

Bazy danych I

Dokumentacja projektu

Temat projektu:
Aplikacja do zarządzania firmą

Hleb Shypula

Informatyka Stosowana

Wydział Fizyki i Informatyki Stosowanej
Akademia Górniczo-Hutnicza w Krakowie

Spis treści:

1. Projekt koncepcji, założenia
 - 1.1. Zdefiniowanie tematu projektu
 - 1.2. Analiza wymagań
 - 1.3. Zaprojektowanie funkcji
2. Projekt diagramu
 - 2.1. Budowa diagramu ERD
 - 2.2. Zdefiniowanie encji
 - 2.3. Zaprojektowanie relacji pomiędzy encjami
3. Projekt logiczny
 - 3.1. Projektowanie tabel, kluczy, indeksów
 - 3.2. Walidacja danych
 - 3.3. Operacje na danych
4. Projekt funkcjonalny
5. Dokumentacja

1. Projekt koncepcji, założenia

1.1. Zdefiniowanie tematu projektu

Tematem projektu z przedmiotu Bazy Danych I jest stworzenie aplikacji do zarządzania firmą głównie specjalizującą się na aptekach.

1.2. Analiza wymagań

Baza danych powinna posiadać takie funkcjonalności jak:

- ✦ Dodawanie nowych pracowników
- ✦ Usuwanie pracowników
- ✦ Wylistowanie danych każdego pracownika
- ✦ Dodawanie kursu do pracownika
- ✦ Dodawanie nowych obiektów detalicznych
- ✦ Dodawanie pracowników do obiektów
- ✦ Wylistowanie pracowników dla obiektów
- ✦ Wylistowanie pracowników według wynagrodzenia
- ✦ Dodawanie nowych inspekcji do obiektów
- ✦ Dodawanie nowych sprawdzianów do inspekcji
- ✦ Dodawanie nowych świątecznych dni roboczych do obiektów

1.3. Zaprojektowanie funkcji

Podstawowymi funkcjami tego serwisu jest dodawanie oraz usuwanie rekordów z odpowiednich tabel tak. Baza danych powinna także zapobiegać wprowadzaniu nieprawidłowych danych np. dodawanie pracownika o istniejących danych (imię, nazwisko, username) i td. Także powinno istnieć okienko logowania do serwisu, natomiast bazodanowa funkcja powinna sprawdzać poprawność danych przy logowaniu, a aplikacja decydować o dostępnych opcjach na podstawie uprawnień użytkownika.

2. Projekt diagramu

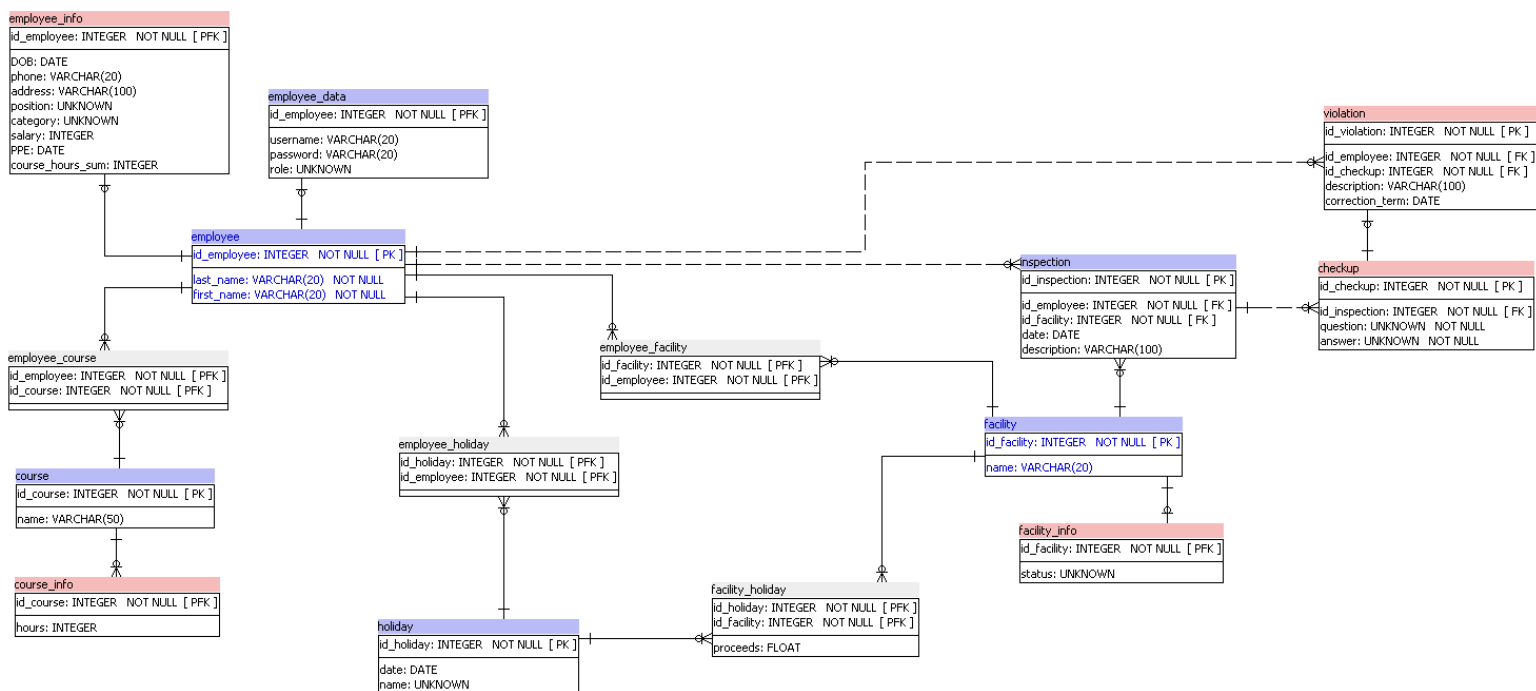
2.1. Budowa diagramu ERD

Występują zarówno relacje 1:N, 1:1, jak i N:M. Tabele powiązane relacjami N:M mają między sobą tabele asocjacyjne.

2.2. Zdefiniowanie encji

Jak na diagramie powyżej.

2.3. Zaprojektowanie relacji pomiędzy encjami



3. Projekt logiczny

3.1. Projektowanie tabel, kluczy, indeksów

Employee		
Kolumna	Typ danych	Opis
Id_employee	Integer	Klucz główny
First_name	Varchar(20)	Imię pracownika
Last_name	Varchar(20)	Nazwisko pracownika

Employee_info		
Kolumna	Typ danych	Opis
Id_employee	Integer	Klucz główny (Obcy)
DOB	Date	Data urodzenia pracownika
Phone	Varchar(20)	Telefon pracownika
Address	Varchar(100)	Adres pracownika
Position	Employee_position(enum)	Stanowisko pracownika
Category	Employee_category(enum)	Kategoria pracownika
Salary	Integer	Wynagrodzenie pracownika
PPE	Date	Data wydania środków ochrony osobowej pracownikowi
Course_hours_sum	Integer	Suma godzin na kursach dla pracownika

Employee_data		
Kolumna	Typ danych	Opis
Id_employee	Integer	Klucz główny (Obcy)
Username	Varchar(20)	Nazwa użytkownika dla pracownika
Password	Varchar(20)	Hasło dla użytkownika
Role	Employee_role(enum)	Rola użytkownika (dla wydawania uprawnień)

Course		
Kolumna	Typ danych	Opis
Id_course	Integer	Klucz główny
Name	Varchar(50)	Nazwa kursu

Course_info		
Kolumna	Typ danych	Opis
Id_course	Integer	Klucz główny (Obcy)
Hours	Integer	Godziny na kursie

Employee_course		
Kolumna	Typ danych	Opis
Id_employee	Integer	Klucz główny (Obcy)
Id_course	Integer	Klucz główny (Obcy)

Facility		
Kolumna	Typ danych	Opis
Id_facility	Integer	Klucz główny
Name	Varchar(20)	Nazwa obiektu

Facility_info		
Kolumna	Typ danych	Opis
Id_facility	Integer	Klucz główny (Obcy)
Status	Facility_status(enum)	Status obiektu

Employee_facility		
Kolumna	Typ danych	Opis
Id_employee	Integer	Klucz główny (Obcy)
Id_facility	Integer	Klucz główny (Obcy)

Holiday		
Kolumna	Typ danych	Opis
Id_holiday	Integer	Klucz główny
Date	Date	Data święta
Name	Holiday_name(enum)	Nazwa święta

Employee_holiday		
Kolumna	Typ danych	Opis
Id_employee	Integer	Klucz główny (Obcy)
Id_holiday	Integer	Klucz główny (Obcy)

Facility_holiday		
Kolumna	Typ danych	Opis
Id_holiday	Integer	Klucz główny (Obcy)
Id_facility	Integer	Klucz główny (Obcy)
Proceeds	Float	Dochód na obiekcie w to święto

Inspection		
Kolumna	Typ danych	Opis
Id_inspection	Integer	Klucz główny
Id_employee	Integer	Klucz obcy
Id_facility	Integer	Klucz obcy
Date	Date	Data przeprowadzenia inspekcji
Description	Varchar(100)	Opis inspekcji

Checkup		
Kolumna	Typ danych	Opis
Id_checkup	Integer	Klucz główny
Id_inspection	Integer	Klucz obcy
Question	Checkup_question(enum)	Pytanie dla inspekcji
Answer	Checkup_answer(enum)	Odpowiedź na pytanie

violation		
Kolumna	Typ danych	Opis
Id_violation	Integer	Klucz główny
Id_employee	Integer	Klucz obcy
Id_checkup	Integer	Klucz obcy
Description	Varchar(100)	Opis poruszenia
Correction_term	Date	Termin naprawy

Własne typy enum:

Employee_position
Dyrektor
Menedżer
Pracownik

Employee_position
Dyrektor
Menedżer
Pracownik

Employee_role
CEO
Manager
Employee

Facility_status
Apteka
Biuro
Mgazyn

Holiday_name
Wielkanoc
Boże Narodzenie

Checkup_question
Czy wilgotność w pomieszczeniu jest zgodna z zasadami?
Czy wszystkie lodówki są sprawne?
Czy wszyscy pracownicy są wyposażeni w ŚOO?

Checkup_answer
Tak
Nie

Widoki (znajdują się w pliku views.sql):

Employee_data_view
Id_employee
Role
First_name
Last_name
Dob
Category
Position
Ppe
Salary
Phone
Address
Course_hours_sum

Facility_data_view
Id_facility
Name
status

Employee_enums_view
Typname
enumlabel

Fill_facility_view
CONCAT(status, ' ', name)

Employee_resposibility
Id_employee
Violation.description
Violation.correction_term
Facility.name

Inspection_data_view
Id_inspection
Id_facility
Employee.first_name
Employee.last_name
Date
description

3.2. Walidacja danych

Aplikacja jest tak stworzona (mam nadzieję), że nie da się wprowadzić niepoprawne dane. Pokazałem umiejętność napisania wyzwalacza na przykładzie imienia i nazwiska pracownika. Walidacja ta jest przeprowadzana przez ten wyzwalacz przed każdorazowym dodaniem nowego rekordu to tabeli employee.

3.3. Operacje na danych

Do modyfikowania danych utworzone zostały takie funkcje i wyzwalacze jak:

- ✦ Wyzwalacz **validate_f_l_name** — wywołuje funkcję validate_first_name_last_name() sprawdza wartości imienia i nazwiska przy próbie wprowadzenia danych do tabeli employee, wyrzuca wyjątek w razie niepoprawnych wartości (zawierają znaki puste, znaki numeryczne i td.).
- ✦ Funkcja **addFacility**(nameAdd varchar, statusAdd facility_status) — sprawdza czy obiekt o takim imieniu i statusie istnieje. Jeżeli nie, to dodaje obiekt. Jeżeli tak, to wyrzuca wyjątek.
- ✦ Funkcja **addEmployee**(firstNameAdd varchar, lastNameAdd varchar, usernameAdd varchar, passwordAdd varchar, roleAdd employee_role, dobAdd date, phoneAdd varchar, addressAdd varchar, positionAdd employee_position, categoryAdd employee_category, salaryAdd integer, ppeAdd date, courseHourseSumAdd integer) — sprawdza czy nazwa użytkownika jest zajęta, jeżeli tak, to wyrzuca wyjątek, jeżeli nie, to wprowadza dane do tabel employee, employee_data, employee_info.
- ✦ Funkcja **addEmployeeToFacility**(employeeId integer, facilityStatus facility_status, facilityName varchar) — sprawdza czy pracownik istnieje na tym obiekcie. Jeżeli nie, to dodaje pracownika na ten obiekt. Jeżeli tak, to wyrzuca wyjątek.
- ✦ Funkcja **loginAndFillInfo**(employeeUsername varchar, employeePassword varchar) — próbuje znaleźć pracownika o takich nazwie użytkownika i hasle. Jeżeli udało się, to wypełnia tabelę danymi tego użytkownika korzystając z widoka i zwraca do aplikacji. Jeżeli użytkownik podał błędne dane, to wyrzuca wyjątek.
- ✦ Funkcja **addCheckup**(inspectionIdAdd integer, questionAdd checkup_question, answerAdd checkup_answer, dateAdd date, employeeIdAdd integer, descriptionAdd varchar) — sprawdza odpowiedź na pytanie inspekcyjne. Jeżeli „Nie”, to funkcja tworzy record w tabeliach checkup i violation. W innym przypadku tworzy tylko record w tabeli checkup.
- ✦ Funkcja **addHolidayForFacility**(facilityIdAdd integer, employeeIdAdd integer, holidayNameAdd holiday_name, holidayDateAdd date, holidayProceedsAdd float) — tworzy rekordy w tabelach holiday, employee_holiday, facility_holiday.
- ✦ Funkcja **deleteEmployee**(employeeId integer) — sprawdza, czy taki użytkownik istnieje. Jeżeli nie, to rzuca wyjątek. Jeżeli tak, to usuwa użytkownika i wszystkie związane z nim dane.

- ✦ Funkcja **addCourse**(employeeIdAdd integer, courseNameAdd varchar, courseHoursAdd integer) — tworzy nowy kurs, dodaje go do użytkownika, edytuje wartość course_hours_sum w tabeli employee w zależności od podanych godzin w nowym kursie.

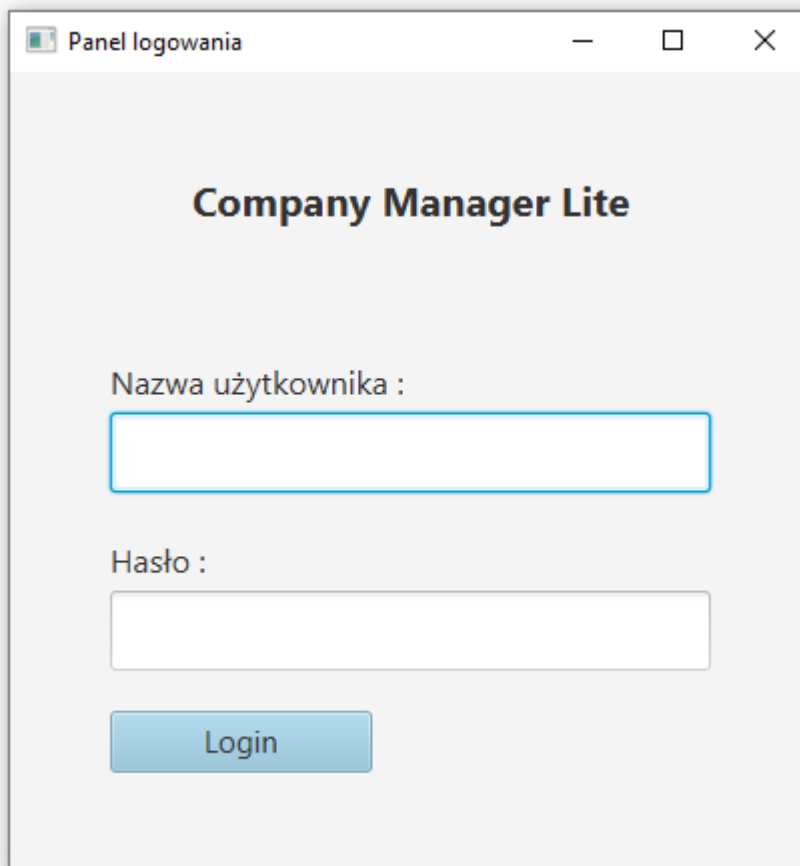
Kod wszystkich funkcji i wyzwalaczy został umieszczony w plikach functions.sql oraz triggers.sql odpowiednio.

4. Projekt funkcjonalny

Testowe dane dla administratora: nazwa użytkownika – **admin**, hasło – **admin**

Testowe dane dla pracownika: nazwa użytkownika – **adam**, hasło – **babacki**

Okno logowania:



Panel logowania

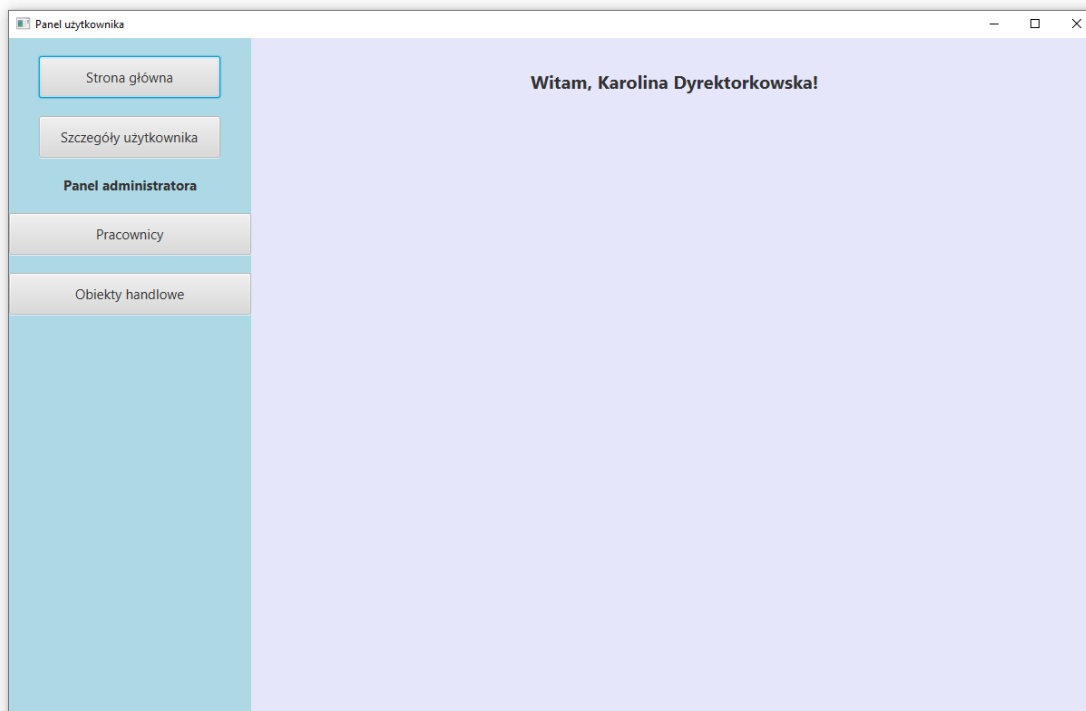
Company Manager Lite

Nazwa użytkownika :

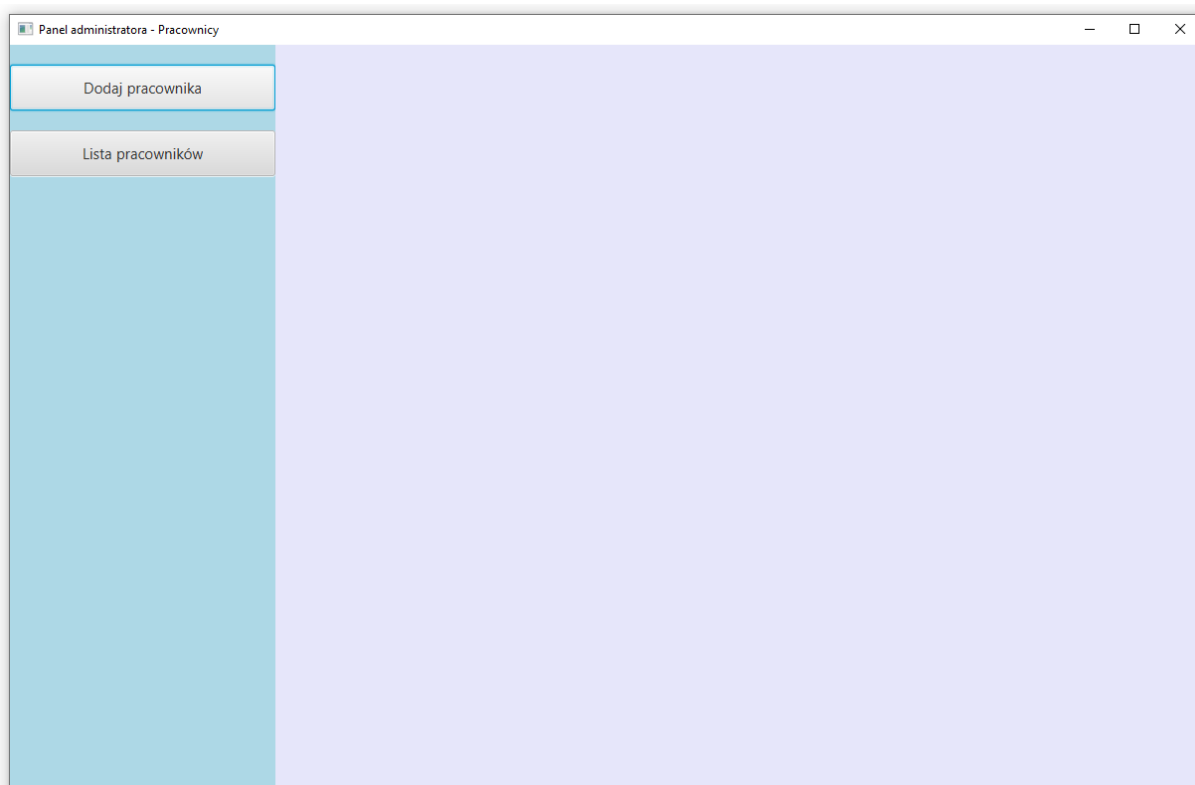
Hasło :

Login

Główne okno – dla administratora jest dostępny panel administratora, natomiast dla użytkownika z wartością „employee” w tabeli employee_data to nie jest widoczne:



Okno zarządzania pracownikami – można dodać nowego pracownika oraz wylistować wszystkich:



Okno dodawania pracowników – jeżeli wszystkie dane są wpisane poprawnie, pracownik zostanie dodany. W przypadku próby wysłania błędnych danych zostanie wyrzucony alert z prośbą o uzupełnienie:

Panel administratora - Pracownicy

Wprowadź dane:

Hleb
Shypula
CEO
hleb
shypula
07.10.1997
+48123456789
Białoruś, Borysów, ul. Podlesnaya 48a.
Dyrektor
wyższa
5499
10.01.2021
50

Dodaj

Lista pracowników – możliwe jest filtrowanie według wynagrodzenia oraz wylistowanie dla konkretnego obiektu. Po wybraniu pracownika z listy będą dostępne guziki do pokazania danych pracownika oraz dodania do obiektu:

Panel administratora - Pracownicy

	Imię	Nazwisko	Data urodzenia	Stanowisko	Kategoria	Wynagrodzenie	Otrzymano 500
Dodaj pracownika	Karolina	Dyrektorkowska	1990-05-05	Dyrektor	wyższa	9800	
	Hleb	Shypula	1997-10-07	Dyrektor	wyższa	5499	1997-10-07
	Adam	Babacki	2001-03-15	Pracownik	2	3300	2001-03-15
Lista pracowników	Galina	Ivanova	1973-05-10	Menedżer	1	4560	1973-05-10
	Tamara	Zayceva	1965-05-12	Menedżer	wyższa	5332	1965-05-12
	Filip	Rudzik	1984-08-09	Pracownik	2	7800	1984-08-09

min. wynagrodzenie: 3300 max. wynagrodzenie: 9800 Filtruj według wynagrodzenia

Dodaj pracownika do obiektu Wylistuj pracowników dla obiektu Pokaż dane pracownika

Wybierz obiekt Wylistuj

Panel administratora - Pracownicy

Dodaj pracownika

Lista pracowników

Imię	Nazwisko	Data urodzenia	Stanowisko	Kategoria	Wynagrodzenie	Otrzymano ŚOO
Karolina	Dyrektorkowska	1990-05-05	Dyrektor	wyższa	9800	
Hleb	Shypula	1997-10-07	Dyrektor	wyższa	5499	1997-10-07
Adam	Babacki	2001-03-15	Pracownik	2	3300	2001-03-15
Galina	Ivanova	1973-05-10	Menedżer	1	4560	1973-05-10
Tamara	Zayceva	1965-05-12	Menedżer	wyższa	5332	1965-05-12
Filip	Rudzik	1984-08-09	Pracownik	2	7800	1984-08-09

min. wynagrodzenie: 3300

max. wynagrodzenie: 9800

Filtruj według wynagrodzenia

Dodaj pracownika do obiektu

Wylistuj pracowników dla obiektu

Pokaż dane pracownika

Wybierz obiekt

Wybierz obiekt

Dodaj

Wylistuj

Guzik „Pokaż dane pracownika” z poprzedniego obrazka – można dodać kurs dla pracownika (godziny w polu „Suma godzin na kursach” zmienia się odpowiednio) oraz usunąć pracownika:

Panel administratora - Obiekty Handlowe

Dodaj obiekt

Lista obiektów

Dane użytkownika: Adam Babacki

Imię: Adam

Nazwisko: Babacki

Data urodzenia: 2001-03-15

Telefon: +48999888123

Adres: Polska, Warszawa

Stanowisko: Pracownik

Kategoria: 2

Wynagrodzenie: 3300

Data otrzymania ŚOO: 2001-03-15

Suma godzin na kursach 140

Odpowiedzialność:

Iodówka nie działa (na obiekcie: Apteka 2, termin do 2021-01-29)

Usuń pracownika

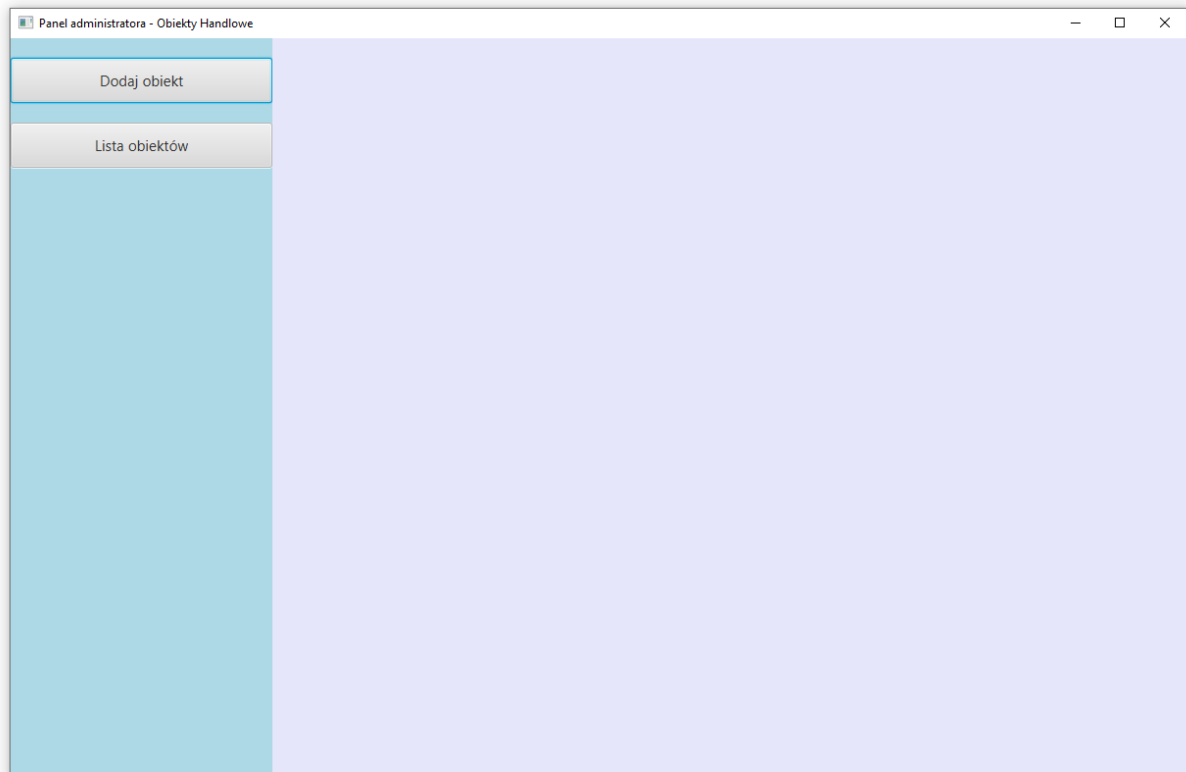
Dodaj kurs dla pracownika

Kurs angielskiego

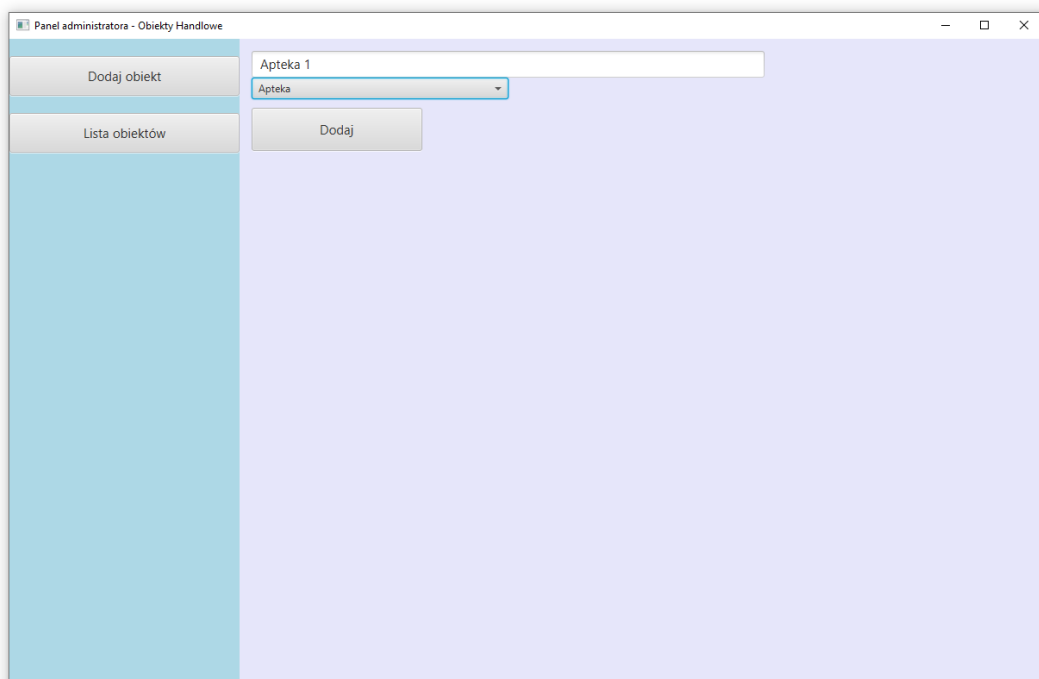
50

Dodaj

Menu zarządzania obiektami – można dodać nowy obiekt oraz wylistować wszystkie:



Dodawanie nowego obiektu:



Lista obiektów – po wybraniu obiektu z tabeli zostaną uruchomione guziki do pokazu inspekcji oraz świąt, guziki do dodawania inspekcji oraz świąt (w polach na wpisanie użytkownika działa automatyczne uzupełnianie tekstu, proszę zacząć wpisywać), guzik do wylistowania pracowników tego obiektu (po wybraniu pracownika można przejść do jego danych):

Nazwa	Status
Apteka 1	Apteka
Apteka 2	Apteka
Apteka 3	Apteka
Biuro	Biuro
Biuro ds. klientów	Biuro
Magazyn produkcji	Magazyn

Wylistuj inspekcje Wylistuj święta Wylistuj pracowników Pokaz dane pracownika

Imię	Nazwisko	Stanowisko
No content in table		

Dodaj inspekcję Dodaj święto

Wylistuj inspekcje Wylistuj święta Wylistuj pracowników Pokaz dane pracownika

Osoba odpowiedzialna: Hleb Shypula; Data: 2021-01-13; Opis: Inspekcja 1;

Imię	Nazwisko	Stanowisko
Adam	Babacki	Pracownik
Hleb	Shypula	Dyrektor

Dodaj inspekcję Dodaj święto

Babacki Adam (Pracownik)

13.01.2021

Inspekcja 1

Pytania:

Czy wilgotność w pomieszczeniu jest zgodna z zasadami? Tak

Czy wszystkie lodówki są sprawne? Nie

Czy wszyscy pracownicy są wyposażeni w ŚOO? Tak

Dodaj

Wielkanoc

04.04.2021

3570

Babacki Adam (Pracownik)

Dodaj

lódówka nie działa Dyrektorkowska Karolina (Dyrekt)

29.01.2021

5. Dokumentacja

Folder **Hleb_Shypula_Projekt_BD1** z archiwum należy wyciągnąć w dowolne miejsce na komputerze. Dalej w tym folderze należy uruchomić plik Application.jar

Uwaga – Dla uruchomienia pliku Application.jar jest potrzebne JRE wersji 8 i wyżej.

W plikach views.sql, functions.sql, triggers.sql znajdują się odpowiednie kody źródłowe.

Aplikacja jest napisana w Java za pomocą JavaFX.

Wszystkie połączenia oraz operacje z bazą danych są napisane w klasie z pliku Database.java

Baza danych jest umieszczona na darmowym hostingu ElephantSQL.