



Autorzy projektu:

Jakub Piotrowski

Bartłomiej Trzask

SPIS TREŚCI

Encje/tabele/modele

Wprowadzenie

Założenia

Relacje bazy danych programu MS Access

Zestawienie encji wraz z opisem

Zestawienie atrybutów

 Klient / Dokument tożsamości

 Karnet / Płeć

 Pracownik / Rezerwacja zajęć

 Rodzaj ulgi / Sale

 Trening personalny / Typ karnetu

 Typ stanowiska / Wejście jednorazowe

 Zniżka lojalnościowa

Związki

Reguły biznesowe

Model logiczny

Model fizyczny

Model koncepcyjny

Kwerendy

Baza karnetów

Baza klientów

Baza klientów z miejscowości

Baza pracowników

Historia rezerwacji treningów dla IDKlienta

Lista karnetów klienta

Lista osób zapisanych na zajęcia w danym dniu

Lista rezerwacji zajęć

Lista trenerów

Lista trenerów i zarobki

Lista treningów danego trenera

Lista treningów pers danego dnia

Lista wejściówek

Lista wejściówek na dany dzień

Formularze

Formularz startowy

Baza klientów

Dokumenty tożsamości

Baza karnetów

Baza treningów personalnych

Baza rezerwacji zajęć

Baza wejść jednorazowych

Typ karnetów/Rodzaje ulg/Zniżki lojalnościowe

Baza sal

Baza pracowników

Typy stanowisk

Raporty

Raporty

Baza klientów z miejscowości

Baza karnetów w miesiącu

Historia rezerwacji treningów dla IDKlienta

Lista karnetów klienta

Lista osób zapisanych na zajęcia w danym dniu

Lista trenerów i zarobki

Lista treningów danego trenera

Lista treningów pers danego dnia

Lista wejściówek na dany dzień

Rezerwacja zajęć według dnia i sali

Przewodnik po aplikacji

Zakończenie



Celem projektu jest utworzenia bazy danych obsługującej klientów naszego centrum fitness „Vega”. Oferujemy dostęp do otwartej strefy, na której znajduje się siłownia i odbywają się treningi personalne z zatrudnionymi u nas trenerami. Możliwy jest również zapis na zajęcia typu Fitness, Cross, Squash, trening rowery itp., które zaplanowane są codziennie o określonych godzinach. Aby wejść na teren centrum każdy klient musi wypełnić formularz informacyjny zawierający jego dane osobowe, a następnie wykupić jednorazową wejściówkę lub karnet.

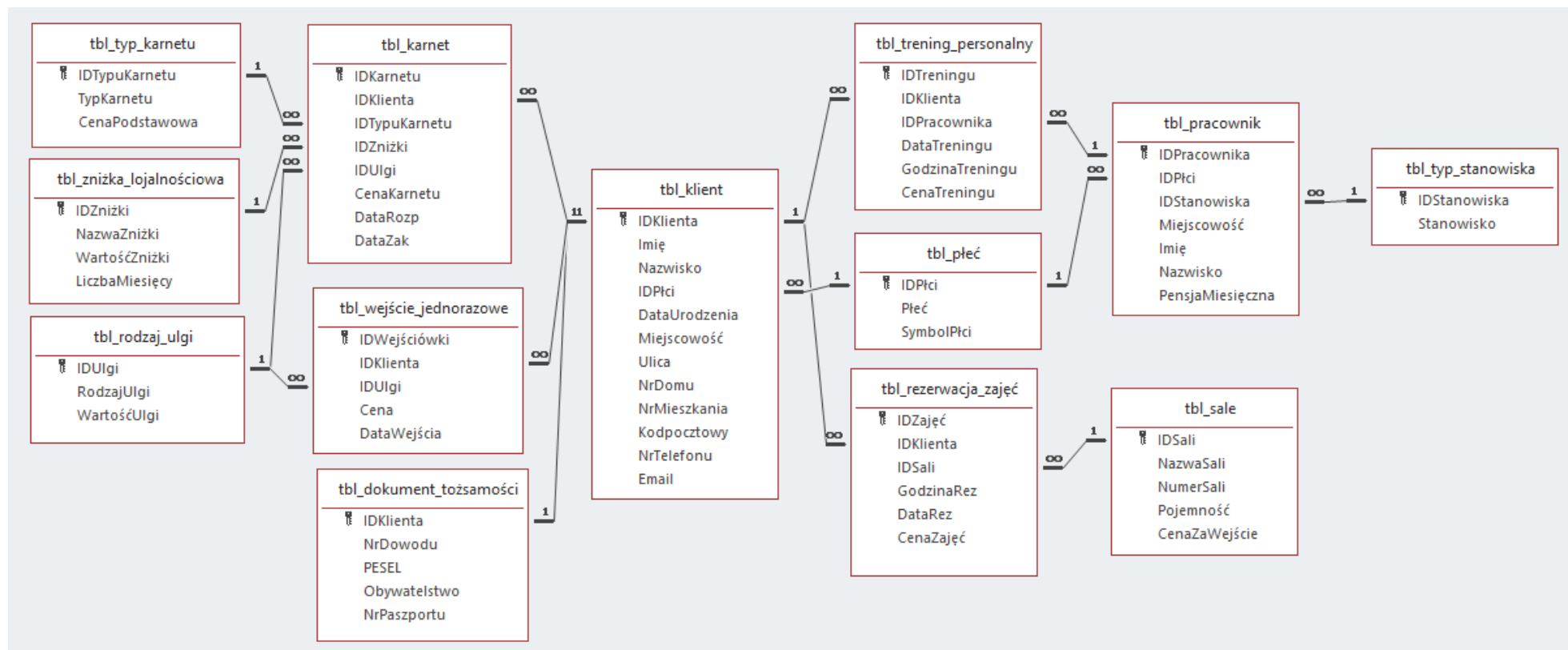
Nasza baza wraz z formularzami i raportami przeznaczona jest dla pracowników recepcji i kierowników, którzy dbają o prawidłowe wprowadzanie wszystkich danych dotyczących klientów, pracowników, opłat, treningów personalnych i zajęć dodatkowych.

Założenia

Aby nasz system działał prawidłowo, poza odpowiednim projektem struktury bazy danych, należy w naszym rozważaniu uwzględnić również w jaki sposób baza będzie użytkowana i jakie możliwe błędy mogą zostać popełnione podczas korzystania z systemu. Postaramy się o wprowadzenie jak największej liczby rozwiązań i zabezpieczeń zapobiegających ewentualnym problemom podczas korzystania z programu. Będą one dotyczyć sytuacji gdzie błąd jest spowodowany czynnikiem ludzkim, przykładowo: zapobiegnięcie sytuacji, w której liczba osób zapisanych na zajęcia przekracza pojemność sali treningowej.

Aby ułatwić pracę na końcu dokumentacji umieszczamy przewodnik po aplikacji, z którym każda osoba używająca bazy powinna się zapoznać.

Relacje bazy danych programu MS Access



Zestawienie encji wraz z opisem

Nazwa encji	Opis
tbl_klient	Zawiera informacje na temat klientów.
tbl_dokument_tożsamości	Zawiera informacje na temat dokumentów tożsamości klientów.
tbl_wejście_jednorazowe	Zawiera informacje o wykupionych wejściówkach jednorazowych.
tbl_karnet	Zawiera informacje na temat karnetów wykupionych przez klientów.
tbl_typ_karnetu	Zawiera informacje o różnych typach karnetów i ich cenach.
tbl_zniżka_lojalnościowa	Zawiera informacje o zniżkach przysługującym klientom kupującym karnet na okres większy niż 1 miesiąc.
tbl_rodzaj_ulgi	Zawiera informacje o ulgach przysługujących klientom i wartościach ulg.
tbl_płeć	Zawiera informacje o płci.
tbl_pracownik	Zawiera informacje o pracownikach pracujących w centrum fitness.
tbl_typ_stanowiska	Zawiera informacje o stanowiskach istniejących w firmie.
tbl_trening_personalny	Zawiera informacje o rezerwacjach treningów personalnych.
tbl_rezerwacja_zajęć	Zawiera informacje o rezerwacjach zajęć.
tbl_sale	Zawiera informacje o salach znajdujących się w centrum fitness.

Zestawienie atrybutów dla tbl_klient

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDKlienta	Liczba całkowita		T	Przechowuje identyfikator klienta.
Imię	Tekst	15	T	Przechowuje identyfikator płci klienta.
Nazwisko	Tekst	30	T	Przechowuje imię klienta.
IDPłci	Liczba całkowita		T	Przechowuje nazwisko klienta.
DataUrodzenia	Data/Godzina		T	Przechowuje datę urodzenia klienta.
Miejscowość	Tekst	30	T	Przechowuje miejscowość, którą zamieszkuje klient.
Ulica	Tekst	50	T	Przechowuje ulicę, przy której mieszka klient.
NrDomu	Liczba całkowita		T	Przechowuje numer domu klienta.
NrMieszkania	Tekst	20	N	Przechowuje numer mieszkania klienta.
Kodpocztowy	Tekst	6	T	Przechowuje kod pocztowy klienta.
NrTelefonu	Tekst	11	T	Przechowuje numer kontaktowy do klienta.
Email	Tekst	50	N	Przechowuje adres e-mail klienta.

Zestawienie atrybutów dla tbl_dokument_tożsamości

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDKlienta	Liczba całkowita		T	Przechowuje identyfikator klienta.
NrDowodu	Tekst	9	T	Przechowuje numer dowodu osobistego klienta.
PESEL	Tekst	11	T	Przechowuje numer PESEL.
Obywatelstwo	Tekst	30	T	Przechowuje narodowość klienta.
NrPaszportu	Tekst	25	N	Przechowuje numer paszportu klienta.

Zestawienie atrybutów dla tbl_karnet

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDKarnetu	Liczba całkowita		T	Przechowuje identyfikator karnetu.
IDKlienta	Liczba całkowita		T	Przechowuje identyfikator klienta.
IDTypuKarnetu	Liczba całkowita		T	Przechowuje rodzaj karnetu.
IDZniżki	Liczba całkowita		T	Przechowuje rodzaj przydzielonej zniżki.
IDUlgi	Liczba całkowita		T	Przechowuje rodzaj przydzielonej ulgi.
CenaKarnetu	Liczba całkowita		T	Przechowuje ostateczną cenę karnetu.
DataRozp	Data/Godzina		T	Przechowuje datę rozpoczęcia działania karnetu.
DataZak	Data/Godzina		T	Przechowuje datę zakończenia obowiązywania karnetu.

Zestawienie atrybutów dla tbl_płeć

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDPłci	Liczba całkowita		T	Przechowuje identyfikator płci.
Płeć	Tekst	16	T	Przechowuje dostępne do wyboru płcie.
SymbolPłci	Tekst	1	T	Przechowuje symbol przypisany do płci.

Zestawienie atrybutów dla tbl_pracownik

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDPracownika	Liczba całkowita		T	Przechowuje identyfikator pracownika.
IDPłci	Liczba całkowita		T	Przechowuje identyfikator płci pracownika.
IDStanowiska	Liczba całkowita		T	Przechowuje identyfikator stanowiska pracownika.
Miejscowość	Tekst	30	T	Przechowuje informacje o miejscowości, w której mieszka pracownik.
Imię	Tekst	15	T	Przechowuje imię pracownika.
Nazwisko	Tekst	30	T	Przechowuje nazwisko pracownika.
PensjaMiesięczna	Podwójna precyzja		T	Przechowuje informacje o wysokości wynagrodzenia pracownika.

Zestawienie atrybutów dla tbl_rezerwacja_zajęć

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDZajęć	Liczba całkowita		T	Przechowuje identyfikator zajęć.
IDKlienta	Liczba całkowita		T	Przechowuje identyfikator klienta.
IDSali	Liczba całkowita		T	Przechowuje identyfikator sali.
GodzinaRez	Data/Godzina		T	Przechowuje godzinę rezerwacji.
DataRez	Data/Godzina		T	Przechowuje dzień rezerwacji.
CenaZajęć	Podwójna precyzja		T	Przechowuje cenę zajęć.

Zestawienie atrybutów dla tbl_rodzaj_ulgi

Encja	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDUlgi	Liczba całkowita		T	Przechowuje identyfikator ulgi.
RodzajUlgi	Tekst	30	T	Przechowuje opis ulgi.
WartośćUlgi	Podwójna precyzja		T	Przechowuje wysokość ulgi.

Zestawienie atrybutów dla tbl_sale

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDSali	Liczba całkowita		T	Przechowuje identyfikator sali.
NazwaSali	Tekst	40	T	Przechowuje nazwę sali.
NumerSali	Liczba całkowita		T	Przechowuje numer sali
Pojemność sali	Liczba całkowita		T	Przechowuje ilość ludzi mogących jednocześnie przebywać na sali.
CenaZaWejście	Podwójna precyzja		T	Przechowuje cenę za wejście na salę.

Zestawienie atrybutów dla tbl_trening_personalny

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDTreningu	Liczba całkowita		T	Przechowuje identyfikator treningu.
IDKlienta	Liczba całkowita		T	Przechowuje identyfikator klienta.
IDPracownika	Liczba całkowita		T	Przechowuje identyfikator pracownika.
DataTreningu	Data/Godzina		T	Przechowuje datę, kiedy trening ma się odbyć.
GodzinaTreningu	Data/Godzina		T	Przechowuje godzinę, kiedy trening ma się odbyć.
CenaTreningu	Liczba całkowita		T	Przechowuje cenę treningu.

Zestawienie atrybutów dla tbl_typ_karnetu

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDTypuKarnetu	Liczba całkowita		T	Przechowuje identyfikator typu karnetu.
TypKarnetu	Tekst	30	T	Przechowuje typ karnetu.
Cena Podstawowa	Podwójna precyzja		T	Przechowuje cenę podstawową karnetu.

Zestawienie atrybutów dla tbl_typ_stanowiska

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDStanowiska	Liczba całkowita		T	Przechowuje identyfikator stanowiska.
Stanowisko	Tekst	30	T	Przechowuje nazwę stanowiska.

Zestawienie atrybutów dla tbl_wejście_jednorazowe

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDWejściówki	Liczba całkowita		T	Przechowuje identyfikator wejścia jednorazowego.
IDKlienta	Liczba całkowita		T	Przechowuje identyfikator klienta.
IDUlgi	Liczba całkowita		T	Przechowuje identyfikator ulgi.
Cena	Podwójna precyzja		T	Przechowuje ostateczną cenę wejścia jednorazowego.
DataWejścia	Data/Godzina		T	Przechowuje datę kiedy wejście jednorazowe zostało zakupione.

Zestawienie atrybutów dla tbl_zniżka_lojalnościowa

Nazwa pola	Typ danych	Rozmiar pola	Czy wymagany?	Opis
IDZniżki	Liczba całkowita		T	Przechowuje identyfikator zniżki.
NazwaZniżki	Tekst	30	T	Przechowuje nazwę zniżki.
WartośćZniżki	Podwójna precyzja		T	Przechowuje wysokość zniżki na karnet.
LiczbaMiesięcy	Liczba całkowita		T	Przechowuje liczbę miesięcy, którą trzeba wykupić aby konkretna zniżka obowiązywała.

Związki

Encja podrzędna	Encja nadrzędna	Typ związku	Opcje związku	Typ relacji	Opis
Karnet	Klient	1:N	NOB,NI	Jeden do wielu	Przyporządkowanie karnetów/karnetu do klienta.
Wejście jednorazowe	Klient	1:N	NOB,NI	Jeden do wielu	Przyporządkowanie wejściówki do klienta.
Dokument tożsamości	Klient	1:1	OB,I	Jeden do jednego	Przyporządkowanie dokumentu do klienta.
Trening personalny	Klient	1:N	NOB,NI	Jeden do wielu	Przyporządkowanie treningu personalnego do klienta.
Płeć	Klient	1:N	OB.,NI	Jeden do wielu	Przyporządkowanie płci do klienta.
Rezerwacja zajęć	Klient	1:N	NOB,NI	Jeden do wielu	Przyporządkowanie rezerwacji sali do klienta.
Typ karnetu	Karnet	1:N	OB,NI	Jeden do wielu	Przyporządkowanie typu karnetu do karnetu.
Zniżka lojalnościowa	Karnet	1:N	OB,NI	Jeden do wielu	Przyporządkowanie zniżki lojalnościowej do karnetu.
Rodzaj ulgi	Karnet	1:N	OB,NI	Jeden do wielu	Przyporządkowanie rodzaju ulgi do karnetu.
Rodzaj ulgi	Wejście jednorazowe	1:N	OB,NI	Jeden do wielu	Przyporządkowanie rodzaju ulgi do jednorazowego wejścia.
Stanowiska	Pracownik	1:N	OB,NI	Jeden do wielu	Przyporządkowanie stanowiska do pracownika.
Płcie	Pracownik	1:N	OB,NI	Jeden do wielu	Przyporządkowanie płci do pracownika.
Trening personalny	Pracownik	1:N	NOB,NI	Jeden do wielu	Przyporządkowanie treningu personalnego do pracownika.
Sale	Rezerwacja zajęć	1:N	OB,NI	Jeden do wielu	Przyporządkowanie Sali do zajęć.

Reguły biznesowe

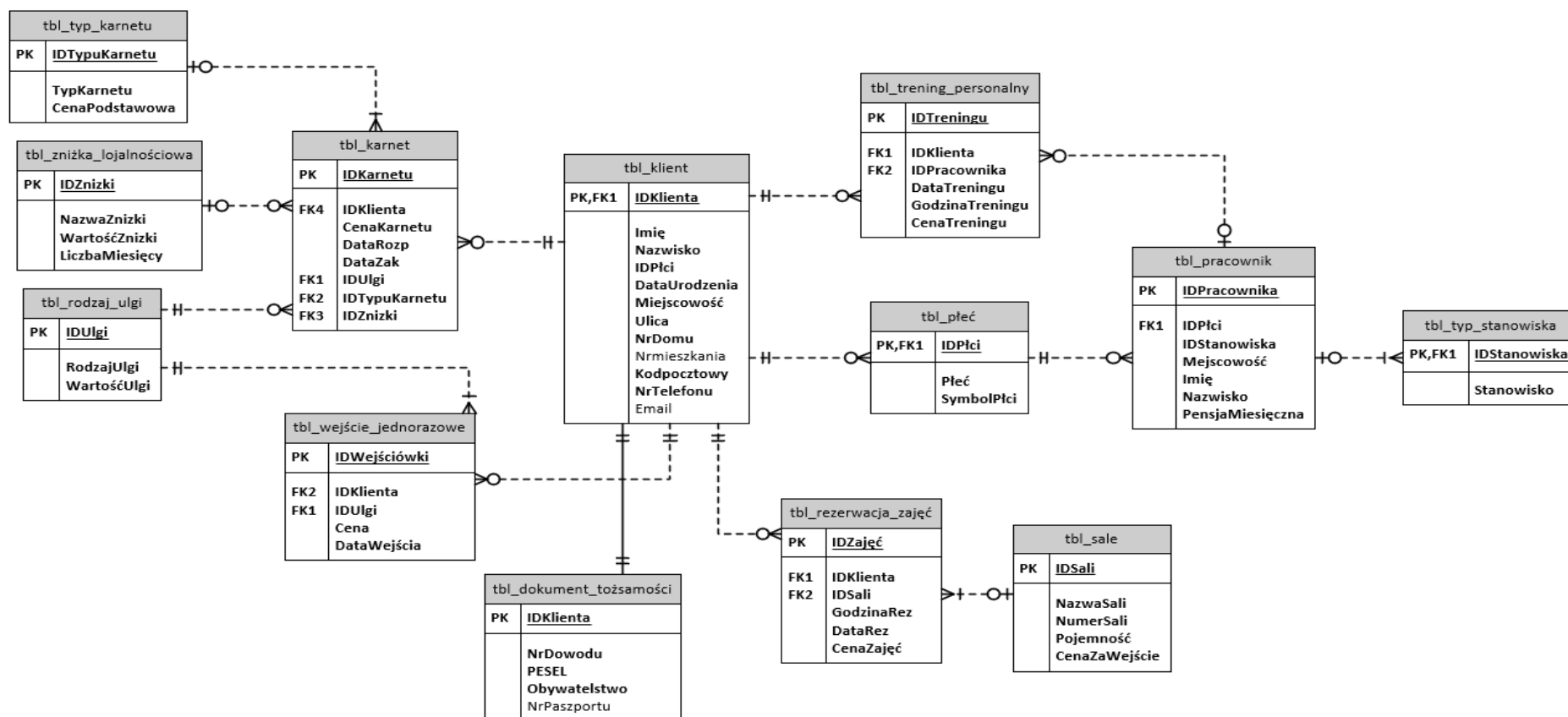
Cel użycia	Opis implementacji	Przykład praktycznego użycia
Sprawdzenie poprawności numeru dowodu.	CHAR(10)	AVC1234567
	XXXXXXXXXX	
Sprawdzenie poprawności Peselu	CHAR(11)	98765432112
	XXXXXXXXXXXX	
Sprawdzenie poprawności daty treningu/zajęć/karnetu/urodzenia klienta/wejścia jednorazowego.	DATETIME	10-12-2007
	XX-XX-XXXX	
Sprawdzenie poprawności godziny treningu/zajęć.	DATETIME	13:50
	XX:XX	
Sprawdzenie poprawności telefonu komórkowego.	CHAR(12)	502-555-526
	XXX-XXX-XXX	
Sprawdzenie poprawności kodu pocztowego.	CHAR(6)	93-586
	XX-XXX	

Model logiczny

Model logiczny reprezentuje podstawowe założenia systemu bazy danych.

Zawiera on zestaw tabel wymaganych do prawidłowego spełnienia jego oczekiwań.

Na zaprezentowanej grafice opisano podstawowe cechy i nazewnictwo. Model logiczny składa się z 13 rozbudowanych tabel, między którymi zachodzą relacyjne związki. Będą one szczegółowo opisane z pozycji programu MS Access.

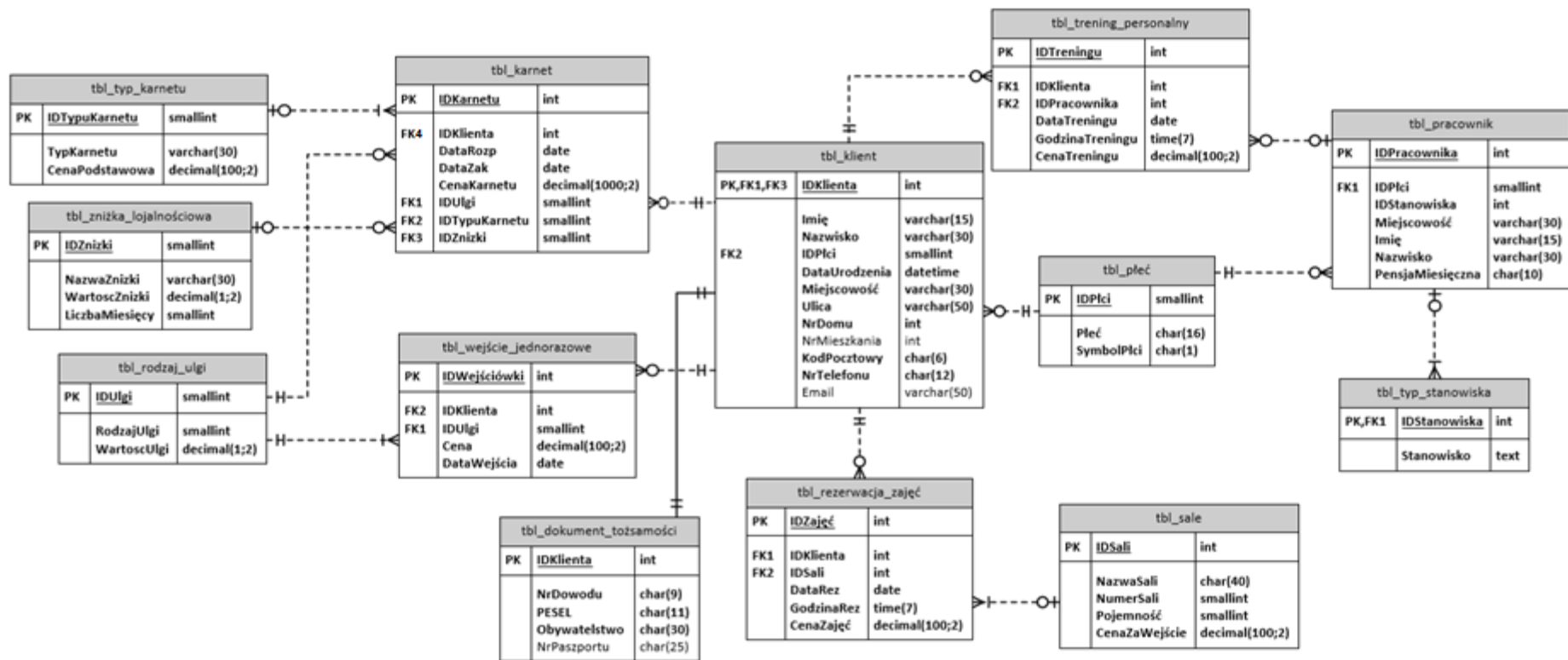


Model fizyczny

Model fizyczny prezentuje złożoność systemu i różnorodność zastosowanych w nim typów danych.

Pozwala łatwo i czytelnie określić, jakich informacji wymagamy od poszczególnych danych w naszej bazie.

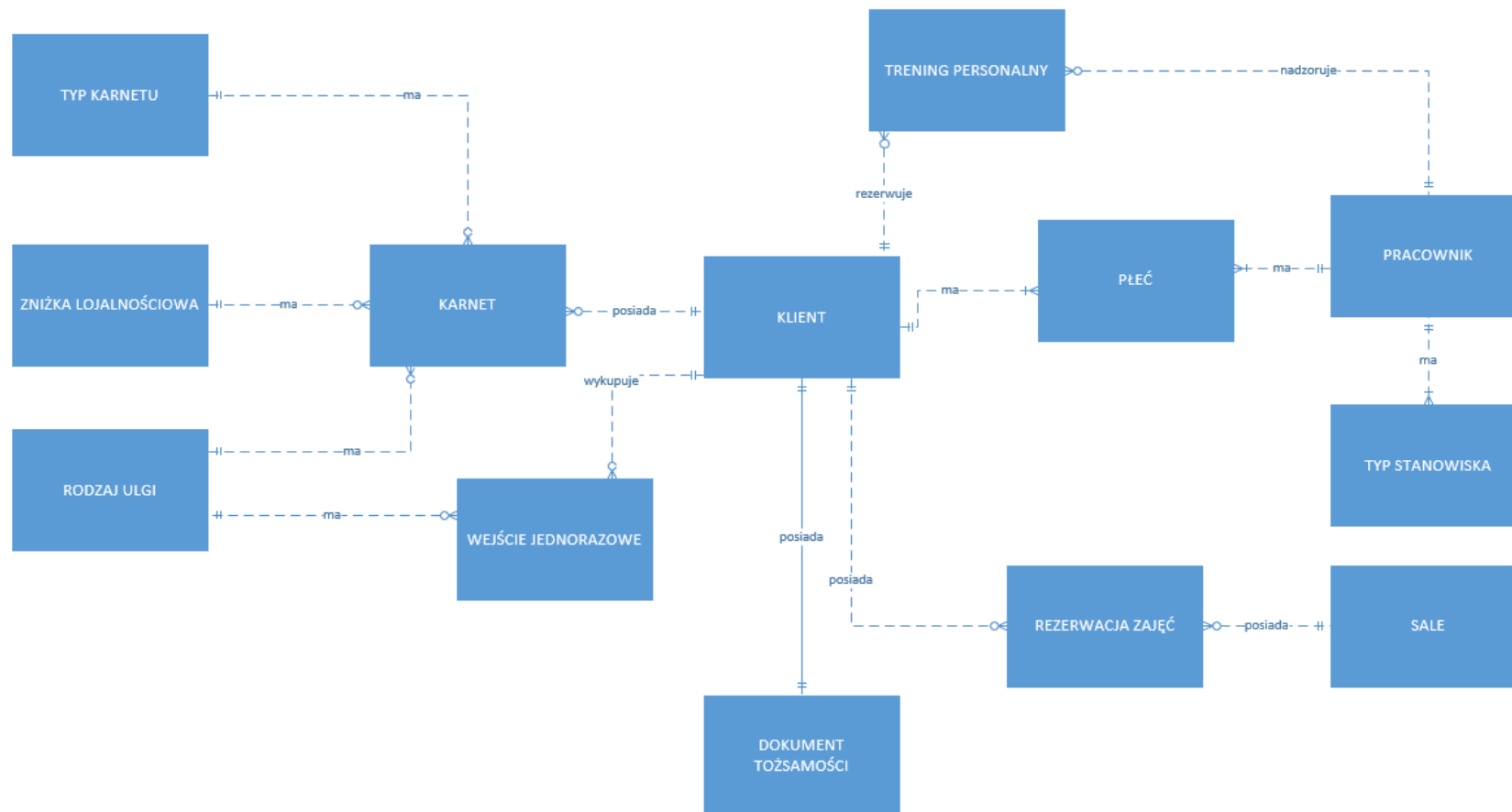
System składa się z wielu fizycznych typów, które będą opisane w dalszej części projektu z poziomu programu MS Access



Model konceptualny

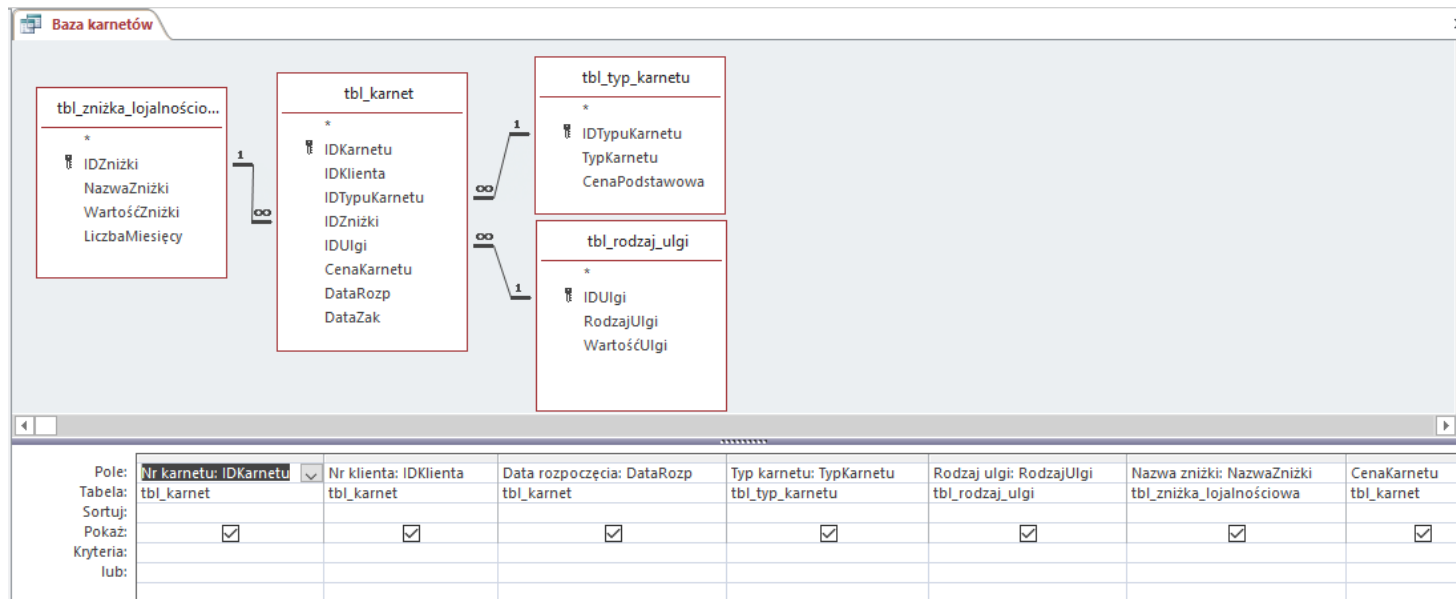
Model konceptualny jest podstawą schematu bazy danych.

Reprezentuje obiekty w uniwersalnym modelu niezależnym od modelu implementacyjnego. Jest wykorzystywany jako fundamentalny model przy projekcie relacyjnym baz danych.



Kwerendy

Kwerenda „Baza kartonów” zwraca informacje o wykupionych kartonach.



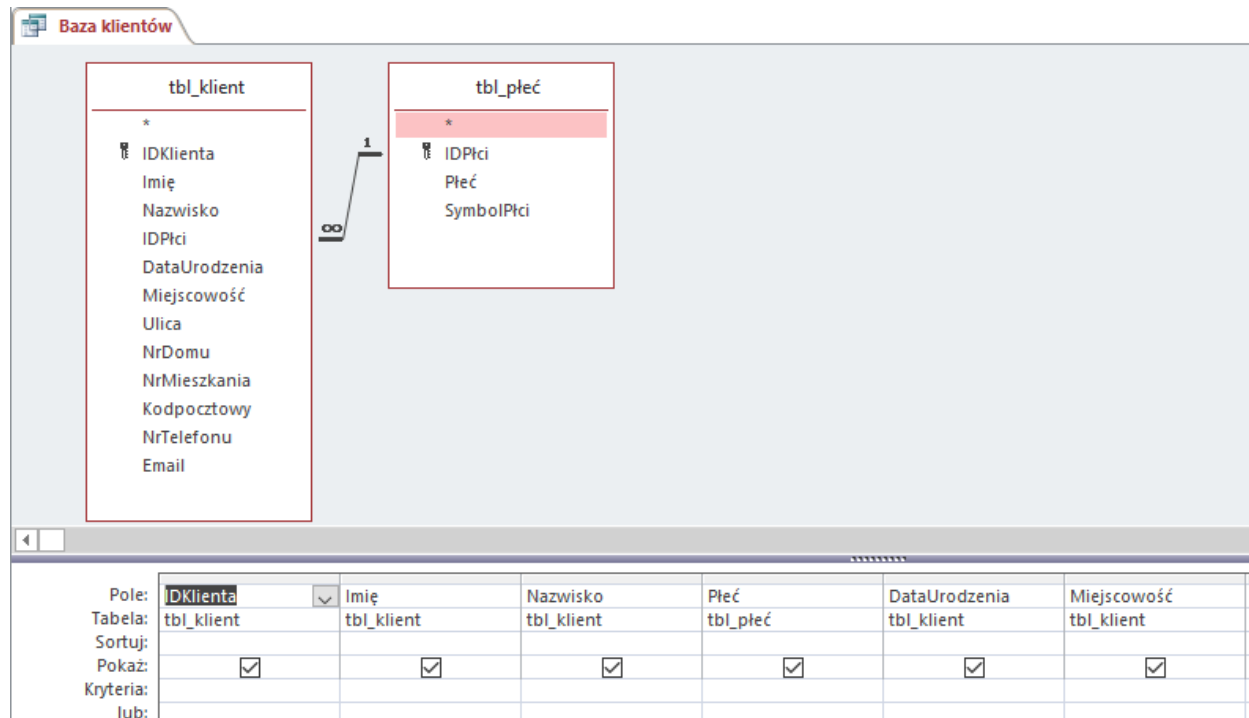
KOD SQL:

```
SELECT tbl_karnet.IDKarnetu AS [Nr karnetu], tbl_karnet.IDKlienta AS [Nr klienta], tbl_karnet.DataRozp AS [Data rozpoczęcia],
tbl_typ_karnetu.TypKarnetu AS [Typ karnetu], tbl_rodzaj_ulgi.RodzajUłgi AS [Rodzaj ulgi], tbl_zniżka_lojalnościowa.NazwaZniżki AS [Nazwa
zniżki], tbl_karnet.CenaKarnetu

FROM tbl_zniżka_lojalnościowa INNER JOIN (tbl_typ_karnetu INNER JOIN (tbl_rodzaj_ulgi INNER JOIN tbl_karnet ON tbl_rodzaj_ulgi.IDUłgi =
tbl_karnet.IDUłgi) ON tbl_typ_karnetu.IDTypuKarnetu = tbl_karnet.IDTypuKarnetu) ON tbl_zniżka_lojalnościowa.IDZniżki = tbl_karnet.IDZniżki;
```

Kwerendy

Kwerenda „Baza klientów” zwraca informacje o klientach naszego centrum.



KOD SQL:

```
SELECT tbl_klient.IDKlienta, tbl_klient.Imię, tbl_klient.Nazwisko, tbl_plec.Płeć, tbl_klient.DataUrodzenia, tbl_klient.Miejscowość  
FROM tbl_plec INNER JOIN tbl_klient ON tbl_plec.IDPłci = tbl_klient.IDPłci;
```

Kwerendy

Kwerenda „Baza klientów z miejscowości” zwraca informacje o klientach naszego centrum z podanej przez użytkownika miejscowości.

Baza klientów z miejscowości

tbl_klient

- IDKlienta
- Imię
- Nazwisko
- IDPłci
- DataUrodzenia
- Miejscowość
- Ulica
- NrDomu
- NrMieszkania
- Kodpocztowy
- NrTelefonu
- Email

tbl_plec

- IDPłci
- Płeć
- SymbolPłci

Pole: IDKlienta Imię Nazwisko Płeć DataUrodzenia Miejscowość

Tabela: tbl_klient tbl_klient tbl_klient tbl_plec tbl_klient tbl_klient

Sortuj:

Pokaż: ☒ ☒ ☒ ☒ ☒ ☒

Kryteria:

lub:

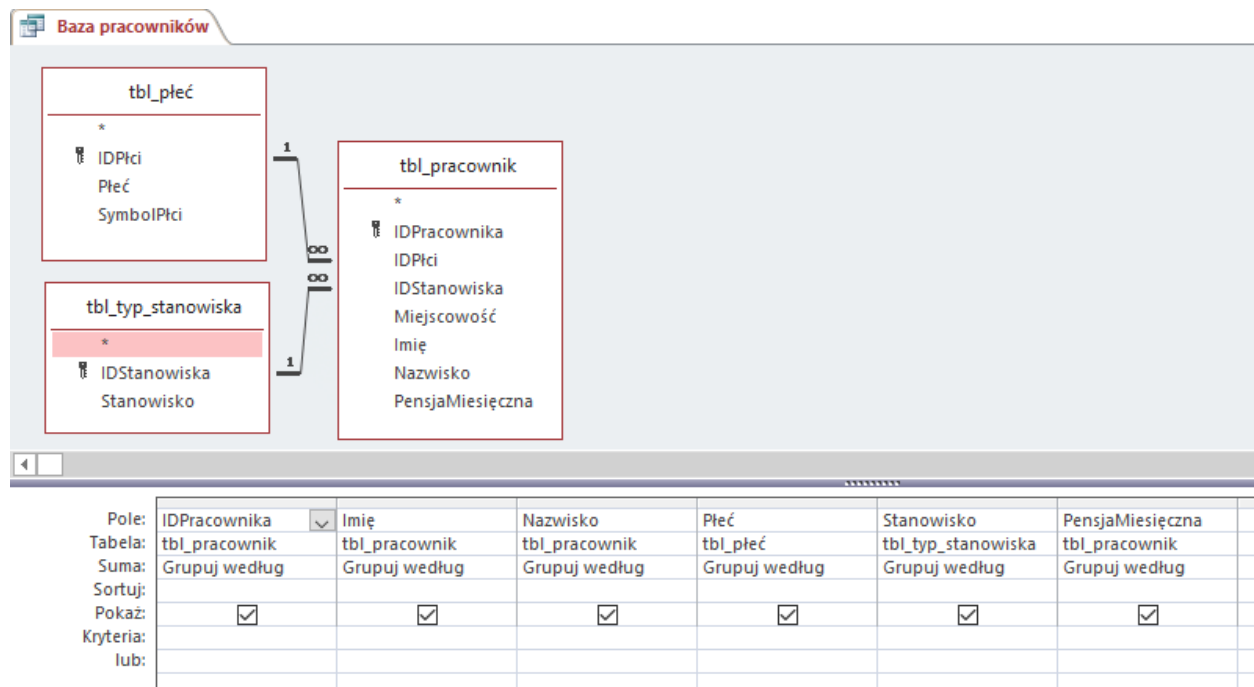
[Podaj miejscowość]

KOD SQL:

```
SELECT tbl_klient.IDKlienta, tbl_klient.Imię, tbl_klient.Nazwisko, tbl_plec.Płeć, tbl_klient.DataUrodzenia, tbl_klient.Miejscowość
FROM tbl_plec INNER JOIN tbl_klient ON tbl_plec.IDPłci = tbl_klient.IDPłci
WHERE (((tbl_klient.Miejscowość)=[Podaj miejscowość]));
```

Kwerendy

Kwerenda „Baza pracowników” zwraca informacje o pracownikach naszego centrum.



KOD SQL:

```
SELECT tbl_pracownik.IDPracownika, tbl_pracownik.Imię, tbl_pracownik.Nazwisko, tbl_plec.Płeć, tbl_typ_stanowiska.Stanowisko,
tbl_pracownik.PensjaMiesięczna
FROM tbl_typ_stanowiska INNER JOIN (tbl_plec INNER JOIN tbl_pracownik ON tbl_plec.IDPłci = tbl_pracownik.IDPłci) ON tbl_typ_stanowiska.IDStanowiska =
tbl_pracownik.IDStanowiska
GROUP BY tbl_pracownik.IDPracownika, tbl_pracownik.Imię, tbl_pracownik.Nazwisko, tbl_plec.Płeć, tbl_typ_stanowiska.Stanowisko,
tbl_pracownik.PensjaMiesięczna;
```

Kwerendy

Kwerenda „Historia rezerwacji treningów dla IDKlienta” zwraca informacje o zarezerwowanych treningach klienta o podanym ID.

Historia rezerwacji treningów dla IDKlienta

tbl_trening_personalny

- IDTreningu
- IDKlienta
- IDPracownika
- DataTreningu
- GodzinaTreningu
- CenaTreningu

tbl_pracownik

- IDPracownika
- IDPłci
- IDStanowiska
- Miejscowość
- Imię
- Nazwisko
- PensjaMiesięczna

1

∞

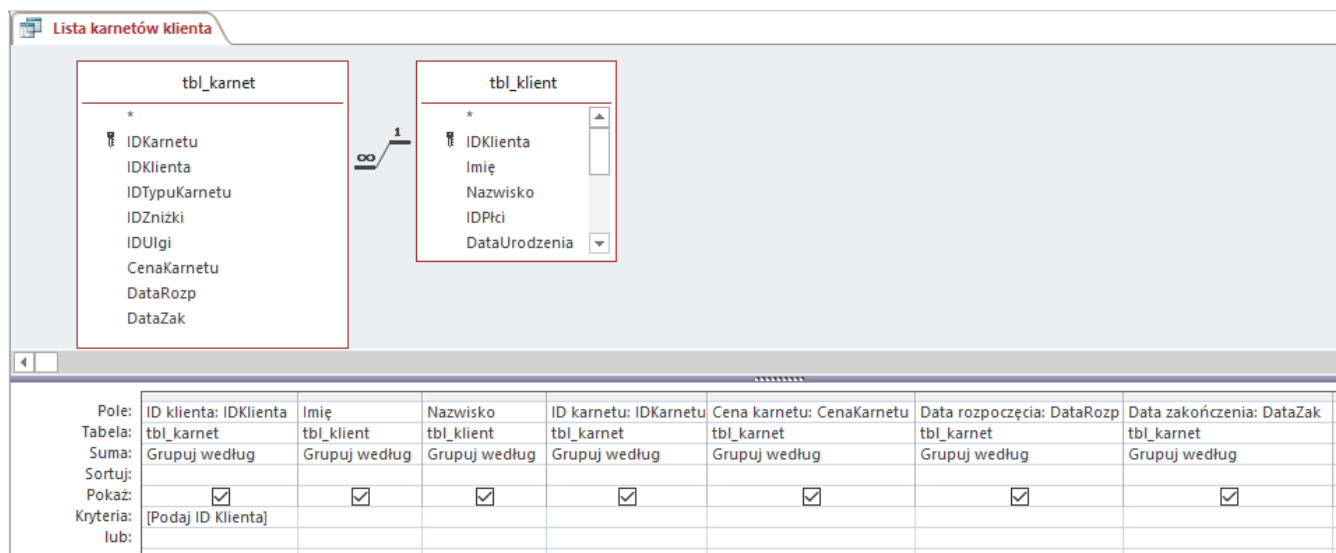
Pole:	ID treningu: IDTreningu	ID klienta: IDKlienta	Data treningu: DataTreningu	Godzina treningu: GodzinaTreningu	Trener: [Imię] + " " + [Nazwisko]	CenaTreningu
Tabela:	tbl_trening_personalny	tbl_trening_personalny	tbl_trening_personalny	tbl_trening_personalny		tbl_trening_personalny
Sortuj:						
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:		[Podaj ID klienta]				
lub:						

KOD SQL:

```
SELECT tbl_trening_personalny.IDTreningu AS [ID treningu], tbl_trening_personalny.IDKlienta AS [ID klienta],  
tbl_trening_personalny.DataTreningu AS [Data treningu], tbl_trening_personalny.GodzinaTreningu AS [Godzina treningu], [Imię] + "  
" + [Nazwisko] AS Trener, tbl_trening_personalny.CenaTreningu  
FROM tbl_pracownik INNER JOIN tbl_trening_personalny ON tbl_pracownik.IDPracownika = tbl_trening_personalny.IDPracownika  
WHERE ((tbl_trening_personalny.IDKlienta)=[Podaj ID klienta]);
```

Kwerendy

Kwerenda „Lista karnetów klienta” zwraca informacje o wszystkich wykupionych przez danego klienta karnetach.

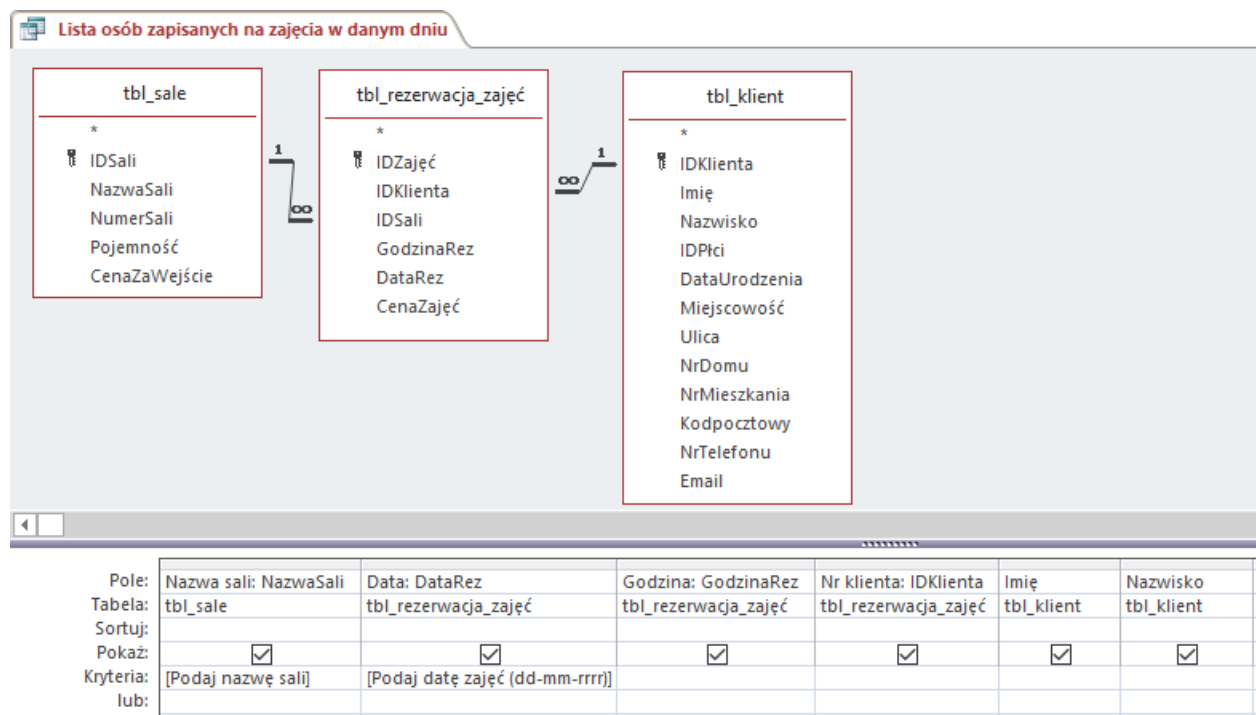


KOD SQL:

```
SELECT tbl_karnet.IDKlienta AS [ID klienta], tbl_klient.Imię, tbl_klient.Nazwisko, tbl_karnet.IDKarnetu AS [ID karnetu], tbl_karnet.CenaKarnetu AS [Cena karnetu], tbl_karnet.DataRozp AS [Data rozpoczęcia], tbl_karnet.DataZak AS [Data zakończenia]
FROM tbl_klient INNER JOIN tbl_karnet ON tbl_klient.IDKlienta = tbl_karnet.IDKlienta
GROUP BY tbl_karnet.IDKlienta, tbl_klient.Imię, tbl_klient.Nazwisko, tbl_karnet.IDKarnetu, tbl_karnet.CenaKarnetu, tbl_karnet.DataRozp,
tbl_karnet.DataZak
HAVING (((tbl_karnet.IDKlienta)=[Podaj ID Klienta]));
```


Kwerendy

Kwerenda „Lista osób zapisanych na zajęcia w danym dniu” zwraca informacje o klientach zapisanych na dane zajęcia w podanym dniu.

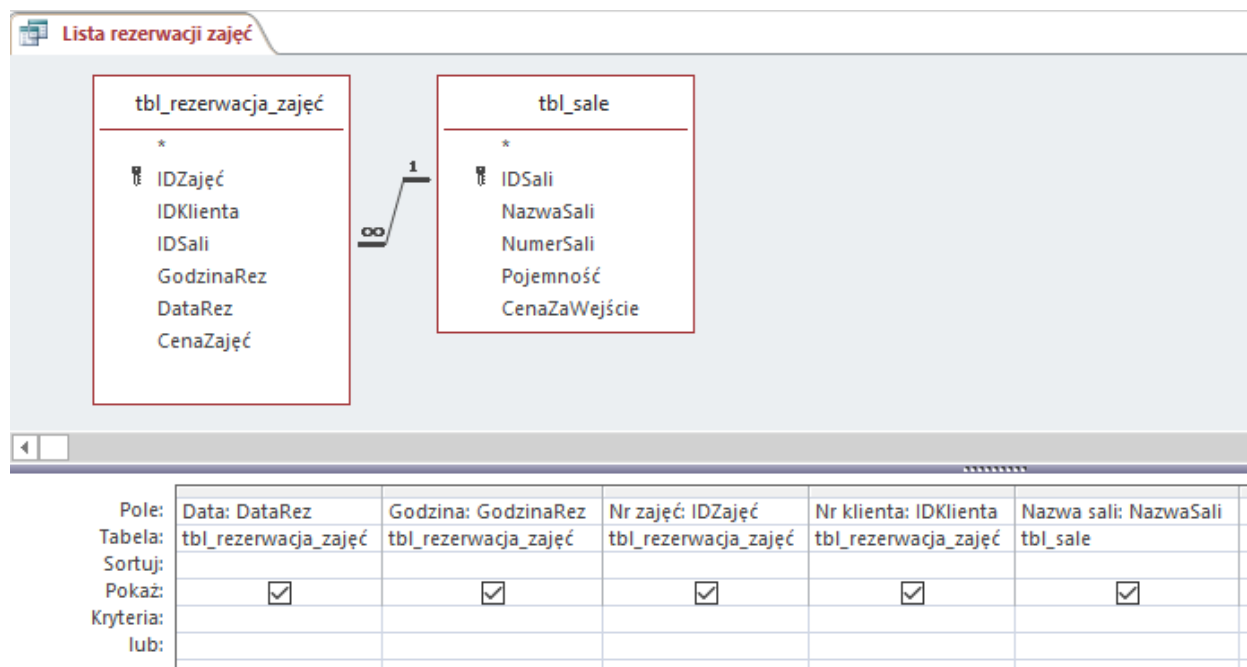


KOD SQL:

```
SELECT tbl_sale.NazwaSali AS [Nazwa sali], tbl_rezerwacja_zajęć.DataRez AS Data, tbl_rezerwacja_zajęć.GodzinaRez AS Godzina,
tbl_rezerwacja_zajęć.IDKlienta AS [Nr klienta], tbl_klient.Imię, tbl_klient.Nazwisko
FROM tbl_sale INNER JOIN (tbl_klient INNER JOIN tbl_rezerwacja_zajęć ON tbl_klient.IDKlienta = tbl_rezerwacja_zajęć.IDKlienta) ON tbl_sale.IDSali =
tbl_rezerwacja_zajęć.IDSali WHERE (((tbl_sale.NazwaSali)=[Podaj nazwę sali]) AND ((tbl_rezerwacja_zajęć.DataRez)=[Podaj datę zajęć (dd-mm-rrrr)]));
```

Kwerendy

Kwerenda „Lista rezerwacji zajęć” zwraca informacje o zarezerwowanych zajęciach.



KOD SQL:

```
SELECT tbl_rezerwacja_zajęc.DataRez AS Data, tbl_rezerwacja_zajęc.GodzinaRez AS Godzina, tbl_rezerwacja_zajęc.IDZajęc AS [Nr zajęć],  
tbl_rezerwacja_zajęc.IDKlienta AS [Nr klienta], tbl_sale.NazwaSali AS [Nazwa sali]  
FROM tbl_sale INNER JOIN tbl_rezerwacja_zajęc ON tbl_sale.IDSali = tbl_rezerwacja_zajęc.IDSali;
```

Kwerendy

Kwerenda „Lista trenerów” zwraca informacje o trenerach pracujących w naszym centrum.

Lista trenerów

tbl_pracownik

- IDPracownika
- IDPłci
- IDStanowiska
- Miejscowość
- Imię
- Nazwisko
- PensjaMiesięczna

tbl_płeć

- IDPłci
- Płeć
- SymbolPłci

1

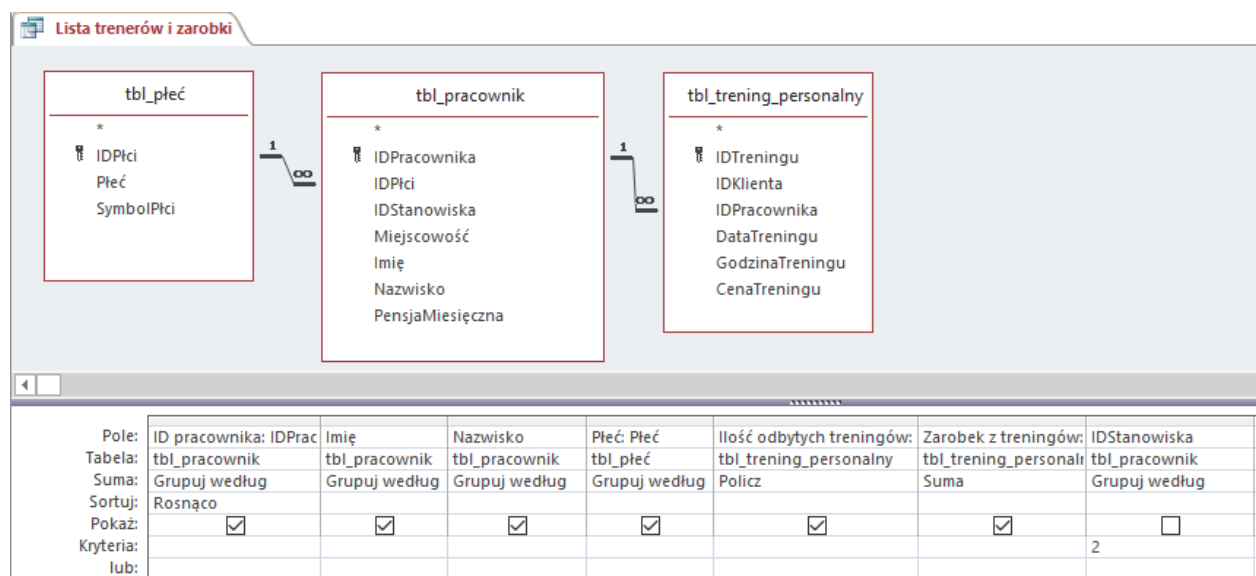
Pole:	ID pracownika: IDPłci	Imię	Nazwisko	Płeć: Płeć	Miejscowość	IDStanowiska
Tabela:	tbl_pracownik	tbl_pracownik	tbl_pracownik	tbl_płeć	tbl_pracownik	tbl_pracownik
Sortuj:	Rosnąco					
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Kryteria:						2
Iub:						

KOD SQL:

```
SELECT tbl_pracownik.IDPracownika AS [ID pracownika], tbl_pracownik.Imię, tbl_pracownik.Nazwisko, tbl_płeć.Płeć AS Płeć,
tbl_pracownik.Miejscowość
FROM tbl_płeć INNER JOIN tbl_pracownik ON tbl_płeć.IDPłci = tbl_pracownik.IDPłci
WHERE (((tbl_pracownik.IDStanowiska)=2)) ORDER BY tbl_pracownik.IDPracownika;
```

Kwerendy

Kwerenda „Lista trenerów i zarobki” zwraca informacje o trenerach pracujących w naszym centrum, liczbę wykonanych przez każdego trenera treningów oraz zarobek z treningów.



KOD SQL:

```
SELECT tbl_pracownik.IDPracownika AS [ID pracownika], tbl_pracownik.Imię, tbl_pracownik.Nazwisko, tbl_plec.Płeć AS Płeć, Count(tbl_trening_personalny.IDPracownika) AS [Ilość odbytych treningów], Sum(tbl_trening_personalny.CenaTreningu) AS [Zarobek z treningów]
FROM (tbl_plec INNER JOIN tbl_pracownik ON tbl_plec.IDPłci = tbl_pracownik.IDPłci) INNER JOIN tbl_trening_personalny ON tbl_pracownik.IDPracownika =
tbl_trening_personalny.IDPracownika
GROUP BY tbl_pracownik.IDPracownika, tbl_pracownik.Imię, tbl_pracownik.Nazwisko, tbl_plec.Płeć, tbl_pracownik.IDStanowiska
HAVING (((tbl_pracownik.IDStanowiska)=2))
ORDER BY tbl_pracownik.IDPracownika;
```

Kwerendy

Kwerenda „Lista treningów danego trenera” zwraca informacje o treningach wykonanych przez danego trenera.

Lista treningów danego trenera

Diagram illustrating the relationship between the **tbl_pracownik** and **tbl_trening_personalny** tables. The relationship is 1-to-many, indicated by the '1' and '∞' symbols. The **tbl_pracownik** table contains fields: IDPracownika, IDPłci, IDStanowiska, Miejscowość, Imię, Nazwisko, PensjaMiesięczna. The **tbl_trening_personalny** table contains fields: IDTreningu, IDKlienta, IDPracownika, DataTreningu, GodzinaTreningu, CenaTreningu.

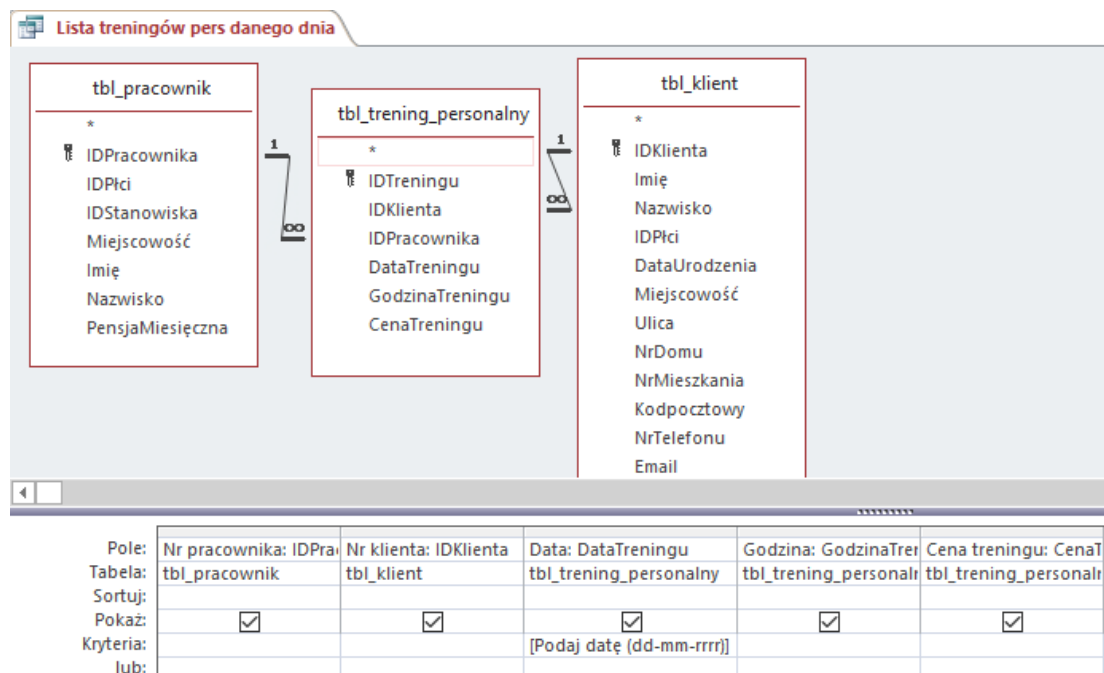
Pole:	Nr pracownika: IDPracownik	Data: DataTreningu	Godzina: GodzinaTreningu	Cena treningu: CenaTreningu
Tabela:	tbl_pracownik	tbl_trening_personalny	tbl_trening_personalny	tbl_trening_personalny
Sortuj:				
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:	[Podaj nr pracownika (trenera)]			
lub:				

KOD SQL:

```
SELECT tbl_pracownik.IDPracownika AS [Nr pracownika], tbl_trening_personalny.DataTreningu AS Data,  
tbl_trening_personalny.GodzinaTreningu AS Godzina, tbl_trening_personalny.CenaTreningu AS [Cena treningu]  
FROM tbl_pracownik INNER JOIN tbl_trening_personalny ON tbl_pracownik.IDPracownika = tbl_trening_personalny.IDPracownika  
WHERE (((tbl_pracownik.IDPracownika)=[Podaj nr pracownika (trenera)]));
```

Kwerendy

Kwerenda „Lista treningów pers danego dnia” zwraca informacje o treningach personalnych, które miały miejsce danego dnia.

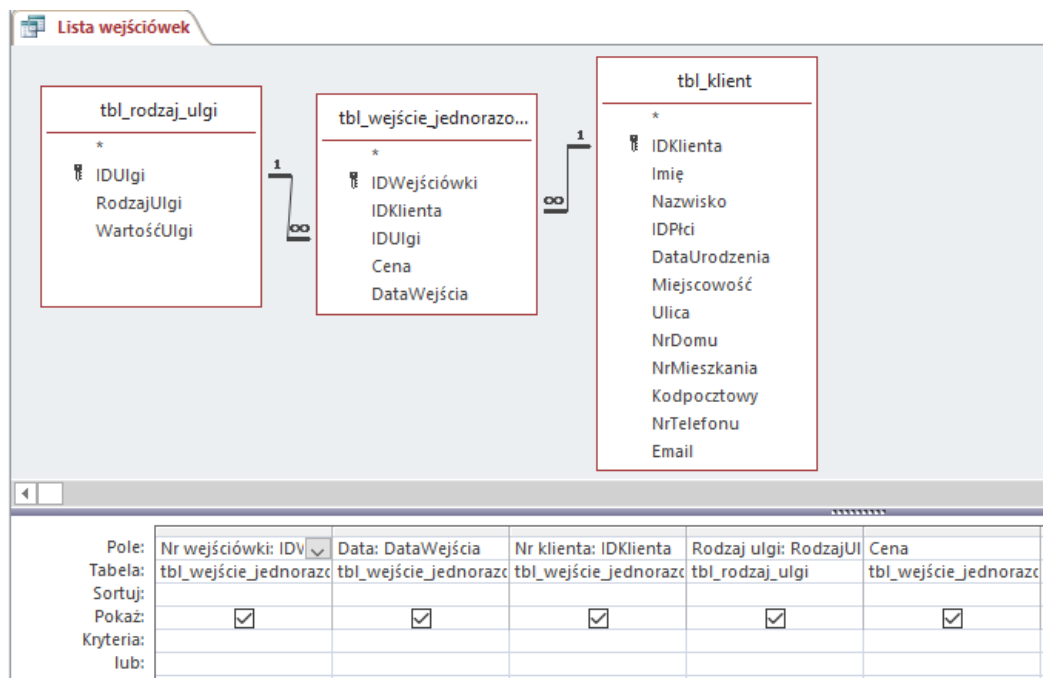


KOD SQL:

```
SELECT tbl_pracownik.IDPracownika AS [Nr pracownika], tbl_klient.IDKlienta AS [Nr klienta], tbl_trening_personalny.DataTreningu AS Data,
tbl_trening_personalny.GodzinaTreningu AS Godzina, tbl_trening_personalny.CenaTreningu AS [Cena treningu]
FROM tbl_pracownik INNER JOIN (tbl_klient INNER JOIN tbl_trening_personalny ON tbl_klient.IDKlienta = tbl_trening_personalny.IDKlienta) ON
tbl_pracownik.IDPracownika = tbl_trening_personalny.IDPracownika
WHERE (((tbl_trening_personalny.DataTreningu)=[Podaj datę (dd-mm-rrrr)]));
```

Kwerendy

Kwerenda „Lista wejściówek” zwraca informację o wejściówkach jednorazowych wykupionych w naszym centrum.

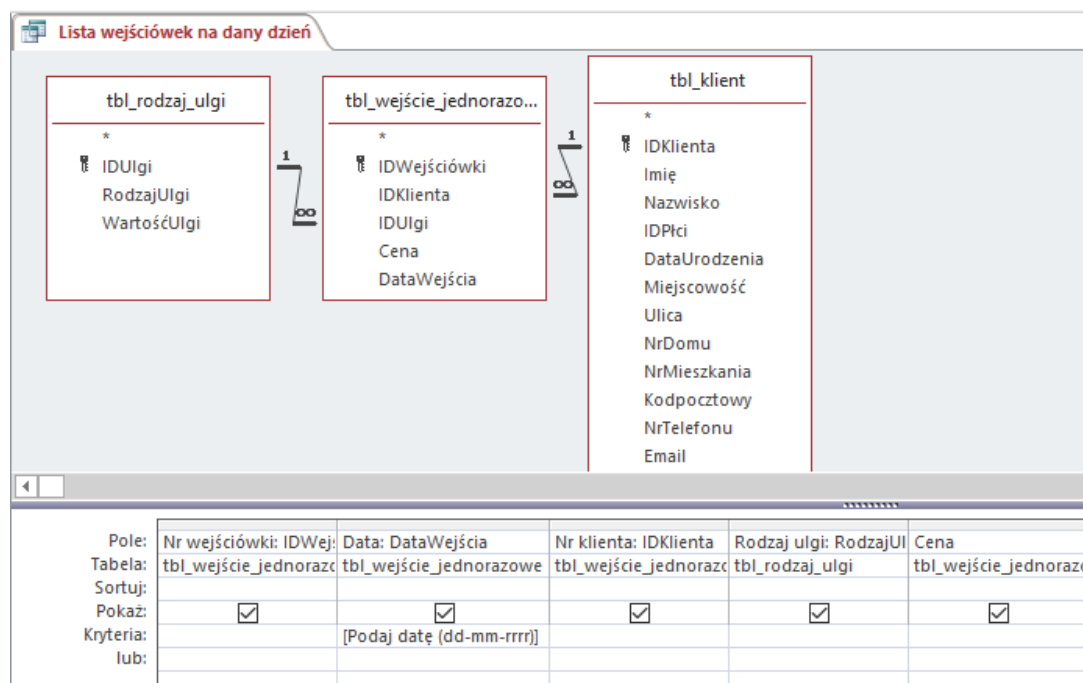


KOD SQL:

```
SELECT tbl_wejście_jednorazowe.IDWejściówki AS [Nr wejściówki], tbl_wejście_jednorazowe.DataWejścia AS Data,  
tbl_wejście_jednorazowe.IDKlienta AS [Nr klienta], tbl_rodzaj_ulgi.RodzajUlgi AS [Rodzaj ulgi], tbl_wejście_jednorazowe.Cena  
FROM tbl_rodzaj_ulgi INNER JOIN (tbl_klient INNER JOIN tbl_wejście_jednorazowe ON tbl_klient.IDKlienta = tbl_wejście_jednorazowe.IDKlienta)  
ON tbl_rodzaj_ulgi.IDUlgi = tbl_wejście_jednorazowe.IDUlgi;
```

Kwerendy

Kwerenda „Lista wejściówek na dany dzień” zwraca informacje o wejściówkach wykupionych danego dnia.

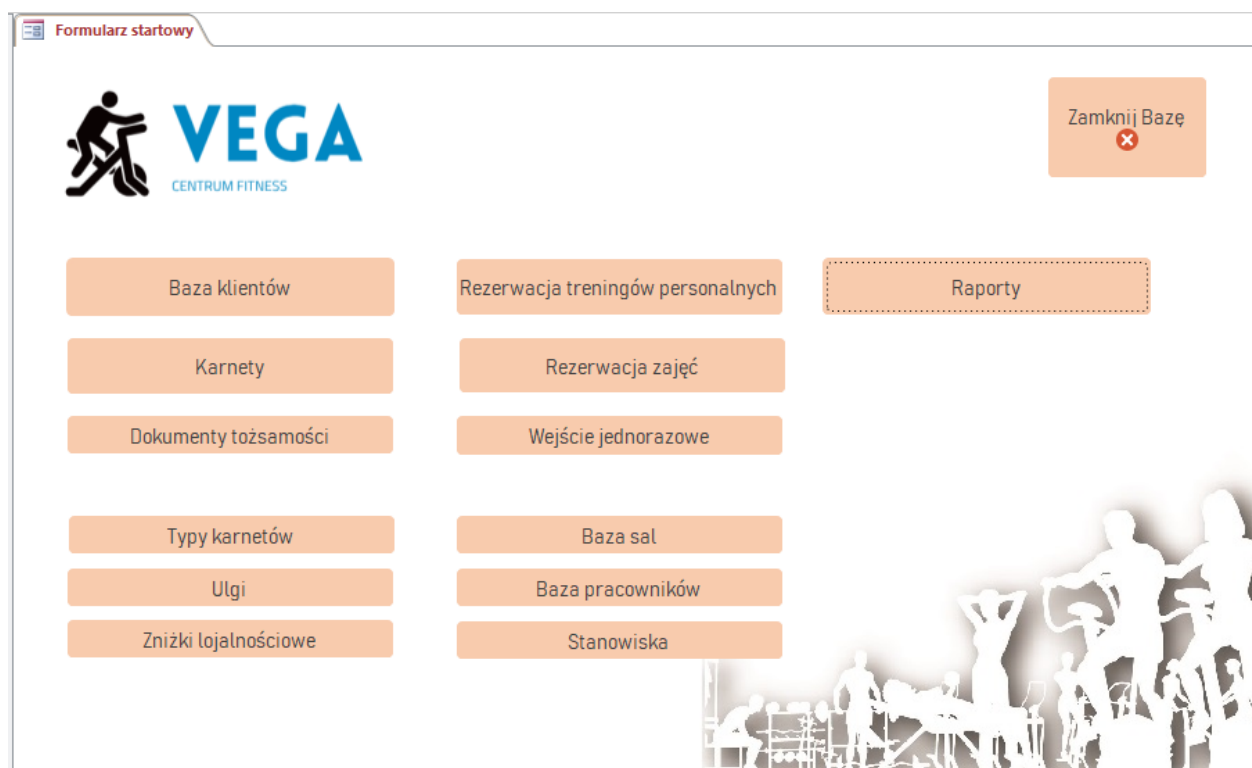


KOD SQL:

```
SELECT tbl_wejście_jednorazowe.IDWejściówki AS [Nr wejściówki], tbl_wejście_jednorazowe.DataWejścia AS Data, tbl_wejście_jednorazowe.IDKlienta AS [Nr klienta], tbl_rodzaj_ulgi.RodzajUlg AS [Rodzaj ulgi], tbl_wejście_jednorazowe.Cena
FROM tbl_rodzaj_ulgi INNER JOIN (tbl_klient INNER JOIN tbl_wejście_jednorazowe ON tbl_klient.IDKlienta = tbl_wejście_jednorazowe.IDKlienta) ON
tbl_rodzaj_ulgi.IDUlg = tbl_wejście_jednorazowe.IDUlg
WHERE (((tbl_wejście_jednorazowe.DataWejścia)=[Podaj datę (dd-mm-rrrr)]));
```


Formularze


Formularz startowy zawiera odnośniki (buttony) do wszystkich innych dostępnych formularzy. Jest formularzem wyjściowym do pracy z bazą danych.



Formularze

Formularz „Baza klientów” służy do wprowadzania danych o nowych klientach. Zawiera odnośniki do innych formularzy potrzebnych w przypadku potrzeby dodania do bazy nowego dowodu osobistego lub karnetu.

Baza klientów



ID Klienta	<input type="text" value="1"/>
Płeć	<input type="text" value="Mężczyzna"/>
Imię	<input type="text" value="Jakub"/>
Nazwisko	<input type="text" value="Piotrowski"/>
Data urodzenia	<input type="text" value="11.08.1996"/>
Miejscowość	<input type="text" value="Łódź"/>
Ulica	<input type="text" value="Jaracza"/>
Numer domu	<input type="text" value="20"/>
Numer mieszkania	<input type="text" value="1"/>
Kod pocztowy	<input type="text" value="95-354"/>
Numer telefonu kom.	<input type="text" value="525-259-852"/>
E-mail	<input type="text" value="kuba@gmail.com"/>

DOKUMENTY TOŻSAMOŚCI

NOWY dokument

Edytuj dokument klienta

Baza dokumentów

Nowy

Usuń

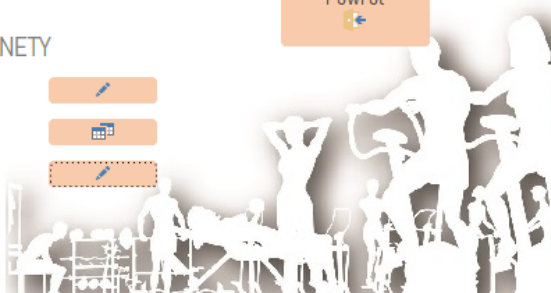
Powrót

KARNETY

NOWY karnet

Historia karnetów

Baza karnetów




Formularze

Formularz „Dokumenty tożsamości” umożliwia dodanie lub edycję informacji o dokumencie osobistym klienta.

Formularz startowy

Dokumenty tożsamości



ID klienta

Nr dowodu

PESEL

Obywatelstwo

Nr paszportu

1

CDS543634

96081186954

polskie

◀◀

◀


▶

▶▶

Nowy

Usuń

Powrót




Formularze

Formularz „Baza kartetów” służy do dodawania nowych kartetów oraz do obliczenia ceny kartetu.

Formularz startowy

Baza kartetów



ID kartetu

1

ID klienta

1

Typ kartetu

Od 8 do 16

Typ zniżki

3 miesiące

Typ ulgi

Student

Wprowadź datę rozpoczęcia i oblicz cenę

Cena kartetu

183,33 zł

Data rozpoczęcia

10.02.2019

Data zakończenia

10.05.2019

NOWY kartet


Historia kartetów

Baza klientów

Nowy

Usuń

Powrót



Wprowadź datę rozpoczęcia i oblicz cenę

Przycisk służy do obliczenia dat: daty rozpoczęcia i daty zakończenia okresu ważności karnetu. Następnie obliczana jest finalna cena karnetu.

KOD VBA:


```
Private Sub Polecenie20_Click()  
Dim FirstDate As Date  
Dim IntervalType As String  
Dim Number As Integer  
Dim DataDzis As Date  
DataDzis = Date  
IntervalType = "m"  
FirstDate = InputBox("Wprowadź datę rozpoczęcia w formacie: (dd-mm-rrrr)")  
If FirstDate >= DataDzis Then  
Me.DataRozp.Value = FirstDate  
Number = Me.KombiIDZnizki.Column(3)  
Me.DataZak.Value = DateAdd(IntervalType, Number, FirstDate)  
End If  
If FirstDate < DataDzis Then  
MsgBox ("Wprowadź poprawną datę w formacie: (dd-mm-rrrr)" & vbCrLf & "Nie wolno wprowadzać dat wcześniejszych niż dzisiejsza!")  
Exit Sub  
End If  
If IsNull(Me.KombiTypKarnetu) Or IsNull(Me.KombiDULgi) Or IsNull(Me.KombiIDZnizki) Then  
MsgBox ("Aby obliczyć cenę należy wybrać wszystkie pola!")  
Else  
Me.CenaKarnetu.Value = Me.KombiTypKarnetu.Column(2) * Me.KombiIDZnizki.Column(2) * Me.KombiDULgi.Column(2) * Me.KombiIDZnizki.Column(3)  
End If  
End Sub
```

Formularze

Formularz „Baza treningów personalnych” umożliwia dodanie nowej rezerwacji treningu personalnego.

Formularz startowy

Baza treningów personalnych



Numer treningu

1

ID klienta

1

Numer trenera

7

Data treningu

10.02.2019

Godzina treningu

08:00

Cena treningu

50,00 zł

Sprawdź dostępność terminu

Baza klientów


Lista trenerów

Historia treningów

Nowy

Usuń

Powrót



Sprawdź dostępność terminu

Przycisk służy do sprawdzenia czy wybrany termin treningu personalnego jest dostępny. Jeżeli dany trener ma już zarezerwowany trening lub klient ma już zarezerwowany w tym czasie trening, program wyświetli stosowny komunikat.

KOD VBA:

```
Private Sub Polecenie15_Click()  
Dim i As Integer  
i = DCount("[IDTreningu]", "tbl_trening_personalny", "[IDPracownika] = IDPracownika.Value AND [GodzinaTreningu]=GodzinaTreningu.Value  
AND [DataTreningu] = DataTreningu.Value")  
j = DCount("[IDTreningu]", "tbl_trening_personalny", "[IDKlienta] = IDKlienta.Value AND [GodzinaTreningu]=GodzinaTreningu.Value AND  
[DataTreningu] = DataTreningu.Value")  
If (i = 0 And j = 0) Then  
    MsgBox ("Termin jest dostępny!")  
End If  
If (i > 0) Then  
    MsgBox ("W bazie jest już zarezerwowany trening na tę datę!" & vbCrLf & vbCrLf & "ID pracownika: " & IDPracownika.Value & " Data: " &  
DataTreningu.Value & " Godzina: " & GodzinaTreningu.Value & vbCrLf & vbCrLf & "Zmień datę lub trenera.")  
    IDPracownika.Value = 0  
End If  
If (j > 0) Then  
    MsgBox ("Klient o ID: " & IDKlienta.Value & " w tym dniu i godzinie ma już zarezerwowany trening!")  
    IDPracownika.Value = 0  
End If  
End Sub
```

Formularze

Formularz „Baza rezerwacji zajęć” umożliwia dodanie nowej rezerwacji na zajęcia.

Formularz startowy

Baza rezerwacji zajęć



ID zajęć

1

ID klienta

1

Sala

Squash

Godzina rezerwacji

10:00

Data rezerwacji

14.02.2019

Cena

40,00 zł

Baza klientów

◀◀ ◀ ▶ ▶▶

Lista sal

Nowy

Usuń

Powrót

Sprawdź czy są wolne miejsca i oblicz cenę



Sprawdź czy są wolne miejsca i oblicz cenę

Przycisk służy do sprawdzenia czy na wybrane zajęcia jest jeszcze wolne miejsce. W przypadku gdy pojemność sali została przekroczona zostanie wyświetlony odpowiedni komunikat. Jeżeli klient ma w tym czasie zarezerwowane inne zajęcia wyświetli się komunikat.

KOD VBA:

```
Private Sub Polecenie15_Click()  
i = DCount("[IDKlienta]", "tbl_rezerwacja_zajęć", "[IDSali] = IDSali.Value AND [GodzinaRez]=GodzinaRez.Value AND [DataRez] = DataRez.Value")  
j = DLookup("[Pojemność]", "tbl_sale", "IDSali.Value = [IDSali]")  
k = DCount("[IDKlienta]", "tbl_rezerwacja_zajęć", "[IDKlienta]=IDKlienta.Value AND [GodzinaRez]=GodzinaRez.Value AND [DataRez] =  
DataRez.Value")  
If (IDSali.Value = 0) Then  
    MsgBox ("Wybierz salę!")  
    Exit Sub  
ElseIf (i >= j) Then  
    MsgBox ("Na zajęcia w dniu " & DataRez.Value & " godzina " & GodzinaRez.Value & " nie ma wolnych miejsc!" & vbCrLf & vbCrLf & "Wybierz  
inny termin lub salę.")  
    IDSali.Value = 0  
    Exit Sub  
End If  
If (k = 0) Then  
    MsgBox ("Są miejsca!")  
    CenaZajęć.Value = DLookup("[CenaZaWejście]", "tbl_sale", "IDSali.Value = [IDSali]")  
ElseIf (k > 0) Then  
    MsgBox ("Klient o ID: " & IDKlienta.Value & " ma już zarezerwowane zajęcia w tym dniu i godzinie!")  
    IDSali.Value = 0  
End If  
End Sub
```

Formularze

Formularz „Baza wejść jednorazowych” umożliwia dodanie nowej wejściówki jednorazowej.

Formularz startowy Baza wejść jednorazowych

VEGA
CENTRUM FITNESS

ID wejściówki: 1
ID klienta: 4
Typ ulgi: Student
Cena: 18,00 zł
Data wejścia: 10.02.2019

Baza klientów

Nowy Usuń
Powrót

Cena wprowadzana jest po wybraniu wartości z Combobox'a zawierającego typ ulgi.

KOD VBA:

```
Private Sub IDUlgiv_Change()
```

```
Dim j As Integer
```

```
j = 20 ' cena standardowej wejściówki bez ulg
```

```
Me.Cena.Value = j * Me.IDUlgiv.Column(2)
```

```
End Sub
```


Formularze

Formularze „Typy karnetów”, „Rodzaje ulg” i „Zniżki lojalnościowe” umożliwiają wprowadzenie lub edycję rodzajów karnetów, rodzajów ulg i zniżek lojalnościowych przysługujących klientom wykupującym karnet lub wejściówkę jednorazową.

Typy kartonów

Rodzaje ulg

Zniżki lojalnościowe



ID typu kartonu

Typ kartonu

Cena podstawowa

1

Całodobowy

120,00 zł

◀◀

◀


▶

▶▶

Nowy

Usuń

Powrót



ID Ulg

Rodzaj ulgi

Wartość ulgi

1

Brak

1

◀◀

◀

▶

▶▶

Nowy


Usuń

Powrót

Typy kartonów

Rodzaje ulg

Zniżki lojalnościowe



ID zniżki

Nazwa zniżki

Wartość zniżki

Liczba miesięcy

1

Brak (1 miesiąc)

1

1

◀◀

◀

▶

▶▶

Nowy

Usuń


Powrót

Formularze

Formularz „Baza sal” umożliwia dodanie lub edycję sal zajęciowych w naszym centrum.

Formularz startowy

Baza sal



ID sali

Nazwa sali

Numer sali

Pojemność sali

Cena za wejście

1

Fitness

12

8

25,00 zł

◀◀

◀


▶

▶▶

Nowy

Usuń

Powrót




Formularze

Formularz „Baza pracowników” umożliwia dodanie lub edycję pracowników naszego centrum.

Formularz startowy

Baza pracowników



ID pracownika

Płeć

Stanowisko

Imię

Nazwisko

Miejscowość

Pensja miesięczna

1

Mężczyzna

Technik serwisant

Kamil

Grosicki

Zgierz

3 200,00 zł

⏮

⏪


⏩

⏭

Nowy

Usuń

Powrót




Formularze

Formularz „Typy stanowisk” umożliwia dodanie lub edycję nazw stanowisk w naszym centrum.

Formularz startowy

Typy stanowisk



ID Stanowiska

Stanowisko

1

Kierownik

◀◀

◀


▶

▶▶

Nowy

Usuń

Powrót



Formularze

Formularz „Raporty” umożliwia szybki dostęp do wszystkich raportów znajdujących się w naszej bazie danych.

Formularz startowy

Raporty



Klienci

Baza klientów

Baza klientów z miejscowości

Karnety

Baza karnetów w miesiącu

Lista karnetów klienta

Wejściówki

Lista wejściówek

Lista wejściówek na dany dzień

Treningi personalne

Historia rezerwacji treningów dla ID klienta

Lista treningów danego dnia

Lista treningów danego trenera

Lista trenerów i zarobki

Zajęcia grupowe

Lista zapisanych na zajęcia w danym dniu

Rezerwacja zajęć według dnia i sali

Pracownicy

Baza pracowników

Baza trenerów

Sale

Baza sal

Powrót



Raporty

W dokumentacji zostały przedstawione tylko najważniejsze raporty.

Raport „Baza klientów z miejscowości” umożliwia wyświetlenie danych klientów z danej miejscowości. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

Przed wyświetleniem raportu należy w oknie dialogowym wpisać nazwę interesującej nas miejscowości.

 Formularz startowy

 Baza klientów raport

**VEGA**
CENTRUM FITNESS

Baza klientów z danej miejscowości

IDKlienta	Imię	Nazwisko	Płeć	Data urodzenia	Miejscowość
1	Jakub	Piotrowski	Mężczyzna	11.08.1996	Łódź
2	Bartłomiej	Trzask	Mężczyzna	12.02.1996	Łódź
5	Jim	Berkeley	Mężczyzna	15.07.1990	Łódź
7	Katarzyna	Wawrzyniak	Kobieta	30.06.1988	Łódź
12	Natan	Mazurkiewicz	Mężczyzna	26.09.1998	Łódź

Raporty

Raport „Baza karnetów w miesiącu” umożliwia wyświetlenie danych o sprzedanych karnetach, w podziale na miesiące. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

Formularz startowy

Baza karnetów w miesiącu



Baza karnetów w danym miesiącu

Miesiąc	Nr karnetu	Nr klienta	Data rozpoczęcia	Typ karnetu	Rodzaj ulgi	Nazwa zniżki	CenaKarnetu
luty 2019							
	1	1	10.02.2019	Od 8 do 16	Student	3 miesiące	183,33 zł
	2	2	11.02.2019	Całodobowy	Student	6 miesięcy	596,16 zł
	3	6	10.02.2019	Od 16 do 24	Senior	3 miesiące	232,80 zł
	5	3	10.02.2019	Całodobowy	Brak	Brak (1 miesiąc)	120,00 zł
	7	4	12.02.2019	Od 16 do 24	Student	12 miesięcy	950,40 zł
	8	5	16.02.2019	Od 16 do 24	Senior	3 miesiące	232,80 zł
	9	7	20.02.2019	Całodobowy	Brak	Brak (1 miesiąc)	120,00 zł
	10	9	11.02.2019	Od 16 do 24	Senior	3 miesiące	232,80 zł
	11	10	11.02.2019	Od 8 do 16	Student	3 miesiące	183,33 zł
marzec 2019							
	6	3	10.03.2019	Od 8 do 16	Student	3 miesiące	183,33 zł
	12	11	19.03.2019	Od 8 do 16	Senior	Brak (1 miesiąc)	56,00 zł
maj 2019							
	14	1	10.05.2019	Od 8 do 16	Student	Brak (1 miesiąc)	63,00 zł

niedziela, 10 lutego 2019

Strona 1 z 1

Raporty

Raport „Historia rezerwacji treningów dla IDKlienta” umożliwia wyświetlenie danych o treningach zarezerwowanych przez danego klienta. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

Przed wyświetleniem raportu należy w oknie dialogowym wpisać ID klienta dla którego chcemy zobaczyć raport.

Historia rezerwacji treningów dla IDKlienta

<div><div>Historia rezerwacji treningów klienta</div></div>					
ID treningu	ID klienta	Data treningu	Godzina treningu	Trener	Cena treningu
4	4	10.02.2019	14:00	Oskar Orzechowski	60,00 zł
niedziela, 10 lutego 2019					Strona 1 z 1

Raporty

Raport „Lista karnetów klienta ” umożliwia wyświetlenie danych o wykupionych przez danego klienta karnetach. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

Przed wyświetleniem raportu należy w oknie dialogowym wpisać ID klienta dla którego chcemy zobaczyć raport.

 Lista karnetów klienta raport

		Lista karnetów danego klienta				
ID karnetu	ID klienta	Imię	Nazwisko	Cena karnetu	Data rozpoczęcia	Data zakończenia
1	1	Jakub	Piotrowski	183,33 zł	10.02.2019	10.05.2019
14	1	Jakub	Piotrowski	63,00 zł	10.05.2019	10.06.2019


niedziela, 10 lutego 2019

Strona 1 z 1

Raporty

Raport „Lista osób zapisanych na zajęcia w danym dniu ” umożliwia wyświetlenie danych o osobach, które zapisały się na dane zajęcia w danym dniu. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

Przed wyświetleniem raportu należy w oknie dialogowym wpisać nazwę sali, a w następnym oknie – datę zajęć.

 Lista osób zapisanych na zajęcia w danym dniu raport

<div><div>Lista osób zapisanych na zajęcia w danym dniu</div></div>					
Data	Godzina	Nr klienta	Nazwa sali	Imię	Nazwisko
14.02.2019	10:00	1	Squash	Jakub	Piotrowski
14.02.2019	10:00	2	Squash	Bartłomiej	Trzask
14.02.2019	10:00	3	Squash	Katarzyna	Jurewicz
14.02.2019	10:00	4	Squash	Magdalena	Brzezińska
14.02.2019	12:00	4	Squash	Magdalena	Brzezińska

niedziela, 10 lutego 2019

Strona 1 z 1

Raporty

Raport „Lista trenerów i zarobki” umożliwia wyświetlenie danych o trenerach, ilości treningów, które prowadzili oraz o sumie zarobków z treningów, w podziale na trenerów. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

 Lista trenerów i zarobki raport



Lista trenerów i zarobki

ID pracownika	Imię	Nazwisko	Płeć	Ilość odbytych treningów	Zarobek z treningów
7	Jan	Tacikiewicz	Mężczyzna	2	100,00 zł
8	Katarzyna	Koneczna	Kobieta	2	110,00 zł
9	Kamil	Kosowski	Mężczyzna	2	160,00 zł
18	Oskar	Orzechowski	Mężczyzna	1	60,00 zł
19	Gabriela	Pająk	Kobieta	1	90,00 zł

niedziela, 10 lutego 2019


Strona 1 z 1

Raporty

Raport „Lista treningów danego trenera” umożliwia wyświetlenie danych o treningach wykonanych przez danego trenera. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

Przed wyświetleniem raportu należy wpisać w oknie dialogowym ID pracownika (trenera), którego raport ma dotyczyć.

 Lista treningów danego trenera raport

<div><div>Lista treningów danego trenera</div></div>				
Data	Godzina	Nr pracownika	Nr klienta	Cena treningu
10.02.2019	08:00	8	2	60,00 zł
10.02.2019	14:00	8	5	50,00 zł

niedziela, 10 lutego 2019

Strona 1 z 1

Raporty

Raport „Lista treningów pers danego dnia” umożliwia wyświetlenie danych o treningach odbytych danego dnia. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

Przed wyświetleniem raportu należy wpisać w oknie dialogowym datę , dla której chcemy zobaczyć listę treningów.

 Lista treningów pers danego dnia raport

<div><div>Lista treningów personalnych danego dnia</div></div>				
Data	Godzina	Nr pracownika	Nr klienta	Cena treningu
10.02.2019	08:00	9	3	70,00 zł
10.02.2019	08:00	8	2	60,00 zł
10.02.2019	08:00	7	1	50,00 zł
10.02.2019	14:00	8	5	50,00 zł
10.02.2019	14:00	19	5	90,00 zł
10.02.2019	14:00	18	4	60,00 zł
10.02.2019	20:00	7	5	50,00 zł


niedziela, 10 lutego 2019

Strona 1 z 1

Raporty

Raport „Lista wejściówek na dany dzień” umożliwia wyświetlenie danych o wykupionych danego dnia wejściówkach. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

Przed wyświetleniem raportu należy wpisać w oknie dialogowym datę , dla której chcemy zobaczyć listę wejściówek.

 Lista wejściówek na dany dzień raport

<div><div>Lista wejściówek na dany dzień</div></div>				
Data	Nr wejściówki	Nr klienta	Rodzaj ulgi	Cena
niedziela, 10 luty 2019	1	4	Student	18,00 zł
	3	6	Brak	20,00 zł
	11	7	Brak	20,00 zł

niedziela, 10 lutego 2019

Strona 1 z 1

Raporty

Raport „Rezerwacja zajęć według dnia i sali” umożliwia wyświetlenie danych o zajęciach odbywających się w naszym centrum, w podziale na datę i rodzaj sali. Raport korzysta z przedstawionej wcześniej kwerendy o tej samej nazwie.

 Rezerwacja zajęć według dnia i sali				
Data	Nazwa sali	Godzina	Nr zajęć	Nr klienta
niedziela, 10 luty				
	Fitness	10:00	10	9
	Rowerowa	10:00	9	6
	Squash 2	10:00	7	7
		10:00	5	5
czwartek, 14 luty				
	Squash	10:00	4	4
		10:00	3	3
		10:00	2	2
		10:00	1	1
		12:00	11	4

niedziela, 10 lutego 2019

Strona 1 z 1

Przewodnik po aplikacji

Po otwarciu bazy danych pojawi się formularz startowy. Jest on swojego rodzaju punktem wyjściowym do poruszania się po aplikacji.

Zawiera przyciski przekierowujące do każdego formularza w programie.

Przewodnik jest przeznaczony dla recepcjonistów i kierowników chcących wprowadzić zmiany w systemie.



1. Dodawanie danych nowego klienta

Aby rozpocząć proces rejestracji nowego klienta należy przejść do „Bazy klientów”.

Lista kroków:

1) Wypełnienie formularza danymi nowego klienta w „Bazie klientów”.

ID Klienta	<input type="text" value="1"/>
Płeć	<input type="text" value="Mężczyzna"/>
Imię	<input type="text" value="Jakub"/>
Nazwisko	<input type="text" value="Piotrowski"/>
Data urodzenia	<input type="text" value="11.08.1996"/>
Miejscowość	<input type="text" value="Łódź"/>
Ulica	<input type="text" value="Jaracza"/>
Numer domu	<input type="text" value="20"/>
Numer mieszkania	<input type="text" value="1"/>
Kod pocztowy	<input type="text" value="95-354"/>
Numer telefonu kom.	<input type="text" value="525-259-852"/>
E-mail	<input type="text" value="kuba@gmail.com"/>

2) Przejście do formularza tworzenia nowego dokumentu tożsamości poprzez kliknięcie przycisku „NOWY dokument”.

NOWY dokument



3) W formularzu „Dowody osobiste” w polu „ID klienta” należy przepisać „ID klienta” przypisane automatycznie w formularzu „Baza klientów”.

4) Powrót do „Bazy klientów” poprzez kliknięcie przycisku „Powrót”.

2. Edycja danych istniejącego w bazie klienta

Aby rozpocząć proces edycji danych istniejącego już w bazie klienta należy przejść do „Bazy klientów”.

Lista kroków:

1) Należy wyszukać rekord zawierający dane klienta. Należy wprowadzić ID Klienta w polu przycisków nawigacyjnych znajdującym się na dole formularza lub frazę ułatwiającą odszukanie danego klienta w polu „Wyszukaj” i kliknąć Enter.

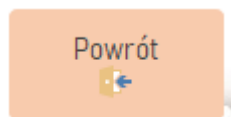


2) Edytować dane klienta w formularzu.

3) W przypadku chęci zmiany informacji o dokumencie tożsamości klienta należy kliknąć przycisk „Edytuj dokument klienta”. Formularz edycji otworzy się na rekordzie zawierającym dane wybranego przez nas wcześniej klienta.



4) Powrót do „Bazy klientów” poprzez kliknięcie przycisku „Powrót”.



3. Dodanie nowej wejściówki jednorazowej

Aby utworzyć nową wejściówkę musimy najpierw utworzyć profil klienta (dodać go do odpowiednio „Bazy danych klientów” i „Dokumentów tożsamości”. Jeżeli dane klienta nie zostały wprowadzone cofnij się do punktu 1.

Aby dodać nowe wejście jednorazowe należy kliknąć przycisk „Wejście jednorazowe”. Zostaniemy przekierowani do formularza, w którym należy wpisać informacje dotyczące wejściówki tj. wprowadzić ID klienta, wybrać typ ulgi mu przysługującej i datę wejścia.

ID wejściówki	<input type="text" value="1"/>
ID klienta	<input type="text" value="4"/>
Typ ulgi	<input type="text" value="Student"/> ▼
Cena	<input type="text" value="18,00 zł"/>
Data wejścia	<input type="text" value="10.02.2019"/>

Lista kroków:

0) Jeżeli nie jesteś pewien co do numeru ID klienta, skorzystaj z przycisku przekierowującego do „Bazy klientów”.

Baza klientów

- 1) Wprowadź ID klienta.
- 2) Wybierz odpowiadający mu typ ulgi. Cena zostanie automatycznie obliczona.
- 3) Wprowadź datę wejściówki.

Wejściówki mogą być kupowane z wyprzedzeniem na przyszłe dni.

4. Rezerwacja treningu personalnego

Aby zarezerwować trening musimy najpierw utworzyć profil klienta (dodać go do odpowiednio „Bazy danych klientów” i „Dokumentów tożsamości”. Jeżeli dane klienta nie zostały wprowadzone cofnij się do punktu 1.

0) Jeżeli nie jesteś pewien co do numeru ID klienta, skorzystaj z przycisku przekierowującego do „Bazy klientów”.

Baza klientów

0) Na życzenie klienta można sprawdzić historię jego treningów. Należy kliknąć przycisk „Historia treningów” i wprowadzić ID Klienta.

Historia treningów



0) Lista trenerów dostępna jest po kliknięciu przycisku „Lista trenerów”.

Lista trenerów



1) Wypełnij formularz danymi ustalonymi po rozmowie z klientem.

2) Kliknij w przycisk „Sprawdź dostępność terminu”.

a) Jeżeli wybrany trener ma już zarezerwowany inny trening w tym czasie na ekranie wyskoczy stosowny komunikat. Zmień trenera lub datę.

b) Jeżeli klient ma już zarezerwowany trening w tym czasie na ekranie wyskoczy stosowny komunikat. Zmień datę.

3) Jeżeli termin jest dostępny, na ekranie pojawi się komunikat „Termin dostępny!”.

5. Rezerwacja na zajęcia

Aby zarezerwować klienta na zajęcia musimy najpierw utworzyć profil klienta (dodać go do odpowiednio „Bazy danych klientów” i „Dokumentów tożsamości”. Jeżeli dane klienta nie zostały wprowadzone cofnij się do punktu 1.

Lista kroków:

0) Jeżeli nie jesteś pewien co do numeru ID klienta, skorzystaj z przycisku przekierowującego do „Bazy klientów”.

Baza klientów

1) Wypełnij formularz danymi ustalonymi po rozmowie z klientem.

ID zajęć	<input type="text" value="1"/>
ID klienta	<input type="text" value="1"/>
Sala	<input type="text" value="Squash"/>
Godzina rezerwacji	<input type="text" value="10:00"/>
Data rezerwacji	<input type="text" value="14.02.2019"/>
Cena	<input type="text" value="40,00 zł"/>

a) Lista sal dostępna jest po kliknięciu przycisku „Lista sal”.

Lista sal

2) Kliknij w przycisk „Sprawdź czy są wolne miejsca i oblicz cenę”.

Sprawdź czy są wolne miejsca i oblicz cenę

a) Jeżeli nie ma już wolnych miejsc na zajęcia (tj. liczba zapisanych klientów jest równa pojemności sali) na ekranie pojawi się stosowny komunikat.

b) Jeżeli klient ma już zarezerwowane zajęcia w tym czasie na ekranie pojawi się stosowny komunikat.

3) Cena zostanie automatycznie wprowadzona.

6. Dodanie nowego karnetu

Aby dodać nowy karnet musimy najpierw utworzyć profil klienta (dodać go do odpowiednio „Bazy danych klientów” i „Dokumentów tożsamości”. Jeżeli dane klienta nie zostały wprowadzone cofnij się do punktu 1.

Lista kroków:

0) Jeżeli nie jesteś pewien co do numeru ID klienta, skorzystaj z przycisku przekierowującego do „Bazy klientów”.

0) Na życzenie klienta można skorzystać z przycisku „Historia karnetów” i wprowadzić ID Klienta aby dowiedzieć się jakie karnety zakupił w przeszłości.

Historia karnetów



1) Kliknij w przycisk „NOWY karnet”.

NOWY karnet



2) Wypełnij formularz danymi klienta i informacjami dotyczącymi typu karnetu i okresu na jaki zostanie zakupiony.

3) Kliknij przycisk „Wprowadź datę rozpoczęcia i oblicz cenę”. Cena jak i okres ważności karnetu zostaną automatycznie obliczone i wprowadzone.

Wprowadź datę rozpoczęcia i oblicz cenę

Cena karnetu obliczana jest ze wzoru – $a*b*c*d$ np. 100 zł (karnet całodobowy) * 0,9 (student) * 0,97 (3 miesiące z góry) * 3 (miesiące)

a - cena podstawowa karnetu (wprowadzana w typie karnetu)

b - wartości ulgi (wprowadzana w rodzajach ulgi)

c - wartość zniżki lojalnościowej (wprowadzana w zniżkach lojalnościowych)

d - ilość wykupionych miesięcy (wprowadzana w zniżkach lojalnościowych)

7. Dodanie i edycja typów karnetów

Aby dodać lub edytować typy karnetów należy kliknąć przycisk „Typy karnetów”.

Nowemu typowi karnetu należy przypisać nowy numer ID. Wprowadzamy informacje o typie karnetu i jego cenę podstawową.

Typ karnetu - może to być typ np. całodobowy, od 8 do 16, od 16 do 24 itp.

ID typu karnetu	<input type="text" value="1"/>
Typ karnetu	<input type="text" value="Całodobowy"/>
Cena podstawowa	<input type="text" value="120,00 zł"/>

8. Dodanie i edycja rodzaju ulg

Aby dodać lub edytować rodzaje ulg należy kliknąć przycisk „Rodzaje ulg”.

Nowemu typowi ulgi należy przypisać nowy numer ID. Wprowadzamy informacje o rodzaju ulgi i wartość ulgi.

Rodzaj ulgi – może to być to być ulga studencka, seniorska, rodzinna itp.

ID Ulgi	<input type="text" value="3"/>
Rodzaj ulgi	<input type="text" value="Senior"/>
Wartość ulgi	<input type="text" value="0,8"/>

Jaką wpisać „Wartość ulgi”? Przykładowe wartości. Stosować analogicznie do przykładu.

Procent obniżonej ceny	Wartość, którą należy wpisać w polu „Wartość ulgi”
0%	1
5%	0,95
10%	0,90
20%	0,80

9. Dodanie i edycja zniżek lojalnościowych

Zniżka lojalnościowa to rabat udzielany przy zakupie karnetu na kilka miesięcy.

Aby dodać lub edytować zniżki lojalnościowe należy kliknąć przycisk „Zniżki lojalnościowe”.

Nowej zniżce lojalnościowej przypisujemy nowy numer ID. Wprowadzamy nazwę zniżki, wartość zniżki i liczbę miesięcy.

ID zniżki	<input type="text" value="2"/>
Nazwa zniżki	<input type="text" value="3 miesiące"/>
Wartość zniżki	<input type="text" value="0,97"/>
Liczba miesięcy	<input type="text" value="3"/>

Jaką wpisać „Wartość zniżki” i „Liczbę miesięcy”? Przykładowe wartości. Stosować analogicznie do przykładu.

Procent obniżonej ceny	Wartość, którą należy wpisać w polu „Wartość zniżki”	Wartość, która należy wpisać w polu „Liczba miesięcy”
0%	1	1
3%	0,97	3
6%	0,94	6
12%	0,88	12

10. Dodanie i edycja sal

W salach odbywają się zajęcia, każda sala ma określone wyposażenie, przeznaczenie, pojemność i cenę za jednorazowe wejście.

Aby dodać lub edytować bazę sal należy kliknąć przycisk „Baza sal”.

Należy wprowadzić nowe ID Sali, nazwę (opis) sali, podać numer sali, jej pojemność i cenę.

ID sali	<input type="text" value="1"/>
Nazwa sali	<input type="text" value="Fitness"/>
Numer sali	<input type="text" value="12"/>
Pojemność sali	<input type="text" value="8"/>
Cena za wejście	<input type="text" value="25,00 zł"/>

11. Dodanie i edycja bazy pracowników

Nasze centrum zatrudnia wielu pracowników. Formularz umożliwia wprowadzenie danych na ich temat.

Aby dodać lub edytować dane pracownika należy kliknąć w przycisk „Baza pracowników”.

ID pracownika	<input type="text" value="1"/>
Płeć	<input type="text" value="Mężczyzna"/>
Stanowisko	<input type="text" value="Technik serwisant"/>
Imię	<input type="text" value="Kamil"/>
Nazwisko	<input type="text" value="Grosicki"/>
Miejscowość	<input type="text" value="Zgierz"/>
Pensja miesięczna	<input type="text" value="3 200,00 zł"/>

12. Dodanie i edycja stanowisk

Każdy pracownik posiada określoną rolę i funkcję. Formularz umożliwia tworzenie opisów stanowisk.

Aby dodać lub edytować informację o danym stanowisku należy kliknąć w przycisk „Stanowiska”.

Nowemu stanowisku należy przypisać numer ID i wprowadzić jego opis w polu „Stanowisko”.

ID Stanowiska	<input type="text" value="1"/>
Stanowisko	<input type="text" value="Kierownik"/>

13. Widok raportów

Kliknięcie przycisku „Raporty” przeniesie nas do nowego formularza, gdzie możemy wyświetlać interesujące nas raporty.

Klienci	Wejściówki	Zajęcia grupowe	
<input type="button" value="Baza klientów"/>	<input type="button" value="Lista wejściówek"/>	<input type="button" value="Lista zapisanych na zajęcia w danym dniu"/>	<input type="button" value="Powrót"/>
<input type="button" value="Baza klientów z miejscowości"/>	<input type="button" value="Lista wejściówek na dany dzień"/>	<input type="button" value="Rezerwacja zajęć według dnia i sali"/>	
Karnety	Treningi personalne	Pracownicy	
<input type="button" value="Baza karnetów w miesiącu"/>	<input type="button" value="Historia rezerwacji treningów dla ID klienta"/>	<input type="button" value="Baza pracowników"/>	
<input type="button" value="Lista karnetów klienta"/>	<input type="button" value="Lista treningów danego dnia"/>	<input type="button" value="Baza trenerów"/>	
	<input type="button" value="Lista treningów danego trenera"/>	<input type="button" value="Sale"/>	
	<input type="button" value="Lista trenerów i zarobki"/>	<input type="button" value="Baza sal"/>	

Zakończenie

Dzięki pracy wykonanej przy naszym projekcie nauczyliśmy się czym są relacyjne bazy danych i jak projektować je od podstaw, aby następnie przystąpić do efektywnej implementacji naszych rozwiązań. Dowiedzieliśmy się jak korzystać z programów takich jak: MS Access, MS Visio, aby stworzyć strukturę bazy wraz z interfejsem ułatwiającym korzystanie z niej.

Jesteśmy świadomi, że niektóre napotkane przez nas problemy mogłyby zostać rozwiązane w inny, lepszy, ale także możliwie trudniejszy dla nas do wykonania sposób. Dotyczy to kwestii zarówno lepszego projektu struktury bazy i relacji, jak i zwiększenia przejrzystości i łatwości korzystania z interfejsu użytkownika.

Wraz z rozwojem naszego centrum fitness pojawiłyby się możliwości i pomysły odnośnie dalszego ulepszania systemu. Mogłyby one uwzględniać dodanie nowych lub rozbudowanie już istniejących funkcjonalności. Przykładowo: utworzenie nowego, drugiego i każdego kolejnego centrum pociągałoby za sobą konieczność przebudowy bazy w taki sposób, aby była w stanie obsługiwać klientów wszystkich placówek. Nie byłaby wtedy konieczna ponowna rejestracja osoby, która korzystała już z naszych usług w innym centrum. Ulepszeniu mogłyby ulec również raporty, tak aby prezentowały większą liczbę informacji i były jeszcze bardziej praktyczne w użyciu.