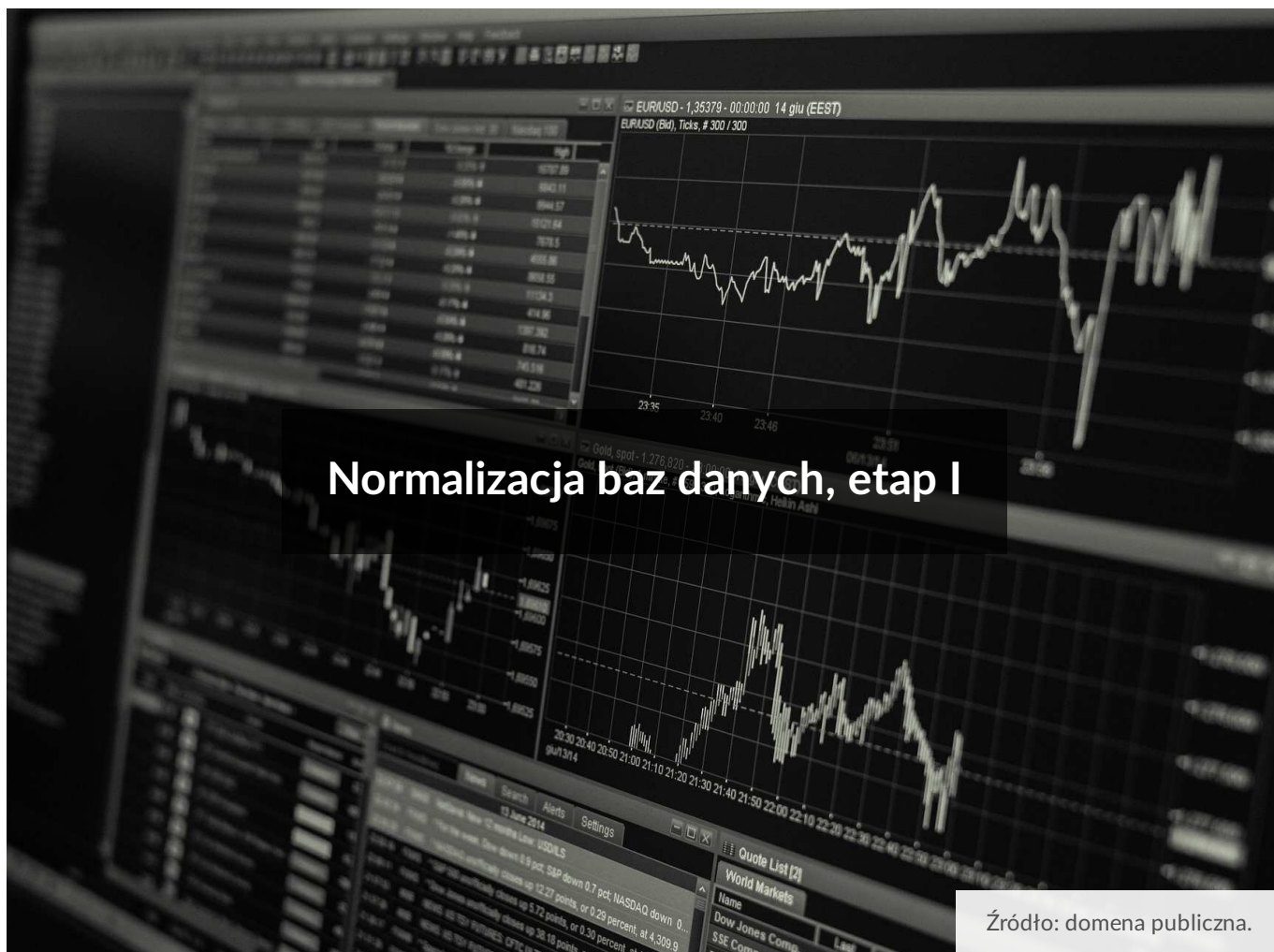




## Normalizacja baz danych, etap I

- Wprowadzenie
- Przeczytaj
- Audiobook
- Sprawdź się
- Dla nauczyciela



## Normalizacja baz danych, etap I

Źródło: domena publiczna.

Normalizacja baz danych ma wiele zalet. Zwiększa przejrzystość modelu bazy, a baza w postaci normalnej jest prosta w utrzymaniu i modyfikacji. Jednak przede wszystkim jest ona sposobem opanowania chaosu związanego z przechowywaniem i utrzymaniem tysięcy rekordów i dziesiątek tabel.

Proces normalizacji opiera się na modyfikacji struktury bazy, z uwzględnieniem tak zwanych postaci normalnych. Są to zbiory reguł, jakimi powinien kierować się programista tworzący bazy, aby były one jak najbardziej spójne i proste w rozbudowie.

Więcej informacji o normalizacji baz danych znajdziesz w e-materiałach:

- [Normalizacja baz danych, etap II,](#)
- [Normalizacja baz danych, etap III.](#)

### Twoje cele

- Prześledzisz, jak funkcjonuje praktyczne zastosowanie normalizacji oraz poznasz jego potencjalne wady.
- Przeanalizujesz zasady pierwszej postaci normalnej.
- Przeprowadzisz normalizację przykładowych relacji.

# Przeczytaj

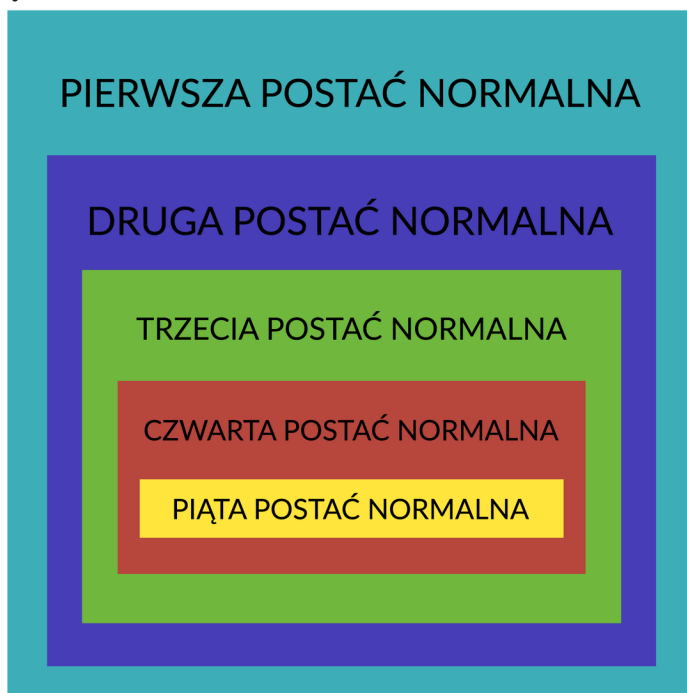
---

## Postacie normalne

Normalizacja polega na sprowadzaniu bazy danych do jednej z tak zwanych **postaci normalnych**. Postać normalna to zbiór reguł, które bazy danych powinny spełniać. W sumie istnieje pięć takich postaci (pięć zbiorów reguł), przy czym większość rzeczywistych baz danych ogranicza się do trzeciej postaci normalnej.

Każda kolejna postać normalna jest szczególnym przypadkiem poprzedniej. Oznacza to, że baza danych, która jest w trzeciej postaci normalnej, spełnia także warunki pierwszej oraz drugiej postaci normalnej. Oznacza to także, że baza w piątej postaci normalnej spełnia warunki wszystkich postaci normalnych.

Przedstawia to poniższy schemat:



Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

## Reguły pierwszej postaci normalnej

Pierwsza postać normalna zdefiniowana jest przez kilka prostych zasad.

1. Jedna tabela opisuje jeden obiekt.
2. Wartości atrybutów są **atomowe**.
3. Tabela nie zawiera powtarzających się informacji.
4. Znaczenie kolumn nie zależy od ich kolejności.

Tabelę, która nie spełnia któregoś z powyższych warunków, nazywamy **nieznormalizowaną**. Poniżej, znajduje się przykład tabeli opisującej gości oraz pokoje w hotelu, ta tabela **łamie** reguły od 1 do 3.

### Nieznormalizowana tabela *Goscie i Pokoje*:

Nr_pokoju	Cena_pokoju	Goscie	Biezaca_liczba_gosci	Maksymalna_liczba_gosci
1	250,00 zł	Adam Nowak, Jan Kowalski	2	2
2	200,00 zł	Ewa Malinowska	1	2

### W jaki sposób ta tabela łamie powyższe reguły?

Po pierwsze, tabela ta miesza informację o pokojach z informacjami o gościach, w ten sposób łamiąc zasadę nr 1.

Po drugie, atrybut *Goscie* zawiera listę, co łamie zasadę 2. Tę i poprzednią kwestię można rozwiązać poprzez rozdzielenie aktualnej tabeli na dwie mniejsze, jedną opisującą gości a drugą pokoje.

Po trzecie, informacja o bieżącej liczbie gości w pokojach znajduje się już w kolumnie *Goscie*, co oznacza, że kolumna *Biezaca\_liczba\_gosci* zawiera powtarzające się informacje.

#### Ciekawostka

Tabela nie łamie reguły nr 4, ponieważ, jeśli zamienimy kolejność kolumn, nie zmieni się ich znaczenie. W praktyce złamanie tej reguły może być wynikiem nieuwagi ze strony programisty. Istnieją przykłady łamania tej reguły – lewa kolumna *Liczba* opisuje aktualną liczbę gości, a prawa – maksymalną.

Nr_pokoju	Cena_pokoju	Goscie	Liczba	Liczba
1	250,00 zł	Adam Nowak, Jan Kowalski	2	2
2	200,00 zł	Ewa Malinowska	1	2

Teraz kolejność kolumn ma znaczenie (rozdzielamy je na prawą i lewą). W związku z tym ta tabela łamie regułę nr 4 postaci normalnej, ponieważ jeśli zamienimy je miejscem, rekordy zmienią sens - zmieni się informacja o liczbie gości w pokoju.

## Doprowadzenie do pierwszej postaci normalnej

Głównym problemem tabel jest próba zamieszczenia zbyt wielu informacji. Może się wydawać, że trzymanie wszystkich informacji w jednym miejscu będzie wygodne. Jednak praktyka pokazuje, że przy takim podejściu w miarę rozbudowywania w bazie pojawiają się wskazane problemy.

Istnieją techniki pozwalające na przekształcenie dowolnej bazy do postaci normalnej.

Pierwszym krokiem jest usunięcie zbędnych informacji. W tym przypadku z tabeli *Goscie i Pokoje* należy usunąć pole *Biezaca\_liczba\_gosci*.

**Tabela *Goscie i Pokoje* po pierwszej poprawce:**

Nr_pokoju	Cena_pokoju	Goscie	Maksymalna_liczba_gosci
1	250,00 zł	Adam Now	2
2	200,00 zł	Ewa Malin	2

W kolejnym kroku należy rozdzielić tabelę *Goscie i Pokoje* na dwie osobne tabele po to, aby pozbyć się listy w kolumnie *Goscie* (a dzięki temu ułatwić przyszłą rozbudowę bazy). Jak to zrobić?

Każdy gość może być przypisany do tylko jednego pokoju, ale każdy pokój może być przypisany do wielu gości. Jest to tak zwana relacja 1-N (jeden pokój - wiele osób).

Po stworzeniu osobnej tabeli *Goscie* musi być w niej przechowywana kolumna numerów pokoju (po jednym numerze dla każdego gościa).

**Nowa tabela *Goscie* po zastosowaniu zmian:**

Id	Imie	Nazwisko	Pokoj
1	Adam	Nowak	1
2	Jan	Kowalski	1
3	Ewa	Malinowska	2

Zwróć uwagę na to, że teraz każdy gość posiada swój identyfikator (**klucz główny**), w formie numeru. Nie możemy używać imienia i nazwiska jako klucza głównego, ponieważ istnieje (stosunkowo mała, ale realna) szansa, że dwóch gości będzie miało to samo imię i nazwisko.

**Tabela *Pokoje* po modyfikacji:**

Nr_pokoju	Cena_pokoju	Maksymalna_liczba_gosci
1	250,00 zł	2
2	200,00 zł	2

Kolumna *Biezaca\_liczba\_gosci* została usunięta, ponieważ była powtórzoną informacją. Jak teraz uzyskać informację o liczbie gości w pokoju, np. w pokoju 1.? Wystarczy policzyć liczbę rekordów w kolumnie *Pokoj* w tabeli *Goscie*, w których numer pokoju to 1.

W praktyce pisze się zapytania do bazy danych, aby otrzymać tego typu informacje.

Dzięki poprawkom tabele są teraz w pierwszej postaci normalnej.

### Dla zainteresowanych

Ktoś dociekliwy mógłby zapytać, czy powtórzenia numerów pokoi w kolumnie *Pokoje* łamią zasady o niepowtarzaniu informacji.

Powtórzona liczba to nie to samo co powtórzona informacja. Można porównać tę sytuację z oryginalną tabelą *Goscie i Pokoje*, w której usunięcie kolumny *Biezaca\_liczba\_gosci* nie prowadziło do utraty żadnych danych, bo można było policzyć gości w kolumnie *Goscie*. Natomiast w nowej tabeli *Goscie* usunięcie numeru pokoju usunie z bazy danych jakąkolwiek informację o tym, gdzie dany gość jest zameldowany.

W tym przypadku numery w kolumnie *Pokoje* tabeli *Goscie* nie stwierdzają wyłącznie istnienia danego pokoju (gdyby tak było, istotnie doszłoby do powtórzenia). Kolumna *Pokoje* mówi o tym, gdzie dany gość jest zameldowany i jest jedyną instancją tej informacji.

## Anomalie

Ważnym aspektem normalizacji jest uniknięcie tak zwanych anomalii wstawiania, usuwania i modyfikacji. Anomalie powodują problemy podczas wykonywania tych operacji.

Tabela przedstawia nauczycieli w pewnej szkole, sale, w których uczą oraz przedmioty przez nich prowadzone.

Id	Imie	Nazwisko	Przedmiot	Sala
1	Joachim	Mróz	j. polski	2
2	Joachim	Mróz	j. angielski	2
3	Halina	Czarnecka	matematyka	3
4	Marek	Stępień	historia	4

### Anomalia wstawiania

Zachodzi kiedy nie możemy zamieścić jakiejś informacji w bazie danych bez dodatkowego zamieszczenia innej informacji.

Jeśli szkoła zatrudni nowego nauczyciela, to nie może go zarejestrować w systemie bez przypisania sali.

### Anomalia usuwania

Taka anomalia zachodzi kiedy podczas usunięcia niepotrzebnej informacji jesteśmy zmuszeni usunąć inną, przydatną informację.

Na przykład kiedy nauczyciel tymczasowo przestanie nauczać swoich przedmiotów, to usunięcie informacji o przedmiotach automatycznie usunie wszystkie wzmianki o tym nauczycielu z bazy danych.

## Anomalia modyfikacji

Anomalia modyfikacji pojawia się w przypadku, kiedy istnieje możliwość stworzenia sprzeczności wskutek nieprzemyślanej modyfikacji danych.

Założmy, że jeden nauczyciel może uczyć wszystkich przedmiotów tylko w jednej sali. Teraz da się zmodyfikować tylko jeden wpis z numerem sali i spowodować sprzeczność w bazie danych.

## Słownik

### atrybut atomowy

atrybut niepodzielny; na przykład liczba całkowita, ciąg znaków, wartość logiczna

### klucz główny

kolumna w tabeli, w której żaden rekord nie może się powtarzać; służy on jako identyfikator wierszy; może to także być wiele kolumn, takich, że żadne dwa wiersze nie posiadają wszystkich wartości takich samych w tych kolumnach (oznacza to, że wartości klucza głównego nie mogą być takie same dla dwóch różnych rekordów)

# Audiobook

---

## Polecenie 1

Zapoznaj się z audiobookiem. Wykonaj ćwiczenie.

Audiobook można wysłuchać pod adresem: <https://zpe.gov.pl/b/PawapAA11>

---

Dlaczego warto przestrzegać reguł postaci normalnych?

Zasada mówiąca, że jedna tabela może opisywać tylko jeden obiekt, gwarantuje przejrzystość bazy danych. Dzięki tej regule łatwo można domyślić się, gdzie przechowywana jest dana informacja.

Druga zasada, mówiąca o tworzeniu wyłącznie atrybutów atomowych, ułatwia modyfikowanie bazy danych. Na przykład w sekcji „Przeczytaj” w tabeli Goście i Pokoje dane dotyczące gości przechowywane były w formie listy. Załóżmy, że właściciel hotelu daje każdemu klientowi możliwość wykupienia śniadań na całą długość wizyty. Najłatwiej jest odzwierciedlić to w tabeli poprzez przechowywanie dla każdego gościa wartości logicznej (Prawda lub Fałsz), która symbolizuje, czy wykupił posiłek.

Zwróć uwagę, że w omawianej tabeli dodanie tej opcji jest wyjątkowo kłopotliwe, ponieważ przechowywanie listy gości zmusza nas jednocześnie do przechowywania listy wartości logicznych. Mało tego, w obu listach kolejność ma znaczenie, a dodatkowo może ona być zaburzona podczas usuwania tylko niektórych gości z pokoju.

Problem ten rozwiązuje rozbicie tabeli Goście i Pokoje na dwie mniejsze. Dzięki czemu każdy gość odnotowany jest w osobnym wierszu i nie musimy się przejmować kolejnością ani listami.

Trzecia zasada, która zakazuje powtarzania się informacji, służy dwóm celom. Po pierwsze oszczędza pamięć twardą serwera poprzez przechowywanie mniejszej liczby pól. Po drugie, co w praktyce często okazuje się ważniejsze, zasada ta zapewnia spójność i zapobiega przechowywaniu sprzecznych informacji. Posiadanie ich kilku kopii rodzi ryzyko, że w wyniku błędu ludzkiego zostanie zmodyfikowana tylko jedna, co z kolei może wywołać problemy.

Na przykład, jeśli w wierszu przechowywana jest jednocześnie liczba gości oraz ich lista, istnieje szansa, że jakiś gość zostanie usunięty z listy, ale liczba gości nie zostanie

zmniejszona. W tym momencie w bazie znajdują się sprzeczne informacje, które mogą uniemożliwić zarezerwowanie pokoju przez nowych klientów.

Potencjalne wady postaci normalnych.

Głównym minusem postaci normalnej jest to, że baza w tej formie zazwyczaj potrzebuje nieco większej ilości zasobów serwera niż (dobrze przygotowana) baza nieznormalizowana. Na przykład kreowanie nowych tabel często wymaga utworzenia nowych identyfikatorów, co zaś skutkuje użyciem dodatkowej pamięci twardej serwera.

Z drugiej strony, źle przygotowane bazy nieznormalizowane cechują się zarówno gorszą spójnością, jak i gorszą optymalizacją.

W praktyce dopiero druga i trzecia postać normalna mają istotnie negatywny wpływ na prędkość działania bazy danych, jednak o nich dowiesz się z kolejnych e-materiałów.

Wadą postaci normalnej jest również większe zużycie pamięci obliczeniowej oraz wolniejsza prędkość odczytu danych. Wynika to z faktu, że w bazach znormalizowanych często konieczne jest wykonanie kwerendy w celu otrzymania jakiejś informacji.

Opisane wady postaci normalnej nie zawsze występują, a nawet jeśli mają miejsce, to zdecydowanie ważniejsze okazują się ich zalety. Od oszczędności zasobów istotniejsze są przejrzystość baz danych oraz ich otwartość na modyfikacje.

Jednak jeśli priorytetem jest prędkość, warto pozostawić bazę nieznormalizowaną. W praktyce takie bazy wykorzystują wiele technik przyspieszających ich działanie, takich jak indeksowanie, odpowiednie partycjonowanie danych, stosowanie własnych struktur oraz bardzo często rezygnacja z wykorzystania języka SQL na rzecz szybszych alternatyw.

---

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY-SA 3.0.

## Ćwiczenie 1

Poszukaj w sieci różnych baz danych. Na ich przykładzie przedstaw anomalie oraz normalizację.

# Sprawdź się

Pokaż ćwiczenia:   

Ćwiczenie 1



Ćwiczenie 2



Ćwiczenie 3



Zapoznaj się z grafiką i wykonaj ćwiczenie.

Nr_ucznia	Imie	Nazwisko	Oceny	Srednia
1	Bartłomiej	Krzyżak	4,5,3	4
2	Anna	Tulewska	3,5,5,3	4
3	Hubert	Petrowski	2,5,2	3

Ćwiczenie 4



Zapoznaj się z grafiką i wykonaj ćwiczenie.

Nr_ucznia	Imie	Nazwisko	Oceny	Srednia
1	Bartłomiej	Krzyżak	4,5,3	4
2	Anna	Tulewska	3,5,5,3	4
3	Hubert	Petrowski	2,5,2	3

Ćwiczenie 5



Ćwiczenie 6



Ćwiczenie 7



## Ćwiczenie 8



Zapoznaj się z grafiką i wykonaj ćwiczenie.

Id	Nazwa_samochodu	Wynajmujacy	Data_wynajmu	Cena_wynajmu
1	Suzuki Foreza	Lidia Szymańska	11.01.2021	299,00 zł
2	Volkswagen Golf	Marlena Mróz	19.04.2021	499,00 zł
3	Ford Focus	Cyprian Sikora	01.05.2021	399,00 zł

# Dla nauczyciela

---

**Autor:** Maurycy Gast

**Przedmiot:** Informatyka

**Temat:** Normalizacja baz danych, etap I

**Grupa docelowa:**

Szkoła ponadpodstawowa, liceum ogólnokształcące, technikum, zakres rozszerzony

**Podstawa programowa:**

Cele kształcenia – wymagania ogólne

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

Treści nauczania – wymagania szczegółowe

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

4) przygotowując opracowania rozwiązań złożonych problemów, posługuje się wybranymi aplikacjami w stopniu zaawansowanym:

d) projektuje i tworzy relacyjną bazę złożoną z wielu tabel oraz sieciową aplikację bazodanową dla danych związanych z rozwiązywanym problemem, formułuje kwerendy, tworzy i modyfikuje formularze oraz raporty, stosuje język SQL do wyszukiwania informacji w bazie i do jej modyfikacji, uwzględnia kwestie integralności danych, bezpieczeństwa i ochrony danych w bazie,

**Kształtowane kompetencje kluczowe:**

- kompetencje cyfrowe;
- kompetencje osobiste, społeczne i w zakresie umiejętności uczenia się;
- kompetencje matematyczne oraz kompetencje w zakresie nauk przyrodniczych, technologii i inżynierii.

**Cele operacyjne (językiem ucznia):**

- Prześledzisz, jak funkcjonuje praktyczne zastosowanie normalizacji oraz poznasz jego potencjalne wady.
- Przeanalizujesz zasady pierwszej postaci normalnej.
- Przeprowadzisz normalizację przykładowych relacji.

### **Strategie nauczania:**

- konstruktywizm;
- konektywizm.

### **Metody i techniki nauczania:**

- dyskusja;
- rozmowa nauczająca z wykorzystaniem multimediu i ćwiczeń interaktywnych;
- metody aktywizujące.

### **Formy pracy:**

- praca indywidualna;
- praca w parach;
- praca w grupach;
- praca całego zespołu klasowego.

### **Środki dydaktyczne:**

- komputery z głośnikami, słuchawkami i dostępem do internetu;
- zasoby multimedialne zawarte w e-materiale;
- tablica interaktywna/tablica, pisak/kreda.

### **Przebieg lekcji**

#### **Przed lekcją:**

1. **Przygotowanie do zajęć.** Nauczyciel loguje się na platformie i udostępnia e-materiał: „Normalizacja baz danych, etap I”. Uczniowie zapoznają się z treściami w sekcji „Przeczytaj”.

#### **Faza wstępna:**

1. Chętna lub wybrana osoba podsumowuje najważniejsze informacje dotyczące baz danych.
2. Nauczyciel wyświetla uczniom temat zajęć oraz cele. Prosi, by na ich podstawie uczniowie sformułowali kryteria sukcesu.

#### **Faza realizacyjna:**

1. **Praca z tekstem.** Jeżeli przygotowanie uczniów do lekcji jest niewystarczające, nauczyciel prosi o indywidualne zapoznanie się z treścią zawartą w sekcji „Przeczytaj”. Każdy uczestnik zajęć podczas cichego czytania wynotowuje najważniejsze kwestie poruszane w tekście. Nauczyciel odpowiada na ew. pytania.
2. **Praca z multimedium.** Nauczyciel prezentuje zawartość sekcji „Audiobook”. Uczniowie wspólnie zapoznają się z multimedium. Zapisują ewentualne problemy i pytania.
3. **Ćwiczenie umiejętności.** Nauczyciel przechodzi do sekcji „Sprawdź się”. Uczniowie indywidualnie rozwiązują ćwiczenia nr 1-8 na czas. Osoba, która poprawnie rozwiąże zadania jako pierwsza, wygrywa, a nauczyciel może nagrodzić ją oceną za aktywność. Nauczyciel przedstawia uczniom przykłady baz danych. Prosi o wskazanie, w jaki sposób można je znormalizować.

#### **Faza podsumowująca:**

1. Nauczyciel inicjuje dyskusję na temat tego, jakie są korzyści płynące z normalizacji tabel.

#### **Praca domowa:**

1. Przygotuj notatkę, która podsumuje, kiedy przydają się znormalizowane, a kiedy nieznormalizowane bazy danych.

#### **Wskazówki metodyczne:**

- Serię materiałów dotyczącą normalizacji baz danych można wykorzystać w ramach dwugodzinnych zajęć warsztatowych.