
	Uniwersytet Technologiczno-Przyrodniczy im. J. i J. Śniadeckich w Bydgoszczy Wydział Telekomunikacji, Informatyki i Elektrotechniki	
Przedmiot	Bazy danych	
Prowadzący	Dr inż. Zbigniew Lutowski	
Temat projektu	System sprzedaży biletów lotniczych	
Student	Łukasz Farulewski	

Wprowadzenie

Celem projektu jest przygotowanie systemu sprzedaży biletów lotniczych na serwerze Firebird. Serwer **Firebird** to wolnodostępowy serwer służący do zarządzania relacyjnymi bazami danych. Obok MySQL oraz PostgreSQL należy do najpopularniejszych serwerów bazodanowych typu Opensource.

Projekt składa się z 3 części:

Część 1 to diagramy modelu ER w notacji Chena, Barkera oraz UML a także założenia projektowe.

Część 2 to instalacja serwera Firebird oraz wykaz utworzonych relacji z uwzględnieniem określonych zapytań wewnątrz poszczególnych relacji, m.in. CREATE, INSERT, UPDATE, ORDER BY, LIKE, BETWEEN.

Część 3 to połączenia zewnętrzne i wykaz powiązań między relacjami i tworzenie takich zapytań, które w odpowiedzi zestawiają dane z różnych tabel.

Część 1

Założenia do diagramów E-R:

- 1. Pasażer może posiadać 1 lub więcej bagaży, ale każdy bagaż musi należeć do 1 pasażera.*
- 2. Klient może dokonać 1 lub więcej rezerwacji, ale każda rezerwacja musi zostać dokonana przez 1 klienta.*
- 3. Pracownik może dokonać 1 lub więcej rezerwacji, ale każda rezerwacja musi zostać dokonana przez 1 pracownika.*
- 4. Pracownik może obsługiwać 1 lub więcej samolotów (np. obsługa naziemna), ale każdy samolot musi być obsługiwany przez 1 lub więcej pracowników (pilot, 2 pilot, obsługa). Z*

uwagi na określony harmonogram lotów (każdy lot wykonywany jest innego dnia) przyjąłem, że dany pilot, może obsługiwać więcej niż 1 samolot.

5. Klient może dokonać 1 lub więcej płatności, ale każda płatność musi być dokonana przez 1 klienta.

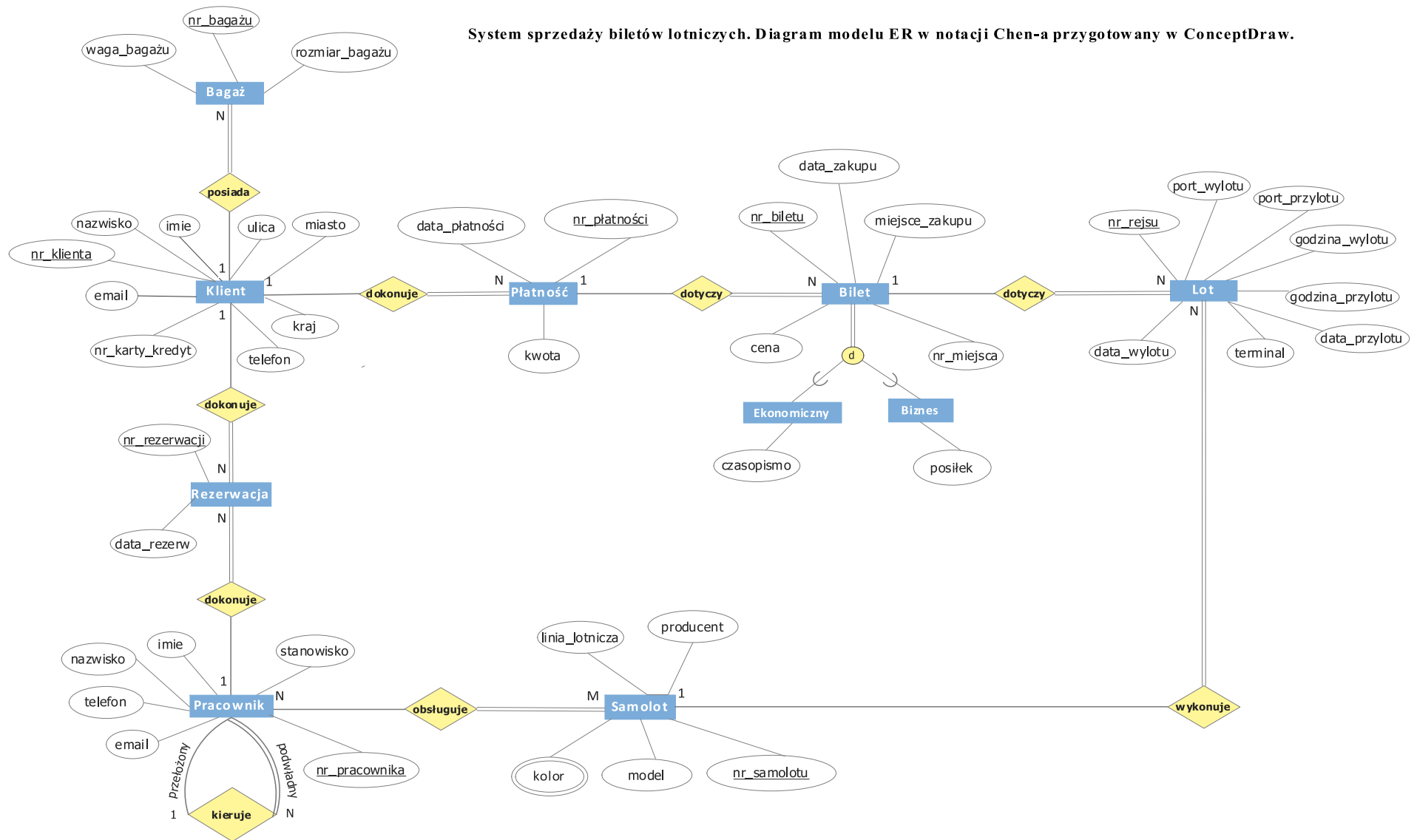
6. Płatność może dotyczyć więcej niż 1 biletu, ale zakup każdego biletu musi być dokonana przez 1 płatność.

7. 1 bilet można wykorzystać np. do lotu w dwie strony lub z przesiadką. Natomiast każdy lot musi mieć 1 bilet.

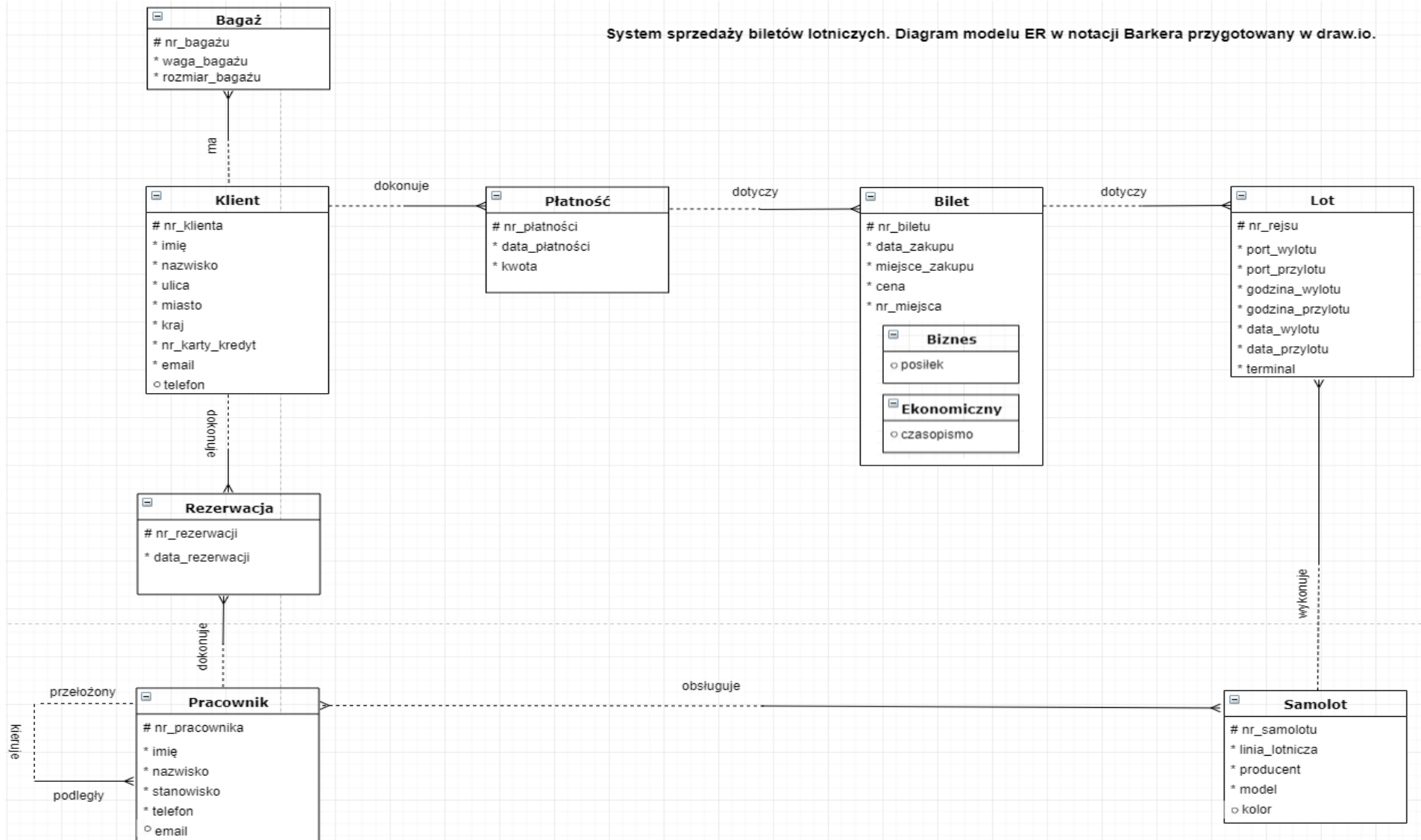
8. Samolot może wykonywać 1 lub więcej lotów, ale każdy lot musi być wykonywany przez 1 samolot.

9. Przełożony może kierować więcej niż 1 pracownikiem ale podwładny nie może mieć więcej niż 1 przełożonego.

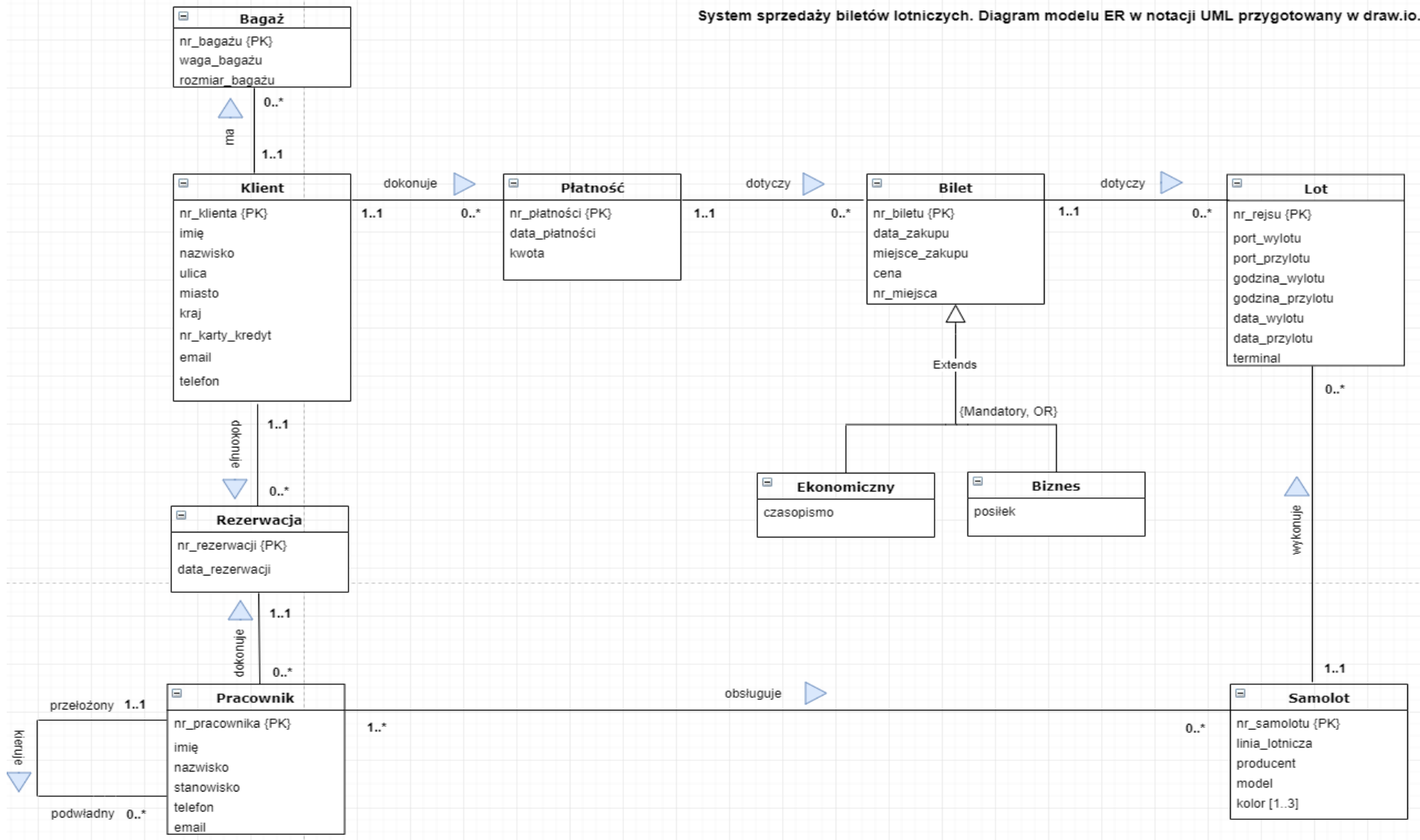
System sprzedaży biletów lotniczych. Diagram modelu ER w notacji Chen-a przygotowany w ConceptDraw.



System sprzedaży biletów lotniczych. Diagram modelu ER w notacji Barkera przygotowany w draw.io.



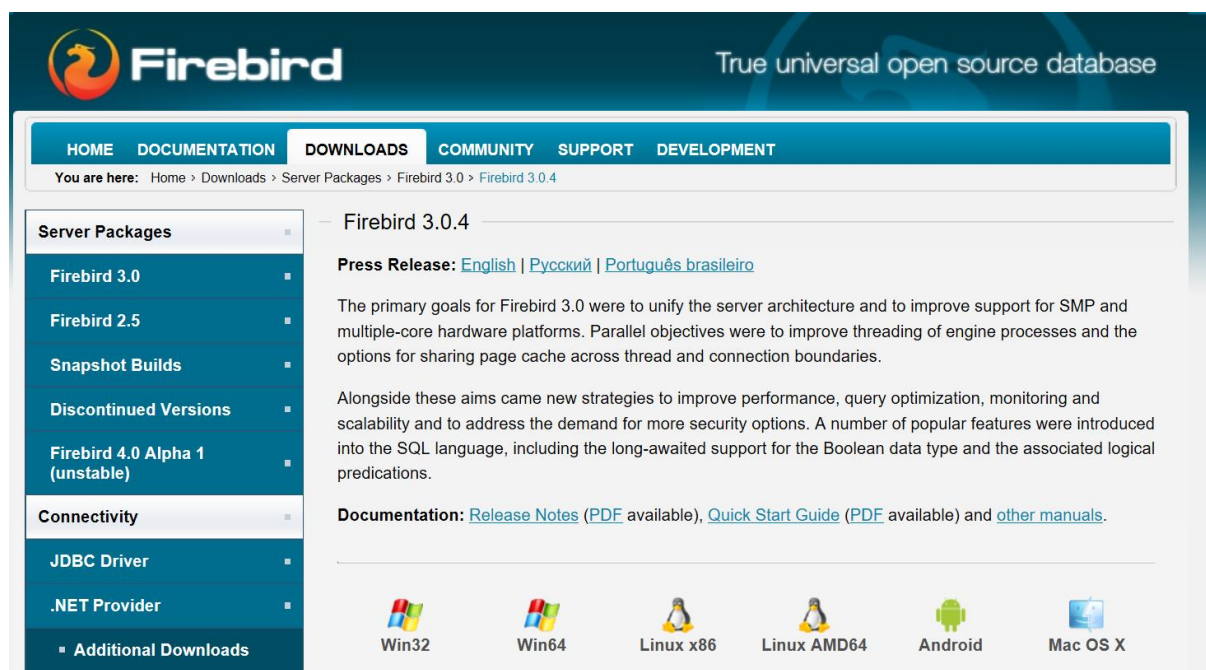
System sprzedaży biletów lotniczych. Diagram modelu ER w notacji UML przygotowany w draw.io.



Cześć 2

1. Instalacja serwera bazodanowego *Firebird 3.0.4* jako usługi na win 64.

<https://firebirdsql.org/en/firebird-3-0-4/>



Stworzenie nowej bazy danych **System_Sprzedazy**.

```
Firebird ISQL Tool
ISQL Version: WI-V3.0.4.33054 Firebird 3.0
Use CONNECT or CREATE DATABASE to specify a database
SQL> create database 'C:\Program Files\Firebird\Projects\System_Sprzedazy.FDB' page_size 8192 user 'sysdba' password 'firebirdcom79';
Server version:
WI-V3.0.4.33054 Firebird 3.0
WI-V3.0.4.33054 Firebird 3.0/XNet (HAPIK)/P14
WI-V3.0.4.33054 Firebird 3.0/XNet (HAPIK)/P14
SQL>
```

Wykaz utworzonych tabel:

```
WybierzFirebird ISQL Tool
SQL> SHOW TABLES;
      BAGAZ      BILET
      KLIENT      LOT
      PLATNOSC    PRACOWNIK
      REZERWACJA  SAMOLOT
SQL>
```

Logowanie do bazy i utworzenie pierwszej tabeli – Klient.

```
SQL> connect 'C:\Program Files\Firebird\Projects\SYSTEM_SPRZEDAZy.FDB' user sysdba password firebirdcom79;
```

```
SQL> connect 'C:\Program Files\Firebird\Projects\SYSTEM_SPRZEDAZy.FDB' user sysdba password firebirdcom79;
Server version:
WI-V3.0.4.33054 Firebird 3.0
WI-V3.0.4.33054 Firebird 3.0/XNet (HAPIK)/P14
WI-V3.0.4.33054 Firebird 3.0/XNet (HAPIK)/P14
Database: 'C:\Program Files\Firebird\Projects\SYSTEM_SPRZEDAZy.FDB', User: SYSDBA
SQL> DROP TABLE KLIENT;
SQL> SHOW TABLES;
There are no tables in this database
SQL> CREATE TABLE KLIENT (
CON> NRKLIENTA INTEGER NOT NULL PRIMARY KEY,
CON> IMIE VARCHAR (10) NOT NULL,
CON> NAZWISKO VARCHAR (10) NOT NULL,
CON> ULICA VARCHAR (10) NOT NULL,
CON> MIASTO VARCHAR (10) NOT NULL,
CON> KRAJ VARCHAR (10) NOT NULL,
CON> NRKARTYKREDYT INTEGER NOT NULL,
CON> TELEFON INTEGER NOT NULL,
CON> EMAIL VARCHAR (10) NOT NULL);
SQL>
```

Tabela Klient

Primary Key: NRKLIENTA

Firebird SQL Tool

```
SQL> select * from klient;
```

NRKLIENTA	IMIE	NAZWISKO	ULICA	MIASTO	KRAJ	NRKARTYKREDYT	TELEFON	EMAIL
1	Lukasz	Farulewski	Berlinga	Bydgoszcz	Polska	111111	52111111	lfarulewski(at)onet.pl
2	Pawel	Dominik	Kisiela	Warszawa	Polska	222222	22222222	pdominik(at)wp.pl
3	Tomasz	Marzec	Gorska	Katowice	Polska	333333	32333333	tmarzec(at)hotmail.com
4	Damian	Kwiecien	Morska	Gdansk	Polska	444444	58444444	dkwiecien(at)gmail.com
5	Karol	Lipiec	Zielona	Olsztyn	Polska	555555	89555555	klipiec(at)onet.pl
6	Maciej	Maj	Strumykowa	Wroclaw	Polska	666666	72666666	mmaj(at)wp.pl
7	Joanna	Wrzesien	Karpacka	Lodz	Polska	777777	42777777	jwrzesien(at)gmail.com
8	Katarzyna	Lato	Graniczna	Bialystok	Polska	888888	82888888	klato(at)yahoo.com
9	Anna	Mrozik	Wiosenna	Poznan	Polska	999999	61999999	amrozik(at)wp.pl
10	Barbara	Karolak	Sniezna	Krakow	Polska	101010	12101010	bkarolak(at)op.pl
11	James	Rodrigez	Abbestr	Berlin	Niemcy	910498	49823514	jr(at)gmail.com
12	David	Coperfield	MainAve	Waszyngton	USA	769023	580235148	d.coperfield(at)miracle.com
13	Vijay	Singh	MahatGandi	Bangalore	Indie	683090	989891207	vijay.singh(at)hotmail.com
14	Shirioku	Nagasaki	Commercial	Tokjo	Japonia	769091	37789102	snagasaki(at)jakuza.com
15	Katherine	Johns	HighAve	Nowy Jork	USA	903711	580899029	katherine.johns(at)aa.com

INSERT. Dodawanie danych do bazy. Dodanie nowego klienta.

Firebird ISQL Tool

```
SQL> INSERT INTO KLIENT VALUES ('16','Janet','Clarkson','SunnyAve','Durban','RPA','980434','296908222','j.clarkson(at)gmail.com');
SQL> select * from klient;
```

NRKLIENTA	IMIE	NAZWISKO	ULICA	MIASTO	KRAJ	NRKARTYKREDYT	TELEFON	EMAIL
1	Lukasz	Farulewski	Berlinga	Bydgoszcz	Polska	111111	52111111	lfarulewski(at)onet.pl
16	Janet	Clarkson	SunnyAve	Durban	RPA	980434	296908222	j.clarkson(at)gmail.com
2	Pawel	Dominik	Kisiela	Warszawa	Polska	222222	22222222	pdominik(at)wp.pl
3	Tomasz	Marzec	Gorska	Katowice	Polska	333333	32333333	tmarzec(at)hotmail.com
4	Damian	Kwiecien	Morska	Gdansk	Polska	444444	58444444	dkwiecien(at)gmail.com
5	Karol	Lipiec	Zielona	Olsztyn	Polska	555555	89555555	klipiec(at)onet.pl
6	Maciej	Maj	Strumykowa	Wroclaw	Polska	666666	72666666	mmaj(at)wp.pl
7	Joanna	Wrzesien	Karpacka	Lodz	Polska	777777	42777777	jwrzesien(at)gmail.com
8	Katarzyna	Lato	Graniczna	Bialystok	Polska	888888	82888888	klato(at)yahoo.com
9	Anna	Mrozik	Wiosenna	Poznan	Polska	999999	61999999	amrozik(at)wp.pl
10	Barbara	Karolak	Sniezna	Krakow	Polska	101010	12101010	bkarolak(at)op.pl
11	James	Rodríguez	Abbestr	Berlin	Niemcy	910498	49823514	jr(at)gmail.com
12	David	Coperfield	MainAve	Waszyngton	USA	769023	580235148	d.coperfield(at)miracle.com
13	Vijay	Singh	MahatGandhi	Bangalore	Indie	683090	989891207	vijay.singh(at)hotmail.com
14	Shirioku	Nagasaki	Commercial	Tokjo	Japonia	769091	37789102	snagasaki(at)jakuza.com
15	Katherine	Johns	HighAve	Nowy Jork	USA	903711	580899029	katherine.johns(at)aa.com

ORDER BY. Sortowanie wyników zapytań. Sortowanie po nrklienta.

Firebird ISQL Tool

```
SQL> select * from klient order by nrklienta asc;
```

NRKLIENTA	IMIE	NAZWISKO	ULICA	MIASTO	KRAJ	NRKARTYKREDYT	TELEFON	EMAIL
1	Lukasz	Farulewski	Berlinga	Bydgoszcz	Polska	111111	52111111	lfarulewski(at)onet.pl
2	Pawel	Dominik	Kisiela	Warszawa	Polska	222222	22222222	pdominik(at)wp.pl
3	Tomasz	Marzec	Gorska	Katowice	Polska	333333	32333333	tmarzec(at)hotmail.com
4	Damian	Kwiecien	Morska	Gdansk	Polska	444444	58444444	dkwiecien(at)gmail.com
5	Karol	Lipiec	Zielona	Olsztyn	Polska	555555	89555555	klipiec(at)onet.pl
6	Maciej	Maj	Strumykowa	Wroclaw	Polska	666666	72666666	mmaj(at)wp.pl
7	Joanna	Wrzesien	Karpacka	Lodz	Polska	777777	42777777	jwrzesien(at)gmail.com
8	Katarzyna	Lato	Graniczna	Bialystok	Polska	888888	82888888	klato(at)yahoo.com
9	Anna	Mrozik	Wiosenna	Poznan	Polska	999999	61999999	amrozik(at)wp.pl
10	Barbara	Karolak	Sniezna	Krakow	Polska	101010	12101010	bkarolak(at)op.pl
11	James	Rodríguez	Abbestr	Berlin	Niemcy	910498	49823514	jr(at)gmail.com
12	David	Coperfield	MainAve	Waszyngton	USA	769023	580235148	d.coperfield(at)miracle.com
13	Vijay	Singh	MahatGandhi	Bangalore	Indie	683090	989891207	vijay.singh(at)hotmail.com
14	Shirioku	Nagasaki	Commercial	Tokjo	Japonia	769091	37789102	snagasaki(at)jakuza.com
15	Katherine	Johns	HighAve	Nowy Jork	USA	903711	580899029	katherine.johns(at)aa.com
16	Janet	Clarkson	SunnyAve	Durban	RPA	980434	296908222	j.clarkson(at)gmail.com

DELETE. Usuwanie danych z bazy danych. Usunięcie nowo dodanej osoby.

Firebird ISQL Tool

```
SQL> delete from klient where nrklienta = 16;
SQL> select * from klient;
```

NRKLIENTA	IMIE	NAZWISKO	ULICA	MIASTO	KRAJ	NRKARTYKREDYT	TELEFON	EMAIL
1	Lukasz	Farulewski	Berlinga	Bydgoszcz	Polska	111111	52111111	lfarulewski(at)onet.pl
2	Pawel	Dominik	Kisiela	Warszawa	Polska	222222	22222222	pdominik(at)wp.pl
3	Tomasz	Marzec	Gorska	Katowice	Polska	333333	32333333	tmarzec(at)hotmail.com
4	Damian	Kwiecien	Morska	Gdansk	Polska	444444	58444444	dkwiecien(at)gmail.com
5	Karol	Lipiec	Zielona	Olsztyn	Polska	555555	89555555	klipiec(at)onet.pl
6	Maciej	Maj	Strumykowa	Wroclaw	Polska	666666	72666666	mmaj(at)wp.pl
7	Joanna	Wrzesien	Karpacka	Lodz	Polska	777777	42777777	jwrzesien(at)gmail.com
8	Katarzyna	Lato	Graniczna	Bialystok	Polska	888888	82888888	klato(at)yahoo.com
9	Anna	Mrozik	Wiosenna	Poznan	Polska	999999	61999999	amrozik(at)wp.pl
10	Barbara	Karolak	Sniezna	Krakow	Polska	101010	12101010	bkarolak(at)op.pl
11	James	Rodríguez	Abbestr	Berlin	Niemcy	910498	49823514	jr(at)gmail.com
12	David	Coperfield	MainAve	Waszyngton	USA	769023	580235148	d.coperfield(at)miracle.com
13	Vijay	Singh	MahatGandhi	Bangalore	Indie	683090	989891207	vijay.singh(at)hotmail.com
14	Shirioku	Nagasaki	Commercial	Tokjo	Japonia	769091	37789102	snagasaki(at)jakuza.com
15	Katherine	Johns	HighAve	Nowy Jork	USA	903711	580899029	katherine.johns(at)aa.com

UPDATE. Modyfikacja danych w bazie. Modyfikacja ulicy, miasta i kraju dla klienta nr 8

```
Firebird ISQL Tool
SQL> update klient set ulica = 'Mrozna', miasto = 'Moskwa', kraj = 'Rosja' where nrklienta = 8;
SQL> select * from klient;
```

NRKLIENTA	IMIE	NAZWISKO	ULICA	MIASTO	KRAJ	NRKARTYKREDYT	TELEFON	EMAIL
1	Lukasz	Farulewski	Berlinga	Bydgoszcz	Polska	111111	52111111	lfarulewski(at)onet.pl
2	Pawel	Dominik	Kisiela	Warszawa	Polska	222222	22222222	pdominik(at)wp.pl
3	Tomasz	Marzec	Gorska	Katowice	Polska	333333	32333333	tmarzec(at)hotmail.com
4	Damian	Kwiecien	Morska	Gdansk	Polska	444444	58444444	dkwiecien(at)gmail.com
5	Karol	Lipiec	Zielona	Olsztyn	Polska	555555	89555555	klipiec(at)onet.pl
6	Maciej	Maj	Strumykowa	Wroclaw	Polska	666666	72666666	mmaj(at)wp.pl
7	Joanna	Wrzesien	Karpacka	Lodz	Polska	777777	42777777	jwrzesien(at)gmail.com
8	Katarzyna	Lato	Mrozna	Moskwa	Rosja	888888	82888888	klato(at)yahoo.com
9	Anna	Mrozik	Wiosenna	Poznan	Polska	999999	61999999	amrozik(at)wp.pl
10	Barbara	Karolak	Sniezna	Krakow	Polska	101010	12101010	bkarolak(at)op.pl
11	James	Rodriguez	Abbestr	Berlin	Niemcy	910498	49823514	jr(at)gmail.com
12	David	Coperfield	MainAve	Waszyngton	USA	769023	580235148	d.coperfield(at)miracle.com
13	Vijay	Singh	MahatGandi	Bangalore	Indie	683090	989891207	vijay.singh(at)hotmail.com
14	Shirioku	Nagasaki	Commercial	Tokjo	Japonia	769091	37789102	snagasaki(at)jakuza.com
15	Katherine	Johns	HighAve	Nowy Jork	USA	903711	580899029	katherine.johns(at)aa.com

```
SQL>
```

IN. Wybór klientów mieszkających w Polsce i poza Polską.

```
Firebird ISQL Tool
SQL> select * from klient where kraj in ('Polska');
```

NRKLIENTA	IMIE	NAZWISKO	ULICA	MIASTO	KRAJ	NRKARTYKREDYT	TELEFON	EMAIL
1	Lukasz	Farulewski	Berlinga	Bydgoszcz	Polska	111111	52111111	lfarulewski(at)onet.pl
2	Pawel	Dominik	Kisiela	Warszawa	Polska	222222	22222222	pdominik(at)wp.pl
3	Tomasz	Marzec	Gorska	Katowice	Polska	333333	32333333	tmarzec(at)hotmail.com
4	Damian	Kwiecien	Morska	Gdansk	Polska	444444	58444444	dkwiecien(at)gmail.com
5	Karol	Lipiec	Zielona	Olsztyn	Polska	555555	89555555	klipiec(at)onet.pl
6	Maciej	Maj	Strumykowa	Wroclaw	Polska	666666	72666666	mmaj(at)wp.pl
7	Joanna	Wrzesien	Karpacka	Lodz	Polska	777777	42777777	jwrzesien(at)gmail.com
9	Anna	Mrozik	Wiosenna	Poznan	Polska	999999	61999999	amrozik(at)wp.pl
10	Barbara	Karolak	Sniezna	Krakow	Polska	101010	12101010	bkarolak(at)op.pl

```
SQL> select * from klient where kraj not in ('Polska');
```

NRKLIENTA	IMIE	NAZWISKO	ULICA	MIASTO	KRAJ	NRKARTYKREDYT	TELEFON	EMAIL
8	Katarzyna	Lato	Mrozna	Moskwa	Rosja	888888	82888888	klato(at)yahoo.com
11	James	Rodriguez	Abbestr	Berlin	Niemcy	910498	49823514	jr(at)gmail.com
12	David	Coperfield	MainAve	Waszyngton	USA	769023	580235148	d.coperfield(at)miracle.com
13	Vijay	Singh	MahatGandi	Bangalore	Indie	683090	989891207	vijay.singh(at)hotmail.com
14	Shirioku	Nagasaki	Commercial	Tokjo	Japonia	769091	37789102	snagasaki(at)jakuza.com
15	Katherine	Johns	HighAve	Nowy Jork	USA	903711	580899029	katherine.johns(at)aa.com

```
SQL>
```

Tabela Pracownik

Primary Key: NRPRACOWNIKA

Firebird ISQL Tool

```
SQL> CREATE TABLE PRACOWNIK (  
CON> NRPRACOWNIKA INTEGER NOT NULL PRIMARY KEY,  
CON> IMIE VARCHAR (10) NOT NULL,  
CON> NAZWISKO VARCHAR (10) NOT NULL,  
CON> STANOWISKO VARCHAR (25) NOT NULL,  
CON> TELEFON INTEGER NOT NULL,  
CON> EMAIL VARCHAR (30));  
SQL> INSERT INTO PRACOWNIK VALUES('1','Dagmara','Biskup','Pilot','668223657','dbiskup(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('2','Marta','Kowal','Obsluga naziemna','668112758','mkowal(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('3','Mateusz','Nowak','Pilot','668978546','mnowak(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('4','Bogdan','Poreba','Obsluga techniczna','668858342','bporeba(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('5','Monika','Makowska','Stewardessa','668896445','mmakowska(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('6','Karolina','Stankowska','Obsluga naziemna','668998780','kstankowska(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('7','Kamila','Pankowska','Stewardessa','668789667','kpankowska(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('8','Waldemar','Noskowski','Obsluga techniczna','668564231','wnoskowski(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('9','Adam','Krupa','Pilot','668590021','akrupa(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('10','Iwona','Waniek','Stewardessa','668021465','iwaniek(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('11','Kamil','Matowski','Obsluga naziemna','668289054','kmatowski(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('12','Johann','Wolf','Pilot','155512345','johann.wolf(at)lufthansa.de');  
SQL> INSERT INTO PRACOWNIK VALUES('13','Krystian','Pomorski','Obsluga naziemna','668787657','kpomorski(at)lot.pl');  
SQL> INSERT INTO PRACOWNIK VALUES('14','Matheus','Ubner','Pilot','155739092','matheus.ubner(at)lufthansa.de');  
SQL> INSERT INTO PRACOWNIK VALUES('15','Lea','Nahmen','Stewardessa','155543506','lea.nahmen(at)lufthansa.de');  
SQL>
```

Firebird ISQL Tool

```
SQL> SELECT * FROM PRACOWNIK;  
  
NRPRACOWNIKA IMIE      NAZWISKO      STANOWISKO      TELEFON EMAIL  
-----  
1 Dagmara      Biskup        Pilot           668223657 dbiskup(at)lot.pl  
2 Marta        Kowal         Obsluga naziemna 668112758 mkowal(at)lot.pl  
3 Mateusz      Nowak         Pilot           668978546 mnowak(at)lot.pl  
4 Bogdan       Poreba        Obsluga techniczna 668858342 bporeba(at)lot.pl  
5 Monika       Makowska      Stewardessa     668896445 mmakowska(at)lot.pl  
6 Karolina     Stankowska    Obsluga naziemna 668998780 kstankowska(at)lot.pl  
7 Kamila       Pankowska     Stewardessa     668789667 kpankowska(at)lot.pl  
8 Waldemar     Noskowski     Obsluga techniczna 668564231 wnoskowski(at)lot.pl  
9 Adam         Krupa         Pilot           668590021 akrupa(at)lot.pl  
10 Iwona        Waniek        Stewardessa     668021465 iwaniek(at)lot.pl  
11 Kamil        Matowski      Obsluga naziemna 668289054 kmatowski(at)lot.pl  
12 Johann      Wolf          Pilot           155512345 johann.wolf(at)lufthansa.de  
13 Krystian     Pomorski      Obsluga naziemna 668787657 kpomorski(at)lot.pl  
14 Matheus      Ubner         Pilot           155739092 matheus.ubner(at)lufthansa.de  
15 Lea          Nahmen        Stewardessa     155543506 lea.nahmen(at)lufthansa.de  
  
SQL>
```

LIKE. Wybór pracowników, których imiona zaczynają się na literę K.

Firebird ISQL Tool

```
SQL> select * from pracownik where imie like 'K%';  
  
NRPRACOWNIKA IMIE      NAZWISKO      STANOWISKO      TELEFON EMAIL  
-----  
6 Karolina     Stankowska    Obsluga naziemna 668998780 kstankowska(at)lot.pl  
7 Kamila       Pankowska     Stewardessa     668789667 kpankowska(at)lot.pl  
11 Kamil        Matowski      Obsluga naziemna 668289054 kmatowski(at)lot.pl  
13 Krystian     Pomorski      Obsluga naziemna 668787657 kpomorski(at)lot.pl  
  
SQL>
```

ALTER TABLE ADD NEW COLUMN. Dodanie nowej kolumny NRSZEFA reprezentującej związek unarny

Firebird ISQL Tool

```
SQL> ALTER TABLE PRACOWNIK ADD NRSZEFA INTEGER REFERENCES PRACOWNIK (NRPRACOWNIKA);
SQL> SELECT * FROM PRACOWNIK;
```

NRPRACOWNIKA	IMIE	NAZWISKO	STANOWISKO	TELEFON	EMAIL	NRSZEFA
1	Dagmara	Biskup	Pilot	668223657	dbiskup(at)lot.pl	<null>
2	Marta	Kowal	Obsługa naziemna	668112758	mkowal(at)lot.pl	<null>
3	Mateusz	Nowak	Pilot	668978546	mnowak(at)lot.pl	<null>
4	Bogdan	Poreba	Obsługa techniczna	668858342	bporeba(at)lot.pl	<null>
5	Monika	Makowska	Stewardessa	668896445	mmakowska(at)lot.pl	<null>
6	Karolina	Stankowska	Obsługa naziemna	668998780	kstankowska(at)lot.pl	<null>
7	Kamila	Pankowska	Stewardessa	668789667	kpankowska(at)lot.pl	<null>
8	Waldemar	Noskowski	Obsługa techniczna	668564231	wnoskowski(at)lot.pl	<null>
9	Adam	Krupa	Pilot	668590021	akrupa(at)lot.pl	<null>
10	Iwona	Waniek	Stewardessa	668021465	iwaniek(at)lot.pl	<null>
11	Kamil	Matowski	Obsługa naziemna	668289054	kmatowski(at)lot.pl	<null>
12	Johann	Wolf	Pilot	155512345	johann.wolf(at)lufthansa.de	<null>
13	Krystian	Pomorski	Obsługa naziemna	668787657	kpomorski(at)lot.pl	<null>
14	Matheus	Ubner	Pilot	155739092	matheus.ubner(at)lufthansa.de	<null>
15	Lea	Nahmen	Stewardessa	155543506	lea.nahmen(at)lufthansa.de	<null>

UPDATE. Dodawanie nowych krotek do kolumny NRSZEFA.

Firebird ISQL Tool

```
SQL> UPDATE PRACOWNIK SET NRSZEFA = '3' WHERE NRPRACOWNIKA = '1';
SQL> UPDATE PRACOWNIK SET NRSZEFA = '3' WHERE NRPRACOWNIKA = '9';
SQL> UPDATE PRACOWNIK SET NRSZEFA = '3' WHERE NRPRACOWNIKA = '3';
SQL> UPDATE PRACOWNIK SET NRSZEFA = '14' WHERE NRPRACOWNIKA = '14';
SQL> UPDATE PRACOWNIK SET NRSZEFA = '12' WHERE NRPRACOWNIKA = '12';
SQL> UPDATE PRACOWNIK SET NRSZEFA = '7' WHERE NRPRACOWNIKA = '15';
SQL> UPDATE PRACOWNIK SET NRSZEFA = '7' WHERE NRPRACOWNIKA = '5';
SQL> UPDATE PRACOWNIK SET NRSZEFA = '7' WHERE NRPRACOWNIKA = '7';
SQL> UPDATE PRACOWNIK SET NRSZEFA = '7' WHERE NRPRACOWNIKA = '10';
SQL> SELECT * FROM PRACOWNIK ORDER BY STANOWISKO ASC;
```

NRPRACOWNIKA	IMIE	NAZWISKO	STANOWISKO	TELEFON	EMAIL	NRSZEFA
2	Marta	Kowal	Obsługa naziemna	668112758	mkowal(at)lot.pl	2
11	Kamil	Matowski	Obsługa naziemna	668289054	kmatowski(at)lot.pl	2
13	Krystian	Pomorski	Obsługa naziemna	668787657	kpomorski(at)lot.pl	2
6	Karolina	Stankowska	Obsługa naziemna	668998780	kstankowska(at)lot.pl	2
8	Waldemar	Noskowski	Obsługa techniczna	668564231	wnoskowski(at)lot.pl	8
4	Bogdan	Poreba	Obsługa techniczna	668858342	bporeba(at)lot.pl	8
1	Dagmara	Biskup	Pilot	668223657	dbiskup(at)lot.pl	3
9	Adam	Krupa	Pilot	668590021	akrupa(at)lot.pl	3
3	Mateusz	Nowak	Pilot	668978546	mnowak(at)lot.pl	3
12	Johann	Wolf	Pilot	155512345	johan.wolf(at)lufthansa.de	12
14	Matheus	Ubner	Pilot	155739092	matheus.ubner(at)lufthansa.de	14
15	Lea	Nahmen	Stewardessa	155543506	lea.nahmen(at)lufthansa.de	7
5	Monika	Makowska	Stewardessa	668896445	mmakowska(at)lot.pl	7
7	Kamila	Pankowska	Stewardessa	668789667	kpankowska(at)lot.pl	7
10	Iwona	Waniek	Stewardessa	668021465	iwaniek(at)lot.pl	7

SQL>

Tabela Lot

Primary Key: NRREJSU

CREATE. Budowa tabeli Lot.

```
Firebird ISQL Tool
SQL> create table Lot (
CON> NRREJSU INTEGER NOT NULL PRIMARY KEY,
CON> PORTWYLOTU VARCHAR(20) NOT NULL,
CON> PORTPRZYLOTU VARCHAR (20) NOT NULL,
CON> DATAWYLOTU DATE NOT NULL,
CON> GODZINAWYLOTU TIME NOT NULL,
CON> DATAPRZYLOTU DATE NOT NULL,
CON> GODZINAPRZYLOTU TIME NOT NULL,
CON> TERMINAL VARCHAR(10) NOT NULL);
SQL> INSERT INTO LOT VALUES('1','Bydgoszcz','Warszawa','2018-12-01','8:55','2018-12-01','10:00','A');
SQL> INSERT INTO LOT VALUES('2','Warszawa','Nowy Jork','2018-12-01','9:00','2018-12-01','15:00','D');
SQL> INSERT INTO LOT VALUES('3','Warszawa','Katowice','2018-12-02','10:00','2018-12-02','11:30','A');
SQL> INSERT INTO LOT VALUES('4','Chicago','Warszawa','2018-12-02','10:20','2018-12-02','17:30','D');
SQL> INSERT INTO LOT VALUES('5','Warszawa','Monachium','2018-12-03','7:20','2018-12-03','9:30','C');
SQL> INSERT INTO LOT VALUES('5','Berlin','Warszawa','2018-12-03','11:20','2018-12-03','13:30','C');
```

INSERT. Dodawanie ostatnich rekordów do relacji Lot.

```
Firebird ISQL Tool
SQL> INSERT INTO LOT VALUES('14','Bombaj','Warszawa','2018-12-07','2:30','2018-12-07','10:15','E');
SQL> select*from lot;

  NRREJSU  PORTWYLOTU      PORTPRZYLOTU      DATAWYLOTU  GODZINAWYLOTU  DATAPRZYLOTU  GODZINAPRZYLOTU  TERMINAL
-----
1 Bydgoszcz      Warszawa
2 Warszawa      Nowy Jork
3 Warszawa      Katowice
4 Chicago        Warszawa
5 Warszawa      Monachium
6 Berlin         Warszawa
7 Krakow         Warszawa
8 Warszawa      Wroclaw
9 Tokjo          Warszawa
10 Bangalore     Warszawa
11 Warszawa      Gdansk
12 Warszawa      Pune
13 Waszyngton    Warszawa
14 Bombaj        Warszawa
15 Warszawa      Frankfurt

SQL> INSERT INTO LOT VALUES('15','Warszawa','Frankfurt','2018-12-08','6:30','2018-12-08','8:50','C');
SQL> select*from lot;

  NRREJSU  PORTWYLOTU      PORTPRZYLOTU      DATAWYLOTU  GODZINAWYLOTU  DATAPRZYLOTU  GODZINAPRZYLOTU  TERMINAL
-----
1 Bydgoszcz      Warszawa
2 Warszawa      Nowy Jork
3 Warszawa      Katowice
4 Chicago        Warszawa
5 Warszawa      Monachium
6 Berlin         Warszawa
7 Krakow         Warszawa
8 Warszawa      Wroclaw
9 Tokjo          Warszawa
10 Bangalore     Warszawa
11 Warszawa      Gdansk
12 Warszawa      Pune
13 Waszyngton    Warszawa
14 Bombaj        Warszawa
15 Warszawa      Frankfurt
```

BETWEEN. Wybór rejsów, których data wylotu jest pomiędzy 2 grudnia a 5 grudnia 2018.

Firebird ISQL Tool

```
SQL> select* from lot where datawylotu between '2018-12-02' and '2018-12-05';
```

NRREJSU	PORTWYLOTU	PORTPRZYLOTU	DATAWYLOTU	GODZINAWYLOTU	DATAPRZYLOTU	GODZINAPRZYLOTU	TERMINAL
3	Warszawa	Katowice	2018-12-02	10:00:00.0000	2018-12-02	11:30:00.0000	A
4	Chicago	Warszawa	2018-12-02	10:20:00.0000	2018-12-02	17:30:00.0000	D
5	Warszawa	Monachium	2018-12-03	07:20:00.0000	2018-12-03	09:30:00.0000	C
6	Berlin	Warszawa	2018-12-03	11:20:00.0000	2018-12-03	13:30:00.0000	C
7	Krakow	Warszawa	2018-12-04	15:20:00.0000	2018-12-04	17:30:00.0000	A
8	Warszawa	Wroclaw	2018-12-05	17:30:00.0000	2018-12-05	19:00:00.0000	A
9	Tokjo	Warszawa	2018-12-05	06:30:00.0000	2018-12-05	14:30:00.0000	E

SQL>

UPDATE. Zmiana portu wylotu i portu przylotu dla rejsu nr 10 i 12.

Firebird ISQL Tool

```
SQL> UPDATE LOT SET PORTWYLOTU = 'Toronto' WHERE NRREJSU = '10';
```

```
SQL> SELECT * FROM LOT;
```

NRREJSU	PORTWYLOTU	PORTPRZYLOTU	DATAWYLOTU	GODZINAWYLOTU	DATAPRZYLOTU	GODZINAPRZYLOTU	TERMINAL
1	Bydgoszcz	Warszawa	2018-12-01	08:55:00.0000	2018-12-01	10:00:00.0000	A
2	Warszawa	Nowy Jork	2018-12-01	09:00:00.0000	2018-12-01	15:00:00.0000	D
3	Warszawa	Katowice	2018-12-02	10:00:00.0000	2018-12-02	11:30:00.0000	A
4	Chicago	Warszawa	2018-12-02	10:20:00.0000	2018-12-02	17:30:00.0000	D
5	Warszawa	Monachium	2018-12-03	07:20:00.0000	2018-12-03	09:30:00.0000	C
6	Berlin	Warszawa	2018-12-03	11:20:00.0000	2018-12-03	13:30:00.0000	C
7	Krakow	Warszawa	2018-12-04	15:20:00.0000	2018-12-04	17:30:00.0000	A
8	Warszawa	Wroclaw	2018-12-05	17:30:00.0000	2018-12-05	19:00:00.0000	A
9	Tokjo	Warszawa	2018-12-05	06:30:00.0000	2018-12-05	14:30:00.0000	E
10	Toronto	Warszawa	2018-12-06	04:30:00.0000	2018-12-06	12:30:00.0000	E
11	Warszawa	Gdansk	2018-12-06	18:30:00.0000	2018-12-06	20:15:00.0000	A
12	Warszawa	Pune	2018-12-07	07:30:00.0000	2018-12-07	15:20:00.0000	E
13	Waszyngton	Warszawa	2018-12-07	05:30:00.0000	2018-12-07	12:15:00.0000	D
14	Bombaj	Warszawa	2018-12-07	02:30:00.0000	2018-12-07	10:15:00.0000	E
15	Warszawa	Frankfurt	2018-12-08	06:30:00.0000	2018-12-08	08:50:00.0000	C

```
SQL> UPDATE LOT SET PORTPRZYLOTU = 'Singapur' WHERE NRREJSU = '12';
```

```
SQL> SELECT * FROM LOT;
```

NRREJSU	PORTWYLOTU	PORTPRZYLOTU	DATAWYLOTU	GODZINAWYLOTU	DATAPRZYLOTU	GODZINAPRZYLOTU	TERMINAL
1	Bydgoszcz	Warszawa	2018-12-01	08:55:00.0000	2018-12-01	10:00:00.0000	A
2	Warszawa	Nowy Jork	2018-12-01	09:00:00.0000	2018-12-01	15:00:00.0000	D
3	Warszawa	Katowice	2018-12-02	10:00:00.0000	2018-12-02	11:30:00.0000	A
4	Chicago	Warszawa	2018-12-02	10:20:00.0000	2018-12-02	17:30:00.0000	D
5	Warszawa	Monachium	2018-12-03	07:20:00.0000	2018-12-03	09:30:00.0000	C
6	Berlin	Warszawa	2018-12-03	11:20:00.0000	2018-12-03	13:30:00.0000	C
7	Krakow	Warszawa	2018-12-04	15:20:00.0000	2018-12-04	17:30:00.0000	A
8	Warszawa	Wroclaw	2018-12-05	17:30:00.0000	2018-12-05	19:00:00.0000	A
9	Tokjo	Warszawa	2018-12-05	06:30:00.0000	2018-12-05	14:30:00.0000	E
10	Toronto	Warszawa	2018-12-06	04:30:00.0000	2018-12-06	12:30:00.0000	E
11	Warszawa	Gdansk	2018-12-06	18:30:00.0000	2018-12-06	20:15:00.0000	A
12	Warszawa	Singapur	2018-12-07	07:30:00.0000	2018-12-07	15:20:00.0000	E
13	Waszyngton	Warszawa	2018-12-07	05:30:00.0000	2018-12-07	12:15:00.0000	D
14	Bombaj	Warszawa	2018-12-07	02:30:00.0000	2018-12-07	10:15:00.0000	E
15	Warszawa	Frankfurt	2018-12-08	06:30:00.0000	2018-12-08	08:50:00.0000	C

SQL>

UPDATE. Zmiana portu przylotu dla rejsu nr 8 z Wrocław na Sharm EL Sheikh.

Firebird ISQL Tool

```
SQL> UPDATE LOT SET PORTPRZYLOTU = 'SHARM EL SHEIKH' WHERE NRREJSU = '8';
SQL> SELECT * FROM LOT;
```

NRREJSU	PORTWYLOTU	PORTPRZYLOTU	DATAWYLOTU	GODZINAWYLOTU	DATAPRZYLOTU	GODZINAPRZYLOTU	TERMINAL
1	Bydgoszcz	Warszawa	2018-12-01	08:55:00.0000	2018-12-01	10:00:00.0000	A
2	Warszawa	Nowy Jork	2018-12-01	09:00:00.0000	2018-12-01	15:00:00.0000	D
3	Warszawa	Katowice	2018-12-02	10:00:00.0000	2018-12-02	11:30:00.0000	A
4	Chicago	Warszawa	2018-12-02	10:20:00.0000	2018-12-02	17:30:00.0000	D
5	Warszawa	Monachium	2018-12-03	07:20:00.0000	2018-12-03	09:30:00.0000	C
6	Berlin	Warszawa	2018-12-03	11:20:00.0000	2018-12-03	13:30:00.0000	C
7	Krakow	Warszawa	2018-12-04	15:20:00.0000	2018-12-04	17:30:00.0000	A
8	Warszawa	SHARM EL SHEIKH	2018-12-05	17:30:00.0000	2018-12-05	19:00:00.0000	A
9	Tokjo	Warszawa	2018-12-05	06:30:00.0000	2018-12-05	14:30:00.0000	E
10	Toronto	Warszawa	2018-12-06	04:30:00.0000	2018-12-06	12:30:00.0000	E
11	Warszawa	Gdansk	2018-12-06	18:30:00.0000	2018-12-06	20:15:00.0000	A
12	Warszawa	Singapur	2018-12-07	07:30:00.0000	2018-12-07	15:20:00.0000	E
13	Waszyngton	Warszawa	2018-12-07	05:30:00.0000	2018-12-07	12:15:00.0000	D
14	Bombaj	Warszawa	2018-12-07	02:30:00.0000	2018-12-07	10:15:00.0000	E
15	Warszawa	Frankfurt	2018-12-08	06:30:00.0000	2018-12-08	08:50:00.0000	C

SQL>

Tabela Bilet

Primary Key: NRBILETU

Foreign Key: NRKLIENTA (KLIENT)

Foreign Key: NRREJSU (LOT)


```

SQL> CREATE TABLE BILET (
CON> NRBILETU INTEGER NOT NULL PRIMARY KEY,
CON> NRKLIENTA INTEGER NOT NULL,
CON> NRREJSU INTEGER NOT NULL,
CON> DATAZAKUPU DATE NOT NULL,
CON> MIEJSCEZAKUPU VARCHAR (15) NOT NULL,
CON> TYPBILETU VARCHAR (15) NOT NULL,
CON> CENA DECIMAL (5, 2) NOT NULL,
CON> NRMIEJSKA INTEGER NOT NULL,
CON> CONSTRAINT FK7 FOREIGN KEY (NRKLIENTA) REFERENCES KLIENT (NRKLIENTA)
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT,
CON> CONSTRAINT FK8 FOREIGN KEY (NRREJSU) REFERENCES LOT (NRREJSU)
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT);
SQL> INSERT INTO BILET VALUES ('1','10','7','2018-11-20','Warszawa','Biznes','1700','37');
SQL> INSERT INTO BILET VALUES ('2','9','5','2018-10-27','Warszawa','Biznes','1300','98');
SQL> INSERT INTO BILET VALUES ('3','1','1','2018-11-17','Bydgoszcz','Ekonomiczny','450','23');
SQL> INSERT INTO BILET VALUES ('4','2','6','2018-11-28','Warszawa','Ekonomiczny','750','15');
SQL> INSERT INTO BILET VALUES ('5','8','12','2018-10-15','Warszawa','Biznes','2500','128');
SQL> INSERT INTO BILET VALUES ('6','3','3','2018-11-12','Katowice','Ekonomiczny','650','18');
SQL> INSERT INTO BILET VALUES ('7','7','4','2018-10-29','Warszawa','Biznes','2800','257');
SQL> INSERT INTO BILET VALUES ('8','4','11','2018-12-01','Warszawa','Ekonomiczny','600','37');
SQL> INSERT INTO BILET VALUES ('9','5','15','2018-11-10','Warszawa','Biznes','1200','174');
SQL> INSERT INTO BILET VALUES ('10','6','8','2018-10-19','Warszawa','Biznes','2900','210');
SQL> INSERT INTO BILET VALUES ('11','14','9','2018-11-08','Tokjo','Biznes','2800','98');
SQL> INSERT INTO BILET VALUES ('12','15','2','2018-11-23','Warszawa','Biznes','2300','175');
SQL> INSERT INTO BILET VALUES ('13','13','10','2018-12-02','Toronto','Biznes','2600','235');
SQL> INSERT INTO BILET VALUES ('14','12','13','2018-11-17','Waszyngton','Biznes','2400','127');
SQL> INSERT INTO BILET VALUES ('15','11','14','2018-11-20','Bombaj','Biznes','2200','143');
SQL> SELECT * FROM BILET;

```

NRBILETU	NRKLIENTA	NRREJSU	DATAZAKUPU	MIEJSCEZAKUPU	TYPBILETU	CENA	NRMIEJSKA
1	10	7	2018-11-20	Warszawa	Biznes	1700.00	37
2	9	5	2018-10-27	Warszawa	Biznes	1300.00	98
3	1	1	2018-11-17	Bydgoszcz	Ekonomiczny	450.00	23
4	2	6	2018-11-28	Warszawa	Ekonomiczny	750.00	15
5	8	12	2018-10-15	Warszawa	Biznes	2500.00	128
6	3	3	2018-11-12	Katowice	Ekonomiczny	650.00	18
7	7	4	2018-10-29	Warszawa	Biznes	2800.00	257
8	4	11	2018-12-01	Warszawa	Ekonomiczny	600.00	37
9	5	15	2018-11-10	Warszawa	Biznes	1200.00	174
10	6	8	2018-10-19	Warszawa	Biznes	2900.00	210
11	14	9	2018-11-08	Tokjo	Biznes	2800.00	98
12	15	2	2018-11-23	Warszawa	Biznes	2300.00	175
13	13	10	2018-12-02	Toronto	Biznes	2600.00	235
14	12	13	2018-11-17	Waszyngton	Biznes	2400.00	127
15	11	14	2018-11-20	Bombaj	Biznes	2200.00	143

SQL>

SELECT MAX i MIN. Wybór biletu, który kosztuje najmniej i najwięcej.

```
SQL> SELECT * FROM BILET WHERE CENA = (SELECT MIN(CENA) FROM BILET);
```

NRBILETU	NRKLIENTA	NRREJSU	DATAZAKUPU	MIEJSCEZAKUPU	TYPBILETU	CENA	NRMIEJSKA
3	1	1	2018-11-17	Bydgoszcz	Ekonomiczny	450.00	23

```
SQL> SELECT * FROM BILET WHERE CENA = (SELECT MAX(CENA) FROM BILET);
```

NRBILETU	NRKLIENTA	NRREJSU	DATAZAKUPU	MIEJSCEZAKUPU	TYPBILETU	CENA	NRMIEJSKA
10	6	8	2018-10-19	Warszawa	Biznes	2900.00	210

SQL>

EXISTS. Wybór klientów których cena biletu przekroczyła 2000 zł w relacji Klient, Bilet.

```
WybierzFirebird ISQL Tool
SQL> SELECT IMIE, NAZWISKO
CON> FROM KLIENT
CON> WHERE EXISTS (SELECT CENA FROM BILET WHERE NRKLIENTA = KLIENT.NRKLIENTA AND CENA > 2000);

IMIE      NAZWISKO
=====
Maciej     Maj
Joanna     Wrzesien
Katarzyna  Lato
James      Rodrigez
David      Coperfield
Vijay      Singh
Shirioku   Nagasaki
Katherine  Johns
```

Modyfikacja tabeli Bilet. Atrybuty podklasy Biznes – Nrmiejsca oraz Ekonomiczny – Cena zostały włączone do nadklasy, natomiast w ich miejsce zostały utworzone nowe atrybuty podklas, które w większym stopniu odpowiadają specyfice podklas.

```
Firebird ISQL Tool
SQL> SELECT* FROM BILET;

NRBILETU  NRKLIENTA  NRREJSU  DATAZAKUPU  MIEJSCEZAKUPU  TYPBILETU  CENA  NRMIEJSCA
-----
1          10         7  2018-11-20  Warszawa      Biznes      1700.00  37
2          9          5  2018-10-27  Warszawa      Biznes      1300.00  98
3          1          1  2018-11-17  Bydgoszcz     Ekonomiczny  450.00   23
4          2          6  2018-11-28  Warszawa      Ekonomiczny  750.00   15
5          8         12  2018-10-15  Warszawa      Biznes      2500.00  128
6          3          3  2018-11-12  Katowice      Ekonomiczny  650.00   18
7          7          4  2018-10-29  Warszawa      Biznes      2800.00  257
8          4         11  2018-12-01  Warszawa      Ekonomiczny  600.00   37
9          5         15  2018-11-10  Warszawa      Biznes      1200.00  174
10         6          8  2018-10-19  Warszawa      Biznes      2900.00  210
11         14         9  2018-11-08  Tokyo         Biznes      2800.00  98
12         15         2  2018-11-23  Warszawa      Biznes      2300.00  175
13         13         10  2018-12-02  Toronto       Biznes      2600.00  235
14         12         13  2018-11-17  Waszyngton    Biznes      2400.00  127
15         11         14  2018-11-20  Bombaj        Biznes      2200.00  143

SQL>
```


DROP column. Usunięcie kolumny TypBiletu.

```
Firebird ISQL Tool
SQL> ALTER TABLE BILET DROP TYPBILETU;
SQL> SELECT* FROM BILET;
```

NRBILETU	NRKLIENTA	NRREJSU	DATAZAKUPU	MIEJSCEZAKUPU	CENA	NRMIEJSCA
1	10	7	2018-11-20	Warszawa	1700.00	37
2	9	5	2018-10-27	Warszawa	1300.00	98
3	1	1	2018-11-17	Bydgoszcz	450.00	23
4	2	6	2018-11-28	Warszawa	750.00	15
5	8	12	2018-10-15	Warszawa	2500.00	128
6	3	3	2018-11-12	Katowice	650.00	18
7	7	4	2018-10-29	Warszawa	2800.00	257
8	4	11	2018-12-01	Warszawa	600.00	37
9	5	15	2018-11-10	Warszawa	1200.00	174
10	6	8	2018-10-19	Warszawa	2900.00	210
11	14	9	2018-11-08	Tokjo	2800.00	98
12	15	2	2018-11-23	Warszawa	2300.00	175
13	13	10	2018-12-02	Toronto	2600.00	235
14	12	13	2018-11-17	Waszyngton	2400.00	127
15	11	14	2018-11-20	Bombaj	2200.00	143

```
SQL>
```

ADD column. Dodanie kolumny TypBiletu z domyślnie określoną wartością „Ekonomiczny”.

```
Firebird ISQL Tool
SQL> ALTER TABLE BILET ADD TYPBILETU VARCHAR (15) DEFAULT 'Ekonomiczny' NOT NULL;
SQL> SELECT* FROM BILET;
```

NRBILETU	NRKLIENTA	NRREJSU	DATAZAKUPU	MIEJSCEZAKUPU	CENA	NRMIEJSCA	TYPBILETU
1	10	7	2018-11-20	Warszawa	1700.00	37	Ekonomiczny
2	9	5	2018-10-27	Warszawa	1300.00	98	Ekonomiczny
3	1	1	2018-11-17	Bydgoszcz	450.00	23	Ekonomiczny
4	2	6	2018-11-28	Warszawa	750.00	15	Ekonomiczny
5	8	12	2018-10-15	Warszawa	2500.00	128	Ekonomiczny
6	3	3	2018-11-12	Katowice	650.00	18	Ekonomiczny
7	7	4	2018-10-29	Warszawa	2800.00	257	Ekonomiczny
8	4	11	2018-12-01	Warszawa	600.00	37	Ekonomiczny
9	5	15	2018-11-10	Warszawa	1200.00	174	Ekonomiczny
10	6	8	2018-10-19	Warszawa	2900.00	210	Ekonomiczny
11	14	9	2018-11-08	Tokjo	2800.00	98	Ekonomiczny
12	15	2	2018-11-23	Warszawa	2300.00	175	Ekonomiczny
13	13	10	2018-12-02	Toronto	2600.00	235	Ekonomiczny
14	12	13	2018-11-17	Waszyngton	2400.00	127	Ekonomiczny
15	11	14	2018-11-20	Bombaj	2200.00	143	Ekonomiczny

```
SQL>
```

UPDATE. Ustawienie klasy Biznes dla ceny biletu >= 2000 zł.

Firebird ISQL Tool

```
SQL> UPDATE BILET SET TYPBILETU = 'Biznes' WHERE CENA >= 2000;
SQL> SELECT* FROM BILET;
```

NRBILETU	NRKLIENTA	NRREJSU	DATAZAKUPU	MIEJSCEZAKUPU	CENA	NRMIEJSCA	TYPBILETU
1	10	7	2018-11-20	Warszawa	1700.00	37	Ekonomiczny
2	9	5	2018-10-27	Warszawa	1300.00	98	Ekonomiczny
3	1	1	2018-11-17	Bydgoszcz	450.00	23	Ekonomiczny
4	2	6	2018-11-28	Warszawa	750.00	15	Ekonomiczny
5	8	12	2018-10-15	Warszawa	2500.00	128	Biznes
6	3	3	2018-11-12	Katowice	650.00	18	Ekonomiczny
7	7	4	2018-10-29	Warszawa	2800.00	257	Biznes
8	4	11	2018-12-01	Warszawa	600.00	37	Ekonomiczny
9	5	15	2018-11-10	Warszawa	1200.00	174	Ekonomiczny
10	6	8	2018-10-19	Warszawa	2900.00	210	Biznes
11	14	9	2018-11-08	Tokjo	2800.00	98	Biznes
12	15	2	2018-11-23	Warszawa	2300.00	175	Biznes
13	13	10	2018-12-02	Toronto	2600.00	235	Biznes
14	12	13	2018-11-17	Waszyngton	2400.00	127	Biznes
15	11	14	2018-11-20	Bombaj	2200.00	143	Biznes

SQL>

ADD column. Dodanie dwóch atrybutów podklas – Czasopismo i posiłek

Firebird ISQL Tool

```
SQL> ALTER TABLE BILET ADD CZASOPISMO VARCHAR (15);
SQL> ALTER TABLE BILET ADD POSILEK VARCHAR (20);
SQL> SELECT* FROM BILET;
```

NRBILETU	NRKLIENTA	NRREJSU	DATAZAKUPU	MIEJSCEZAKUPU	CENA	NRMIEJSCA	TYPBILETU	CZASOPISMO	POSILEK
1	10	7	2018-11-20	Warszawa	1700.00	37	Ekonomiczny	<null>	<null>
2	9	5	2018-10-27	Warszawa	1300.00	98	Ekonomiczny	<null>	<null>
3	1	1	2018-11-17	Bydgoszcz	450.00	23	Ekonomiczny	<null>	<null>
4	2	6	2018-11-28	Warszawa	750.00	15	Ekonomiczny	<null>	<null>
5	8	12	2018-10-15	Warszawa	2500.00	128	Biznes	<null>	<null>
6	3	3	2018-11-12	Katowice	650.00	18	Ekonomiczny	<null>	<null>
7	7	4	2018-10-29	Warszawa	2800.00	257	Biznes	<null>	<null>
8	4	11	2018-12-01	Warszawa	600.00	37	Ekonomiczny	<null>	<null>
9	5	15	2018-11-10	Warszawa	1200.00	174	Ekonomiczny	<null>	<null>
10	6	8	2018-10-19	Warszawa	2900.00	210	Biznes	<null>	<null>
11	14	9	2018-11-08	Tokjo	2800.00	98	Biznes	<null>	<null>
12	15	2	2018-11-23	Warszawa	2300.00	175	Biznes	<null>	<null>
13	13	10	2018-12-02	Toronto	2600.00	235	Biznes	<null>	<null>
14	12	13	2018-11-17	Waszyngton	2400.00	127	Biznes	<null>	<null>
15	11	14	2018-11-20	Bombaj	2200.00	143	Biznes	<null>	<null>

SQL>

UPDATE. Ustawienie atrybutów podklas.

Firebird ISQL Tool

```
SQL> UPDATE BILET SET CZASOPISMO = '-' WHERE TYPBILETU = 'Biznes';
SQL> SELECT* FROM BILET;
```

NRBILETU	NRKLIENTA	NRREJSU	DATAZAKUPU	MIEJSCEZAKUPU	CENA	NRMIEJSCA	TYPBILETU	CZASOPISMO	POSILEK
1	10	7	2018-11-20	Warszawa	1700.00	37	Ekonomiczny	Wprost	-
2	9	5	2018-10-27	Warszawa	1300.00	98	Ekonomiczny	Wprost	-
3	1	1	2018-11-17	Bydgoszcz	450.00	23	Ekonomiczny	Wprost	-
4	2	6	2018-11-28	Warszawa	750.00	15	Ekonomiczny	Wprost	-
5	8	12	2018-10-15	Warszawa	2500.00	128	Biznes	-	Risotto
6	3	3	2018-11-12	Katowice	650.00	18	Ekonomiczny	Wprost	-
7	7	4	2018-10-29	Warszawa	2800.00	257	Biznes	-	Risotto
8	4	11	2018-12-01	Warszawa	600.00	37	Ekonomiczny	Wprost	-
9	5	15	2018-11-10	Warszawa	1200.00	174	Ekonomiczny	Wprost	-
10	6	8	2018-10-19	Warszawa	2900.00	210	Biznes	-	Risotto
11	14	9	2018-11-08	Tokjo	2800.00	98	Biznes	-	Risotto
12	15	2	2018-11-23	Warszawa	2300.00	175	Biznes	-	Risotto
13	13	10	2018-12-02	Toronto	2600.00	235	Biznes	-	Risotto
14	12	13	2018-11-17	Waszyngton	2400.00	127	Biznes	-	Risotto
15	11	14	2018-11-20	Bombaj	2200.00	143	Biznes	-	Risotto

SQL>

Tabela Rezerwacja

Primary Key: NRREZERWACJI

Foreign Key: NRKLIENTA (Klient)

Foreign Key: NRPRACOWNIKA (Pracownik)

CREATE. Tworzenie tabeli Rezerwacja i dodawanie nowych rekordów.

Firebird ISQL Tool

```
SQL> CREATE TABLE REZERWACJA (  
CON> NRREZERWACJI INTEGER NOT NULL PRIMARY KEY,  
CON> NRKLIENTA INTEGER NOT NULL,  
CON> NRPRACOWNIKA INTEGER NOT NULL,  
CON> NRREJSU INTEGER NOT NULL,  
CON> DATAREZERWACJI DATE NOT NULL,  
CON> CONSTRAINT FK5 FOREIGN KEY (NRKLIENTA) REFERENCES KLIENT (NRKLIENTA)  
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT,  
CON> CONSTRAINT FK6 FOREIGN KEY (NRPRACOWNIKA) REFERENCES PRACOWNIK (NRPRACOWNIKA)  
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT);  
SQL> INSERT INTO REZERWACJA VALUES ('1','1','2','1','2018-11-10');  
SQL> INSERT INTO REZERWACJA VALUES ('2','2','6','6','2018-11-23');  
SQL> INSERT INTO REZERWACJA VALUES ('3','3','11','3','2018-11-07');  
SQL> INSERT INTO REZERWACJA VALUES ('4','4','13','11','2018-11-15');  
SQL> INSERT INTO REZERWACJA VALUES ('5','5','2','15','2018-10-15');  
SQL> INSERT INTO REZERWACJA VALUES ('6','6','6','8','2018-10-10');  
SQL> INSERT INTO REZERWACJA VALUES ('7','7','11','4','2018-10-12');  
SQL> INSERT INTO REZERWACJA VALUES ('8','8','13','12','2018-09-12');  
SQL> INSERT INTO REZERWACJA VALUES ('9','9','2','5','2018-10-02');  
SQL> INSERT INTO REZERWACJA VALUES ('10','10','6','7','2018-10-07');  
SQL> INSERT INTO REZERWACJA VALUES ('11','11','11','14','2018-10-18');  
SQL> INSERT INTO REZERWACJA VALUES ('12','12','13','13','2018-10-01');  
SQL> INSERT INTO REZERWACJA VALUES ('13','13','2','10','2018-11-19');  
SQL> INSERT INTO REZERWACJA VALUES ('14','14','6','9','2018-09-19');  
SQL> INSERT INTO REZERWACJA VALUES ('15','15','11','2','2018-11-20');  
SQL> SELECT * FROM REZERWACJA;
```

NRREZERWACJI	NRKLIENTA	NRPRACOWNIKA	NRREJSU	DATAREZERWACJI
1	1	2	1	2018-11-10
2	2	6	6	2018-11-23
3	3	11	3	2018-11-07
4	4	13	11	2018-11-15
5	5	2	15	2018-10-15
6	6	6	8	2018-10-10
7	7	11	4	2018-10-12
8	8	13	12	2018-09-12
9	9	2	5	2018-10-02
10	10	6	7	2018-10-07
11	11	11	14	2018-10-18
12	12	13	13	2018-10-01
13	13	2	10	2018-11-19
14	14	6	9	2018-09-19
15	15	11	2	2018-11-20

SQL>

BETWEEN i ORDER BY. Wybór rezerwacji dokonanych w listopadzie.

```
Firebird ISQL Tool
SQL> SELECT * FROM REZERWACJA WHERE DATAREZERWACJI BETWEEN '2018-11-01' AND '2018-11-30' ORDER BY DATAREZERWACJI ASC;
NRREZERWACJI  NRKLIENTA  NRPRACOWNIKA  NRREJSU  DATAREZERWACJI
=====
          3           3           11           3  2018-11-07
          1           1            2           1  2018-11-10
          4           4           13          11  2018-11-15
         13          13            2          10  2018-11-19
         15          15           11           2  2018-11-20
          2           2            6           6  2018-11-23
SQL>
```

Tabela Bagaz

Primary Key: NRBAGAZU

Foreign Key: NRKLIENTA (Klient)

Foreign Key: NRREJSU (Lot)

CREATE. Tabela Bagaz

```
Firebird ISQL Tool
SQL> CREATE TABLE BAGAZ (
CON> NRBAGAZU INTEGER NOT NULL PRIMARY KEY,
CON> NRKLIENTA INTEGER NOT NULL,
CON> NRREJSU INTEGER NOT NULL,
CON> WAGABAGAZU DECIMAL (5, 2) NOT NULL,
CON> ROZMIARBAGAZU CHAR (10) NOT NULL,
CON> CONSTRAINT FK1 FOREIGN KEY (NRKLIENTA) REFERENCES KLIENT (NRKLIENTA)
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT,
CON> CONSTRAINT FK2 FOREIGN KEY (NRREJSU) REFERENCES LOT (NRREJSU)
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT);
SQL> INSERT INTO BAGAZ VALUES ('1','1','1','5','75x30');
SQL> select * from bagaz;
NRBAGAZU  NRKLIENTA  NRREJSU  WAGABAGAZU  ROZMIARBAGAZU
=====
          1           1           1          5.00  75x30

SQL> INSERT INTO BAGAZ VALUES ('2','10','7','15','100x40');
SQL> INSERT INTO BAGAZ VALUES ('3','9','5','10','90x30');
SQL> INSERT INTO BAGAZ VALUES ('4','2','6','10','80x40');
SQL> INSERT INTO BAGAZ VALUES ('5','8','12','30','100x50');
SQL> INSERT INTO BAGAZ VALUES ('6','3','3','12','50x30');
SQL> INSERT INTO BAGAZ VALUES ('7','7','4','25','100x45');
SQL> INSERT INTO BAGAZ VALUES ('8','4','11','5','50x25');
SQL> INSERT INTO BAGAZ VALUES ('9','5','15','15','90x35');
SQL> INSERT INTO BAGAZ VALUES ('10','6','8','30','100x35');
SQL> INSERT INTO BAGAZ VALUES ('11','14','9','25','100x40');
SQL> INSERT INTO BAGAZ VALUES ('12','15','2','40','120x50');
SQL> INSERT INTO BAGAZ VALUES ('13','13','10','27','100x50');
SQL> INSERT INTO BAGAZ VALUES ('14','12','13','20','90x30');
SQL> INSERT INTO BAGAZ VALUES ('15','11','14','10','50x20');
SQL> select * from bagaz;
NRBAGAZU  NRKLIENTA  NRREJSU  WAGABAGAZU  ROZMIARBAGAZU
=====
          1           1           1          5.00  75x30
          2          10           7          15.00  100x40
          3           9           5          10.00  90x30
          4           2           6          10.00  80x40
          5           8          12          30.00  100x50
          6           3           3          12.00  50x30
          7           7           4          25.00  100x45
          8           4          11           5.00  50x25
          9           5          15          15.00  90x35
         10           6           8          30.00  100x35
         11          14           9          25.00  100x40
         12          15           2          40.00  120x50
         13          13          10          27.00  100x50
         14          12          13          20.00  90x30
         15          11          14          10.00  50x20
```

SELECT MAX. Wybór klienta, który ma najcięższy bagaż.

```
Firebird ISQL Tool
SQL> select nrklienta, wagabagazu from bagaz where wagabagazu = (select max(wagabagazu) from bagaz);

   NRKLIENTA   WAGABAGAZU
=====
      15         40.00

SQL>
```

Tabela Samolot

Primary Key: NRSAMOTU

Foreign Key: NRREJSU (Lot)

Foreign Key: NRPRACOWNIKA (Pracownik)

CREATE. Budowa tabeli Samolot

```
Firebird ISQL Tool
SQL> CREATE TABLE SAMOLOT (
CON> NRSAMOTU INTEGER NOT NULL PRIMARY KEY,
CON> NRREJSU INTEGER NOT NULL,
CON> NRPRACOWNIKA INTEGER NOT NULL,
CON> LINIALOTNICZA VARCHAR (25) NOT NULL,
CON> PRODUCENT VARCHAR (15) NOT NULL,
CON> MODEL VARCHAR (15) NOT NULL,
CON> KOLOR VARCHAR (25),
CON> CONSTRAINT FK3 FOREIGN KEY (NRREJSU) REFERENCES LOT (NRREJSU)
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT,
CON> CONSTRAINT FK4 FOREIGN KEY (NRPRACOWNIKA) REFERENCES PRACOWNIK (NRPRACOWNIKA)
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT);
SQL> INSERT INTO SAMOLOT VALUES ('1','1','1','LOT','Embraer','ERJ 145 XR','Bialo-niebieski');
SQL> INSERT INTO SAMOLOT VALUES ('2','2','3','United Airlines','Boeing','737 MAX','Bialo-niebieski');
SQL> INSERT INTO SAMOLOT VALUES ('3','3','9','LOT','Embraer','ERJ 145 XR','Bialo-niebieski');
SQL> INSERT INTO SAMOLOT VALUES ('4','4','9','LOT','Boeing','737-800','Bialo-niebieski');
SQL> INSERT INTO SAMOLOT VALUES ('5','5','9','Lufthansa','Embraer','E195','Bialy');
SQL> INSERT INTO SAMOLOT VALUES ('6','6','9','Lufthansa','Airbus','A320 neo','Bialy');
SQL> INSERT INTO SAMOLOT VALUES ('7','7','5','LOT','Embraer','E195','Bialo-nienieski');
SQL> INSERT INTO SAMOLOT VALUES ('8','8','10','LOT','Embraer','E195','Bialo-nienieski');
SQL> INSERT INTO SAMOLOT VALUES ('9','9','15','United Airlines','Boeing','737 MAX','Bialo-nienieski');
SQL> INSERT INTO SAMOLOT VALUES ('10','10','12','Air Canada','Boeing','787-900','Bialo-czerwony');
SQL> INSERT INTO SAMOLOT VALUES ('11','11','7','LOT','Embraer','E195','Bialo-niebieski');
SQL> INSERT INTO SAMOLOT VALUES ('12','12','5','Singapore Airlines','Boeing','787-800','Bialo-niebieski');
SQL> INSERT INTO SAMOLOT VALUES ('13','13','8','United Airlines','Boeing','737 MAX','Bialo-niebieski');
SQL> INSERT INTO SAMOLOT VALUES ('14','14','1','LOT','Boeing','737-800','Bialo-niebieski');
SQL> INSERT INTO SAMOLOT VALUES ('15','15','14','Lufthansa','Embraer','E195','Bialy');
```

Firebird ISQL Tool

```
SQL> SELECT * FROM SAMOLOT;
```

NRSAMOTU	NRREJSU	NRPRACOWNIKA	LINIALOTNICZA	PRODUCENT	MODEL	KOLOR
1	1	1	LOT	Embraer	ERJ 145 XR	Bialo-niebieski
2	2	3	United Airlines	Boeing	737 MAX	Bialo-niebieski
3	3	9	LOT	Embraer	ERJ 145 XR	Bialo-niebieski
4	4	9	LOT	Boeing	737-800	Bialo-niebieski
5	5	9	Lufthansa	Embraer	E195	Bialy
6	6	9	Lufthansa	Airbus	A320 neo	Bialy
7	7	5	LOT	Embraer	E195	Bialo-nienieski
8	8	10	LOT	Embraer	E195	Bialo-nienieski
9	9	15	United Airlines	Boeing	737 MAX	Bialo-nienieski
10	10	12	Air Canada	Boeing	787-900	Bialo-czerwony
11	11	7	LOT	Embraer	E195	Bialo-niebieski
12	12	5	Singapore Airlines	Boeing	787-800	Bialo-niebieski
13	13	8	United Airlines	Boeing	737 MAX	Bialo-niebieski
14	14	1	LOT	Boeing	737-800	Bialo-niebieski
15	15	14	Lufthansa	Embraer	E195	Bialy

SQL>

UPDATE. Zmiana nr pracownika dla samolotów 5 i 6 obsługujących rejsy 5 i 6.

Firebird ISQL Tool

```
SQL> UPDATE SAMOLOT SET NRPRACOWNIKA = '12' WHERE NRSAMOTU = '5';
SQL> UPDATE SAMOLOT SET NRPRACOWNIKA = '14' WHERE NRSAMOTU = '6';
SQL> SELECT * FROM SAMOLOT;
```

NRSAMOTU	NRREJSU	NRPRACOWNIKA	LINIALOTNICZA	PRODUCENT	MODEL	KOLOR
1	1	1	LOT	Embraer	ERJ 145 XR	Bialo-niebieski
2	2	3	United Airlines	Boeing	737 MAX	Bialo-niebieski
3	3	9	LOT	Embraer	ERJ 145 XR	Bialo-niebieski
4	4	9	LOT	Boeing	737-800	Bialo-niebieski
5	5	12	Lufthansa	Embraer	E195	Bialy
6	6	14	Lufthansa	Airbus	A320 neo	Bialy
7	7	5	LOT	Embraer	E195	Bialo-niebieski
8	8	10	LOT	Embraer	E195	Bialo-niebieski
9	9	15	United Airlines	Boeing	737 MAX	Bialo-niebieski
10	10	12	Air Canada	Boeing	787-900	Bialo-czerwony
11	11	7	LOT	Embraer	E195	Bialo-niebieski
12	12	5	Singapore Airlines	Boeing	787-800	Bialo-niebieski
13	13	8	United Airlines	Boeing	737 MAX	Bialo-niebieski
14	14	1	LOT	Boeing	737-800	Bialo-niebieski
15	15	14	Lufthansa	Embraer	E195	Bialy

UPDATE. Zmiana producenta i modelu samolotu dla rejsu nr 8.

Firebird ISQL Tool

```
SQL> UPDATE SAMOLOT SET PRODUCENT = 'Boeing', MODEL = '787-800' WHERE NRREJSU = '8';
SQL> SELECT * FROM SAMOLOT;
```

NRSAMOTU	NRREJSU	NRPRACOWNIKA	LINIALOTNICZA	PRODUCENT	MODEL	KOLOR
1	1	1	LOT	Embraer	ERJ 145 XR	Bialo-niebieski
2	2	3	United Airlines	Boeing	737 MAX	Bialo-niebieski
3	3	9	LOT	Embraer	ERJ 145 XR	Bialo-niebieski
4	4	9	LOT	Boeing	737-800	Bialo-niebieski
5	5	12	Lufthansa	Embraer	E195	Bialy
6	6	14	Lufthansa	Airbus	A320 neo	Bialy
7	7	5	LOT	Embraer	E195	Bialo-niebieski
8	8	10	LOT	Boeing	787-800	Bialo-niebieski
9	9	15	United Airlines	Boeing	737 MAX	Bialo-niebieski
10	10	12	Air Canada	Boeing	787-900	Bialo-czerwony
11	11	7	LOT	Embraer	E195	Bialo-niebieski
12	12	5	Singapore Airlines	Boeing	787-800	Bialo-niebieski
13	13	8	United Airlines	Boeing	737 MAX	Bialo-niebieski
14	14	1	LOT	Boeing	737-800	Bialo-niebieski
15	15	14	Lufthansa	Embraer	E195	Bialy

SQL>

ANY. Wykaz pracowników obsługujących linie lotnicze LOT w relacji Pracownik, Samolot

Firebird ISQL Tool

```
SQL> SELECT * FROM PRACOWNIK
CON> WHERE NRPRACOWNIKA = ANY(SELECT NRPRACOWNIKA FROM SAMOLOT WHERE LINIALOTNICZA = 'LOT');
```

NRPRACOWNIKA	IMIE	NAZWISKO	STANOWISKO	TELEFON	EMAIL	NRSZEFA
1	Dagmara	Biskup	Pilot	668223657	dbiskup(at)lot.pl	3
5	Monika	Makowska	Stewardessa	668896445	mmakowska(at)lot.pl	7
7	Kamila	Pankowska	Stewardessa	668789667	kpankowska(at)lot.pl	7
9	Adam	Krupa	Pilot	668590021	akrupa(at)lot.pl	3
10	Iwona	Waniek	Stewardessa	668021465	iwaniek(at)lot.pl	7

SQL>

GROUP BY. Wykaz lini lotniczych

Firebird ISQL Tool

```
SQL> SELECT LINIALOTNICZA
CON> FROM SAMOLOT
CON> GROUP BY LINIALOTNICZA;
```

LINIALOTNICZA
Air Canada
LOT
Lufthansa
Singapore Airlines
United Airlines

SQL>

Tabela Płatność

Primary Key: NRPLATNOSCI

Foreign Key: NRKLIENTA (Klient)

Foreign Key: NRREZERWACJI (Rezerwacja)


```
SQL> CREATE TABLE PLATNOSC (
CON> NRPLATNOSCI INTEGER NOT NULL PRIMARY KEY,
CON> NRKLIENTA INTEGER NOT NULL,
CON> NRREZERWACJI INTEGER NOT NULL,
CON> KWOTA DECIMAL (5, 2) NOT NULL,
CON> DATAPLATNOSCI DATE NOT NULL,
CON> CONSTRAINT FK9 FOREIGN KEY (NRKLIENTA) REFERENCES KLIENT (NRKLIENTA)
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT,
CON> CONSTRAINT FK10 FOREIGN KEY (NRREZERWACJI) REFERENCES REZERWACJA (NRREZERWACJI)
CON> ON UPDATE CASCADE ON DELETE SET DEFAULT);
SQL> INSERT INTO PLATNOSC VALUES ('1','1','1','450','2018-11-20');
SQL> INSERT INTO PLATNOSC VALUES ('2','2','2','750','2018-12-04');
SQL> INSERT INTO PLATNOSC VALUES ('3','3','3','650','2018-11-12');
SQL> INSERT INTO PLATNOSC VALUES ('4','4','4','600','2018-12-01');
SQL> INSERT INTO PLATNOSC VALUES ('5','5','5','1200','2018-11-10');
SQL> INSERT INTO PLATNOSC VALUES ('6','6','6','2900','2018-10-19');
SQL> INSERT INTO PLATNOSC VALUES ('7','7','7','2800','2018-10-29');
```

```
SQL> SELECT * FROM PLATNOSC;
```

NRPLATNOSCI	NRKLIENTA	NRREZERWACJI	KWOTA	DATAPLATNOSCI
1	1	1	450.00	2018-11-20
2	2	2	750.00	2018-12-04
3	3	3	650.00	2018-11-12
4	4	4	600.00	2018-12-01
5	5	5	1200.00	2018-11-10
6	6	6	2900.00	2018-10-19
7	7	7	2800.00	2018-10-29
8	8	8	2500.00	2018-10-15
9	9	9	1300.00	2018-10-27
10	10	10	1700.00	2018-11-20
11	11	11	2200.00	2018-11-20
12	12	12	2400.00	2018-11-17
13	13	13	2600.00	2018-12-02
14	14	14	2800.00	2018-11-08
15	15	15	2300.00	2018-11-23

```
SQL>
```


Część 3 Połączenia zewnętrzne

- 1) Wykaz dokonanych rezerwacji przez określonych pracowników dla poszczególnych klientów w relacji: Klient, Pracownik, Rezerwacja

Firebird ISQL Tool

```
SQL> SELECT PRACOWNIK.NRPRACOWNIKA, PRACOWNIK.IMIE, PRACOWNIK.NAZWISKO, PRACOWNIK.STANOWISKO, KLIENT.IMIE, KLIENT.NAZWISKO, REZERWACJA.NRREZERWACJI
CON> FROM PRACOWNIK, REZERWACJA, KLIENT
CON> WHERE KLIENT.NRKLIENTA = REZERWACJA.NRKLIENTA AND PRACOWNIK.NRPRACOWNIKA = REZERWACJA.NRPRACOWNIKA
CON> ORDER BY NRPRACOWNIKA ASC;
```

NRPRACOWNIKA	IMIE	NAZWISKO	STANOWISKO	IMIE	NAZWISKO	NRREZERWACJI
2	Marta	Kowal	Obsługa naziemna	Lukasz	Farulewski	1
2	Marta	Kowal	Obsługa naziemna	Karol	Lipiec	5
2	Marta	Kowal	Obsługa naziemna	Anna	Mrozik	9
2	Marta	Kowal	Obsługa naziemna	Vijay	Singh	13
2	Marta	Kowal	Obsługa naziemna	Joanna	Wrzesien	7
6	Karolina	Stankowska	Obsługa naziemna	Pawel	Dominik	2
6	Karolina	Stankowska	Obsługa naziemna	Maciej	Maj	6
6	Karolina	Stankowska	Obsługa naziemna	Barbara	Karolak	10
6	Karolina	Stankowska	Obsługa naziemna	Shirioku	Nagasaki	14
11	Kamil	Matowski	Obsługa naziemna	Tomasz	Marzec	3
11	Kamil	Matowski	Obsługa naziemna	James	Rodriguez	11
11	Kamil	Matowski	Obsługa naziemna	Katherine	Johns	15
13	Krystian	Pomorski	Obsługa naziemna	Damian	Kwiecien	4
13	Krystian	Pomorski	Obsługa naziemna	Katarzyna	Lato	8
13	Krystian	Pomorski	Obsługa naziemna	David	Coperfield	12

SQL>

- 2) Wykaz obsługi poszczególnych rejsów pod kątem linii lotniczej, samolotu i pracowników w relacji: Lot, Samolot, Pracownik

Firebird ISQL Tool

```
SQL> SELECT LOT.NRREJSU, LOT.PORTWYLOTU, LOT.PORTPRZYLOTU, SAMOLOT.LINIALOTNICZA, SAMOLOT.PRODUCENT, SAMOLOT.MODEL, PRACOWNIK.IMIE, PRACOWNIK.NAZWISKO, PRACOWNIK.STANOWISKO
CON> FROM LOT, SAMOLOT, PRACOWNIK
CON> WHERE PRACOWNIK.NRPRACOWNIKA = SAMOLOT.NRPRACOWNIKA AND SAMOLOT.NRREJSU = LOT.NRREJSU;
```

NRREJSU	PORTWYLOTU	PORTPRZYLOTU	LINIALOTNICZA	PRODUCENT	MODEL	IMIE	NAZWISKO	STANOWISKO
1	Bydgoszcz	Warszawa	LOT	Embraer	ERJ 145 XR	Dagmara	Biskup	Pilot
2	Warszawa	Nowy Jork	United Airlines	Boeing	737 MAX	Mateusz	Nowak	Pilot
3	Warszawa	Katowice	LOT	Embraer	ERJ 145 XR	Adam	Krupa	Pilot
4	Chicago	Warszawa	LOT	Boeing	737-800	Adam	Krupa	Pilot
5	Warszawa	Monachium	Lufthansa	Embraer	E195	Johann	Wolf	Pilot
6	Berlin	Warszawa	Lufthansa	Airbus	A320 neo	Matheus	Ubner	Pilot
7	Krakow	Warszawa	LOT	Embraer	E195	Monika	Makowska	Stewardessa
8	Warszawa	Wroclaw	LOT	Embraer	E195	Iwona	Wanek	Stewardessa
9	Tokjo	Warszawa	United Airlines	Boeing	737 MAX	Lea	Nahmen	Stewardessa
10	Toronto	Warszawa	Air Canada	Boeing	787-900	Johann	Wolf	Pilot
11	Warszawa	Gdansk	LOT	Embraer	E195	Kamila	Pankowska	Stewardessa
12	Warszawa	Singapur	Singapore Airlines	Boeing	787-800	Monika	Makowska	Stewardessa
13	Waszyngton	Warszawa	United Airlines	Boeing	737 MAX	Waldemar	Noskowski	Obsługa techniczna
14	Bombaj	Warszawa	LOT	Boeing	737-800	Dagmara	Biskup	Pilot
15	Warszawa	Frankfurt	Lufthansa	Embraer	E195	Matheus	Ubner	Pilot

SQL>

- 3) Wykaz wagi bagażu dla poszczególnych rejsów z uwzględnieniem producenta samolotu oraz modelu w relacji: Bagaż, Lot, Samolot

```
SQL> SELECT BAGAZ.NRBAGAZU, BAGAZ.WAGABAGAZU, LOT.PORTWYLOTU, LOT.PORTPRZYLOTU, SAMOLOT.PRODUCENT, SAMOLOT.MODEL
CON> FROM LOT, SAMOLOT, BAGAZ
CON> WHERE BAGAZ.NRREJSU = LOT.NRREJSU AND LOT.NRREJSU = SAMOLOT.NRREJSU
CON> ORDER BY BAGAZ.WAGABAGAZU DESC;
```

NRBAGAZU	WAGABAGAZU	PORTWYLOTU	PORTPRZYLOTU	PRODUCENT	MODEL
12	40.00	Warszawa	Nowy Jork	Boeing	737 MAX
10	30.00	Warszawa	Wroclaw	Embraer	E195
5	30.00	Warszawa	Singapur	Boeing	787-800
13	27.00	Toronto	Warszawa	Boeing	787-900
7	25.00	Chicago	Warszawa	Boeing	737-800
11	25.00	Tokjo	Warszawa	Boeing	737 MAX
14	20.00	Waszyngton	Warszawa	Boeing	737 MAX
2	15.00	Krakow	Warszawa	Embraer	E195
9	15.00	Warszawa	Frankfurt	Embraer	E195
6	12.00	Warszawa	Katowice	Embraer	ERJ 145 XR
3	10.00	Warszawa	Monachium	Embraer	E195
4	10.00	Berlin	Warszawa	Airbus	A320 neo
15	10.00	Bombaj	Warszawa	Boeing	737-800
1	5.00	Bydgoszcz	Warszawa	Embraer	ERJ 145 XR
8	5.00	Warszawa	Gdansk	Embraer	E195

SQL>

- 4) Wykaz pracowników pod kątem wartości dokonanej rezerwacji, klienta oraz lotu w relacji:
Pracownik, Rezerwacja, Klient, Płatność, Lot

```
SQL> SELECT PRACOWNIK.IMIE, PRACOWNIK.NAZWISKO, REZERWACJA.NRREZERWACJI, KLIENT.IMIE, KLIENT.NAZWISKO, PLATNOSC.KWOTA, LOT.PORTWYLOTU, LOT.PORTPRZYLOTU
CON> FROM PRACOWNIK, REZERWACJA, KLIENT, PLATNOSC, LOT
CON> WHERE PRACOWNIK.NRPRACOWNIKA = REZERWACJA.NRPRACOWNIKA AND KLIENT.NRKLIENTA = REZERWACJA.NRKLIENTA AND KLIENT.NRKLIENTA = PLATNOSC.NRKLIENTA AND
CON> LOT.NRREJSU = REZERWACJA.NRREJSU
CON> ORDER BY PLATNOSC.KWOTA DESC;
```

IMIE	NAZWISKO	NRREZERWACJI	IMIE	NAZWISKO	KWOTA	PORTWYLOTU	PORTPRZYLOTU
Karolina	Stankowska	6	Maciej	Maj	2900.00	Warszawa	Wroclaw
Marta	Kowal	7	Joanna	Wrzesien	2800.00	Chicago	Warszawa
Karolina	Stankowska	14	Shirioku	Nagasaki	2800.00	Tokjo	Warszawa
Marta	Kowal	13	Vijay	Singh	2600.00	Toronto	Warszawa
Krystian	Pomorski	8	Katarzyna	Lato	2500.00	Warszawa	Singapur
Krystian	Pomorski	12	David	Coperfield	2400.00	Waszyngton	Warszawa
Kamil	Matowski	15	Katherine	Johns	2300.00	Warszawa	Nowy Jork
Kamil	Matowski	11	James	Rodriguez	2200.00	Bombaj	Warszawa
Karolina	Stankowska	10	Barbara	Karolak	1700.00	Krakow	Warszawa
Marta	Kowal	9	Anna	Mrozik	1300.00	Warszawa	Monachium
Marta	Kowal	5	Karol	Lipiec	1200.00	Warszawa	Frankfurt
Karolina	Stankowska	2	Pawel	Dominik	750.00	Berlin	Warszawa
Kamil	Matowski	3	Tomasz	Marzec	650.00	Warszawa	Katowice
Krystian	Pomorski	4	Damian	Kwiecien	600.00	Warszawa	Gdansk
Marta	Kowal	1	Lukasz	Farulewski	450.00	Bydgoszcz	Warszawa

SQL>

