci | 1..1 | Produkuje | R2 | Motocykle | 0..\* |
| Motocykle | 1..1 | Ma | R3 | Egzemplarze | 0..\* |
| TypyMotocykli | 1..1 | JestTypu | R4 | Egzemplarze | 1..\* |
| Kolory | 1..1 | JestKoloru | R5 | Egzemplarze | 1..\* |
| RodzajePaliwa | 1..1 | NaPaliwo | R6 | Egzemplarze | 1..\* |
| Egzemplarze | 1..1 | Zamawiane | R7 | Zamowienia | 0..\* |
| Egzemplarze | 1..1 | JestDostepny | R8 | DostepnoscEgzemplarza | 0..\* |
| Miasta | 1..1 | JestZ | R9 | Klienci | 1..\* |
| Miasta | 1..1 | ZnajdujeSieW | R10 | Oddzialy | 1..\* |
| Oddzialy | 1..1 | Zatrudnia | R11 | Zatrudnienie | 0..\* |
| Oddzialy | 1..1 | Posiada | R12 | DostepnoscEgzemplarza | 0..\* |
| Personel | 1..1 | Zamawia | R13 | Zamowienia | 0..\* |
| Personel | 1..1 | Organizuje | R14 | JazdyProbne | 0..\* |
| Personel | 1..1 | JestZatrudniony | R15 | Zatrudnienia | 0..\* |
| Stanowiska | 1..1 | naStanowisku | R16 | Zatrudnienia | 1..\* |
| Klienci | 1..1 | BioraUdzial | R17 | JazdyProbne | 0..\* |
| Klienci | 1..1 | Zamawiaja | R18 | Zamowienia | 0..\* |
| DostepnoscEgzemplarza | 1..1 | JestWypozyczany | R19 | JazdyProbne | 0..\* |
| Zamowienia | 1..1 | Sprzedany | R20 | Sprzedane | 0..\* |
| Zamowienia | 1..1 | Rezygnacja | R20 | Rezygnacje | 0..\* |

W relacji R16 zakładamy że jeden klient może testować konkretny motocykl dokładnie jeden raz.

## 3.3 Atrybuty i dziedziny atrybutów

### 3.3.1 Atrybuty

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nazwa zbioru encji** | **Atrybut** | **Opis** | **Nadrzędny zbiór encji** | **Typ danych** | **Wart. pusta** | **Wielo wart.** | **Wart. domyślna** | **Wart. trigger.** |
| *Kraje* | **kraj** | **Klucz główny** | - | **Varchar2(20)** | **NIE** | **NIE** | **NIE** | **NIE** |
| *Producenci* | **marka** | **Klucz główny** | - | **Varchar2(20)** | **NIE** | **NIE** | **NIE** | **NIE** |
| kraj | Klucz obcy | Kraje | Varchar2(20) | NIE | NIE | NIE | NIE |
| *Motocykle* | **model** | **Klucz główny** | - | **Varchar2(20)** | **NIE** | **NIE** | **NIE** | **NIE** |
| **marka** | **Klucz główny obcy** | **Producenci** | **Varchar2(20)** | **NIE** | **NIE** | **NIE** | **NIE** |
| *TypyMotocykli* | **typ** | **Klucz główny** | **-** | **Varchar2(20)** | **NIE** | **NIE** | **NIE** | **NIE** |
| *Kolory* | **kolor** | **Klucz główny** | **-** | **Varchar2(20)** | **NIE** | **NIE** | **NIE** | **NIE** |
| *RodzajePaliwa* | **paliwo** | **Klucz główny** | **-** | **Varchar2(20)** | **NIE** | **NIE** | **NIE** | **NIE** |
| *Egzemplarze* | **idMotocyklu** | **Klucz główny** | - | **Number** | **NIE** | **NIE** | **NIE** | **TAK**  **(od 1 z krokiem co 1)** |
| marka | Klucz obcy | Motocykle | Varchar2(20) | NIE | NIE | NIE | NIE |
| model | Klucz obcy | Motocykle | Varchar2(20) | NIE | NIE | NIE | NIE |
| typ | Klucz obcy | TypyMotocykli | Varchar2(20) | NIE | NIE | NIE | NIE |
| paliwo | Klucz obcy | RodzajePaliwa | Varchar2(20) | NIE | NIE | NIE | NIE |
| kolor | Klucz obcy | Kolory | Varchar2(20) | NIE | NIE | NIE | NIE |
| **nrSeryjny** | **Numer seryjny motocyklu** | **-** | **Varchar2(30)** | **NIE** | **NIE** | **NIE** | **NIE** |
| rokProdukcji | Rok produkcji | - | Number(4,0) | NIE | NIE | NIE | NIE |
| pojSilnika | Pojemność silnika [cm3] | - | Number(4,0) | NIE | NIE | NIE | NIE |
| mocSilnika | Moc silnika [Km] | - | Number(3,0) | NIE | NIE | NIE | NIE |
| waga | Waga [kg] | - | Number(4,0) | NIE | NIE | NIE | NIE |
| szerokosc | Szerokosc [cm] | - | Number(3,0) | NIE | NIE | NIE | NIE |
| wysokosc | Wysokosc [cm] | - | Number(3,0) | NIE | NIE | NIE | NIE |
| dlugosc | Dlugosc [cm] | - | Number(3,0) | NIE | NIE | NIE | NIE |
| cenaDetaliczna | Detaliczna cena producenta [zł] | - | Number(8,2) | NIE | NIE | NIE | NIE |
| gwarancja | Okres gwarancji [lat] | - | Number(2,0) | NIE | NIE | NIE | NIE |
| opis | Opis egzemplarza | - | Varchar2(255) | TAK | NIE | „NULL” | NIE |
| *Miasta* | **kodPocztowy** | **Klucz główny** | **-** | **Number(5,0)** | **NIE** | **NIE** | **NIE** | **NIE** |
| miasto | Nazwa miasta | - | Varchar2(30) | NIE | NIE | NIE | NIE |
| *Klienci* | **idKlienta** | **Klucz główny** | **-** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| imie | Imię klienta | - | Varchar2(20) | NIE | NIE | NIE | NIE |
| drugieImie | Drugie imię klienta | - | Varchar2(20) | TAK | NIE | „NULL” | NIE |
| nazwisko | Nazwisko klienta | - | Varchar2(20) | NIE | NIE | NIE | NIE |
| dataUrodzenia | Data urodzenia klienta | - | Date | NIE | NIE | NIE | NIE |
| kodPocztowy | Klucz obcy | Miasta | Number(5,0) | NIE | NIE | NIE | NIE |
| ulica | Ulica zamieszkania klienta | - | Varchar2(30) | NIE | NIE | NIE | NIE |
| nrDomu | Numer domu klienta | - | Varchar2(4) | NIE | NIE | NIE | NIE |
| nrMieszkania | Numer mieszkania klienta | - | Number(3,0) | NIE | NIE | NIE | NIE |
| pesel | Numer PESEL klienta | - | Number(11,0) | TAK | NIE | „NULL” | NIE |
| nrTelefonu | Numer telefonu klienta | - | Number(9,0) | NIE | NIE | NIE | NIE |
| *Oddzialy* | **idOddzialu** | **Klucz główny** | **-** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| kodPocztowy | Klucz obcy | Miasta | Number(5,0) | NIE | NIE | NIE | NIE |
| ulica | Ulica oddziału | - | Varchar2(30) | NIE | NIE | NIE | NIE |
| nrBudynku | Numer budynku oddziału | - | Varchar2(4) | NIE | NIE | NIE | NIE |
| nrLokalu | Numer lokalu oddziału | - | Number(3,0) | TAK | NIE | „NULL” | NIE |
| *Personel* | **idPracownika** | **Klucz główny** | **-** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| imie | Imię pracownika | - | Varchar2(20) | NIE | NIE | NIE | NIE |
| drugieImie | Drugie imię pracownika | - | Varchar2(20) | TAK | NIE | „NULL” |  |
| nazwisko | Nazwisko pracownika | - | Varchar2(20) | NIE | NIE | NIE | NIE |
| dataUrodzenia | Data urodzenia pracownika | - | Date | NIE | NIE | NIE | NIE |
| zatrudnionyOd | Data początku zatrudnienia pracownika | - | Date | NIE | NIE | SYSDATE | NIE |
| zatrudnionyDo | Data końca zatrudnienia pracownika | - | Date | TAK | NIE | NIE | NIE |
| nrTelefonu | Numer telefonu pracownika | - | Number(9,0) | NIE | NIE | NIE | NIE |
| *Stanowiska* | **funkcja** | **Klucz główny. Stanowisko pracownika** | **-** | **Varchar2(20)** | **NIE** | **NIE** | **NIE** | **NIE** |
| *Zatrudnienia* | **idPracownika** | **Klucz główny obcy** | **Personel** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| **idOddzialu** | **Klucz główny obcy** | **Oddzialy** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| funkcja | Klucz obcy | Stanowiska | Varchar2(20) | NIE | NIE | NIE | NIE |
| pensja | Aktualne zarobki pracownika | - | Number(7,2) | NIE | NIE | NIE | NIE |
| *DostEgezemplarza* | **idMotocyklu** | **Klucz główny obcy** | **Motocykle** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| idOddzialu | Klucz obcy | Oddzialy | Number | NIE | NIE | NIE | **TAK (od1 z krokiem co 1)** |
| *JazdyProbne* | **idMotocyklu** | **Klucz główny obcy** | **DostEgzemplarza** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| **idKlienta** | **Klucz główny obcy** | **Klienci** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| idPracownika | Klucz obcy | Personel | Number | NIE | NIE | NIE | **TAK (od1 z krokiem co 1)** |
| od | Data i godzina rozpoczęcia jazdy próbnej | - | Timestamp(0) | NIE | NIE | NIE | NIE |
| do | Data i godzina zakończenia jazdy próbnej | - | Timestamp(0) | NIE | NIE | NIE | NIE |
| uwagi | Uwagi klienta odnośnie jazdy próbnej | - | Varchar2(255) | TAK | NIE | „NULL” | NIE |
| *Zamowienia* | **idZamowienia** | **Klucz główny** | - | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| idMotocyklu | Klucz obcy | Motocykle | Number | NIE | NIE | NIE | TAK (od1 z krokiem co 1) |
| idKlienta | Klucz obcy | Klienci | Number | NIE | NIE | NIE | TAK (od1 z krokiem co 1) |
| idPracownika | Klucz obcy | Personel | Number | NIE | NIE | NIE | TAK (od1 z krokiem co 1) |
| dataZamowienia | Data złożenia zamówienia | - | Date | NIE | NIE | SYSDATE | NIE |
| doOdbioru | Możliwa data odbioru motocyklu | - | Date | NIE | NIE | NIE | NIE |
| cena | Ustalona cena sprzedaży | - | Number(8,2) | NIE | NIE | NIE | NIE |
| zaliczka | Wpłacona zaliczka | - | Number(8,2) | NIE | NIE | ‘0’ | NIE |
| *Sprzedane* | **idZamowienia** | **Klucz główny obcy** | **Zamowienia** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| dataOdbioru | Data odbioru zamówienia | - | Date | NIE | NIE | SYSDATE | NIE |
| *Rezgynacje* | **idZamowienia** | **Klucz główny obcy** | **Zamowienia** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| dataRezygnacji | Data rezygnacji z zamówienia | - | Date | NIE | NIE | SYSDATE | NIE |
| powód | Powód rezygnacji z zamówienia | - | Varchar2(255) | TAK | NIE | „NULL” | NIE |

### 3.3.2 Dziedziny atrybutów

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa zbioru encji** | **Atrybut** | **Dziedzina** | **Format** | **Zakres wartości** |
| *Kraje* | **kraj** | **Ciąg znaków** |  |  |
| *Producenci* | **marka** | **Ciąg znaków** |  |  |
| kraj | Ciąg znaków |  |  |
| *Motocykle* | **model** | **Ciąg znaków** |  |  |
| **marka** | **Ciąg znaków** |  |  |
| *TypyMotocykli* | **typ** | **Ciąg znaków** |  |  |
| *Kolory* | **kolor** | **Ciąg znaków** |  |  |
| *RodzajePaliwa* | **paliwo** | **Ciąg znaków** |  |  |
| *Egzemplarze* | **idMotocyklu** | **Liczby całkowite dodatnie** |  |  |
| marka | Ciąg znaków |  |  |
| model | Ciąg znaków |  |  |
| typ | Ciąg znaków |  |  |
| paliwo | Ciąg znaków |  |  |
| kolor | Ciąg znaków |  |  |
| nrSeryjny | Ciąg znaków |  |  |
| rokProdukcji | Liczby całkowite dodatnie | xxxx | x<0,9> |
| pojSilnika | Liczby całkowite dodatnie | xxxx | x<0,9> |
| mocSilnika | Liczby całkowite dodatnie | xxx | x<0,9> |
| waga | Liczby całkowite dodatnie | xxxx | x<0,9> |
| szerokosc | Liczby całkowite dodatnie | xxx | x<0,9> |
| wysokosc | Liczby całkowite dodatnie | xxx | x<0,9> |
| dlugosc | Liczby całkowite dodatnie | xxx | x<0,9> |
| cenaDetaliczna | Liczby zmiennoprzecinkowe dodatnie | xxxxxx.xx | x<0,9> |
| gwarancja | Liczby całkowite dodatnie | xx | x<0,9> |
| opis | Ciąg znaków |  |  |
| *Miasta* | **kodPocztowy** | **Ciąg znaków** | **xxxxx** | **x<0,9>** |
| miasto | Ciąg znaków |  |  |
| *Klienci* | **idKlienta** | **Liczby całkowite dodatnie** |  |  |
| imie | Ciąg znaków |  |  |
| drugieImie | Ciąg znaków |  |  |
| nazwisko | Ciąg znaków |  |  |
| dataUrodzenia | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| kodPocztowy | Ciąg znaków | xxxxx | x<0,9> |
| ulica | Ciąg znaków |  |  |
| nrDomu | Ciąg znaków |  |  |
| nrMieszkania | Liczby całkowite dodatnie | xxx | x<0,9> |
| pesel | Ciąg znaków | rrmmddxxxxx | rr<00,99>,m<01,12>&<21,32>,x<0,9> |
| nrTelefonu | Ciąg znaków | xxxxxxxxx | x<0,9> |
| *Oddzialy* | **idOddzialu** | **Liczby całkowite dodatnie** |  |  |
| kodPocztowy | Ciąg znaków | xxxxx | x<0,9> |
| ulica | Ciąg znaków |  |  |
| nrBudynku | Ciąg znaków |  |  |
| nrLokalu | Liczby całkowite dodatnie | xxx | x<0,9> |
| *Personel* | **idPracownika** | **Liczby całkowite dodatnie** |  |  |
| imie | Ciąg znaków |  |  |
| drugieImie | Ciąg znaków |  |  |
| nazwisko | Ciąg znaków |  |  |
| dataUrodzenia | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| zatrudnionyOd | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| zatrudnionyDo | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| nrTelefonu | **Liczby całkowite dodatnie** | xxxxxxxxx | x<0,9> |
| *Stanowiska* | funkcja | Ciąg znaków |  | ‘dyrektor’, ‘kierownik, ‘sprzedawca’ |
| *Zatrudnienia* | **idPracownika** | **Liczby całkowite dodatnie** |  |  |
| **idOddzialu** | **Liczby całkowite dodatnie** |  |  |
| funkcja | Ciąg znaków |  | dyrektor’, ‘kierownik, ‘sprzedawca’ |
| pensja | Liczby zmienoprzecinkowe dodatnie | xxxxx.xx | x<0,9> |
| *DostEgezemplarza* | **idMotocyklu** | **Liczby całkowite dodatnie** |  |  |
| idOddzialu | Liczby całkowite dodatnie |  |  |
| *JazdyProbne* | **idMotocyklu** | **Liczby całkowite dodatnie** |  |  |
| **idKlienta** | **Liczby całkowite dodatnie** |  |  |
| idPracownika | Liczby całkowite dodatnie |  |  |
| od | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| do | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| uwagi | Ciąg znaków |  |  |
| *Zamowienia* | **idZamowienia** | **Liczby całkowite dodatnie** |  |  |
| idMotocyklu | Liczby całkowite dodatnie |  |  |
| idKlienta | Liczby całkowite dodatnie |  |  |
| idPracownika | Liczby całkowite dodatnie |  |  |
| dataZamowienia | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| doOdbioru | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| cena | Liczby zmienoprzecinkowe dodatnie | xxxxxx.xx | x<0,9> |
| zaliczka | Liczby zmienoprzecinkowe dodatnie | xxxxxx.xx | x<0,9> |
| *Sprzedane* | **idZamowienia** | **Liczby całkowite dodatnie** |  |  |
| dataOdbioru | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| *Rezgynacje* | **idZamowienia** | **Liczby całkowite dodatnie** |  |  |
| dataRezygnacji | data, zależne od systemu | rr/mm/dd | dd<1,31>, mm<1,12>, rrrr<00, 99> |
| powód | Ciąg znaków |  |  |

## 3.4 Klucze kandydujące i główne

Wybierając klucze kandydujące, staraliśmy się znaleźć taki zbiór atrybutów, który by jednoznacznie identyfikował wystąpienie encji w zbiorze encji. Przy wyborze kluczy głównych kierowaliśmy się jak najmniejszą ilość atrybutów. Zwracaliśmy uwagę na fakt czy dany atrybut jest tekstowy czy liczbowy. Wybieraliśmy te drugie o względnie niskich wartościach.

Klucze kandydujące są pogrubione, a główne podkreślone i pogrubione.

## 3.5Weryfikacja pod kątem wystąpienia redundancji

W naszym projekcie nie występują związki wzajemne jednoznaczne (1:1)

Związek redundantny, to taki związek dostarczający informacje, które można uzyskać na podstawie innego związku.

- Do informacji o tym jakie jazdy próbne są organizowane możemy dojść od strony pracownika poprzez relacje R14 , również klient może uzyskać taką informację poprzez relację R16, ale Klient wie tylko o jazdach próbnych, w których sam bierze udział, a personel o wszystkich jazdach próbnych.

- Podobna sytuacja mogłaby wystąpić przy próbie sprawdzenia, które egzemplarze zostały sprzedane spoglądając od strony relacji R13 i R19, a z drugiej strony od relacji R17 i R19. Tutaj jednak również klient ma dostęp tylko do swoich egzemplarzy zakupionych a personel do wszystkich.

Zatem w naszej bazie nie zaobserwowaliśmy związku redundancyjnego.

## Salon motocyklowy.jpeg3.6 Schemat ER na poziomie konceptualnym

# 4. Model logiczny

## 4.1 Charakterystyka modelu relacyjnego

W modelu relacyjnym każda relacja ma nazwę i składa się z posiadających własne nazwy atrybutów (kolumn) danych. Każda krotka zawiera jedną wartość każdego atrybutu. O sile modelu relacyjnego świadczy prostota struktury logicznej. Jednak główny mechanizm tego modelu jest oparty na matematycznej teorii mnogości i rachunku predykantów pierwszego rzędu.

## 4.2 Usunięcie właściwości niekompatybilnych z modelem relacyjnym - przykłady

### 4.2.1 Usunięcie związków wielu do wielu.

W naszej bazie występowały następujące związki wielu do wielu.

Związek: Oddziały ⬄ Personel

Związek ten rozwiązaliśmy w następujący sposób: poprzez dodanie nowej encji Zatrudnienia oraz związków R11, R15. Bez tej encji Personel mógłby być zatrudniony w różnych oddziałach oraz w danym oddziale mogłoby pracować wielu pracowników.

|  |  |  |
| --- | --- | --- |
| **Nazwa zbioru encji** | **Opis** | **Własności** |
| **Zatrudnienia** | Encja pośrednicząca pomiędzy Encjami Personel i Odziały. | Encja modelująca zatrudnienia w celu uniknięcia związku wielu do wielu. Z jednej strony może być wielu pracowników zatrudnionych obecnie. Z drugiej strony każdy oddział zatrudnia wielu pracowników. |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Nazwa zbioru encji** | **Atrybut** | **Opis** | **Nadrzędny zbiór encji** | **Typ danych** | **Wart. pusta** | **Wielo wart.** | **Wart. domyślna** |
| *Zatrudnienia* | **idPracownika** | **Klucz główny obcy** | **Personel** | **Number** | **NIE** | **NIE** | **NIE** |
| **idOddzialu** | **Klucz główny obcy** | **Oddzialy** | **Number** | **NIE** | **NIE** | **NIE** |
| pensja | Aktualne zarobki pracownika | - | Number(7,2) | NIE | NIE | NIE |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nazwa encji** | **Krotność** | **Związek** | | **Nazwa encji** | **Krotność** |
| Oddzialy | 1..1 | Zatrudnia | R11 | Zatrudnienie | 0..\* |
| Personel | 1..1 | JestZatrudniony | R15 | Zatrudnienia | 0..\* |

### 4.2.2 Usunięcie związków rekurencyjnych wielu do wielu oraz atrybutów złożonych.

W naszej bazie nie ma związków rekurencyjnych wielu do wielu. Nie zaobserwowaliśmy też żadnych związków złożonych. Ponadto nie ma zaobserwowaliśmy żadnych pól atrybutów wielowartościowych. Już na pierwszym etapie projektowania wyeliminowaliśmy pola złożone, np:

* *Imię i Nazwisko* rozdzieliliśmy na dwa pola: *Imię* oraz *Nazwisko*
* *Adres* oddziału (podobnie jak klienta) rozdzieliliśmy na pola: *kodPocztowy, ulica, nrBudynku, nrLokalu*

## 4.3 Usunięcie pułapek szczelinowych i wachlarzowych

### 4.3.1 Pułapki wachlarzowe

Pułapki wachlarzowe mogłyby być związane z encjami Egzemplarze, Personel, Oddziały i Klienci. Po przebadaniu relacji występujących w naszym modelu, nie stwierdziliśmy występowania żadnych pułapek wachlarzowych, ponieważ do wszystkich potrzebnych informacji da się dojść z każdej odpowiedniej encji.

### 4.3.2 Pułapki szczelinowe

Podobnie pułapki szczelinowe mogłyby być związane z Producenci, Motocykle, Egzemplarze lub z encjami Klienci, Zamówienia, Rezerwacje. Jednak podobnie tutaj nie stwierdzamy pominięcia żadnych istotnych informacji w żadnym z rozważanych przypadków.

## 4.4 Proces normalizacji – analiza i przykłady

### 4.4.1Analiza anomalii wstawiania, modyfikacji i usuwania.

Po przeanalizowaniu uznaliśmy, że w naszej bazie danych nie obserwujemy anomalii wstawiania i modyfikacji. Jedynym miejscem, w którym w naszej bazie występowały anomalie usuwania były encje zatrudnienia, personel oraz oddziały. Informacje o personelu takie jak oddziały, w których pracował, ostatnia pensja oraz okresy pracy były tracone wraz z odejściem pracownika z danego oddziału. Zostało to rozwiązane poprzez usunięcie atrybutów *zatrudnionyOd* oraz *zatrudnionyDo* z encji *Personel* oraz utworzenie identycznych w encji *Zatrudnienia,* a także poprzez powiązanie związków R13 i R14 z encją zatrudnienie, zamiast encją Personel.

|  |  |  |
| --- | --- | --- |
| **Nazwa zbioru encji** | **Opis** | **Własności** |
| **Zatrudnienia** | Encja pośrednicząca pomiędzy Encjami Personel i Oddziały opisująca zatrudnionych pracowników. | Encja modelująca zatrudnienia w celu uniknięcia związku wielu do wielu. Dany pracownik może tylko raz być zatrudniony w danym oddziale. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nazwa zbioru encji** | **Atrybut** | **Opis** | **Nadrzędny zbiór encji** | **Typ danych** | **Wart. pusta** | **Wielo wart.** | **Wart. domyślna** | **Wart. trigger.** |
| *Zatrudnienia* | **idPracownika** | **Klucz główny obcy** | **Personel** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| **idOddzialu** | **Klucz główny obcy** | **Oddzialy** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| pensja | Aktualne zarobki pracownika | - | Number(7,2) | NIE | NIE | NIE | NIE |
| zatrudnionyOd | Data początku zatrudnienia pracownika | - | Date | NIE | NIE | NIE | NIE |
| zatrudnionyDo | Data końca zatrudnienia pracownika | - | Date | TAK | NIE | NIE | NIE |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nazwa encji** | **Krotność** | **Związek** | | **Nazwa encji** | **Krotność** |
| Zatrudnienia | 1..1 | Zamawia | R13 | Zamowienia | 0..\* |
| Zatrudnienia | 1..1 | Organizuje | R14 | JazdyProbne | 0..\* |

### 4.4.2 Pierwsza postać normalna

Relacja jest w pierwszej postaci normalnej, jeśli każda wartość atrybutu w każdej krotce tej relacji jest wartością elementarną, czyli nierozkładalną . W trakcie projektowania naszej bazy zwracaliśmy uwagę na fakt, żeby atrybuty w każdej krotce były wartościami nierozkładalnymi. Zatem nasza baza jest w pierwszej postaci normalnej.

### 4.4.3 Druga postać normalna

Relacja jest w drugiej postaci normalnej jeżeli każdy atrybut nie wchodzący w skład klucza zależy od klucza a nie od jego części. W każdej encji, która ma klucz główny w postaci prostej to niewątpliwie każdy atrybut nie wchodzący w skład klucza, zależy od całego klucza a nie od części.

Przeanalizowaliśmy encje zawierające następujące klucze główne:

|  |  |
| --- | --- |
| **Encja** | **Klucz główny** |
| *Motocykle* | *model* + *marka* |
| *Zatrudnienia* | *idPracownika* + *idOddzialu* |
| *JazdyProbne* | *idKlienta* + *idMotocyklu* |

Stwierdzamy, że każda kolumna zależy funkcyjnie od całego klucza głównego a nie od jego części. Nasza baza jest zatem w drugiej postaci normalnej.

### 4.4.4 Trzecia postać normalna

Dana relacja jest w trzeciej postaci normalnej, jeśli jest ona w drugiej postaci normalnej i każdy jej atrybut nie wchodzący w skład żadnego klucza potencjalnego nie jest przechodnio funkcyjnie zależny od żadnego klucza potencjalnego tej relacji. Inaczej mówiąc, wszystkie niekluczowe kolumny są określone kluczem, całym kluczem i tylko kluczem.

W encji *Motocykle* poza wyznaczonym kluczem głównym *id*Motocykla istnieje również klucz kandydujący *nrSeryjny,* który jest unikatowy dla każdego motocykla i jednoznacznie identyfikuje każdy motocykl. Zatem w encji tej mamy do czynienia z zależnościami przechodnimi. Ponadto zauważyliśmy, że może istnieć wiele jednakowych pod względem parametrów motocykli różniących się wyłącznie numerem seryjnym. W związku z tym tworzymy nową encję *KonkretnyMotocykl*.

|  |  |  |
| --- | --- | --- |
| **Nazwa zbioru encji** | **Opis** | **Własności** |
| **KonkretnyMotocykl** | Opisuje dokładnie jeden, konkretny motocykl na podstawie jego numeru seryjnego | Może istnieć wiele jednakowych motocykli różniących się wyłącznie numerem seryjnym. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nazwa zbioru encji** | **Atrybut** | **Opis** | **Nadrzędny zbiór encji** | **Typ danych** | **Wart. pusta** | **Wielo wart.** | **Wart. domyślna** | **Wart. trigger.** |
| *KonkretnyMotocykl* | **nrSeryjny** | **Klucz główny** | - | **Varchar2(30)** | **NIE** | **NIE** | **NIE** | **NIE** |
| idMotocyklu | Klucz obcy | Egzemplarze | Number | NIE | NIE | NIE | TAK (od1 z krokiem co 1) |
| dostepny | Określa czy konkretny motocykl został sprzedany | - | Char(1) | NIE | NIE | ‘t’ | NIE |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nazwa zbioru encji** | **Atrybut** | **Dziedzina** | **Format** | **Zakres wartości** |
| *KonkretnyMotocykl* | dostepny | Znak | x | x<’t’,’n’> t - tak n – nie |

Z encji Egzemplarze zostaje usunięty atrybut *nrSeryjny.* Zostaje utworzony nowy związek pomiędzy encją *Egzemplarze* a *KonkretnyMotocykl* oraz część dotychczasowych związków powiązanych z encją *Egzemplarze* zostaje powiązana z encją *KonkretnyMotocykl.* Uzyskany związek R8 jest związkiem wzajemnie jednoznacznym (1:1), ponieważ jeden konkretny motocykl o unikatowym numerze seryjnym może być na stanie tylko i wyłącznie w jednym oddziale. Związek ten nie wprowadza do naszej bazy niekompatybilności.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nazwa encji** | **Krotność** | **Związek** | | **Nazwa encji** | **Krotność** |
| Egzemplarze | 1..1 | Istnieje | R22 | KonkretnyMotocykl | 1..\* |
| KonkretnyMotocykl | 1..1 | Zamawiane | R7 | Zamowienia | 0..\* |
| KonkretnyMotocykl | 1..1 | JestDostepny | R8 | DostepnoscEgzemplarza | 0..1 |

Ponadto już na etapie projektowania modelu konceptualnego stworzyliśmy encję *Miasta*, aby uniknąć zależności nazwy *miasta* od *kodu pocztowego* w zbiorze encji *Klienci* oraz *Oddzialy.*Stworzyliśmy również encję *Silniki*, aby uniknąć ewentualnej redundancji danych oraz mieć możliwość uzyskania większej ilości informacji odnośnie stosowanych silników w danych motocyklach. Związek R6 powiązaną z utworzoną encją.

|  |  |  |
| --- | --- | --- |
| **Nazwa zbioru encji** | **Opis** | **Własności** |
| **Silniki** | Encja opisująca silnika. | Reprezentuje właściwości konkretnego silnika, takie jak jego typ, pojemność, moc, spalanie czy wykorzystywane paliwo. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nazwa zbioru encji** | **Atrybut** | **Opis** | **Nadrzędny zbiór encji** | **Typ danych** | **Wart. pusta** | **Wielo wart.** | **Wart. domyślna** | **Wart. trigger.** |
| *Silniki* | **idSilnika** | **Klucz główny** | - | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| Paliwo | Klucz obcy | RodzajePaliwa | Varchar2(20) | NIE | NIE | NIE | NIE |
| typ | Określa typ silnika, tj dwusuw czy czterosuw | - | Char(9) | NIE | NIE | NIE | NIE |
| pojSilnika | Pojemność silnika [cm3] | - | Number(4,0) | NIE | NIE | NIE | NIE |
| mocSilnika | Moc silnika [KM] | - | Number(3,0) | NIE | NIE | NIE | NIE |
| spalanie | Średnie spalanie silnika[l/100km] | - | Number(3,1) | NIE | NIE | NIE | NIE |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nazwa encji** | **Krotność** | **Związek** | | **Nazwa encji** | **Krotność** |
| Silniki | 1..1 | maSilnik | R23 | Egzemplarze | 1..\* |
| RodzajePaliwa | 1..1 | naPaliwo | R6 | Silniki | 1..\* |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nazwa zbioru encji** | **Atrybut** | **Dziedzina** | **Format** | **Zakres wartości** | |
| *Silnik* | typ | Ciąg znaków |  | | ‘dwusuw’, ‘czterosuw’ | |

Po przeprowadzeniu normalizacji, projektowana przez nas baza danych została doprowadzona do trzeciej postaci normalnej.

## Salon motocyklowy koncep.jpeg4.5 Schemat ER na poziomie modelu logicznego

## 4.6 Więzy integralności

Spełnienie warunku zachowania więzów integralności jest jednym z podstawowych zadań każdej bazy danych. W naszym projekcie przyjęliśmy następujące założenia.

### 4.6.1 Wymaganie obecności danych

Ustaliliśmy, że pola oznaczone jako NOT NULL muszą być określone. Natomiast pola takie jak *zatrudnionyDo, powod, zaliczka, uwagi, nrMieszkania, nrLokalu* nie muszą mieć wartości. Reszta pól musi mieć przypisaną wartość.

### 4.6.2 Integralność encji

Integralność encji zakłada, że klucz główny encji nie może przyjmować wartości pustej. Np. *idMotocyklu* musi być określone dla każdej krotki.

### 4.6.3 Integralność referencyjna

Modyfikacja klucza głównego krotki nadrzędnej implikuje zmiany we wszystkich krotkach podrzędnych(metoda Cascade). W przypadku usunięcia rekordu z tabeli nadrzędnej nie można usunąć rekordu z tabeli podrzędnej jeśli istnieje powiązany rekord (Metoda Restrict).

Wyjątkami są: jeśli usuniemy *Oddział* to chcemy również usunąć Dostępne egzemplarze w tym oddziale. (Metoda Cascade)

## 4.7 Proces denormalizacji

W konsekwencji, w pewnych okolicznościach konieczne jest zaakceptowanie utraty części korzyści płynących z całkowicie znormalizowanego projektu na rzecz wydajności systemu. Jest to uzasadnione tylko w przypadku, gdy przewiduje się, że system nie będzie w stanie spełnić wymagań w zakresie wydajności przetwarzania.

W naszej bazie danych odwołania do adresu klienta bądź oddziału pomijające atrybut *miasto*występują bardzo rzadko. Wybieramy więc drugą postać normalną i dokonujemy złączenia relacji *Klient* i *Miasta*,*Oddzialy* i *Miasta.* Usunięte zostały związki R9 i R10.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nazwa zbioru encji** | **Atrybut** | **Opis** | **Nadrzędny zbiór encji** | **Typ danych** | **Wart. pusta** | **Wielo wart.** | **Wart. domyślna** | **Wart. trigger.** |
| *Klienci* | **idKlienta** | **Klucz główny** | **-** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| imie | Imię klienta | - | Varchar2(20) | NIE | NIE | NIE | NIE |
| drugieImie | Drugie imię klienta | - | Varchar2(20) | TAK | NIE | NIE | NIE |
| nazwisko | Nazwisko klienta | - | Varchar2(20) | NIE | NIE | NIE | NIE |
| dataUrodzenia | Data urodzenia klienta | - | Date | NIE | NIE | NIE | NIE |
| kodPocztowy | Kod pocztowy | - | Number(5,0) | NIE | NIE | NIE | NIE |
| miasto | Miasto klienta | - | Varchar2(30) | NIE | NIE | NIE | NIE |
| ulica | Ulica klienta | - | Varchar2(30) | NIE | NIE | NIE | NIE |
| nrDomu | Numer domu klienta | - | Varchar2(4) | NIE | NIE | NIE | NIE |
| nrMieszkania | Numer mieszkania klienta | - | Number(3,0) | NIE | NIE | NIE | NIE |
| pesel | Numer PESEL klienta | - | Number(11,0) | TAK | NIE | NIE | NIE |
| nrTelefonu | Numer telefonu klienta | - | Number(9,0) | NIE | NIE | NIE | NIE |
| *Oddzialy* | **idOddzialu** | **Klucz główny** | **-** | **Number** | **NIE** | **NIE** | **NIE** | **TAK (od1 z krokiem co 1)** |
| kodPocztowy | Kod pocztowy | - | Number(5,0) | NIE | NIE | NIE | NIE |
| miasto | Miasto oddziału | - | Varchar2(30) | NIE | NIE | NIE | NIE |
| ulica | Ulica oddziału | - | Varchar2(30) | NIE | NIE | NIE | NIE |
| nrBudynku | Numer budynku oddziału | - | Varchar2(4) | NIE | NIE | NIE | NIE |
| nrLokalu | Numer lokalu oddziału | - | Number(3,0) | TAK | NIE | NIE | NIE |

# 5. Model fizyczny

## 5.1 Projekt transakcji

### 5.1.1 Perspektywa sprzedawcy:

* Wyszukiwanie motocykli na podstawie podanych parametrów (typ, model, marka itp.),
* Podawanie szczegółowych informacji na temat danego motocyklu,
* Wprowadzanie nowych motocykli,
* Wprowadzanie dostępnych motocykli w oddziale,
* Wprowadzanie danych klientów,
* Sprawdzanie dostępności motocykli,
* Organizowanie jazd próbnych,
* Realizowanie zamówień klientów.

### 5.1.2 Perspektywa kierownika

* Wprowadzanie nowych motocykli,
* Wprowadzanie danych klientów,
* Modyfikacja/usunięcie motocykli,
* Modyfikacja/usunięcie informacji o klientach,
* Wyszukiwanie informacji o odpowiednim motocyklu, części.

### 5.1.3 Perspektywa dyrektora

* Wprowadzanie danych o pracownikach, klientach, motocyklach,
* Modyfikowanie/usuwanie informacji o pracownikach, klientach,
* Modyfikowanie/usuwanie informacji o motocyklach,
* Wyszukiwanie informacji na temat pracowników, klientów oraz motocykli.

## 5.2 Dobór indeksów

Indeks jest to struktura danych umożliwiająca szybszy dostęp do konkretnych rekordów w pliku, a tym samym przyspieszenie realizacji zapytań. Najczęstszymi informacjami wyszukiwanymi w naszej bazie danych są oferowane motocykle, zamówienia i jazdy próbne obsługiwane przez danych pracowników, kupione motocykle przez danego klienta oraz sprzedane motocykle. Do usprawnienia operacji na powyższych danych użyliśmy indeksów, które wspomogą wyszukiwanie przedstawionych danych. Są one zamieszczone w skrypcie z podpunktu 5.3.

Jako, że należy utworzyć indeks dla klucza głównego danej relacji, a ponadto nie należy indeksować małej relacji. Ponadto należy utworzyć indeks pomocniczy dla kolumn, które nie są kluczami głównymi ani kluczami obcymi, ale mogą być używane w złożonych powiązaniach , to wybraliśmy następujące Indeksy:

|  |  |  |
| --- | --- | --- |
| **Nazwa indeksu** | **Tabela** | **Items** |
| Index1 | Egzemplarze | idMotocyklu |
| Index2 | Zamowienia | idZamowienia |
| Index3 | JazdyPprobne | idKlienta, nrSeryjny, idPracownika |
| Index4 | Klienci | idKlienta |
| Index5 | Oddzialy | idOddzialu |
| Index6 | Pracownicy | idPracownika |

## 5.3 Skrypt SQL zakładający bazę danych

-- Create sequences section -------------------------------------------------

CREATE SEQUENCE ID\_motocyklu\_seq

INCREMENT BY 1

START WITH 1

NOMAXVALUE

NOMINVALUE

CACHE 20

/

CREATE SEQUENCE ID\_miasta\_seq

INCREMENT BY 1

START WITH 1

NOMAXVALUE

NOMINVALUE

CACHE 20

/

CREATE SEQUENCE ID\_personelu\_seq

INCREMENT BY 1

START WITH 1

NOMAXVALUE

NOMINVALUE

CACHE 20

/

CREATE SEQUENCE ID\_klienta\_seq

INCREMENT BY 1

START WITH 1

NOMAXVALUE

NOMINVALUE

CACHE 20

/

CREATE SEQUENCE ID\_oddzialu\_seq

INCREMENT BY 1

START WITH 1

NOMAXVALUE

NOMINVALUE

CACHE 20

/

CREATE SEQUENCE ID\_zamowienia\_trig

INCREMENT BY 1

START WITH 1

NOMAXVALUE

NOMINVALUE

CACHE 20

/

CREATE SEQUENCE ID\_silnika\_trig

INCREMENT BY 1

START WITH 1

NOMAXVALUE

NOMINVALUE

CACHE 20

/

-- Create tables section -------------------------------------------------

-- Table Egzemplarze

CREATE TABLE Egzemplarze(

idMotocyklu Number NOT NULL,

marka Varchar2(20 ) NOT NULL,

model Varchar2(20 ) NOT NULL,

typ Varchar2(20 ) NOT NULL,

paliwo Varchar2(20 ) NOT NULL,

kolor Varchar2(20 ) NOT NULL,

idSilnika Number NOT NULL,

rokProdukcji Number(4,0) NOT NULL,

waga Number(4,0) NOT NULL

CONSTRAINT waga\_check CHECK (waga > 0),

szerokosc Number(3,0) NOT NULL

CONSTRAINT CheckConstraint1 CHECK (szerokosc > 0),

wysokosc Number(3,0) NOT NULL

CONSTRAINT CheckConstraint2 CHECK (wysokosc > 0),

dlugosc Number(3,0) NOT NULL

CONSTRAINT CheckConstraint3 CHECK (dlugosc > 0),

cenaDetaliczna Number(8,2) NOT NULL,

gwarancja Number(2,0) NOT NULL

CONSTRAINT CheckConstraint6 CHECK (gwarancja > 0),

opis Varchar2(255 ),

CONSTRAINT rokProdukcji\_check CHECK (rokProdukcji BETWEEN 1900 and 2300)

)

/

-- Create indexes for table Egzemplarze

CREATE INDEX Index1 ON Egzemplarze (idMotocyklu)

/

-- Add keys for table Egzemplarze

ALTER TABLE Egzemplarze ADD CONSTRAINT Key1 PRIMARY KEY (idMotocyklu)

/

-- Table Motocykle

CREATE TABLE Motocykle(

model Varchar2(20 ) NOT NULL,

marka Varchar2(20 ) NOT NULL

)

/

-- Add keys for table Motocykle

ALTER TABLE Motocykle ADD CONSTRAINT Key2 PRIMARY KEY (marka,model)

/

-- Table Producenci

CREATE TABLE Producenci(

marka Varchar2(20 ) NOT NULL,

kraj Varchar2(20 ) NOT NULL

)

/

-- Add keys for table Producenci

ALTER TABLE Producenci ADD CONSTRAINT Key3 PRIMARY KEY (marka)

/

-- Table Sprzedane

CREATE TABLE Sprzedane(

idZamowienia Number NOT NULL,

dataOdbioru Date DEFAULT SYSDATE NOT NULL

)

/

-- Add keys for table Sprzedane

ALTER TABLE Sprzedane ADD CONSTRAINT Key4 PRIMARY KEY (idZamowienia)

/

-- Table JazdyProbne

CREATE TABLE JazdyProbne(

idKlienta Number NOT NULL,

nrSeryjny Varchar2(20 ) NOT NULL,

idPracownika Number NOT NULL,

idOddzialu Number NOT NULL,

od Timestamp(0) NOT NULL,

do Timestamp(0) NOT NULL,

uwagi Varchar2(255 )

)

/

-- Create indexes for table JazdyProbne

CREATE INDEX Index3 ON JazdyProbne (idKlienta,nrSeryjny,idPracownika)

/

-- Add keys for table JazdyProbne

ALTER TABLE JazdyProbne ADD CONSTRAINT Key6 PRIMARY KEY (idKlienta,nrSeryjny)

/

-- Table Klienci

CREATE TABLE Klienci(

idKlienta Number NOT NULL,

imie Varchar2(20 ) NOT NULL,

drugieImie Varchar2(20 ),

nazwisko Varchar2(20 ) NOT NULL,

dataUrodzenia Date NOT NULL,

kodPocztowy Char(5 ) NOT NULL

CONSTRAINT CheckConstraint9 CHECK (kodPocztowy > 0),

miasto Varchar2(30 ) NOT NULL,

ulica Varchar2(30 ) NOT NULL,

nrDomu Varchar2(3 ) NOT NULL,

nrMieszkania Number(3,0)

CONSTRAINT CheckConstraint10 CHECK (nrMieszkania > 0),

pesel Char(11 )

CONSTRAINT CheckConstraint18 CHECK ((SUBSTR(pesel,1,2) BETWEEN '00' AND '99')

AND ((SUBSTR(pesel,3,2) BETWEEN '01' AND '12')

OR (SUBSTR(pesel,3,2) BETWEEN '21' AND '32'))

AND (SUBSTR(pesel,5,2) BETWEEN '01' AND '31')),

nrTelefonu Char(9 ) NOT NULL

CONSTRAINT CheckConstraint11 CHECK (nrTelefonu > 0)

)

/

-- Create indexes for table Klienci

CREATE INDEX Index4 ON Klienci (idKlienta)

/

-- Add keys for table Klienci

ALTER TABLE Klienci ADD CONSTRAINT Key7 PRIMARY KEY (idKlienta)

/

-- Table Oddzialy

CREATE TABLE Oddzialy(

idOddzialu Number NOT NULL,

kodPocztowy Char(5 ) NOT NULL

CONSTRAINT CheckConstraint13 CHECK (kodPocztowy > 0),

miasto Varchar2(30 ) NOT NULL,

ulica Varchar2(30 ) NOT NULL,

nrBudynku Varchar2(4 ) NOT NULL,

nrLokalu Number(3,0)

CONSTRAINT CheckConstraint14 CHECK (nrLokalu > 0)

)

/

-- Create indexes for table Oddzialy

CREATE INDEX Index5 ON Oddzialy (idOddzialu)

/

-- Add keys for table Oddzialy

ALTER TABLE Oddzialy ADD CONSTRAINT Key8 PRIMARY KEY (idOddzialu)

/

-- Table Personel

CREATE TABLE Personel(

idPracownika Number NOT NULL,

imie Varchar2(20 ) NOT NULL,

drugieImie Varchar2(20 ),

nazwisko Char(20 ) NOT NULL,

dataUrodzenia Date NOT NULL,

nrTelefonu Number(9,0) NOT NULL

CONSTRAINT CheckConstraint15 CHECK (nrTelefonu > 0)

)

/

-- Create indexes for table Personel

CREATE INDEX Index6 ON Personel (idPracownika)

/

-- Add keys for table Personel

ALTER TABLE Personel ADD CONSTRAINT idPersonelu PRIMARY KEY (idPracownika)

/

-- Table Stanowiska

CREATE TABLE Stanowiska(

funkcja Varchar2(20 ) NOT NULL

)

/

-- Add keys for table Stanowiska

ALTER TABLE Stanowiska ADD CONSTRAINT Key21 PRIMARY KEY (funkcja)

/

-- Table TypyMotocykli

CREATE TABLE TypyMotocykli(

typ Varchar2(20 ) NOT NULL

)

/

-- Add keys for table TypyMotocykli

ALTER TABLE TypyMotocykli ADD CONSTRAINT Key9 PRIMARY KEY (typ)

/

-- Table Kolory

CREATE TABLE Kolory(

kolor Varchar2(20 ) NOT NULL

)

/

-- Add keys for table Kolory

ALTER TABLE Kolory ADD CONSTRAINT Key10 PRIMARY KEY (kolor)

/

-- Table RodzajePaliwa

CREATE TABLE RodzajePaliwa(

paliwo Varchar2(20 ) NOT NULL

)

/

-- Add keys for table RodzajePaliwa

ALTER TABLE RodzajePaliwa ADD CONSTRAINT Key11 PRIMARY KEY (paliwo)

/

-- Table Kraje

CREATE TABLE Kraje(

kraj Varchar2(20 ) NOT NULL

)

/

-- Add keys for table Kraje

ALTER TABLE Kraje ADD CONSTRAINT Key13 PRIMARY KEY (kraj)

/

-- Table DostEgzemplarza

CREATE TABLE DostEgzemplarza(

nrSeryjny Varchar2(20 ) NOT NULL,

idOddzialu Number NOT NULL

)

/

-- Add keys for table DostEgzemplarza

ALTER TABLE DostEgzemplarza ADD CONSTRAINT Key14 PRIMARY KEY (nrSeryjny)

/

-- Table Zatrudnienia

CREATE TABLE Zatrudnienia(

idPracownika Number NOT NULL,

idOddzialu Number NOT NULL,

funkcja Varchar2(20 ) NOT NULL,

pensja Number(7,2) NOT NULL

CONSTRAINT CheckConstraint12 CHECK (pensja > 0),

zatrudnionyOd Date DEFAULT SYSDATE NOT NULL,

zatrudnionyDo Date

)

/

-- Add keys for table Zatrudnienia

ALTER TABLE Zatrudnienia ADD CONSTRAINT Key15 PRIMARY KEY (idOddzialu,idPracownika)

/

-- Table Zamowienia

CREATE TABLE Zamowienia(

idZamowienia Number NOT NULL,

idKlienta Number NOT NULL,

nrSeryjny Varchar2(20 ) NOT NULL,

idPracownika Number NOT NULL,

idOddzialu Number NOT NULL,

dataZamowienia Date DEFAULT SYSDATE NOT NULL,

doOdbioru Date NOT NULL,

cena Number(8,2) NOT NULL,

zaliczka Number(8,2) DEFAULT 0 NOT NULL,

CONSTRAINT CheckConstraint7 CHECK (cena > 0)

)

/

-- Create indexes for table Zamowienia

CREATE INDEX Index2 ON Zamowienia (idZamowienia)

/

-- Add keys for table Zamowienia

ALTER TABLE Zamowienia ADD CONSTRAINT Key18 PRIMARY KEY (idZamowienia)

/

-- Table Rezygnacje

CREATE TABLE Rezygnacje(

idZamowienia Number NOT NULL,

dataRezygnacji Date DEFAULT SYSDATE NOT NULL,

powod Varchar2(255 )

)

/

-- Add keys for table Rezygnacje

ALTER TABLE Rezygnacje ADD CONSTRAINT Key19 PRIMARY KEY (idZamowienia)

/

-- Table KonkretnyEgzemplarz

CREATE TABLE KonkretnyEgzemplarz(

nrSeryjny Varchar2(20 ) NOT NULL,

idMotocyklu Number NOT NULL,

dostepny Char(1 ) DEFAULT 't' NOT NULL

CONSTRAINT CheckConstraint16 CHECK (dostepny IN ('t','n','r'))

)

/

-- Add keys for table KonkretnyEgzemplarz

ALTER TABLE KonkretnyEgzemplarz ADD CONSTRAINT Key20 PRIMARY KEY (nrSeryjny)

/

-- Table Silniki

CREATE TABLE Silniki(

idSilnika Number NOT NULL,

paliwo Varchar2(20 ) NOT NULL,

typ Char(9 ) NOT NULL

CONSTRAINT CheckConstraint17 CHECK (typ IN ('dwusuw','czterosuw')),

pojSilnika Number(4,0) NOT NULL

CONSTRAINT CheckConstraint19 CHECK (pojSilnika > 0),

mocSilnika Number(3,0) NOT NULL

CONSTRAINT CheckConstraint20 CHECK (mocSilnika > 0),

spalanie Number(3,1) NOT NULL

CONSTRAINT CheckConstraint21 CHECK (spalanie > 0)

)

/

-- Add keys for table Silniki

ALTER TABLE Silniki ADD CONSTRAINT Key22 PRIMARY KEY (idSilnika)

/

-- Trigger for sequence ID\_motocyklu\_seq for column idMotocyklu in table Egzemplarze ---------

CREATE OR REPLACE TRIGGER ts\_Egzemplarze\_ID\_motocyklu\_\_0 BEFORE INSERT

ON Egzemplarze FOR EACH ROW

BEGIN

:new.idMotocyklu := ID\_motocyklu\_seq.nextval;

END;

/

CREATE OR REPLACE TRIGGER tsu\_Egzemplarze\_ID\_motocyklu\_0 AFTER UPDATE OF idMotocyklu

ON Egzemplarze FOR EACH ROW

BEGIN

RAISE\_APPLICATION\_ERROR(-20010,'Cannot update column idMotocyklu in table Egzemplarze as it uses sequence.');

END;

/

-- Trigger for sequence ID\_klienta\_seq for column idKlienta in table Klienci ---------

CREATE OR REPLACE TRIGGER ts\_Klienci\_ID\_klienta\_seq BEFORE INSERT

ON Klienci FOR EACH ROW

BEGIN

:new.idKlienta := ID\_klienta\_seq.nextval;

END;

/

CREATE OR REPLACE TRIGGER tsu\_Klienci\_ID\_klienta\_seq AFTER UPDATE OF idKlienta

ON Klienci FOR EACH ROW

BEGIN

RAISE\_APPLICATION\_ERROR(-20010,'Cannot update column idKlienta in table Klienci as it uses sequence.');

END;

/

-- Trigger for sequence ID\_oddzialu\_seq for column idOddzialu in table Oddzialy ---------

CREATE OR REPLACE TRIGGER ts\_Oddzialy\_ID\_oddzialu\_seq BEFORE INSERT

ON Oddzialy FOR EACH ROW

BEGIN

:new.idOddzialu := ID\_oddzialu\_seq.nextval;

END;

/

CREATE OR REPLACE TRIGGER tsu\_Oddzialy\_ID\_oddzialu\_seq AFTER UPDATE OF idOddzialu

ON Oddzialy FOR EACH ROW

BEGIN

RAISE\_APPLICATION\_ERROR(-20010,'Cannot update column idOddzialu in table Oddzialy as it uses sequence.');

END;

/

-- Trigger for sequence ID\_personelu\_seq for column idPracownika in table Personel ---------

CREATE OR REPLACE TRIGGER ts\_Personel\_ID\_personelu\_seq BEFORE INSERT

ON Personel FOR EACH ROW

BEGIN

:new.idPracownika := ID\_personelu\_seq.nextval;

END;

/

CREATE OR REPLACE TRIGGER tsu\_Personel\_ID\_personelu\_seq AFTER UPDATE OF idPracownika

ON Personel FOR EACH ROW

BEGIN

RAISE\_APPLICATION\_ERROR(-20010,'Cannot update column idPracownika in table Personel as it uses sequence.');

END;

/

-- Trigger for sequence ID\_zamowienia\_trig for column idZamowienia in table Zamowienia ---------

CREATE OR REPLACE TRIGGER ts\_Zamowienia\_ID\_zamowienia\_\_0 BEFORE INSERT

ON Zamowienia FOR EACH ROW

BEGIN

:new.idZamowienia := ID\_zamowienia\_trig.nextval;

END;

/

CREATE OR REPLACE TRIGGER tsu\_Zamowienia\_ID\_zamowienia\_0 AFTER UPDATE OF idZamowienia

ON Zamowienia FOR EACH ROW

BEGIN

RAISE\_APPLICATION\_ERROR(-20010,'Cannot update column idZamowienia in table Zamowienia as it uses sequence.');

END;

/

-- Trigger for sequence ID\_zamowienia\_trig for column idSilnika in table Silniki ---------

CREATE OR REPLACE TRIGGER ts\_Silniki\_ID\_zamowienia\_trig BEFORE INSERT

ON Silniki FOR EACH ROW

BEGIN

:new.idSilnika := ID\_zamowienia\_trig.nextval;

END;

/

CREATE OR REPLACE TRIGGER tsu\_Silniki\_ID\_zamowienia\_trig AFTER UPDATE OF idSilnika

ON Silniki FOR EACH ROW

BEGIN

RAISE\_APPLICATION\_ERROR(-20010,'Cannot update column idSilnika in table Silniki as it uses sequence.');

END;

/

-- Wlasne triggery zapewniajace spojnosc i integralnosc bazy danych ---------------

CREATE TRIGGER ustawNiedostepny BEFORE INSERT ON Zamowienia  
FOR EACH ROW BEGIN  
 UPDATE KonkretnyEgzemplarz SET dostepny = 'n' WHERE nrseryjny=(:NEW.nrSeryjny);   
END;  
/  
CREATE TRIGGER ustawDostepny BEFORE INSERT ON Rezygnacje   
FOR EACH ROW BEGIN  
 UPDATE KonkretnyEgzemplarz SET dostepny = 't' WHERE nrSeryjny=(  
 SELECT nrSeryjny FROM Zamowienia WHERE idZamowienia=(:NEW.idZamowienia));  
END;  
/  
CREATE TRIGGER dostEgzemplarz BEFORE INSERT ON dostEgzemplarza  
FOR EACH ROW BEGIN  
 UPDATE KonkretnyEgzemplarz SET dostepny = 'n' WHERE nrseryjny=(:NEW.nrSeryjny);  
END;  
/

-- Create relationships section -------------------------------------------------   
ALTER TABLE Motocykle ADD CONSTRAINT produkuje FOREIGN KEY (marka) REFERENCES Producenci (marka)  
/  
ALTER TABLE JazdyProbne ADD CONSTRAINT bioraUdzial FOREIGN KEY (idKlienta) REFERENCES Klienci (idKlienta)  
/  
ALTER TABLE Egzemplarze ADD CONSTRAINT jestTypu FOREIGN KEY (typ) REFERENCES TypyMotocykli (typ)  
/  
ALTER TABLE Producenci ADD CONSTRAINT pochodziZ FOREIGN KEY (kraj) REFERENCES Kraje (kraj)  
/  
ALTER TABLE Egzemplarze ADD CONSTRAINT ma FOREIGN KEY (marka, model) REFERENCES Motocykle (marka, model)  
/  
ALTER TABLE DostEgzemplarza ADD CONSTRAINT posiada FOREIGN KEY (idOddzialu) REFERENCES Oddzialy (idOddzialu) ON DELETE CASCADE  
/  
ALTER TABLE Silniki ADD CONSTRAINT naPaliwo FOREIGN KEY (paliwo) REFERENCES RodzajePaliwa (paliwo)  
/  
ALTER TABLE Egzemplarze ADD CONSTRAINT jestKoloru FOREIGN KEY (kolor) REFERENCES Kolory (kolor)  
/  
ALTER TABLE Zatrudnienia ADD CONSTRAINT jestZatrudniony FOREIGN KEY (idPracownika) REFERENCES Personel (idPracownika)  
/  
ALTER TABLE Zatrudnienia ADD CONSTRAINT zatrudnia FOREIGN KEY (idOddzialu) REFERENCES Oddzialy (idOddzialu)  
/  
ALTER TABLE JazdyProbne ADD CONSTRAINT jestWypozyczany FOREIGN KEY (nrSeryjny) REFERENCES DostEgzemplarza (nrSeryjny)  
/  
ALTER TABLE Rezygnacje ADD CONSTRAINT rezygnacja FOREIGN KEY (idZamowienia) REFERENCES Zamowienia (idZamowienia)  
/  
ALTER TABLE Zamowienia ADD CONSTRAINT zamawiaja FOREIGN KEY (idKlienta) REFERENCES Klienci (idKlienta)  
/  
ALTER TABLE Sprzedane ADD CONSTRAINT sprzedawane FOREIGN KEY (idZamowienia) REFERENCES Zamowienia (idZamowienia)  
/  
ALTER TABLE KonkretnyEgzemplarz ADD CONSTRAINT istnieje FOREIGN KEY (idMotocyklu) REFERENCES Egzemplarze (idMotocyklu)  
/  
ALTER TABLE Zamowienia ADD CONSTRAINT zamawiane FOREIGN KEY (nrSeryjny) REFERENCES KonkretnyEgzemplarz (nrSeryjny)  
/  
ALTER TABLE DostEgzemplarza ADD CONSTRAINT jestDostepny FOREIGN KEY (nrSeryjny) REFERENCES KonkretnyEgzemplarz (nrSeryjny)  
/  
ALTER TABLE JazdyProbne ADD CONSTRAINT organizuje FOREIGN KEY (idOddzialu, idPracownika) REFERENCES Zatrudnienia (idOddzialu, idPracownika)  
/  
ALTER TABLE Zamowienia ADD CONSTRAINT zamawia FOREIGN KEY (idOddzialu, idPracownika) REFERENCES Zatrudnienia (idOddzialu, idPracownika)  
/  
ALTER TABLE Zatrudnienia ADD CONSTRAINT naStanowisku FOREIGN KEY (funkcja) REFERENCES Stanowiska (funkcja)  
/  
ALTER TABLE Egzemplarze ADD CONSTRAINT maSilnik FOREIGN KEY (idSilnika) REFERENCES Silniki (idSilnika);

## 5.4 Przykłady zapytań i poleceń SQL odnoszących się do bazy danych

1. Dodanie koloru do tablicy Kolory.

*insert into Kolory values('Fioletowy');*

analogicznie dodajemy inne kolory zapytaniem:

*select Kolor from Kolory where Kolor like 'F%';*

Wyświetlamy wszystkie kolory zaczynające sięna literę F w tym przypadku SQL Developer wypisał kolor Fioletowy.

1. Polecenie UPDATE

Wpisujemy przykładową wartość encji egzemplarze poleceniem:

*insert into Egzemplarze values ('5', 'Honda', 'jakas', 'sportowy', 'diesel', 'zielony', '1991', '1788', '142', '1210','11','212','145','75000','2');*

Następnie aktualizujemy jedną z danych. Wagę mnożymy razy 1.1, rok produkcji ustawiamy o jeden młodszy, dla motocykli firmy Honda.

*UPDATE Egzemplarze SET Waga=Waga\*1.1, Rok\_produkcji= Rok\_produkcji + 1 WHERE Marka=’Honda’;*

1. Wyświetlenie wszystkich egzemplarzy motocykli (atrybutów ID\_motocyklu, Marka, Model, Typ ) o pojemności silnika pomiędzy 500 a 1500

*SELECT ID\_motocyklu, Marka, Model, Typ FROM Egzemplarze WHERE Pojemnosc\_silnika BETWEEN 500 AND 1500;*

1. Usunięcie przykładowej wartości z relacji Klienci

*DELETE FROM Klienci WHERE Nazwisko IN (‘Kowalski’);*

1. Wyświetlenie w porządku rosnącym według roku produkcji wszystkich egzemplarzy motocykli wraz ze wszystkimi atrybutami

*SELECT \* FROM Egzemplarze ORDER BY Rok\_produkcji ASC;*

# 6. Bibliografia

[1] T.Connoly, C.Begg, *„Systemy baz danych. Praktyczne metody projektowania, implementacji i zarządzania - tom 1”*, Wydawnictwo RM, Warszawa 2004

[2] M.Kowalczyk, *Slajdy wykładowe do przedmiotu Wstęp do Baz Danych,* Warszawa 2012