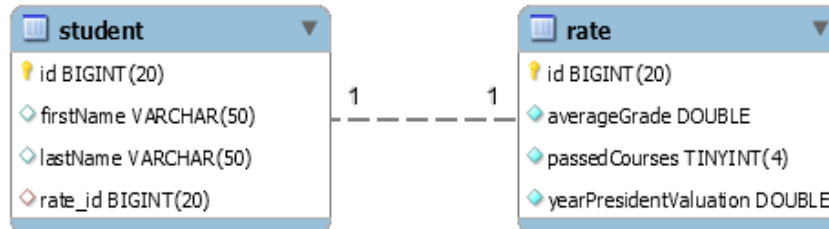


## ĆWICZENIE: Relacje - Hibernate

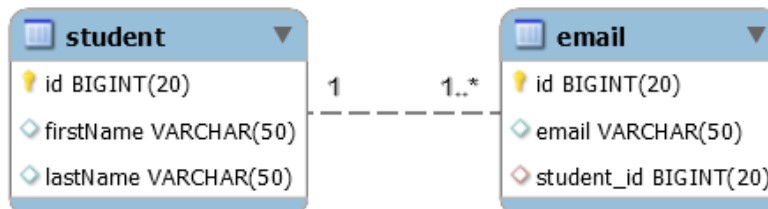
(2 godziny zajęciowe)

1. Jakie mamy podstawowe rodzaje relacji w JPA?
2. Stwórz za pomocą Hibernate jednokierunkową relację jeden do jednego zgodnie z diagramem poniżej i wprowadź przykładowe dane. Sprawdź wynik w bazie danych.

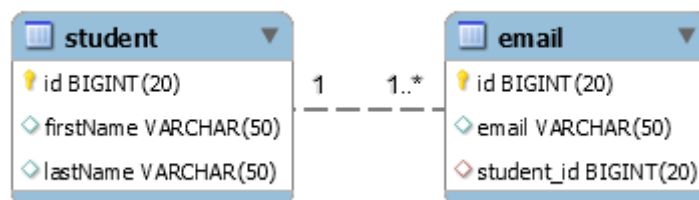


Ctrl+Shift+o import potrzebnych bibliotek

3. Stwórz za pomocą Hibernate jednokierunkową relację jeden do wielu zgodnie z diagramem poniżej i wprowadź przykładowe dane. Sprawdź wynik w bazie danych.



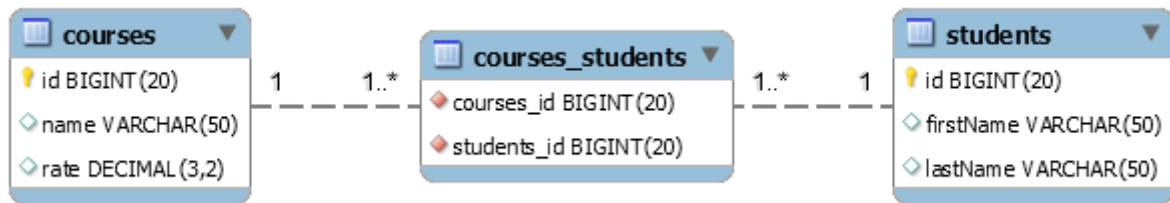
4. Co powoduje zastosowanie/usunięcie adnotacji `@JoinColumn` w powyższym przykładzie?
5. Stwórz za pomocą Hibernate dwukierunkową relację jeden do wielu zgodnie z diagramem poniżej i wprowadź przykładowe dane. Sprawdź wynik w bazie danych.



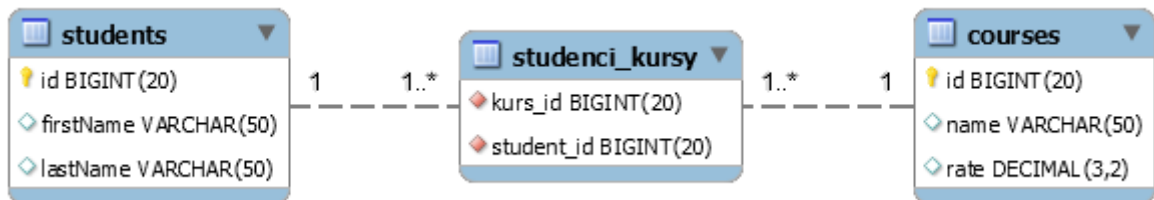
Jak można zauważyć, nie ma różnicy na poziomie bazy danych pomiędzy tymi rozwiązaniami. Jaki jest cel stosowania relacji dwukierunkowej? Odpowiedź zawrzyj w sprawozdaniu.

6. Za pomocą Hibernate dodaj Studenta "Jan Kowalski" oraz jego dwa emaile. Następnie za pomocą JPQL (Java Persistence Query Language) pobierz dane tego studenta z bazy danych i wyświetl w konsoli (imię, nazwisko i wszystkie jego emaile).

7. Stwórz za pomocą Hibernate relację wiele do wielu zgodnie z diagramem poniżej i wprowadź przykładowe dane. Sprawdź wynik w bazie danych.



8. Używając adnotacji `@JoinTable(name, joinColumns, inverseJoinColumns)` zmodyfikuj tabelę `courses_students`, aby wyglądała jak poniżej



Jeden z przydatnych skrótów w programie Eclipse `alt+shift+L` (extract local variable), który warto zapamiętać.