

**Politechnika Łódzka**

Wydział Elektrotechniki, Elektroniki, Informatyki i Automatyki

**Szkoła Muzyczna**

Obiektowa Baza Danych Oracle

Rozproszone i Obiektowe Bazy Danych

**Autorzy:** Igor Typinski (251237)  
Mateusz Mroz (251190)

**Grupa:** 5

**Temat:** Szkoła muzyczna (z naciskiem na rozwój ucznia)

Łódź, styczeń 2026

# Spis treści

<b>1 Opis projektu</b>	<b>3</b>
1.1 Cel i zakres . . . . .	3
1.2 Przyjęte ograniczenia . . . . .	3
1.3 Technologia . . . . .	4
<b>2 Typy obiektowe</b>	<b>4</b>
2.1 t_instrument_obj . . . . .	5
2.2 t_lista_instrumentow . . . . .	5
2.3 t_sala_obj . . . . .	5
2.4 t_nauczyciel_obj . . . . .	5
2.5 t_uczen_obj . . . . .	5
2.6 t_kurs_obj . . . . .	6
2.7 t_lekcja_obj . . . . .	6
2.8 t_ocena_obj . . . . .	6
<b>3 Tabele obiektowe</b>	<b>7</b>
3.1 t_instrument . . . . .	7
3.2 t_sala . . . . .	7
3.3 t_nauczyciel . . . . .	7
3.4 t_uczen . . . . .	8
3.5 t_kurs . . . . .	8
3.6 t_lekcja . . . . .	8
3.7 t_ocena_postepu . . . . .	8
3.8 Referencje (REF/DEREF) . . . . .	8
3.9 Sekwencje . . . . .	9
<b>4 Pakiety PL/SQL</b>	<b>9</b>
4.1 pkg_uczen . . . . .	9
4.2 pkg_lekcja . . . . .	9
4.3 pkg_ocena . . . . .	10
<b>5 Wyzwalacze (Triggery)</b>	<b>10</b>
<b>6 Obsługa błędów</b>	<b>10</b>
6.1 Kody błędów aplikacji . . . . .	10
6.2 Scenariusze testowe . . . . .	11
6.2.1 Scenariusz 1: Dane podstawowe . . . . .	11
6.2.2 Scenariusz 2: Walidacja wieku ucznia . . . . .	11
6.2.3 Scenariusz 3: Dni robocze . . . . .	11
6.2.4 Scenariusz 4: Godziny dla dzieci . . . . .	11
6.2.5 Scenariusz 5: Limit nauczyciela . . . . .	12
6.2.6 Scenariusz 6: Limit ucznia . . . . .	12
6.2.7 Scenariusz 7: Konflikty czasowe . . . . .	12
6.2.8 Scenariusz 8: Blokada usuwania . . . . .	12
6.2.9 Scenariusz 9: Pakiety CRUD . . . . .	12
6.2.10 Scenariusz 10: Metody obiektów . . . . .	12

<b>7</b>	<b>Role użytkowników</b>	<b>12</b>
7.1	Administrator (rola_admin) . . . . .	13
7.2	Nauczyciel (rola_nauczyciel) . . . . .	13
7.3	Sekretariat (rola_sekretariat) . . . . .	13
7.4	Macierz uprawnień . . . . .	13
7.5	Użytkownicy testowi . . . . .	13
<b>8</b>	<b>Diagram relacji obiektów</b>	<b>14</b>
<b>9</b>	<b>Struktura plików projektu</b>	<b>14</b>

# 1 Opis projektu

## 1.1 Cel i zakres

Projekt przedstawia obiektową bazę danych dla szkoły muzycznej, ze szczególnym uwzględnieniem śledzenia rozwoju uczniów. Szkoła zajmuje się wyłącznie nauką muzyki w trybie indywidualnym. System został zaprojektowany z myślą o codziennej pracy sekretariatu, nauczycieli oraz dyrekcji szkoły.

Głównym celem projektu jest stworzenie kompleksowego systemu informatycznego umożliwiającego:

- **Zarządzanie danymi uczniów** – rejestracja nowych uczniów, przechowywanie danych osobowych, automatyczne obliczanie wieku i statusu (dziecko/dorosły)
- **Zarządzanie danymi nauczycieli** – ewidencja kadry pedagogicznej wraz z listą instrumentów, których mogą uczyć (przechowywana jako kolekcja VARRAY)
- **Planowanie lekcji** – tworzenie harmonogramu zajęć z uwzględnieniem dostępności sal, nauczycieli oraz ograniczeń czasowych dla dzieci
- **Rezerwacja sal lekcyjnych** – przydzielanie sal z odpowiednim wyposażeniem (fortepian, perkusja) do poszczególnych lekcji
- **Ocenianie postępu uczniów** – wystawianie ocen w różnych obszarach muzycznych (technika, teoria, słuch, rytm, interpretacja)
- **Kontrola reguł biznesowych** – automatyczna walidacja limitów obciążenia nauczycieli, liczby lekcji uczniów, konfliktów czasowych
- **Generowanie raportów** – plan dnia, obciążenie nauczycieli, historia ocen, raporty postępu uczniów

System uwzględnia specyfikę szkoły muzycznej, w której większość uczniów to dzieci uczęszczające równolegle do szkoły ogólnokształcącej. Dlatego lekcje dla dzieci (poniżej 15 lat) mogą odbywać się wyłącznie w godzinach popołudniowych (14:00–19:00), po zakończeniu zajęć szkolnych.

## 1.2 Przyjęte ograniczenia

W projekcie przyjęto następujące ograniczenia biznesowe:

1. **Minimalny wiek ucznia:** 5 lat – młodsze dzieci nie są przyjmowane do szkoły
2. **Definicja dziecka:** uczeń poniżej 15 lat – podlega dodatkowym ograniczeniom godzinowym
3. **Godziny lekcji dla dzieci:** tylko 14:00–19:00 – po zakończeniu zajęć w szkole ogólnokształcącej
4. **Dni pracy szkoły:** poniedziałek–piątek – szkoła nie prowadzi zajęć w weekendy
5. **Godziny pracy szkoły:** 08:00–20:00 – lekcje mogą być planowane tylko w tych godzinach

6. **Maksymalna liczba instrumentów nauczyciela:** 5 – ograniczenie kolekcji VARRAY
7. **Maksymalne obciążenie nauczyciela:** 6 godzin (360 minut) dziennie
8. **Maksymalna liczba lekcji ucznia:** 2 lekcje dziennie
9. **Czas trwania lekcji:** 30, 45, 60 lub 90 minut
10. **Skala ocen:** 1–6 (polska skala szkolna)
11. **Obszary oceny:** technika, teoria, słuch, rytm, interpretacja, ogólna
12. **Poziomy kursów:** początkujący, średni, zaawansowany
13. **Statusy lekcji:** zaplanowana, odbyta, odwołana
14. **Kategorie instrumentów:** dęte, strunowe, perkusyjne, klawiszowe
15. **Brak konfliktów czasowych:** ta sama sala, nauczyciel lub uczeń nie mogą mieć dwóch lekcji w tym samym czasie
16. **Ochrona danych:** nie można usunąć nauczyciela lub ucznia z zaplanowanymi lekcjami

### 1.3 Technologia

- **Oracle Database** – baza danych obiektowo-relacyjna
- **Podejście obiektowo-relacyjne** – typy obiektowe z metodami, referencje, VARRAY
- **Język PL/SQL** – pakiety, procedury, triggery, walidacja biznesowa

## 2 Typy obiektowe

W projekcie zdefiniowano 8 typów obiektowych z łącznie 13 metodami.

<b>Typ</b>	<b>Metody</b>	<b>Opis</b>
t_instrument_obj	1	Instrument muzyczny
t_lista_instrumentow	–	VARRAY(5) nazw instrumentow
t_sala_obj	1	Sala lekcyjna z wyposażeniem
t_nauczyciel_obj	3	Nauczyciel z lista instrumentow
t_uczen_obj	3	Uczeń szkoły muzycznej
t_kurs_obj	2	Kurs nauki gry na instrumencie
t_lekcja_obj	2	Pojedyncza lekcja muzyki
t_ocena_obj	1	Ocena postępu ucznia
<b>Razem:</b>		<b>8 typów, 13 metod</b>

Tabela 1: Typy obiektowe w projekcie

## **2.1 t\_instrument\_obj**

Reprezentuje instrument muzyczny w słowniku instrumentów.

**Atrybuty:**

- id\_instrumentu (NUMBER) – unikalny identyfikator
- nazwa (VARCHAR2(50)) – nazwa instrumentu (np. Fortepian, Gitara)
- kategoria (VARCHAR2(20)) – kategoria: dęte, strunowe, perkusyjne, klawiszowe

**Metody:**

- info() RETURN VARCHAR2 – zwraca nazwę z kategorią w nawiasie

## **2.2 t\_lista\_instrumentow**

Kolekcja VARRAY przechowująca nazwy instrumentów, których może uczyć nauczyciel.

```
CREATE OR REPLACE TYPE t_lista_instrumentow AS VARRAY(5) OF VARCHAR2(50);
```

Ograniczenie do 5 elementów wynika z założenia, że nauczyciel specjalizuje się w kilku pokrewnych instrumentach. Weryfikacja kompetencji odbywa się przy zatrudnieniu przez dyrektora szkoły.

## **2.3 t\_sala\_obj**

Reprezentuje salę lekcyjną z informacją o wyposażeniu.

**Atrybuty:**

- id\_sali (NUMBER) – unikalny identyfikator
- nazwa (VARCHAR2(50)) – nazwa sali (unikalna, np. Sala A1)
- pojemność (NUMBER) – maksymalna liczba osób (1–20)
- ma\_fortepian (CHAR(1)) – 'T' lub 'N'
- ma\_perkusje (CHAR(1)) – 'T' lub 'N'

**Metody:**

- opis() RETURN VARCHAR2 – zwraca nazwę z informacją o wyposażeniu

## **2.4 t\_nauczyciel\_obj**

Reprezentuje nauczyciela szkoły muzycznej wraz z listą instrumentów.

**Atrybuty:**

- id\_nauczyciela (NUMBER) – unikalny identyfikator
- imię (VARCHAR2(50)) – imię nauczyciela
- nazwisko (VARCHAR2(50)) – nazwisko nauczyciela
- email (VARCHAR2(100)) – adres email (unikalny)
- data\_zatrudnienia (DATE) – data rozpoczęcia pracy
- instrumenty (t\_lista\_instrumentow) – VARRAY instrumentów

**Metody:**

- staz() RETURN NUMBER – oblicza liczbę lat pracy w szkole
- liczba\_instrumentow() RETURN NUMBER – zwraca liczbę elementów w VARRAY
- info() RETURN VARCHAR2 – zwraca imię, nazwisko i staż

## **2.5 t\_uczen\_obj**

Reprezentuje ucznia szkoły muzycznej.

**Atrybuty:**

- id\_ucznia (NUMBER) – unikalny identyfikator
- imię (VARCHAR2(50)) – imię ucznia
- nazwisko (VARCHAR2(50)) – nazwisko ucznia
- data\_urodzenia (DATE) – data urodzenia
- email (VARCHAR2(100)) – adres email (opcjonalny)
- data\_zapisu (DATE) – data rejestracji w szkole

**Metody:**

- wiek() RETURN NUMBER – oblicza aktualny wiek w latach
- czy\_dziecko() RETURN CHAR – zwraca 'T' jeśli wiek < 15 lat, inaczej 'N'
- info() RETURN VARCHAR2 – zwraca imię, nazwisko i wiek

## 2.6 t\_kurs\_obj

Reprezentuje kurs nauki gry na instrumencie.

**Atrybuty:**

- id\_kursu (NUMBER) – unikalny identyfikator
- nazwa (VARCHAR2(100)) – nazwa kursu
- poziom (VARCHAR2(20)) – początkujący, średni, zaawansowany
- cena\_za\_lekcje (NUMBER) – cena jednej lekcji w PLN
- ref\_instrument (REF t\_instrument\_obj) – referencja do instrumentu

**Metody:**

- cena\_brutto() RETURN NUMBER – oblicza cenę z VAT (23%)
- info() RETURN VARCHAR2 – zwraca nazwę z poziomem i ceną

## 2.7 t\_lekcja\_obj

Reprezentuje pojedynczą lekcję muzyki.

**Atrybuty:**

- id\_lekcji (NUMBER) – unikalny identyfikator
- data\_lekcji (DATE) – data lekcji
- godzina\_start (VARCHAR2(5)) – godzina rozpoczęcia (HH:MM)
- czas\_trwania (NUMBER) – czas w minutach (30, 45, 60, 90)
- status (VARCHAR2(20)) – zaplanowana, odbyta, odwołana
- ref\_uczen (REF t\_uczen\_obj) – referencja do ucznia
- ref\_nauczyciel (REF t\_nauczyciel\_obj) – referencja do nauczyciela
- ref\_kurs (REF t\_kurs\_obj) – referencja do kursu
- ref\_sala (REF t\_sala\_obj) – referencja do sali

**Metody:**

- godzina\_konca() RETURN VARCHAR2 – oblicza godzinę zakończenia
- info() RETURN VARCHAR2 – zwraca datę, godzinę i status

## 2.8 t\_ocena\_obj

Reprezentuje ocenę postępu ucznia.

**Atrybuty:**

- id\_oceny (NUMBER) – unikalny identyfikator
- data\_oceny (DATE) – data wystawienia oceny
- ocena (NUMBER) – wartość 1–6

- obszar (VARCHAR2(50)) – obszar oceny
- komentarz (VARCHAR2(500)) – komentarz nauczyciela (opcjonalny)
- ref\_uczen (REF t\_uczen\_obj) – referencja do ucznia
- ref\_nauczyciel (REF t\_nauczyciel\_obj) – referencja do nauczyciela

#### Metody:

- info() RETURN VARCHAR2 – zwraca datę, ocenę i obszar

## 3 Tabele obiektowe

Utworzono 7 tabel obiektowych przechowujących dane.

Tabela	Typ bazowy	Referencje (REF)
t_instrument	t_instrument_obj	–
t_sala	t_sala_obj	–
t_nauczyciel	t_nauczyciel_obj	– (zawiera VARRAY)
t_uczen	t_uczen_obj	–
t_kurs	t_kurs_obj	ref_instrument
t_lekcja	t_lekcja_obj	ref_uczen, ref_nauczyciel, ref_kurs, ref_sala
t_ocena_postepu	t_ocena_obj	ref_uczen, ref_nauczyciel

Tabela 2: Tabele obiektowe i ich referencje

### 3.1 t\_instrument

Słownik instrumentów muzycznych.

- Klucz główny: id\_instrumentu
- Ograniczenia NOT NULL: nazwa, kategoria
- CHECK: kategoria IN ('dęte', 'strunowe', 'perkusyjne', 'klawiszowe')

### 3.2 t\_sala

Informacje o salach lekcyjnych z wyposażeniem.

- Klucz główny: id\_sali
- Ograniczenie UNIQUE: nazwa
- CHECK: pojemność BETWEEN 1 AND 20
- CHECK: ma\_fortepian IN ('T', 'N')
- CHECK: ma\_perkusje IN ('T', 'N')

### 3.3 t\_nauczyciel

Dane nauczycieli wraz z kolekcją instrumentów (VARRAY).

- Klucz główny: id\_nauczyciela
- Ograniczenie UNIQUE: email
- Ograniczenia NOT NULL: imię, nazwisko, email, data\_zatrudnienia
- CHECK: email LIKE '%@%'
- Zawiera kolekcję t\_lista\_instrumentow

### **3.4 t\_uczen**

Dane uczniów szkoły muzycznej.

- Klucz główny: id\_ucznia
- Ograniczenie UNIQUE: email
- Ograniczenia NOT NULL: imię, nazwisko, data\_urodzenia, data\_zapisu
- CHECK: email IS NULL OR email LIKE '%@%'
- Indeks: idx\_uczen\_nazwisko (nazwisko)

### **3.5 t\_kurs**

Oferta kursów nauki gry na instrumentach.

- Klucz główny: id\_kursu
- Ograniczenia NOT NULL: nazwa, poziom, cena\_za\_lekcje
- CHECK: poziom IN ('początkujący', 'średni', 'zaawansowany')
- CHECK: cena\_za\_lekcje > 0
- Referencja: ref\_instrument SCOPE IS t\_instrument

### **3.6 t\_lekcja**

Zaplanowane i odbyte lekcje muzyki.

- Klucz główny: id\_lekcji
- Ograniczenia NOT NULL: data\_lekcji, godzina\_start, czas\_trwania, status
- CHECK: czas\_trwania IN (30, 45, 60, 90)
- CHECK: status IN ('zaplanowana', 'odbyta', 'odwołana')
- CHECK: godzina\_start >= '08:00' AND godzina\_start <= '20:00'
- Referencje: ref\_uczen, ref\_nauczyciel, ref\_kurs, ref\_sala (wszystkie ze SCOPE IS)
- Indeksy: idx\_lekcja\_data, idx\_lekcja\_status

### **3.7 t\_ocena\_postepu**

Oceny postępu uczniów w różnych obszarach.

- Klucz główny: id\_oceny
- Ograniczenia NOT NULL: data\_oceny, ocena, obszar
- CHECK: ocena BETWEEN 1 AND 6
- CHECK: obszar IN ('technika', 'teoria', 'słuch', 'rytm', 'interpretacja', 'ogólna')
- Referencje: ref\_uczen, ref\_nauczyciel (ze SCOPE IS)
- Indeks: idx\_ocena\_data

## **3.8 Referencje (REF/DEREF)**

W projekcie zastosowano 7 referencji do modelowania relacji między obiektami:

- Kurs wskazuje na instrument, którego dotyczy
- Lekcja wskazuje na ucznia, nauczyciela, kurs i salę
- Ocena wskazuje na ucznia i nauczyciela wystawiającego

Dzięki DEREF możliwe jest odwołanie się do atrybutów i metod obiektu wskazywanego:

```
SELECT DEREF(l.ref_uczen).imię || ' ' || DEREF(l.ref_uczen).nazwisko AS uczeń,
```

```

        DEREF(l.ref_sala).nazwa AS sala,
        DEREF(l.ref_kurs).nazwa AS kurs
FROM t_lekcja l WHERE l.status = 'zaproponowana';

```

### 3.9 Sekwencje

Utworzono 7 sekwencji do generowania identyfikatorów:

- seq\_instrument – dla t\_instrument
- seq\_sala – dla t\_sala
- seq\_nauczyciel – dla t\_nauczyciel
- seq\_uczen – dla t\_uczen
- seq\_kurs – dla t\_kurs
- seq\_lekcja – dla t\_lekcja
- seq\_ocena – dla t\_ocena\_postepu

## 4 Pakiety PL/SQL

Logika biznesowa zaimplementowana w 3 pakietach z łącznie 15 podprogramami.

Pakiet	Podprogramy	Funkcjonalności
pkg_uczen	6	Zarządzanie uczniami
pkg_lekcja	6	Zarządzanie lekcjami
pkg_ocena	3	Zarządzanie ocenami
<b>Razem:</b>		<b>15 podprogramów</b>

Tabela 3: Pakiety PL/SQL

### 4.1 pkg\_uczen

Pakiet do zarządzania uczniami szkoły muzycznej.

**Procedury:**

- dodaj(imię, nazwisko, data\_urodzenia, email) – dodaje nowego ucznia z walidacją wieku (min. 5 lat)
- lista() – wyświetla listę wszystkich uczniów z wiekiem i statusem
- lista\_dzieci() – wyświetla listę uczniów poniżej 15 lat
- info(id\_ucznia) – wyświetla szczegółowe informacje o uczniu

**Funkcje:**

- srednia\_ocen(id\_ucznia) RETURN NUMBER – oblicza średnią ocen ucznia
- liczba\_lekcji(id\_ucznia) RETURN NUMBER – zwraca liczbę lekcji ucznia

### 4.2 pkg\_lekcja

Pakiet do zarządzania lekcjami i harmonogramem.

**Procedury:**

- zaplanuj(id\_ucznia, id\_nauczyciela, id\_kursu, id\_sali, data, godzina, czas) – planuje nową lekcję
- oznacz\_odbyta(id\_lekcji) – zmienia status lekcji na 'odbyta'

- odwołaj(id\_lekcji) – zmienia status lekcji na 'odwołana'
- plan\_dnia(data) – wyświetla wszystkie lekcje danego dnia
- plan\_nauczyciela(id\_nauczyciela, data) – wyświetla plan nauczyciela
- raport\_ociazienia(data) – wyświetla obciążenie nauczycieli w minutach

### 4.3 pkg\_ocena

Pakiet do zarządzania ocenami postępu uczniów.

#### Procedury:

- dodaj(id\_uchnia, id\_nauczyciela, ocena, obszar, komentarz) – dodaje nową ocenę
- historia\_uchnia(id\_uchnia) – wyświetla wszystkie oceny ucznia
- raport\_postepu(id\_uchnia) – wyświetla średnie oceny wg obszarów

## 5 Wyzwalacze (Triggers)

Zdefiniowano 10 wyzwalaczy realizujących reguły biznesowe.

Trigger	Typ	Działanie
trg_uczen_wiek	BEFORE I/U	Walidacja minimalnego wieku (5 lat)
trg_lekcja_dni_roboce	BEFORE I/U	Lekcje tylko w dni robocze (Pn-Pt)
trg_lekcja_godziny_dziecka	BEFORE I/U	Dzieci: lekcje tylko 14:00–19:00
trg_lekcja_limit_nauczyciela	BEFORE I/U	Nauczyciel max 360 min/dzień
trg_lekcja_limit_uchnia	BEFORE I/U	Uczeń max 2 lekcje/dzień
trg_lekcja_konflikt_sali	BEFORE I/U	Brak nakładania lekcji w sali
trg_lekcja_konflikt_nauczyciela	BEFORE I/U	Nauczyciel w jednym miejscu
trg_lekcja_konflikt_uchnia	BEFORE I/U	Uczeń w jednym miejscu
trg_blokada_usun_nauczyciela	BEFORE D	Ochrona nauczyciela z lekcjami
trg_blokada_usun_uchnia	BEFORE D	Ochrona ucznia z lekcjami

Tabela 4: Wyzwalacze w projekcie

## 6 Obsługa błędów

W projekcie zastosowano mechanizmy obsługi wyjątków z własnymi kodami błędów.

### 6.1 Kody błędów aplikacji

Kod	Trigger/Procedura	Znaczenie
-20001	pkg_uczen.dodaj	Uczeń musi mieć co najmniej 5 lat
-20010	pkg_lekcja.zaplanuj	Nie znaleziono ucznia, nauczyciela, kursu lub sali
-20011	pkg_lekcja.oznacz_odbyta	Lekcja nie istnieje lub nie jest zaplanowana

Kod	Trigger/Procedura	Znaczenie
-20012	pkg_lekcja.odwołaj	Lekcja nie istnieje lub nie jest zaplanowana
-20020	pkg_ocena.dodaj	Nie znaleziono ucznia lub nauczyciela
-20101	trg_uczen_wiek	Wiek ucznia poniżej 5 lat
-20102	trg_lekcja_dni_robocze	Lekcja zaplanowana w weekend
-20103	trg_lekcja_godziny_dziecka	Dziecko poza godzinami 14:00–19:00
-20104	trg_lekcja_limit_nauczyciela	Nauczyciel przekracza 6h dziennie
-20105	trg_lekcja_limit_ucznia	Uczeń ma już 2 lekcje w danym dniu
-20106	trg_lekcja_konflikt_sali	Konflikt rezerwacji sali
-20107	trg_lekcja_konflikt_nauczyciela	Nauczyciel ma inną lekcję w tym czasie
-20108	trg_lekcja_konflikt_ucznia	Uczeń ma inną lekcję w tym czasie
-20109	trg_blokada_usun_nauczyciela	Próba usunięcia nauczyciela z lekcjami
-20110	trg_blokada_usun_ucznia	Próba usunięcia ucznia z lekcjami

Tabela 5: Kody błędów aplikacji

## 6.2 Scenariusze testowe

Projekt zawiera kompleksowe testy w pliku 06\_testy.sql obejmujące 10 scenariuszy:

### 6.2.1 Scenariusz 1: Dane podstawowe

Weryfikacja poprawnego załadowania danych (instrumenty, sale, nauczyciele, uczniowie, kursy).

### 6.2.2 Scenariusz 2: Walidacja wieku ucznia

Test triggera trg\_uczen\_wiek:

- 3-latek: odrzucony (błąd -20101)
- 4-latek: odrzucony (błąd -20101)
- 5-latek: akceptowany
- 10-latek: akceptowany

### 6.2.3 Scenariusz 3: Dni robocze

Test triggera trg\_lekcja\_dni\_robocze:

- Sobota: odrzucona (błąd -20102)
- Niedziela: odrzucona (błąd -20102)
- Poniedziałek: akceptowany

### 6.2.4 Scenariusz 4: Godziny dla dzieci

Test triggera trg\_lekcja\_godziny\_dziecka:

- Dziecko o 08:00: odrzucone (błąd -20103)
- Dziecko o 13:00: odrzucone (błąd -20103)
- Dziecko o 14:00: akceptowane
- Dziecko 18:15–19:00: akceptowane
- Dziecko 18:30–19:15: odrzucone (przekracza 19:00)

- Dorosły o 08:00: akceptowany

#### **6.2.5 Scenariusz 5: Limit nauczyciela**

Test triggera trg\_lekcja\_limit\_nauczyciela:

- 6 lekcji po 60 minut (360 min): akceptowane
- 7. lekcja (30 min): odrzucona (błąd -20104)

#### **6.2.6 Scenariusz 6: Limit ucznia**

Test triggera trg\_lekcja\_limit\_ucznia:

- 1. lekcja: akceptowana
- 2. lekcja: akceptowana
- 3. lekcja: odrzucona (błąd -20105)

#### **6.2.7 Scenariusz 7: Konflikty czasowe**

Test triggerów konfliktów:

- Ta sama sala, nakładający się czas: odrzucone (błąd -20106)
- Ten sam nauczyciel, nakładający się czas: odrzucone (błąd -20107)
- Ten sam uczeń, nakładający się czas: odrzucone (błąd -20108)

#### **6.2.8 Scenariusz 8: Blokada usuwania**

Test triggerów ochrony danych:

- Usunięcie nauczyciela z lekcjami: odrzucone (błąd -20109)
- Usunięcie ucznia z lekcjami: odrzucone (błąd -20110)

#### **6.2.9 Scenariusz 9: Pakiety CRUD**

Test procedur pakietów:

- pkg\_uczen.lista() – wyświetla listę uczniów
- pkg\_uczen.lista\_dzieci() – wyświetla dzieci
- pkg\_uczen.info(1) – szczegóły ucznia
- pkg\_lekcja.plan\_dnia(data) – plan dnia
- pkg\_ocena.historia\_ucznia(1) – historia ocen

#### **6.2.10 Scenariusz 10: Metody obiektów**

Test metod typów obiektowych:

- t\_uczen\_obj.wiek() – oblicza wiek
- t\_uczen\_obj.czy\_dziecko() – zwraca 'T' lub 'N'
- t\_nauczyciel\_obj.staz() – oblicza staż
- t\_nauczyciel\_obj.liczba\_instrumentow() – liczba instrumentów
- t\_sala\_obj.opis() – opis sali z wyposażeniem

## **7 Role użytkowników**

W projekcie zdefiniowano 3 role z różnymi poziomami uprawnień.

## 7.1 Administrator (rola\_admin)

Pełny dostęp do systemu:

- SELECT, INSERT, UPDATE, DELETE na wszystkich tabelach
- Dostęp do wszystkich sekwencji
- Wykonywanie wszystkich pakietów
- Zarządzanie użytkownikami

## 7.2 Nauczyciel (rola\_nauczyciel)

Prowadzenie lekcji i ocenianie:

- SELECT na wszystkich tabelach
- UPDATE kolumny status w tabeli t\_lekcja
- INSERT do tabeli t\_ocena\_postepu
- Wykonywanie pakietów: pkg\_uczen, pkg\_lekcja, pkg\_ocena

## 7.3 Sekretariat (rola\_sekretariat)

Zarządzanie harmonogramem i uczniami:

- SELECT na wszystkich tabelach
- INSERT, UPDATE na tabeli t\_uczen
- INSERT, UPDATE na tabeli t\_lekcja
- Wykonywanie pakietów: pkg\_uczen, pkg\_lekcja (bez pkg\_ocena)

## 7.4 Macierz uprawnień

Tabela	Admin	Nauczyciel	Sekretariat
t_instrument	SIUD	S	S
t_sala	SIUD	S	S
t_nauczyciel	SIUD	S	S
t_uczen	SIUD	S	SIU
t_kurs	SIUD	S	S
t_lekcja	SIUD	SU*	SIU
t_ocena_postepu	SIUD	SI	S

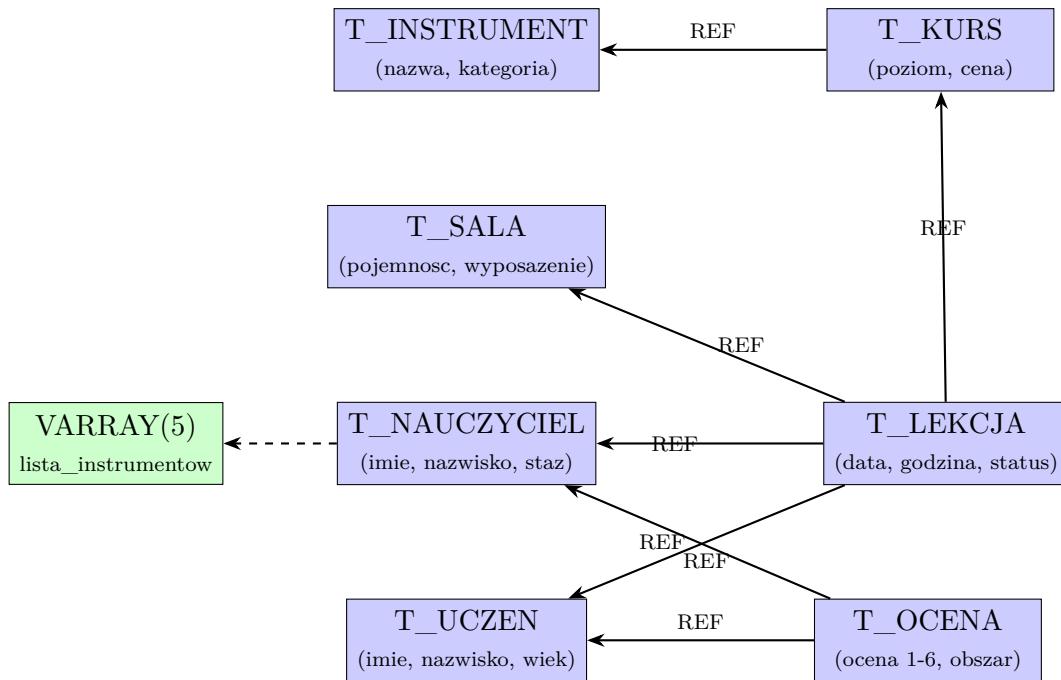
Tabela 6: Macierz uprawnień (S=SELECT, I=INSERT, U=UPDATE, D=DELETE,  
\*tylko status)

## 7.5 Użytkownicy testowi

Użytkownik	Rola	Hasło
usr_admin	rola_admin	Admin123!
usr_nauczyciel	rola_nauczyciel	Naucz123!
usr_sekretariat	rola_sekretariat	Sekr123!

Tabela 7: Użytkownicy testowi

## 8 Diagram relacji obiektów



## Legenda:

- Strzałka ciągła – referencja REF (wskaźnik do obiektu)
  - Strzałka przerywana – zawiera VARRAY (kolekcja)
  - Niebieskie prostokąty – typy obiektowe / tabele
  - Zielony prostokąt – typ kolekcji VARRAY

## 9 Struktura plików projektu

Plik	Zawartość
01_typy.sql	Definicje 8 typów obiektowych z 13 metodami
02_tabele.sql	7 tabel obiektowych, 7 sekwencji, 5 indeksów
03_pakiety.sql	3 pakiety PL/SQL z 15 podprogramami
04_triggery.sql	10 wyzwalaczy walidacyjnych
05_dane.sql	Dane testowe (10 instr., 5 sal, 5 naucz., 10 uczn., 10 kursów)
06_testy.sql	10 scenariuszy testowych
07_uzytkownicy.sql	3 role, 3 użytkowników, synonimy

Tabela 8: Pliki projektu

#### Kolejność uruchamiania:

1. 01\_typy.sql – typy obiektowe
  2. 02\_tabele.sql – tabele i sekwencje
  3. 03\_pakietы.sql – pakietы PL/SQL
  4. 04\_triggery.sql – wyzwalacze
  5. 05\_dane.sql – dane testowe

6. 06\_testy.sql – testy (opcjonalne)
7. 07\_uzlytkownicy.sql – role i użytkownicy (wymaga uprawnień DBA)