



**POLITECHNIKA LUBELSKA
WYDZIAŁ ELEKTROTECHNIKI
I INFORMATYKI**

**KIERUNEK STUDIÓW
INFORMATYKA**

Przedmiot: Wprowadzenie do systemów baz danych

Raport z wykonania projektu pt.

Tytuł projektu:

Baza danych zastosowana w systemie wspomagania funkcjonowania tzw. car-sharingu.

Autorzy:

***Michał Grzeszuk,
Mateusz Izdebski***

Lublin, 2020



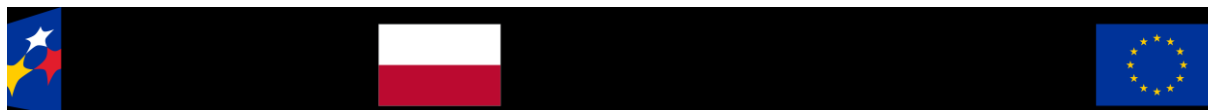
ROZDZIAŁ 1. OPIS WYBRANEGO OBSZARU RZECZYWISTOŚCI ORAZ WSKAZANIE PROBLEMU, KTÓRY ZOSTANIE ROZWIĄZANY PRZY POMOCY SYSTEMU INFORMATYCZNEGO WYKORZYSTUJĄCEGO PROJEKTOWANĄ BAZĘ DANYCH

Dokonaj ogólnej prezentacji wybranej rzeczywistości, wskazując procesy w niej realizowane oraz obiekty uczestniczące w tych procesach. Określ logiczną kolejność realizacji tych procesów.

Określ problem lub problemy jakie należy rozwiązać w wybranym obszarze rzeczywistości poprzez wdrożenie systemu informatycznego, w którym zastosujesz projektowaną bazę danych.

Wybrana rzeczywistość tworzy wypożyczalnia samochodów (car-sharing), w której oferowane są różne rodzaje samochodów. Wypożyczalnia prowadzi działalność na terenie Polski i innych krajów. Klienci mogą dokonywać wypożyczeń w różnych miastach. W celu ułatwienia klientom wypożyczania aut jakie chcą zostały one sklasyfikowane według różnych kategorii. Wypożyczalnia oferuje kanały sprzedaży, takie jak strona webowa oraz aplikacja mobilna. Składając zamówienie na auto klient musi podać, które auto chce wybrać i na jaki okres czasu chce je wypożyczyć (tylko w ilości dni). Każdy klient musi posiadać swój własny profil by móc wypożyczyć auto. Każde wypożyczenie otrzymuje swój unikalny numer identyfikacyjny.

Poprzez wdrożenie systemu informatycznego będziemy wiedzieć jakie auta są na wypożyczeniu, kiedy będą dostępne znowu by klient mógł złożyć zamówienie na konkretną datę, auto oraz na długość terminu wypożyczenia jaki chce. Będziemy również mogli skategoryzować auta po ich cenach wypożyczeń za dzień i ich cechach.



ROZDZIAŁ 2. SFORMUŁOWANIE CELU BUDOWY SYSTEMU INFORMATYCZNEGO, WYKORZYSTUJĄCEGO PROJEKTOWANĄ BAZĘ DANYCH, ORAZ PREZENTACJA WYMAGAŃ FUNKCJONALNYCH STAWIANYCH SYSTEMOWI

Określ cel budowy systemu informatycznego, wykorzystującego projektowaną bazę danych. W definicji celu pamiętaj o zastosowaniu zasady SMART, tj. zapewnij, aby zdefiniowany cel był skonkretyzowany, mierzalny, osiągalny, istotny i określony w czasie.

Przedstaw wymagania funkcjonalne jakie powinny być spełnione przez projektowany system IT, aby możliwe było osiągnięcie celu jego budowy. Jeśli system ma być wykorzystywany przez różnego rodzaju użytkowników (np. klient, pracownik, administrator systemu), dokonaj podziału tych wymagań uwzględniając specyficzne potrzeby każdego z nich.

Omawiana baza danych zawiera dane nt. fikcyjnej firmy zajmującej się Car-Sharingu. Mamy w niej, określone datą, rekordy wypożyczeń, które są powiązane relacyjnie z danymi umieszczonymi w odpowiednich tabelach. Zawierają one informacje nt. wypożyczanych aut, kierowców oraz pracowników obsługujących “transakcję” i poszczególnych oddziałów sieci wypożyczalni. Pozwala to określić, kiedy, komu, przez kogo i gdzie została wypożyczony dany samochód. Tego typu baza danych bardzo ułatwiałaby obsługę sieci wypożyczalni – poza oczywistymi zastosowaniami jak przechowywanie danych pracowników, aut oraz kierowców w różnych oddziałach – umożliwia ona np. szybki wgląd w rekordy wypożyczeń, których termin oddania już minął. Możemy wtedy szybko namierzyć kierowcę, który jest za nią odpowiedzialny oraz pracownika, który to auto wydał. Kolejnym przykładowym zastosowaniem takiej bazy jest wgląd w listę pracowników i kierowców, którzy mieli styczność w danym okresie z danym autem w przypadku jej uszkodzenia. Tego typu baza danych jest niezbędna do prowadzenia wypożyczalni, szczególnie w przypadku zajmowania się wieloma jej oddziałami, a korzystanie z niej w formie elektronicznej jest, w dzisiejszych czasach - ze względu na dogodność, niezawodność działania oraz szybkość przeglądania rekordów - właściwie koniecznością. Utworzenie bazy danych takiej firmy ma za równo pomóc pracownikom jak i klientom. Pracownicy będą z niej czerpać wiele korzyści jakie płyną z łatwej obsługi i szybkiego dostępu do wszystkich informacji na temat kierowców oraz aut, mogą w każdej chwili określić kto, kiedy, gdzie i jakie auto wypożyczył oraz kiedy musi je zwrócić. Klienci za pomocą względu do bazy danych mają dostęp do informacji na temat każdego auta, żeby dowiedzieć się szczegółów na temat obiektu jaki wypożyczają w celu dokładnego rozeznania się w produkcie jaki chce wypożyczyć. Dodatkowo dla klienta plusem może być świadomość, że dzięki bazie danych mniejsze prawdopodobieństwo jest zgubienia bądź złego zapisu danych osobowych co mogło by potem źle odbić się w razie jakiegoś wypadku bądź incydentu na drodze. Kolejną grupą, która skorzysta z naszej bazy danych jest policja, która podczas ówczesnie podanej sytuacji np. wypadku użyje bazy danych wypożyczalni na temat auta i kierowcy, aby pomóc ustalić informacje i dokładne szczegóły wypadku.



ROZDZIAŁ 3. SZCZEGÓŁOWY OPIS PROCESÓW REALIZOWANYCH W WYBRANEJ RZECZYWISTOŚCI, ZALEŻNOŚCI MIĘDZY NIMI ORAZ OBIEKTÓW W NICH UCZESTNICZĄCYCH

Przedstaw szczegółowo procesy jakie będą zachodzić w wybranej rzeczywistości. Jeżeli dany proces ma charakter złożony, dokonaj jego podziału na działania (akcje) elementarne oraz opisz kolejność ich realizacji.

Opisz zależności pomiędzy poszczególnymi procesami, wskazując logiczną kolejność ich realizacji.

Wskaż (nazwij) obiekty, które będą uczestniczyć w realizacji poszczególnych procesów.

Cały proces „Car-sharingu” rozpocznie się w momencie, kiedy klient poda swoje dane podczas wypożyczania auta, które zostaną wprowadzone do systemu bazy danych wypożyczalni. Klient podaje swoje dane osobowe jak i wybiera auto jakie chce wypożyczyć. Do bazy danych wprowadzane są także dane dotyczące karalności klienta m.in. punkty karne. Można płacić tylko kartą od razu przy wprowadzaniu danych. Przypisane są dane karty do danych klienta. Podczas wybierania auta z bazy danych aut sprawdzana jest numer oddziału, w którym klient się znajduje i sprawdzenie dostępności auta którego poszukuje. Auto ma przypisane numer oddziału, jego adres (kraj, miasto, numer lokalu) oraz dostępność w tym oddziale. Jeżeli w danym punkcie auto jest dostępne, klient podał swoje dane, jego dane na temat karalności nie przeciw skazują wypożyczeniu oraz numer karty jest prawidłowy pojazd zostaje od danej daty i godziny przypisany do klienta. Od tej pory klient staje się odpowiedzialny za auto i ponosi wszelkie szkody i bierze pełną odpowiedzialność za pojazd (Do wypożyczenia nie dojdzie w przypadku zadłużenia lub braku udowodnienia źródła stałego zarobku). W bazie danych auta nalicza się mu przebyte kilometry przez klienta oraz czas jaki był w jego posiadaniu w dniach. Status zamówienia opisuje czy pojazd jest u klienta, w oddziale bądź w serwisie i nie jest dostępny dla klienta. Podczas każdego wypożyczenia i wpisywania danych klienta do bazy ma ogromne znaczenie kto w danym momencie wpisuje te dane. Tabela pracowników opisuje osobę, która jest odpowiedzialna za daną transakcję. Dane pracownika takie jak imię, nazwisko, pesel i numer oddziału w jakim aktualnie pracuje widnieją na umowie o wypożyczeniu auta. Za równo klient jak i pracownik są odpowiedzialni za dane auto podczas wypożyczenia. Baza danych w firmie typu „Car-Sharingu” ma na celu pomóc i ułatwić przechowywanie danych o autach i klientach. Danych tych jest bardzo dużo i sama baza danych ułatwia i przechowywanie takich danych, ale też łatwy dostęp do każdego pojedynczego źródła informacji na temat klienta, jego danych auta oraz jego danych technicznych, ale także mamy informacje o oddziale i pracownikach jacy w danej chwili pracują i odpowiadają za legalne i bezpieczne transakcje pojazdami.

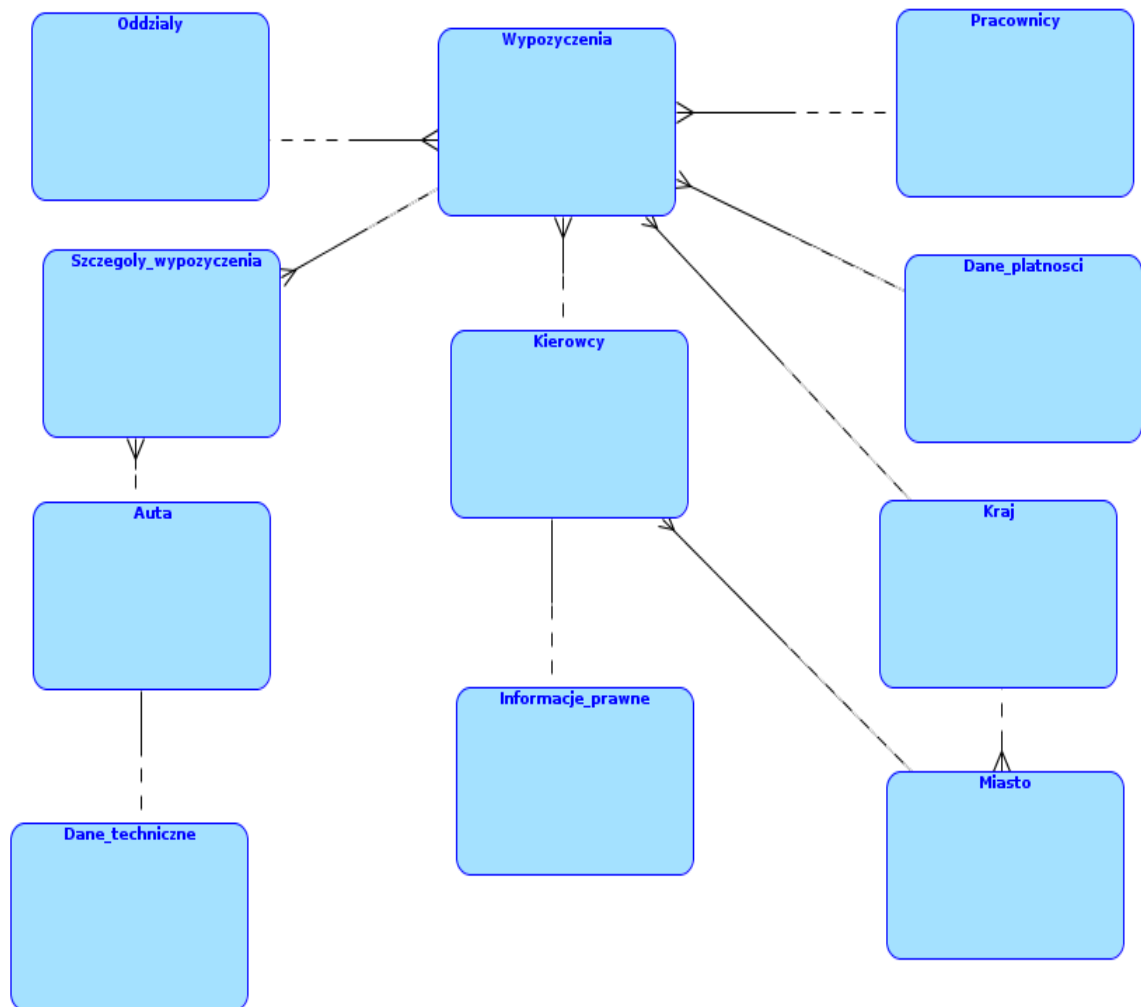


ROZDZIAŁ 4. MODEL KONCEPTUALNY PROJEKTOWANEJ BAZY DANYCH

Przedstaw w formie graficznej model konceptualny projektowanej bazy danych.

Wymień i scharakteryzuj encje, które będą uwzględnione w modelowanej bazie danych. Podaj ich nazwy oraz opisz ogólnie, jakie obiekty rzeczywistości będą one opisywały.

Opisz rodzaje związków pomiędzy poszczególnymi encjami.



Encje, które będą uwzględnione w modelowanej bazie danych to:

- wypożyczenia (opisuje każde transakcje auta dla klienta)**
- kierowcy (opisuje dane klientów)**
- auta (opisuje dane aut wypożyczalni)**
- pracownicy (opisuje pracowników, którzy są odpowiedzialni za transakcje auta)**
- dane płatności (będzie zawierać dane płatności takie jak numer karty, kwota)**
- kraj (będzie zawierać nazwę kraju)**
- miasto (zawiera informacje o mieście i dokładnym adresie oddziału)**
- informacje prawne (będzie zawierać informacje prane danego kierowcy takie jak np. punkty karne**
- dane techniczne (przechowuje dane techniczne każdego auta)**
- oddziały (informuje o adresie oddziału, w którym zostało wypożyczone auto)**
- szczegóły wypożyczenia (zawiera dane transakcji)**

Dane encje mają między sobą relacje, aby sprawnie wszystko działało. Do każdego wypożyczenia przypisane są dane klienta dane pracownika dane auta oraz status, gdzie znajduje się auto. Dzięki tej bazie danych w łatwy sposób możemy przechowywać dane na temat kierowców, którzy wynajmują auta i w łatwy sposób można określić, czy mogą wypożyczyć dane auto i kto to auto im wypożyczył.



ROZDZIAŁ 5. MODEL ZWIĄZKÓW ENCJI PROJEKTOWANEJ BAZY DANYCH

Przedstaw w formie graficznej model logiczny (związków encji) projektowanej bazy danych.

Wymień i opisz znaczenie atrybutów dla poszczególnych encji występujących w prezentowanym modelu logicznym bazy danych. Podaj nazwę i rodzaj każdego atrybutu encji.

Określ dla każdego atrybutu encji jego dziedzinę, maksymalny rozmiar danych, opcjonalność lub konieczność wystąpienia wartości określonej oraz inne ograniczenia integralnościowe, np. unikalność wartości atrybutu bądź też zawężenie jego dziedziny.

Dokonaj szczegółowego opisu każdego związku pomiędzy poszczególnymi encjami.

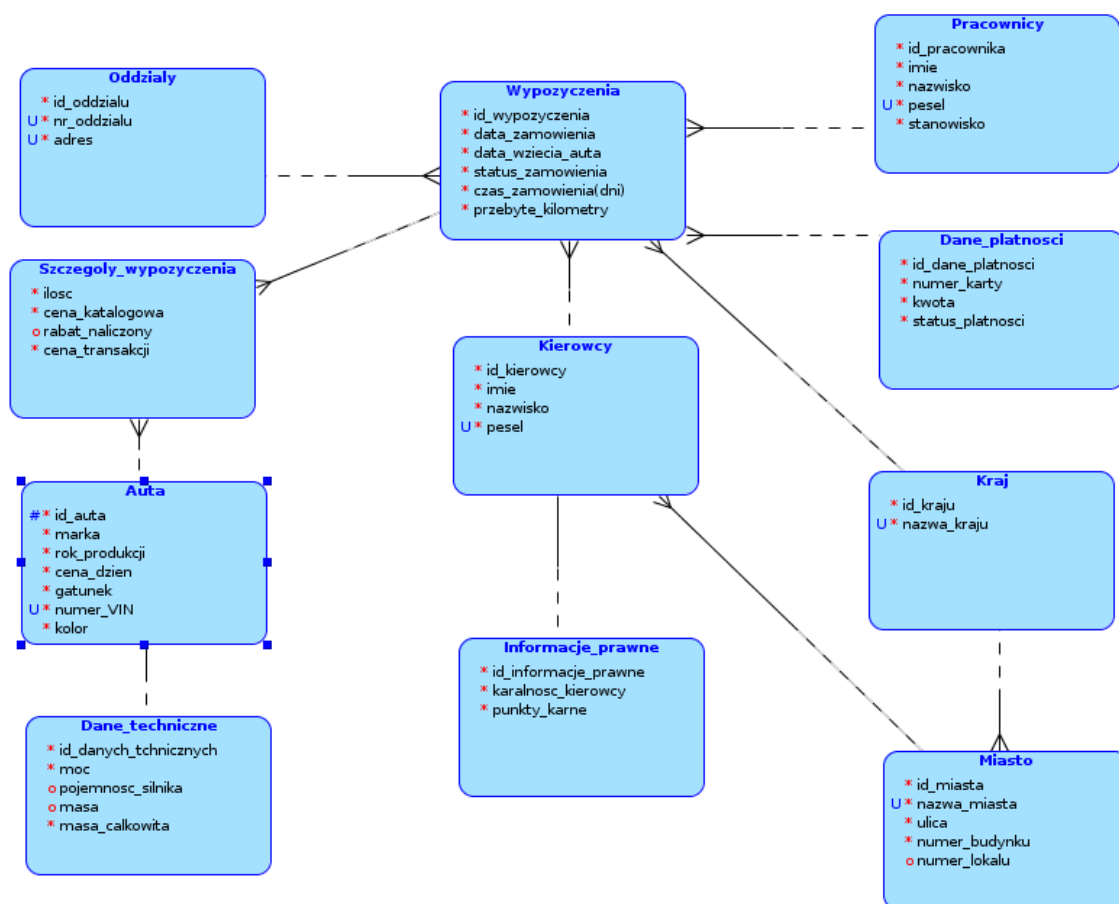


Tabela pracownicy:

id_pracownika	NUMBER(10) NOT NULL,	PRIMARY KEY
imie	VARCHAR2(45) NOT NULL,	
nazwisko	VARCHAR2(45) NOT NULL,	
pesel	NUMBER(11) NOT NULL,	UNIQUE
stanowisko	VARCHAR2(25) NOT NULL	

Tabela dane_platnosci:

id_dane_platosci	NUMBER(10) NOT NULL,	PRIMARY KEY
numer_karty	NUMBER(24) NOT NULL,	
kwota	NUMBER(6) NOT NULL,	
status_platnosci	VARCHAR2(25) NOT NULL	

Tabela kraj:

id_kraju	NUMBER(10) NOT NULL,	PRIMARY KEY
nazwa_kraju	VARCHAR2(25) NOT NULL	UNIQUE

Tabela informacje_prawne:

id_informacje_prawne	NUMBER(10) NOT NULL,	PRIMARY KEY
karalnosc_kierowcy	VARCHAR2(3) NOT NULL,	
punkty_karne	NUMBER(2) NOT NULL	

Tabela dane_techiczne:

id_danych_techicznych	NUMBER(10) NOT NULL,	PRIMARY KEY
moc	NUMBER(4) NOT NULL,	
pojemnosc_silnika	NUMBER(4),	
masa	NUMBER(5),	
masa_calkowita	NUMBER(5) NOT NULL	

Tabela oddzialy:

id_oddzialu	NUMBER(10) NOT NULL,	PRIMARY KEY
nr_oddzialu	NUMBER(3) NOT NULL,	UNIQUE
adres	VARCHAR2(45) NOT NULL	UNIQUE

Tabela auta:

id_auta	NUMBER(10) NOT NULL,	PRIMARY KEY
marka	VARCHAR2(25) NOT NULL,	
rok_produkcji	NUMBER(4) NOT NULL,	
cena_dzien	NUMBER(7, 2) NOT NULL,	
gatunek	VARCHAR2(25) NOT NULL,	
numer_vin	NUMBER(25) NOT NULL,	UNIQUE
kolor	VARCHAR2(20) NOT NULL,	
id_danych_techicznych	NUMBER(10) NOT NULL	FOREIGN KEY

Tabela miasto:

id_miasta	NUMBER(10) NOT NULL,	PRIMARY KEY
nazwa_miasta	VARCHAR2(25) NOT NULL,	UNIQUE
ulica	VARCHAR2(30) NOT NULL,	
numer_budynku	NUMBER(3) NOT NULL,	
numer_lokalu	NUMBER(3),	



id_kraju	NUMBER(10) NOT NULL	FOREIGN KEY
Tabela kierowcy:		
id_kierowcy	NUMBER(10) NOT NULL,	PRIMARY KEY
imie	VARCHAR2(45) NOT NULL,	
nazwisko	VARCHAR2(45) NOT NULL,	
pesel	NUMBER(11) NOT NULL,	UNIQUE
id_informacje_prawne	NUMBER(10) NOT NULL,	FOREIGN KEY
id_miasta	NUMBER(10) NOT NULL	FOREIGN KEY

Tabela wypożyczenia:

id_wypozyczenia	NUMBER(10) NOT NULL,	PRIMARY KEY
data_zamowienia	DATE NOT NULL,	
data_wziecia_auta	DATE NOT NULL,	
status_zamowienia	VARCHAR2(25) NOT NULL,	
"czas_zamowienia_(dni)"	NUMBER(3) NOT NULL,	
przebyte_kilometry	NUMBER(4) NOT NULL,	
id_pracownika	NUMBER(10) NOT NULL,	FOREIGN KEY
id_dane_platosci	NUMBER(10) NOT NULL,	FOREIGN KEY
id_kierowcy	NUMBER(10) NOT NULL,	FOREIGN KEY
id_oddzialu	NUMBER(10) NOT NULL,	FOREIGN KEY
id_kraju	NUMBER(10) NOT NULL	FOREIGN KEY

Tabela szczegoly_wypozyczenia:

id_wypozyczenia	NUMBER(10) NOT NULL,	FOREIGN KEY
id_auta	NUMBER(10) NOT NULL,	FOREIGN KEY
ilosc	NUMBER(2) NOT NULL,	
cena_katalogowa	NUMBER(7, 2) NOT NULL,	
rabat_naliczony	NUMBER(7, 2),	
cena_transakcji	NUMBER(7, 2) NOT NULL	

Relacje między encjami:

Pracownicy: Wypożyczenia (jeden pracownik obliguje kilka wypożyczeń)

Dane płatności: Wypożyczenia (jedna karta można zapłacić za kilka wypożyczeń)

Kraj: Wypożyczenia (w jednym kraju można dokonać kilka wypożyczeń)

Miasto: Kraj (Wkraje jest wiele miast z oddziałami)

Miasto: Kierowcy (W jednym mieście jest wiele kierowców)

Dane prawne: Kierowcy (każdy kierowca posiada jedno informacje prawne o sobie)

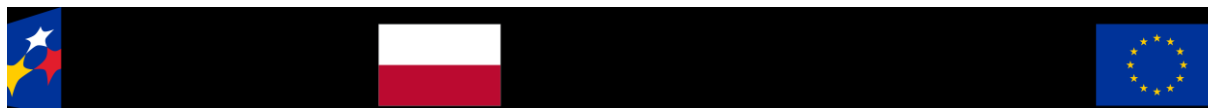
Wypożyczenia: Kierowca (Kierowca może wypożyczyć kilka aut)

Oddziały: Wypożyczenia (Jeden oddział odpowiada z kilka transakcji)

Szczegóły Wypożyczenia: Wypożyczenia (Jedno wypożyczenia ma wiele danych)

Aut: Szczegóły Wypożyczenia (Informacje o wypożyczeniu są przypisane do każdego auta)

Dane techniczne: Aut (Każde auto posiada przypisane do siebie informacje)

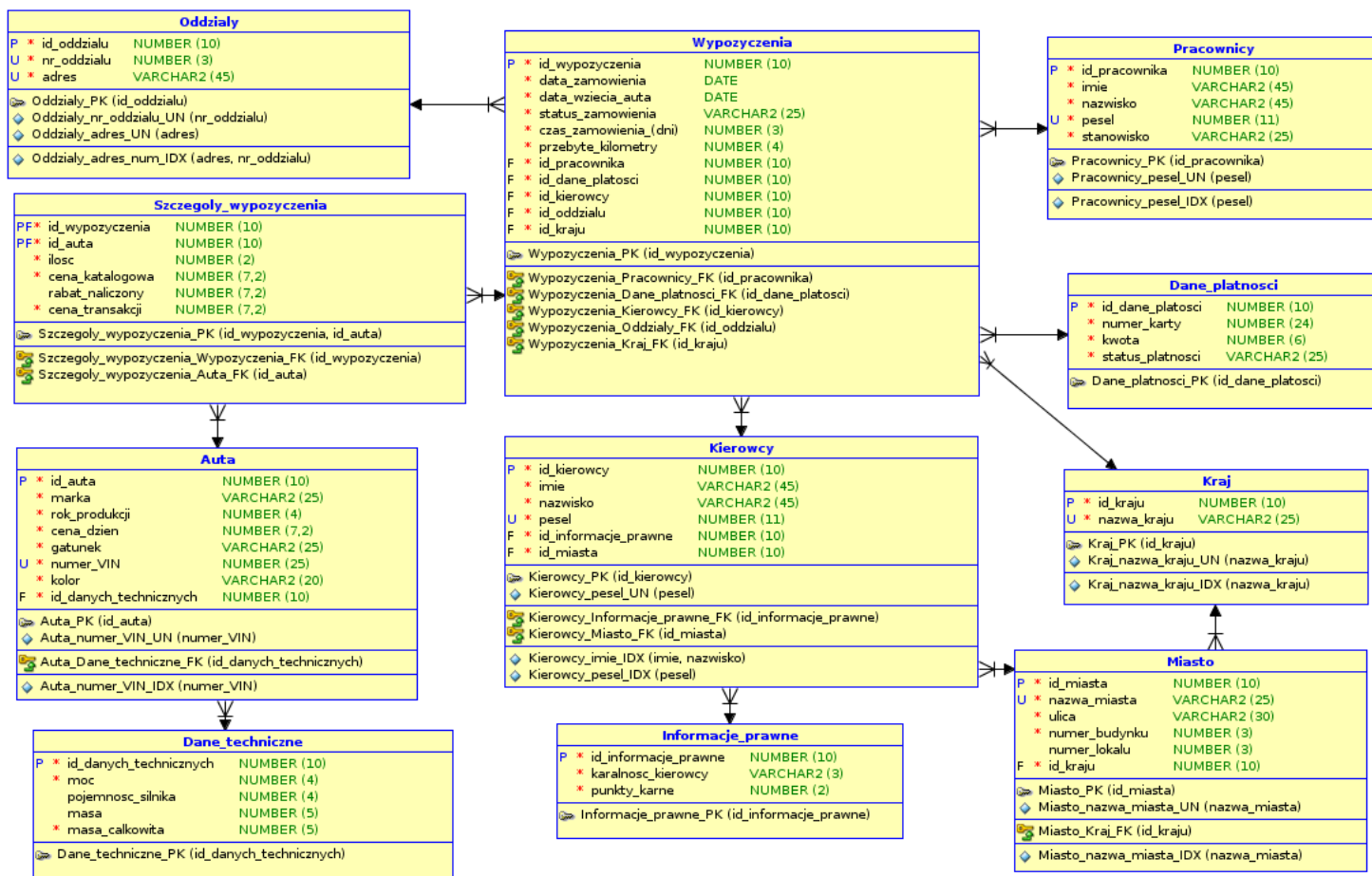


ROZDZIAŁ 6. MODEL RELACYJNY PROJEKTOWANEJ BAZY DANYCH

Przedstaw w formie graficznej model relacyjny projektowanej bazy danych.

Określ ograniczenia integralności we zastosowane dla poszczególnych kolumn poszczególnych tabeli. Podaj nazwy tych ograniczeń oraz ich rodzaj.

Wymień indeksy, jakie należy utworzyć dla poszczególnych tabel. Podaj nazwę indeksu, jego rodzaj oraz kolumny użyte do jego zbudowania.



Ograniczenia integralności:

Tabela pracownicy:

-pracownicy_pk PRIMARY KEY (id_pracownika)

-pracownicy_pesel_un UNIQUE (pesel)

Tabela dane_platnosci:

-dane_platnosci_pk PRIMARY KEY (id_dane_platosci)

Tabela kraj:

-kraj_pk PRIMARY KEY (id_kraju)

-kraj_nazwa_kraju_un UNIQUE (nazwa_kraju)

Tabela informacje_prawne:

-informacje_prawne_pk PRIMARY KEY (id_informacje_prawne)

Tabela dane_teczniczne:



-dane_techiczne_pk PRIMARY KEY (id_danych_techicznych)

Tabela oddzialy:

-oddzialy_pk PRIMARY KEY (id_oddzialu)

-oddzialy_nr_oddzialu_un UNIQUE (nr_oddzialu)

-oddzialy_adres_un UNIQUE (adres)

Tabela auta:

-auta_pk PRIMARY KEY (id_auta)

-auta_numer_vin_un UNIQUE (numer_vin)

-auta_dane_techiczne_fk FOREIGN KEY (id_danych_techicznych)
REFERENCES dane_techiczne (id_danych_techicznych)

Tabela miasto:

-miasto_pk PRIMARY KEY (id_miasta)

-miasto_nazwa_miasta_un UNIQUE (nazwa_miasta)

-miasto_kraj_fk FOREIGN KEY (id_kraju)
REFERENCES kraj (id_kraju)

Tabela kierowcy:

-kierowcy_pk PRIMARY KEY (id_kierowcy)

-kierowcy_pesel_un UNIQUE (pesel)

-kierowcy_informacje_prawne_fk FOREIGN KEY (id_informacje_prawne)
REFERENCES informacje_prawne (id_informacje_prawne)

-kierowcy_miasto_fk FOREIGN KEY (id_miasta)
REFERENCES miasto (id_miasta)

Tabela wypożyczenia:

-wypozyczenia_pk PRIMARY KEY (id_wypozyczenia)

-wypozyczenia_dane_platnosci_fk FOREIGN KEY (id_dane_platosci)
REFERENCES dane_platnosci (id_dane_platosci)

-wypozyczenia_kierowcy_fk FOREIGN KEY (id_kierowcy)
REFERENCES kierowcy (id_kierowcy)

-wypozyczenia_kraj_fk FOREIGN KEY (id_kraju)
REFERENCES kraj (id_kraju)

-wypozyczenia_oddzialy_fk FOREIGN KEY (id_oddzialu)
REFERENCES oddzialy (id_oddzialu)

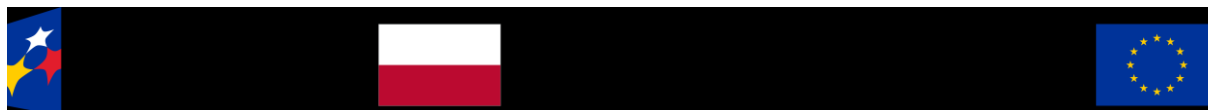
-wypozyczenia_pracownicy_fk FOREIGN KEY (id_pracownika)
REFERENCES pracownicy (id_pracownika)

Tabela szczegoly_wypozyczenia:

-szczegoly_wypozyczenia_pk PRIMARY KEY (id_wypozyczenia,
id_auta)

-szczegoly_wyp_auta_fk FOREIGN KEY (id_auta)
REFERENCES auta (id_auta)

-szczegoly_wyp_wypozyczenia_fk FOREIGN KEY (id_wypozyczenia)
REFERENCES wypozyczenia (id_wypozyczenia)



Indeksy:

Tabela pracownicy:

-pracownicy_pesel_idx – pesel ASC

Tabela kraj:

-kraj_nazwa_kraju_idx - nazwa_kraju ASC

Tabela oddzialy:

-oddzialy_adres_num_idx -adres ASC, nr_oddzialu ASC

Tabela auta:

-auta_numer_vin_idx - numer_vin ASC

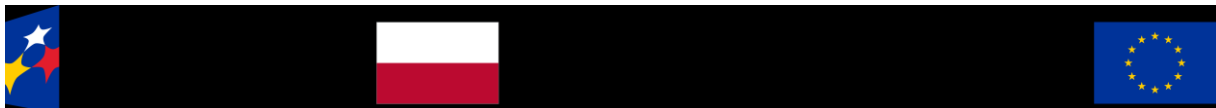
Tabela miasto:

-miasto_nazwa_miasta_idx - nazwa_miasta ASC

Tabela kierowcy:

-kierowcy_imie_idx – imie ASC, nazwisko ASC

-kierowcy_pesel_idx - pesel ASC



ROZDZIAŁ 7. KOD SQL – TWORZENIE BAZY DANYCH

Przedstaw instrukcje do tworzenia poszczególnych tabel bazy danych.

Przedstaw instrukcje do tworzenia więzów integralności zarówno na poziomie poszczególnych kolumn jak i całych tabel.

Przedstaw instrukcje do tworzenia związków między tabelami.

Przedstaw instrukcje do tworzenia indeksów dla poszczególnych tabel.

```
CREATE TABLE pracownicy (  
    id_pracownika  NUMBER(10) NOT NULL,  
    imie           VARCHAR2(45) NOT NULL,  
    nazwisko       VARCHAR2(45) NOT NULL,  
    pesel          NUMBER(11) NOT NULL,  
    stanowisko     VARCHAR2(25) NOT NULL  
);
```

```
CREATE INDEX pracownicy_pesel_idx ON  
    pracownicy (  
        pesel  
    ASC );
```

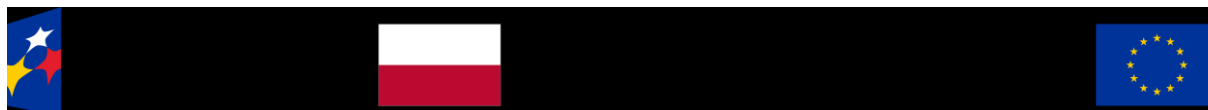
```
ALTER TABLE pracownicy ADD CONSTRAINT pracownicy_pk PRIMARY KEY (  
id_pracownika );
```

```
ALTER TABLE pracownicy ADD CONSTRAINT pracownicy_pesel_un UNIQUE ( pesel );
```

```
CREATE TABLE dane_platnosci (  
    id_dane_platosci  NUMBER(10) NOT NULL,  
    numer_karty       NUMBER(24) NOT NULL,  
    kwota             NUMBER(6) NOT NULL,  
    status_platnosci  VARCHAR2(25) NOT NULL  
);
```

```
ALTER TABLE dane_platnosci ADD CONSTRAINT dane_platnosci_pk PRIMARY KEY (  
id_dane_platosci );
```

```
CREATE TABLE kraj (  
    id_kraju  NUMBER(10) NOT NULL,  
    nazwa_kraju VARCHAR2(25) NOT NULL  
);
```



```
CREATE INDEX kraj_nazwa_kraju_idx ON
  kraj (
    nazwa_kraju
  ASC );
```

```
ALTER TABLE kraj ADD CONSTRAINT kraj_pk PRIMARY KEY ( id_kraju );
```

```
ALTER TABLE kraj ADD CONSTRAINT kraj_nazwa_kraju_un UNIQUE ( nazwa_kraju );
```

```
CREATE TABLE informacje_prawne (
  id_informacje_prawne  NUMBER(10) NOT NULL,
  karalnosc_kierowcy    VARCHAR2(3) NOT NULL,
  punkty_karne          NUMBER(2) NOT NULL
);
```

```
ALTER TABLE informacje_prawne ADD CONSTRAINT informacje_prawne_pk
PRIMARY KEY ( id_informacje_prawne );
```

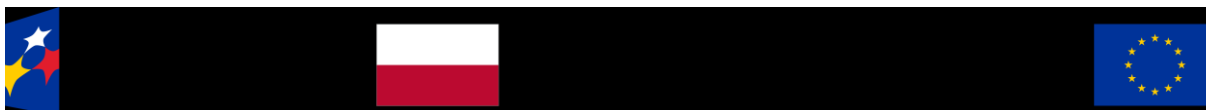
```
CREATE TABLE dane_techiczne (
  id_danych_techicznych  NUMBER(10) NOT NULL,
  moc                    NUMBER(4) NOT NULL,
  pojemnosc_silnika      NUMBER(4),
  masa                  NUMBER(5),
  masa_calkowita        NUMBER(5) NOT NULL
);
```

```
ALTER TABLE dane_techiczne ADD CONSTRAINT dane_techiczne_pk PRIMARY
KEY ( id_danych_techicznych );
```

```
CREATE TABLE oddzialy (
  id_oddzialu  NUMBER(10) NOT NULL,
  nr_oddzialu  NUMBER(3) NOT NULL,
  adres        VARCHAR2(45) NOT NULL
);
```

```
CREATE INDEX oddzialy_adres_num_idx ON
  oddzialy (
    adres
  ASC,
  nr_oddzialu
  ASC );
```

```
ALTER TABLE oddzialy ADD CONSTRAINT oddzialy_pk PRIMARY KEY ( id_oddzialu
);
```



```
ALTER TABLE oddzialy ADD CONSTRAINT oddzialy_nr_oddzialu_un UNIQUE (
nr_oddzialu );
```

```
ALTER TABLE oddzialy ADD CONSTRAINT oddzialy_adres_un UNIQUE ( adres );
```

```
CREATE TABLE auta (
    id_auta          NUMBER(10) NOT NULL,
    marka            VARCHAR2(25) NOT NULL,
    rok_produkcji     NUMBER(4) NOT NULL,
    cena_dzien        NUMBER(7, 2) NOT NULL,
    gatunek           VARCHAR2(25) NOT NULL,
    numer_vin         NUMBER(25) NOT NULL,
    kolor            VARCHAR2(20) NOT NULL,
    id_danych_techicznych NUMBER(10) NOT NULL
```

```
);
```

```
CREATE INDEX auta_numer_vin_idx ON
    auta (
        numer_vin
    ASC );
```

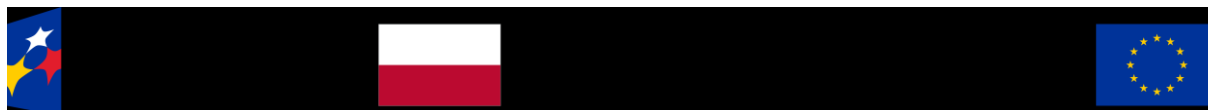
```
ALTER TABLE auta ADD CONSTRAINT auta_pk PRIMARY KEY ( id_auta );
```

```
ALTER TABLE auta ADD CONSTRAINT auta_numer_vin_un UNIQUE ( numer_vin );
```

```
ALTER TABLE auta
    ADD CONSTRAINT auta_dane_techiczne_fk FOREIGN KEY (
id_danych_techicznych )
    REFERENCES dane_techiczne ( id_danych_techicznych );
```

```
CREATE TABLE miasto (
    id_miasta        NUMBER(10) NOT NULL,
    nazwa_miasta     VARCHAR2(25) NOT NULL,
    ulica            VARCHAR2(30) NOT NULL,
    numer_budynku    NUMBER(3) NOT NULL,
    numer_lokalu     NUMBER(3),
    id_kraju         NUMBER(10) NOT NULL
);
```

```
CREATE INDEX miasto_nazwa_miasta_idx ON
    miasto (
```



```
    nazwa_miasta  
ASC );
```

```
ALTER TABLE miasto ADD CONSTRAINT miasto_pk PRIMARY KEY ( id_miasta );
```

```
ALTER TABLE miasto ADD CONSTRAINT miasto_nazwa_miasta_un UNIQUE (  
nazwa_miasta );
```

```
ALTER TABLE miasto  
ADD CONSTRAINT miasto_kraj_fk FOREIGN KEY ( id_kraju )  
REFERENCES kraj ( id_kraju );
```

```
CREATE TABLE kierowcy (
```

```
    id_kierowcy      NUMBER(10) NOT NULL,  
    imie             VARCHAR2(45) NOT NULL,  
    nazwisko         VARCHAR2(45) NOT NULL,  
    pesel            NUMBER(11) NOT NULL,  
    id_informacje_prawne NUMBER(10) NOT NULL,  
    id_miasta        NUMBER(10) NOT NULL
```

```
);
```

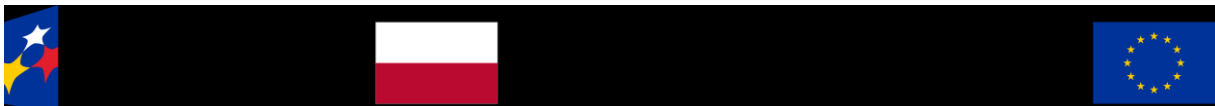
```
CREATE INDEX kierowcy_imie_idx ON  
kierowcy (  
    imie  
ASC,  
    nazwisko  
ASC );
```

```
CREATE INDEX kierowcy_pesel_idx ON  
kierowcy (  
    pesel  
ASC );
```

```
ALTER TABLE kierowcy ADD CONSTRAINT kierowcy_pk PRIMARY KEY (  
id_kierowcy );
```

```
ALTER TABLE kierowcy ADD CONSTRAINT kierowcy_pesel_un UNIQUE ( pesel );
```

```
ALTER TABLE kierowcy
```




```
ADD CONSTRAINT kierowcy_informacje_prawne_fk FOREIGN KEY (
id_informacje_prawne )
REFERENCES informacje_prawne ( id_informacje_prawne );
```

```
ALTER TABLE kierowcy
ADD CONSTRAINT kierowcy_miasto_fk FOREIGN KEY ( id_miasta )
REFERENCES miasto ( id_miasta );
```

```
CREATE TABLE wypozyczenia (
id_wypozyczenia      NUMBER(10) NOT NULL,
data_zamowienia      DATE NOT NULL,
data_wziecia_auta     DATE NOT NULL,
status_zamowienia    VARCHAR2(25) NOT NULL,
"czas_zamowienia_(dni)"  NUMBER(3) NOT NULL,
przebyte_kilometry    NUMBER(4) NOT NULL,
id_pracownika        NUMBER(10) NOT NULL,
id_dane_platosci     NUMBER(10) NOT NULL,
id_kierowcy          NUMBER(10) NOT NULL,
id_oddzialu          NUMBER(10) NOT NULL,
id_kraju             NUMBER(10) NOT NULL
);
```

```
ALTER TABLE wypozyczenia ADD CONSTRAINT wypozyczenia_pk PRIMARY KEY (
id_wypozyczenia );
```

```
ALTER TABLE wypozyczenia
ADD CONSTRAINT wypozyczenia_dane_platnosci_fk FOREIGN KEY (
id_dane_platosci )
REFERENCES dane_platnosci ( id_dane_platosci );
```

```
ALTER TABLE wypozyczenia
ADD CONSTRAINT wypozyczenia_kierowcy_fk FOREIGN KEY ( id_kierowcy )
REFERENCES kierowcy ( id_kierowcy );
```

```
ALTER TABLE wypozyczenia
ADD CONSTRAINT wypozyczenia_kraj_fk FOREIGN KEY ( id_kraju )
REFERENCES kraj ( id_kraju );
```

```
ALTER TABLE wypozyczenia
ADD CONSTRAINT wypozyczenia_oddzialy_fk FOREIGN KEY ( id_oddzialu )
REFERENCES oddzialy ( id_oddzialu );
```

```
ALTER TABLE wypozyczenia
ADD CONSTRAINT wypozyczenia_pracownicy_fk FOREIGN KEY ( id_pracownika )
REFERENCES pracownicy ( id_pracownika );
```



```
CREATE TABLE szczegoly_wypozyczenia (  
    id_wypozyczenia  NUMBER(10) NOT NULL,  
    id_auta          NUMBER(10) NOT NULL,  
    ilosc            NUMBER(2) NOT NULL,  
    cena_katalogowa  NUMBER(7, 2) NOT NULL,  
    rabat_naliczony  NUMBER(7, 2),  
    cena_transakcji  NUMBER(7, 2) NOT NULL  
);  
  
ALTER TABLE szczegoly_wypozyczenia ADD CONSTRAINT  
szczegoly_wypozyczenia_pk PRIMARY KEY ( id_wypozyczenia,  
                                         id_auta );  
  
ALTER TABLE szczegoly_wypozyczenia  
ADD CONSTRAINT szczegoly_wyp_auta_fk FOREIGN KEY ( id_auta )  
REFERENCES auta ( id_auta );  
  
ALTER TABLE szczegoly_wypozyczenia  
ADD CONSTRAINT szczegoly_wyp_wypozyczenia_fk FOREIGN KEY (  
id_wypozyczenia )  
REFERENCES wypozyczenia ( id_wypozyczenia );
```

WYPEŁNIENIE TABEL:

```
INSERT INTO kraj VALUES ('1', 'Polska');  
INSERT INTO kraj VALUES ('2', 'Niemcy');  
INSERT INTO kraj VALUES ('3', 'Szwecja');  
INSERT INTO kraj VALUES ('4', 'Holandia');  
INSERT INTO kraj VALUES ('5', 'Norwegia');  
  
INSERT INTO pracownicy VALUES ('1', 'Michal', 'Kowalski', '98753244552', 'Kierownik');  
INSERT INTO pracownicy VALUES ('2', 'Bartek', 'Nowak', '87456327463', 'Sprzedawca');  
INSERT INTO pracownicy VALUES ('3', 'Jakub', 'Kostka', '98457385601', 'Sprzedawca');  
INSERT INTO pracownicy VALUES ('4', 'Blazej', 'Rolka', '34762350981', 'Sprzedawca');  
INSERT INTO pracownicy VALUES ('5', 'Mateusz', 'Szkłanka', '98345365771',  
'Marketingowiec');  
  
INSERT INTO dane_platnosci VALUES ('1', '837465988973245374512342', '10000',  
'Zaplacono');
```



```
INSERT INTO dane_platnosci VALUES ('2', '365872394609837462841834', '1000',  
'Oczekuje');  
INSERT INTO dane_platnosci VALUES ('3', '432165265277423857665477', '2500',  
'Oczekuje');  
INSERT INTO dane_platnosci VALUES ('4', '432554351657431864781236', '3500',  
'Zaplacono');  
INSERT INTO dane_platnosci VALUES ('5', '342115287674137437254576', '25000',  
'Zaplacono');
```

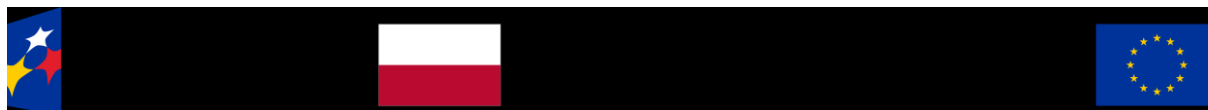
```
INSERT INTO informacje_prawne VALUES ('1', 'Nie', '0');  
INSERT INTO informacje_prawne VALUES ('2', 'Tak', '2');  
INSERT INTO informacje_prawne VALUES ('3', 'Nie', '0');  
INSERT INTO informacje_prawne VALUES ('4', 'Nie', '0');  
INSERT INTO informacje_prawne VALUES ('5', 'Tak', '13');
```

```
INSERT INTO dane_techiczne VALUES ('1', '200', '2500', '2500', '2700');  
INSERT INTO dane_techiczne VALUES ('2', '74', '1400', '1800', '2100');  
INSERT INTO dane_techiczne VALUES ('3', '140', '2000', '1900', '2200');  
INSERT INTO dane_techiczne VALUES ('4', '195', '1999', '1850', '1950');  
INSERT INTO dane_techiczne VALUES ('5', '500', '6200', '1700', '1950');
```

```
INSERT INTO oddzialy VALUES ('1', '123', 'Warszawa Ulica Cynamonowa 21');  
INSERT INTO oddzialy VALUES ('2', '100', 'DGB-Haus, Kapweg 4, 13405 Berlin');  
INSERT INTO oddzialy VALUES ('3', '98', 'Ambasada RP Karlavägen 35');  
INSERT INTO oddzialy VALUES ('4', '23', 'Haga, Alexanderstraat 25, 2514 JM');  
INSERT INTO oddzialy VALUES ('5', '45', 'C. J. Hambros plass 2, 0164 Oslo');
```

```
INSERT INTO miasto(id_miasta, nazwa_miasta, ulica, numer_budynku, numer_lokalu,  
id_kraju) VALUES ('1', 'Warszawa', 'Cynamonowa', '2', '9', '1');  
INSERT INTO miasto(id_miasta, nazwa_miasta, ulica, numer_budynku, numer_lokalu,  
id_kraju) VALUES ('2', 'Berlin', 'Kapweg', '21', '10', '2');  
INSERT INTO miasto(id_miasta, nazwa_miasta, ulica, numer_budynku, numer_lokalu,  
id_kraju) VALUES ('3', 'Stockholm', 'Karlavägen', '22', '11', '3');  
INSERT INTO miasto(id_miasta, nazwa_miasta, ulica, numer_budynku, numer_lokalu,  
id_kraju) VALUES ('4', 'Amsterdam', 'Alexanderstraat', '23', '12', '4');  
INSERT INTO miasto(id_miasta, nazwa_miasta, ulica, numer_budynku, numer_lokalu,  
id_kraju) VALUES ('5', 'Oslo', 'Hambros', '24', '13', '5');
```

```
INSERT INTO auta(id_auta, marka, rok_produkcji, cena_dzien, gatunek, numer_VIN, kolor,  
id_danych_techicznych) VALUES ('1', 'Audi', '2012', '950', 'sedan',  
'4137387461234974652355462', 'czarny', '1');  
INSERT INTO auta(id_auta, marka, rok_produkcji, cena_dzien, gatunek, numer_VIN, kolor,  
id_danych_techicznych) VALUES ('2', 'BMW', '2009', '250', 'sedan',  
'3242313488743276539823647', 'bialy', '2');
```



```
INSERT INTO auta(id_auta, marka, rok_produkcji, cena_dzien, gatunek, numer_VIN, kolor, id_danych_techicznych) VALUES ('3', 'Mercedes', '2003', '400', 'sedan', '4321341324132414321343333', 'zielony', '3');
INSERT INTO auta(id_auta, marka, rok_produkcji, cena_dzien, gatunek, numer_VIN, kolor, id_danych_techicznych) VALUES ('4', 'Citroen', '2019', '700', 'suv', '3413214431413243214231432', 'zielony', '4');
INSERT INTO auta(id_auta, marka, rok_produkcji, cena_dzien, gatunek, numer_VIN, kolor, id_danych_techicznych) VALUES ('5', 'Mercedes', '2000', '1500', 'sport', '4654524252423642364323634', 'srebrny', '5');
```

```
INSERT INTO kierowcy(id_kierowcy, imie, nazwisko, pesel, id_informacje_prawne, id_miasta) VALUES ('1', 'Tomasz', 'Kowalski', '32153465323', '1', '1');
INSERT INTO kierowcy(id_kierowcy, imie, nazwisko, pesel, id_informacje_prawne, id_miasta) VALUES ('2', 'Adam', 'Zielinski', '32474326785', '2', '2');
INSERT INTO kierowcy(id_kierowcy, imie, nazwisko, pesel, id_informacje_prawne, id_miasta) VALUES ('3', 'Ewa', 'Kos', '43243224432', '3', '3');
INSERT INTO kierowcy(id_kierowcy, imie, nazwisko, pesel, id_informacje_prawne, id_miasta) VALUES ('4', 'Pawel', 'Gmor', '42343242322', '4', '4');
INSERT INTO kierowcy(id_kierowcy, imie, nazwisko, pesel, id_informacje_prawne, id_miasta) VALUES ('5', 'Piotr', 'Nowak', '67556433333', '5', '5');
```

```
INSERT INTO wypozyczenia VALUES ('1', TO_DATE('2020-12-20','YYYY-MM-DD'), TO_DATE('2020-12-23','YYYY-MM-DD'), 'Realizuje', '12', '120', '1', '1', '1', '1', '1');
INSERT INTO wypozyczenia VALUES ('2', TO_DATE('2010-11-12','YYYY-MM-DD'), TO_DATE('2010-12-21','YYYY-MM-DD'), 'Zrealizowane', '5', '150', '2', '2', '2', '2', '2');
INSERT INTO wypozyczenia VALUES ('3', TO_DATE('2003-02-13','YYYY-MM-DD'), TO_DATE('2003-02-19','YYYY-MM-DD'), 'Zrealizowane', '4', '200', '2', '3', '3', '3', '3');
INSERT INTO wypozyczenia VALUES ('4', TO_DATE('2020-05-20','YYYY-MM-DD'), TO_DATE('2020-05-23','YYYY-MM-DD'), 'Zrealizowane', '2', '45', '3', '4', '4', '4', '4');
INSERT INTO wypozyczenia VALUES ('5', TO_DATE('2012-03-21','YYYY-MM-DD'), TO_DATE('2012-03-23','YYYY-MM-DD'), 'Zrealizowane', '20', '1200', '4', '5', '5', '5', '5');
```

```
INSERT INTO szczegoly_wypozyczenia VALUES ('1', '1', '1', '10000', '0', '10000');
INSERT INTO szczegoly_wypozyczenia VALUES ('2', '2', '1', '1000', '0', '1000');
INSERT INTO szczegoly_wypozyczenia VALUES ('3', '3', '1', '2500', '0', '2500');
INSERT INTO szczegoly_wypozyczenia VALUES ('4', '4', '1', '3500', '0', '3500');
INSERT INTO szczegoly_wypozyczenia VALUES ('5', '5', '1', '25200', '200', '25000');
```





Raport powstał podczas zajęć laboratoryjnych z przedmiotu
prowadzonego w ramach projektu
„Zintegrowany Program Rozwoju Politechniki Lubelskiej – część druga”,
umowa nr **POWR.03.05.00-00-Z060/18-00**
w ramach Programu Operacyjnego Wiedza Edukacja Rozwój 2014-2020
współfinansowanego ze środków Europejskiego Funduszu Społecznego

