ЗМІСТ

[ВСТУП 4](#_Toc482592904)

[1. ПРЕДМЕТНА ОБЛАСТЬ 5](#_Toc482592905)

[1.1 Постановка завдання 5](#_Toc482592906)

[1.2 Інформаційні потоки 5](#_Toc482592907)

[1.3 Види запитів в інформаційній системі 6](#_Toc482592908)

[2 ПРОЕКТУВАННЯ БАЗИ ДАНИХ 7](#_Toc482592909)

[2.1 Сутності 7](#_Toc482592910)

[2.3. Фізичне проектування 10](#_Toc482592911)

[3. РЕАЛІЗАЦІЯ ПРОЕКТУ 14](#_Toc482592912)

[3.1. Реалізація бази даних 14](#_Toc482592913)

[3.2. Реалізація прикладної програми 14](#_Toc482592914)

[ВИСНОВКИ 20](#_Toc482592915)

[СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 21](#_Toc482592916)

[ДОДАТОК Г 22](#_Toc482592917)

[ДОДАТОК А 2](#_Toc482592918)

[ДОДАТОК Б 2](#_Toc482592919)

[ДОДАТОК B 3](#_Toc482592920)

# ВСТУП

[База даних](http://lib.mdpu.org.ua/e-book/vstup/glos.htm#База даних) (БД) – упорядкований набір логічно взаємопов'язаних даних, що використовується спільно, та призначений для задоволення інформаційних потреб користувачів.

[Система управління базами даних](http://lib.mdpu.org.ua/e-book/vstup/glos.htm#Система управління базою даних) (СУБД) – це комплекс програмних і мовних засобів, необхідних для створення баз даних, підтримання їх в актуальному стані та організації пошуку в них необхідної інформації.

Централізований характер управління даними в базі даних передбачає необхідність існування деякої особи (групи осіб), на яку покладаються функції адміністрування даними, що зберігаються в базі.

Головним завданням [БД](http://lib.mdpu.org.ua/e-book/vstup/glos.htm#База даних) є гарантоване збереження значних обсягів інформації та надання доступу до неї користувачеві або ж прикладній програмі. Таким чином БД складається з двох частин: збереження інформації та системи управління нею.

# 1. ПРЕДМЕТНА ОБЛАСТЬ

* 1. **Постановка завдання**

Темою курсової роботи – є проектування бази даних міжнародних конкурсів танців. Так як видів танців та видів конкурсів існує багато, було обрано спортивні бальні танці. База даних міжнародних конкурсів зі спортивних бальних танців, насамперед, включає в себе, перегляд інформації про змагання та реєстрацію в них: учасників, тренерів та судей. Кожен конкурс включає в себе назву, день проведення, правила, організаторів змагань, країну та точне місце проведення. В кожному конкурсі має обов’язково бути: тип бальної програми.

* 1. **Інформаційні потоки**

Інформаційні потоки в міжнародних конкурсах танців:

* потік «Competitions»: представляє інформацію про змагання;
* потік «BallroomPrograms»: представляє інформацію про типи програм які є у бальних танцях;
* потік «Categories»: представляє інформацію про категорії які будуть на конкрутному змаганні;
* потік «Classes»: представляє інформацію про класи які будуть на конкрутному змаганні;
* потік «Clouples»: представляє інформацію про парі які будуть брати участь у конкрутному на змаганні;
* потік «Partners»: представляє інформацію про партнерів які зареєстровані в базі даних;
* потік «Shepartners»: представляє інформацію про партнерш які зареєстровані в базі даних;
* потік «Coaches»: представляє інформацію про тренерів які зареєстровані в базі даних;
* потік «Adjudicators»: представляє інформацію про суддів які зареєстровані в базі даних.
  1. **Види запитів в інформаційній системі**
* одержати інформацію про міжнародні змагання;
* одержати інформацію про якесь конкретне змагання: місце проведення, правила, організатори, дату проведення, загальну кількість учасників;
* одержати загальну інформацію про кількість танцювальних програм, категорій, класів та учасників у кожній із них;
* одержати інформацію про кожну танцювальну програму, категорію та клас окремо;
* зареєструвати учасника у базу даних;
* зареєструвати тренера у базу даних;
* зареєструвати суддю у базу даних;
* зареєструвати суддю на конкретне змагання;
* зареєструвати учасника на конкретне змагання;
* зареєструвати нове змагання та додати у нього інформацію.

# 2 ПРОЕКТУВАННЯ БАЗИ ДАНИХ

## 2.1 Сутності

Проаналізувавши предметну область, було виділено наступні сутності:

1. Adjudicators
2. BallroomPrograms
3. Categories
4. Classes
5. Coaches
6. Competitions
7. Couples
8. Partners
9. Shepartners

Під поняттям «сутність» ми розуміємо абстракцію деякого об’єкту предметної області. Розроблена концептуальна модель представлена в додатку А.

Розглянемо специфікацію описаних вище сутностей (табл. 1). В таблиці вказано сутність, її опис та первинний ключ. При чому ключ може бути як простий, так і складений.

|  |  |  |  |
| --- | --- | --- | --- |
| **Сутність** | **Опис сутності** | **Ключ** | **Опис атрибута** |
| Competitions | Містить інформацію про всі конкурси | idCompetition | Порядковий номер змагання, який генерується при додаванні в таблицю(первинний ключ) |
| Ballroom  Programs | Містить інформацію про типи програм | idProgram  idCompetition | Слабка сутність, ключ генерується із ключів сутностей idProgram (первинний ключ)  idCompetition |
| Categories | Містить інформацію про типи категорій | CategoryID  idProgram  idCompetition | Слабка сутність, ключ генерується із ключів сутностей CategoryID(первинний ключ)  idProgram  idCompetition |
| consistClass | Містить інформацію про зв’язок між класами та категоріями | idCompetition IdProgram  CategoryID  ClassID | Слабка сутність, ключ генерується із ключів сутностей Classes і Categories |
| Classes | Містить інформацію про класи | idCompetition ClassID | Слабка сутність, ключ генерується із ключів сутностей ClassID  (первинний ключ)  idCompetition |
| Couples | Містить інформацію про пари | idCompetition IdProgram  CategoryID  ClassID  PairNumber | Слабка сутність, ключ генерується із ключів сутностей Classes і Categories  PairNumber (первинний ключ) |
| Coaches | Містить інформацію про тренерів | nameCoach  surname  Coach  codeCoach | Порядковий номер партнера, який генерується при додаванні в таблицю(первинний ключ) |
| Shepartners | Містить інформацію про партнерок | name  Shepartners  surname  Shepartners  codeShepartners | Порядковий номер партнерки, який генерується при додаванні в таблицю (первинний ключ) |
| Partners | Містить інформацію про партнерів | namePartners  surname  Partners  codePartners | Порядковий номер партнера, який генерується при додаванні в таблицю (первинний ключ) |
| judge | Містить інформацію про  суддів та  змаганнь | idCompetitionsurname  Adjudicator name  Adjudicator code  Adjudicator | Слабка сутність, ключ генерується із ключів сутностей Competitions і Adjudicators |
| Adjudicators | Містить інформацію про суддів | surname  Adjudicator name  Adjudicator code  Adjudicator | Порядковий номер судді, який генерується при додаванні в таблицю (первинний ключ) |

Таблиця 1. Опис сутностей

Зв'язок між сутностями характеризується: типом зв'язку (1:M, M:М). Зв’язок 1:M (“один-до-багатьох”) означає, що кожному запису в одній таблиці може відповідати багато записів в іншій таблиці. Зв’язок M:M (“багато-до-багатьох”) означає, що кожен запис в одній таблиці пов'язаний з декількома записами в іншій таблиці і навпаки. Зв’язки між сутностями представлено в таблиці 2.

|  |  |  |  |
| --- | --- | --- | --- |
| **Батьківська сутність** | **Дочірня сутність** | **Опис зв’язку** | **Потужність зв’язку** |
| Competitions | BallroomPrograms | consist | 1:M |
| BallroomPrograms | Classes | consistCategories | 1:M |
| Categories | Class | consistClasses | M:M |
| Class | Couples | haveClass | 1:М |
| Couples | Categories | partIn | М:1 |
| Couples | Partners | consistPartners | М:1 |
| Couples | ShePartners | consistShePartners | М:1 |
| Couples | Coaches | consistCoaches | М:1 |
| Competitions | Adjudicators | judge | M:М |

Таблиця 2. Опис зв’язків

## 2.3. Фізичне проектування

Модель даних на фізичному рівні відрізняється від моделі даних на логічному рівні тим, що вона повністю орієнтована на обрану СУБД, тобто на відміну від логічної моделі, в якій не має значення, який конкретно тип даних має атрибут, у фізичній моделі даних важливо описати інформацію про конкретні фізичні об'єкти – таблиці, поля, індекси, процедури і т.д. В ході проектування фізичного рівня була отримана модель, яка представлена в додатку В. Список атрибутів, типів, первинних та зовнішніх ключів цих відношень наведено в таблицях.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Data Type | Length | P | F | M |
| IdCompetition | INTEGER |  | X |  | X |
| Title | VARCHAR(32) | 32 |  |  | X |
| DateCompetition | VARCHAR(32) | 32 |  |  | X |
| Place | VARCHAR(32) | 32 |  |  | X |
| Rules | VARCHAR(32) | 32 |  |  |  |
| Organizers | VARCHAR(32) | 32 |  |  |  |
| Country | VARCHAR(32) | 32 |  |  |  |

Таблиця. 3 Сутність Competitions

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| IdCompetition | IdCompetition | INTEGER |  | X | X | X |
| IdProgram | IdProgram | INTEGER |  | X |  | X |
| typeOfProgram | typeOfProgram | VARCHAR(32) | 32 |  |  |  |

Таблиця. 3 Сутність BallroomPrograms

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| IdCompetition | IdCompetition | INTEGER |  | X | X | X |
| IdProgram | IdProgram | INTEGER |  | X | X | X |
| CategoryID | CategoryID | INTEGER |  | X |  | X |
| CategoryName | CategoryName | VARCHAR(32) | 32 |  |  |  |

Таблиця. 4 Сутність Categories

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| IdCompetition | IdCompetition | INTEGER |  | X | X | X |
| IdProgram | IdProgram | INTEGER |  | X | X | X |
| CategoryID | CategoryID | INTEGER |  | X | X | X |
| ClassID | ClassID | INTEGER |  | X | X | X |

Таблиця. 5 Сутність consistClass

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| ClassID | ClassID | INTEGER |  | X |  | X |
| ClassName | ClassName | VARCHAR(32) | 32 |  |  |  |

Таблиця. 6 Сутність Classes

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| ClassID | ClassID | INTEGER |  | X | X | X |
| IdCompetition | IdCompetition | INTEGER |  | X | X | X |
| IdProgram | IdProgram | INTEGER |  | X | X | X |
| CategoryID | CategoryID | INTEGER |  | X | X | X |
| PairNumber | PairNumber | INTEGER |  | X |  | X |
| codePartner | codePartner | VARCHAR(32) | 32 |  | X | X |
| namePartner | namePartner | VARCHAR(32) | 32 |  | X | X |
| surnamePartner | surnamePartner | VARCHAR(32) | 32 |  | X | X |
| surnameShepartner | surnameShepartner | VARCHAR(32) | 32 |  | X | X |
| nameShepartner | nameShepartner | VARCHAR(32) | 32 |  | X | X |
| codeShePartner | codeShePartner | INTEGER |  |  | X | X |
| surnameCoach | surnameCoach | VARCHAR(32) | 32 |  | X |  |
| nameCoach | nameCoach | VARCHAR(32) | 32 |  | X |  |
| codeCoach | codeCoach | INTEGER |  |  | X |  |

Таблиця. 7 Сутність Couples

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| Country | Country | VARCHAR(32) | 32 |  |  |  |
| nameCoach | nameCoach | VARCHAR(32) | 32 | X |  | X |
| surnameCoach | surnameCoach | VARCHAR(32) | 32 | X |  | X |
| patronymicCoaches | patronymicCoaches | VARCHAR(32) | 32 |  |  |  |
| codeCoach | codeCoach | INTEGER |  | X |  | X |

Таблиця. 8 Сутність Coaches

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| nameShepartner | nameShepartner | VARCHAR(32) | 32 | X |  | X |
| surnameShepartner | surnameShepartner | VARCHAR(32) | 32 | X |  | X |
| patronymic | patronymic | VARCHAR(32) | 32 |  |  |  |
| codeShePartner | codeShePartner | INTEGER |  | X |  | X |
| Country | Country | VARCHAR(32) | 32 |  |  |  |

Таблиця. 9 Сутність Shepartners

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| Country | Country | VARCHAR(32) | 32 |  |  |  |
| namePartner | namePartner | VARCHAR(32) | 32 | X |  | X |
| surnamePartner | surnamePartner | VARCHAR(32) | 32 | X |  | X |
| patronymic | patronymic | VARCHAR(32) | 32 |  |  |  |
| codePartner | codePartner | VARCHAR(32) | 32 | X |  | X |

Таблиця. 10 Сутність Partners

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| IdCompetition | IdCompetition | INTEGER |  | X | X | X |
| surnameAdjudicator | surnameAdjudicator | VARCHAR(32) | 32 | X | X | X |
| nameAdjudicator | nameAdjudicator | VARCHAR(32) | 32 | X | X | X |
| codeAdjudicator | codeAdjudicator | INTEGER |  | X | X | X |

Таблиця. 11 Сутність judge

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Code | Data Type | Length | P | F | M |
| nameAdjudicator | nameAdjudicator | varchar(32) | 32 | X |  | X |
| surnameAdjudicator | surnameAdjudicator | varchar(32) | 32 | X |  | X |
| patronymicAdjudicator | patronymicAdjudicator | varchar(32) | 32 |  |  |  |
| codeAdjudicator | codeAdjudicator | integer |  | X |  | X |
| Country | Country | varchar(32) | 32 |  |  |  |

Таблиця. 12 Сутність Adjudicators

# 3. РЕАЛІЗАЦІЯ ПРОЕКТУ

## 3.1. Реалізація бази даних

Базу даних було реалізовано за допомогою СУБД Firebird.

**Firebird** – компактна, крос-платформова, [вільна](https://uk.wikipedia.org/wiki/%D0%92%D1%96%D0%BB%D1%8C%D0%BD%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BD%D0%B5_%D0%B7%D0%B0%D0%B1%D0%B5%D0%B7%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%BD%D1%8F) [реляційна система керування базами даних](https://uk.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D1%96%D0%B9%D0%BD%D0%B0_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85), що реалізує більшість функцій стандарту [SQL:2003](https://uk.wikipedia.org/wiki/SQL:2003). Вона може запускатись на більшості [UNIX](https://uk.wikipedia.org/wiki/UNIX)-подібних систем (в тому числі [Linux](https://uk.wikipedia.org/wiki/Linux) та [FreeBSD](https://uk.wikipedia.org/wiki/FreeBSD)) та [Windows](https://uk.wikipedia.org/wiki/Microsoft_Windows).

Версійна архітектура: основна особливість Firebird – версійна архітектура, що дозволяє [серверу](https://uk.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B2%D0%B5%D1%80) обробляти різні версії одного запису в будь-який час таким чином, що кожна [транзакція](https://uk.wikipedia.org/wiki/%D0%A2%D1%80%D0%B0%D0%BD%D0%B7%D0%B0%D0%BA%D1%86%D1%96%D1%8F) бачить свою версію даних, не заважаючи сусіднім. Таким чином читаючі транзакції не блокують пишучі, і навпаки. Окрім того, це дає можливість відмовитись від логу транзакцій і таким чином зменшити ймовірність пошкодження службової інформації бази даних.

Як графічне середовище редагування бази даних викоритовувся Sybase PowerDesigner. Загалом за фізичною моделлю було створено 11 таблиць.

## 3.2. Реалізація прикладної програми

Для реалізації прикладної програми було обрано середовище nodeJS, express, jade, css, GIT.

**Node.js** – платформа з [відкритим](https://uk.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D1%96%D1%82%D0%B8%D0%BA%D0%B0_%D0%B2%D1%96%D0%B4%D0%BA%D1%80%D0%B8%D1%82%D0%BE%D0%B3%D0%BE_%D0%BA%D0%BE%D0%B4%D1%83) [кодом](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%80%D1%86%D0%B5%D0%B2%D0%B8%D0%B9_%D0%BA%D0%BE%D0%B4) для виконання високопродуктивних мережевих [застосунків](https://uk.wikipedia.org/wiki/%D0%97%D0%B0%D1%81%D1%82%D0%BE%D1%81%D1%83%D0%BD%D0%BE%D0%BA), написаних мовою [JavaScript](https://uk.wikipedia.org/wiki/JavaScript). Платформа node.js перетворила мову JavaScript, що в основному використовувалась в браузерах на мову загального використання з великою спільнотою розробників.

Node.js характеризується такими властивостями:

* асинхронна [однопотокова](https://uk.wikipedia.org/wiki/%D0%9D%D0%B8%D1%82%D1%8C) модель виконання запитів;
* неблокуючий ввід/вивід;
* система модулів [CommonJS](https://uk.wikipedia.org/wiki/CommonJS);
* для управління модулями використовується пакетний менеджер [npm](https://uk.wikipedia.org/w/index.php?title=Npm&action=edit&redlink=1) (node package manager).

Express – це мінімалістичний і гнучкий веб-фреймворк для додатків Node.js, що надає великий набір функцій для мобільних і веб-додатків.

Jade – це препроцесор HTML і шаблонизатор, який був написаний на JavaScript для Node.js. Простіше кажучи, Jade – це саме той засіб, який надає вам можливість написання розмітки абсолютно по новому, з цілим рядом переваг у порівнянні зі звичайним HTML.

Git – розподілена система керування версіями.

Перелік усіх змагань, за адресою ‘/competitions’ Рис.1.

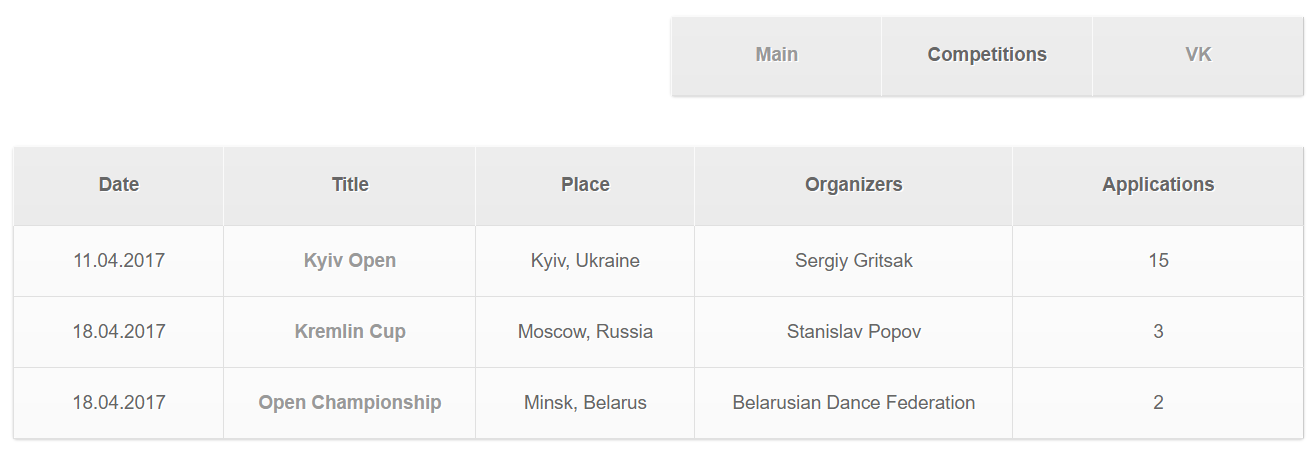


Рис. 1 – Перелік усіх змагань

Перелік усієї інформації про конкретне змагання, за адресою ‘/competition/id\_competition/info’ Рис.2.



Рис. 2 – Перелік усієї інформації про конкретне змагання

Для переходу між розділами додатку використовуємо панель управління Рис. 3.



Рис. 3 – Панель управління

Перелік усієї програми конкурсу та перегляд кількості пар у кожній категорії, за адресою ‘/competition/:id\_competition/details’ Рис. 4.

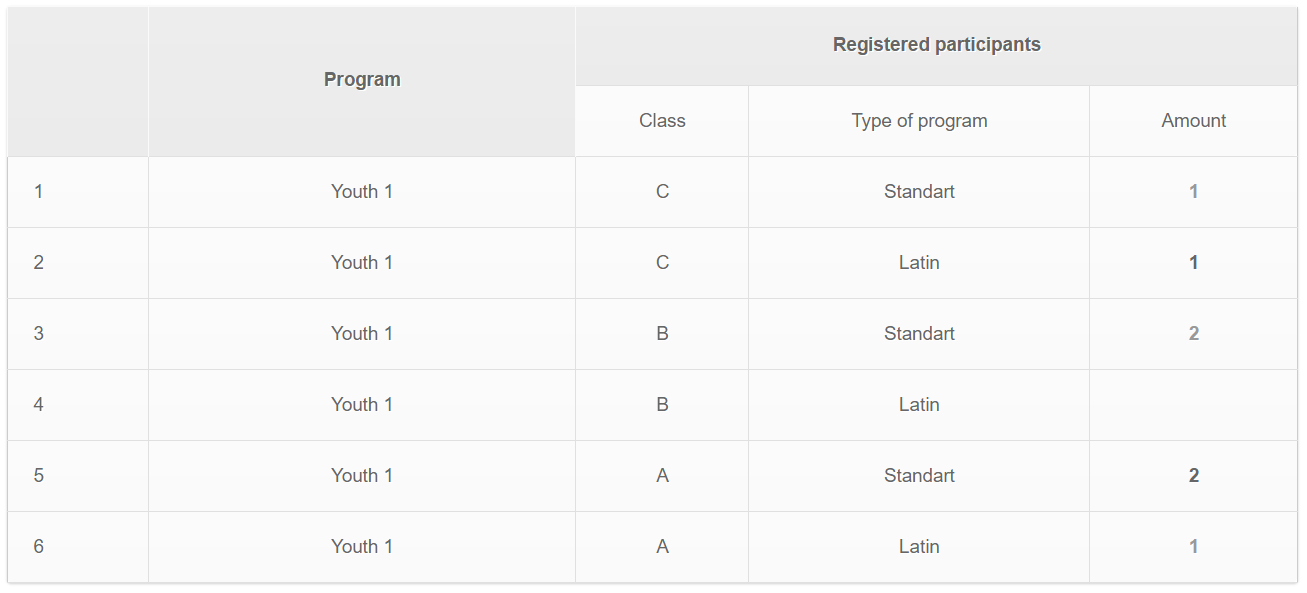


Рис. 4 – Програма конкурсу

Перегляд одної категорії, за адресою ‘/competition/:id\_competition//oneCategory/Youth%201/B/Standart’ Рис. 5.

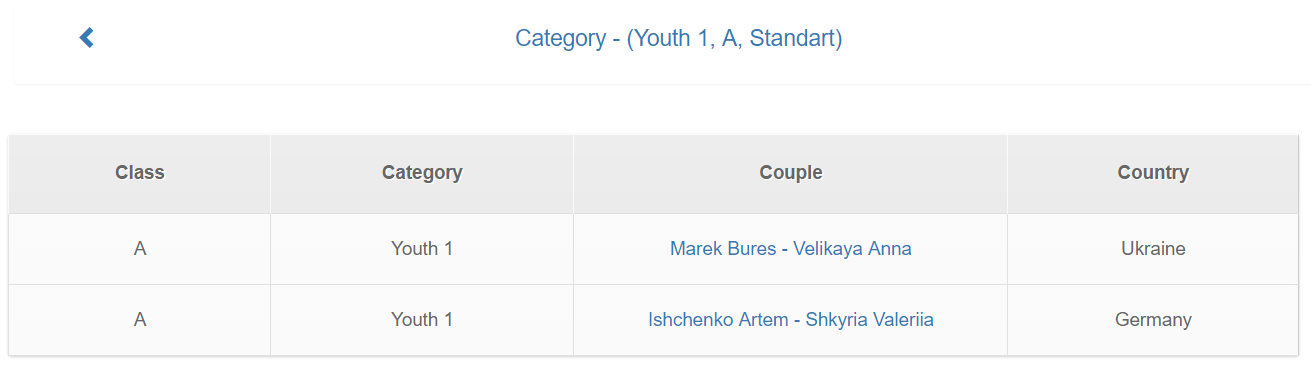


Рис. 5 – Одна категорія

При реєстрації потрібно вибрати, ящо людина є в базі даних то потрібно перейти в розділ In Data Base, якщо ні, то в розділ Not in Data Base, за адресою ‘/competition/:id\_competition/registration’ Рис. 6.

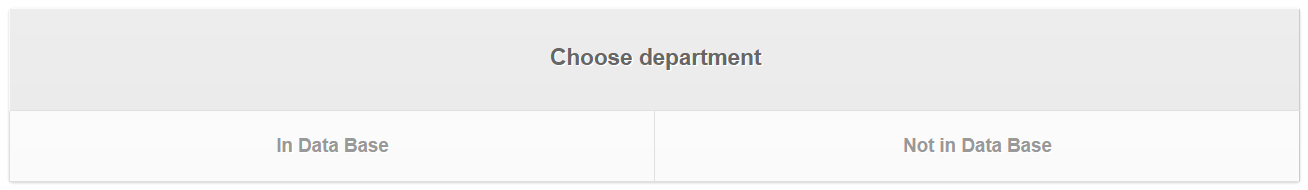


Рис. 6 – Одна категорія

Якщо всі учасники зареєстровані в базі даннх, то можна скласти пару Рис. 7.

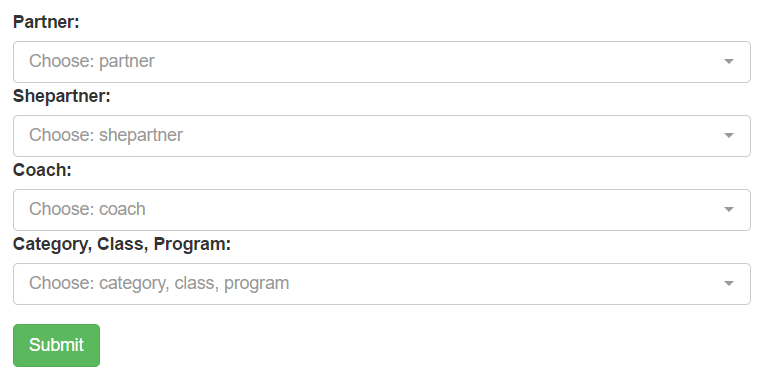


Рис. 7. – Реєстрація пари

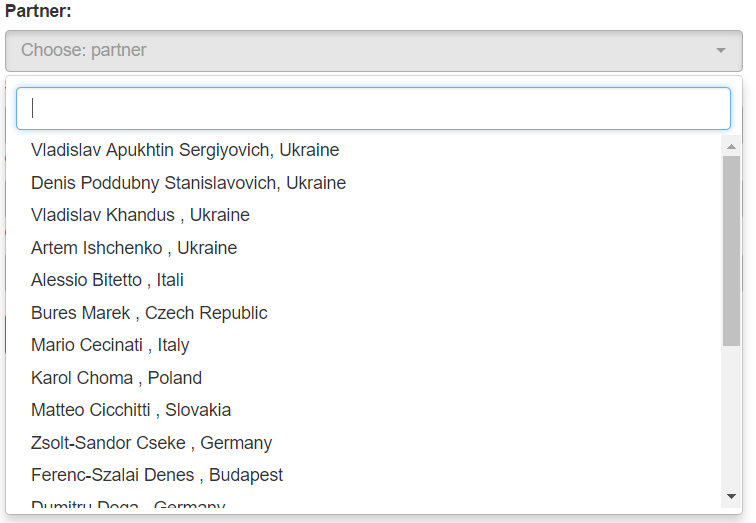


Рис. 8. – Реєстрація пари, вибір з бази даних партнера

При успішній реєстрації з’являється вікно підтвердження реєстрації Рис. 9.



Рис. 9. – Успішна реєстрація

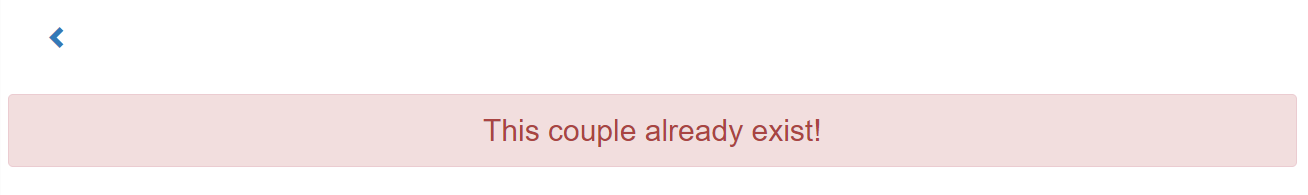


Рис. 10. – Невдала реєстрація

Якщо людини немає в базі даних, то її можна зареєструвати у відповідному підрозділі Рис. 11.

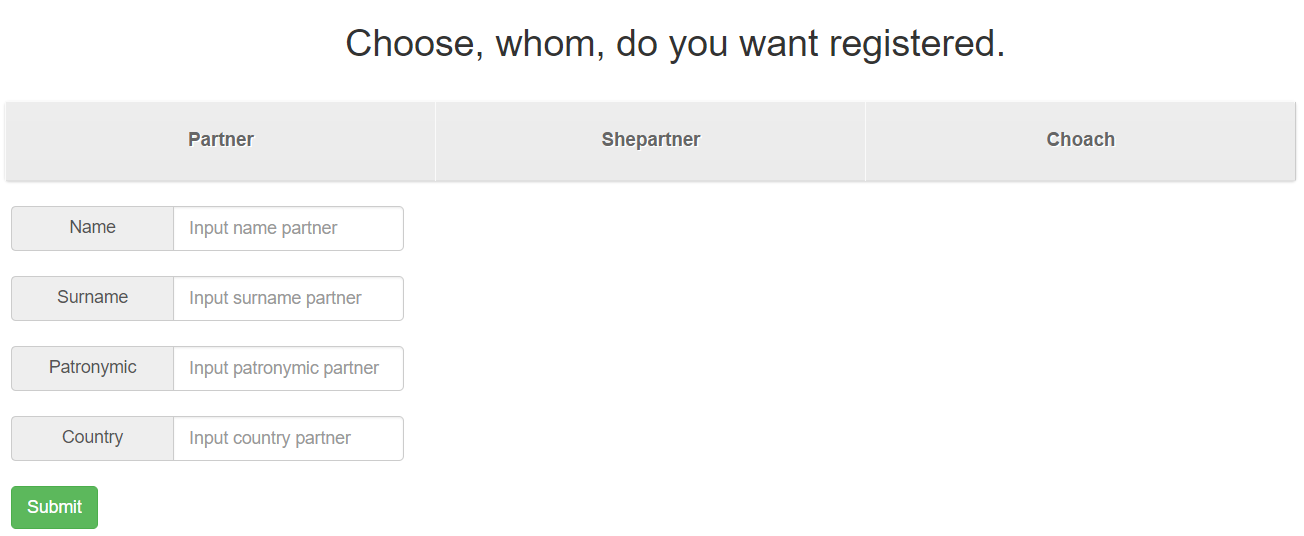


Рис. 11. – Реєстрація в підрозділі Partners

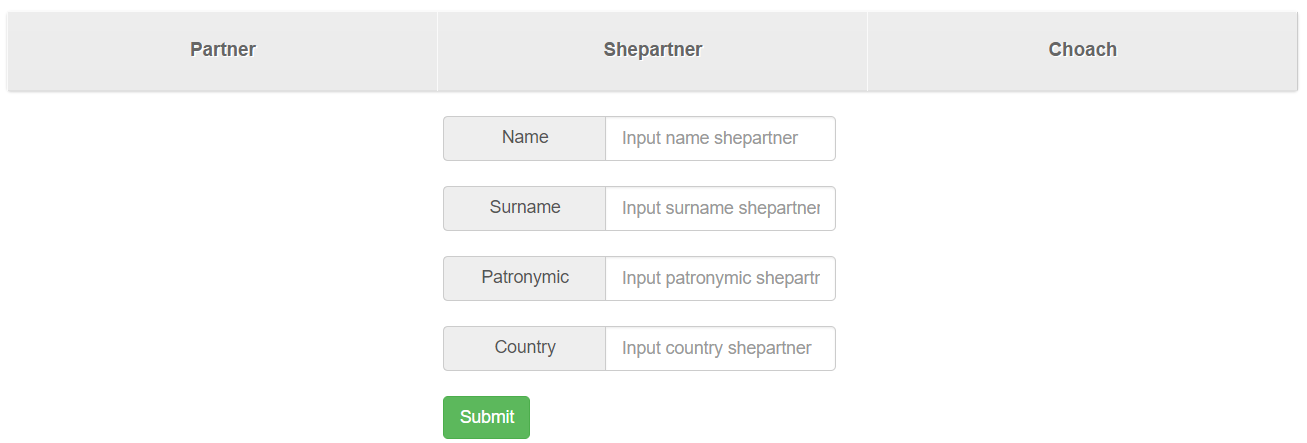
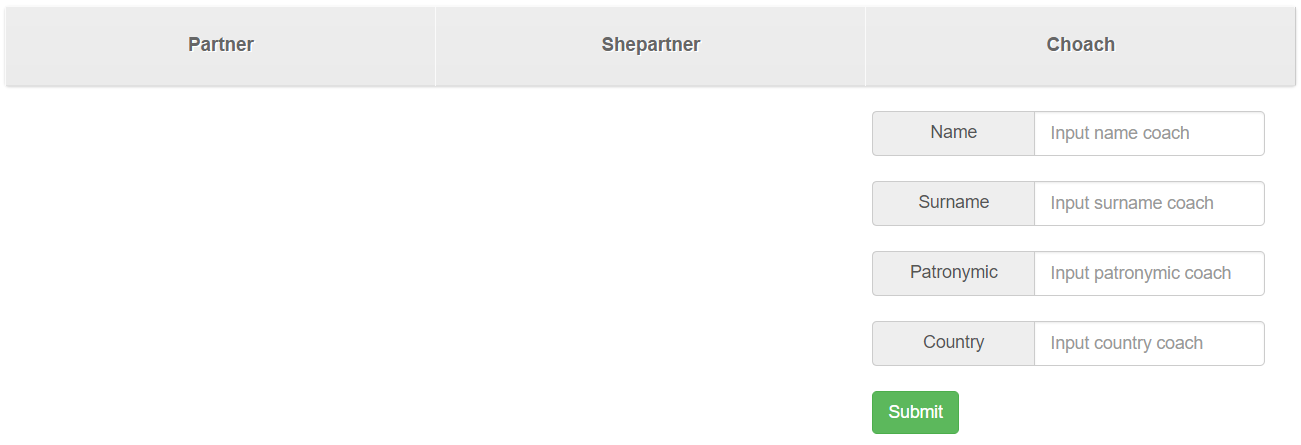


Рис. 12. – Реєстрація в підрозділі Shepartners

 Рис. 13. – Реєстрація в підрозділі Coaches

# ВИСНОВКИ

Виходячи з описуваного вище процесу проектування і побудови бази даних, а також основних цілей проектування баз даних і постановки задачі можна зробити такі висновки: була спроектована і створена база даних для міжнародних конкурсів спортивних бальних танців, створена програма, що забезпечує роботу користувача з базою даних (перегляд, редагування інформації в базі даних і здійснення пошуку).

Виконана робота дозволила набути навичок розробки баз даних від поставленого завдання до кінцевого програмного продукту, придатного до використання. В ході роботи розглянуті і засвоєні навички використання мови SQL для роботи з базами даних, отриманий досвід організації баз даних в СКБД Firebird, зокрема використання запитів. Отриманий досвід програмування на мові JavaScript та досвід роботи з платформою node.JS, та фреймворками node-firebird, Express, та шаблонізатором Jade.

Даний програмний продукт можна використовувати для проведення міжнародних конкурсів танців.

# СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Дейт. Введение в системы баз данных. – М.: Наука , 1980.
2. Конноли, Бегг. Базы даннях: проектирование, реализация и сопровождение. Теория и практика, 3-е изд.: Пер. с англ..: Уч. пос. – М.: Издательский дом “Вильямс”, 2003. – 1440 с.
3. Дж.Ульман Основі систем баз данных. М., Финансы и статистика, 1983. Стэнфордский университет, 1980.
4. Д. Крёнке. Теория и практика построения баз данных. 8-е изд. – СПб.: Питер, 2003. –800с.:ил. – (Серия «Классика computer science»).
5. Роб П., Коронел К. Системы баз данных: проектирование, реализация и управление. – 5-е изд., перераб. и доп.: Пер. с англ. – СПб.: БХВ-Петербург, 2004. – 1040 с: ил.
6. Грабер М. Введение в SQL. – М.: Лори, 1996.
7. Тиори Т., Фрай Дж. Проектирование структур баз данных: В 2-х кн. – М.: Мир, 1985. – Кн. 1. – 287 с.; Кн. 2. – 320 с.

# ДОДАТОК Г

**Скріпт БД :**

CONNECT 'D:\Pars\3 course\2 semester\DataBase\Coursework\git\_Coursework\sql\new\_db\coursework\_ver3.FDB'

USER 'SYSDBA' PASSWORD 'masterkey';

DROP DATABASE;

CREATE DATABASE 'D:\Pars\3 course\2 semester\DataBase\Coursework\git\_Coursework\sql\new\_db\coursework\_ver3.FDB'

USER 'SYSDBA' PASSWORD 'masterkey';

/\*==============================================================\*/

/\* Table: Adjudicators \*/

/\*==============================================================\*/

create table Adjudicators

(

nameAdjudicator VARCHAR(32) not null,

surnameAdjudicator VARCHAR(32) not null,

patronymicAdjudicator VARCHAR(32),

codeAdjudicator INTEGER not null,

Country VARCHAR(32),

primary key (surnameAdjudicator, nameAdjudicator, codeAdjudicator)

);

CREATE GENERATOR GENERATOR\_Adjudicators;

SET TERM ^ ;

CREATE TRIGGER trigger\_Adjudicators FOR Adjudicators ACTIVE

BEFORE INSERT POSITION 1

AS

BEGIN

if (new.codeAdjudicator is null ) then

new.codeAdjudicator = gen\_id (GENERATOR\_Adjudicators, 1);

END^

SET TERM ; ^

/\*==============================================================\*/

/\* Table: BallroomPrograms \*/

/\*==============================================================\*/

create table BallroomPrograms

(

IdCompetition INTEGER not null,

IdProgram INTEGER not null,

typeOfProgram VARCHAR(32),

primary key (IdCompetition, IdProgram)

);

/\*==============================================================\*/

/\* Table: Categories \*/

/\*==============================================================\*/

create table Categories

(

IdCompetition INTEGER not null,

IdProgram INTEGER not null,

CategoryID INTEGER not null,

CategoryName VARCHAR(32),

primary key (IdCompetition, IdProgram, CategoryID)

);

/\*==============================================================\*/

/\* Table: Classes \*/

/\*==============================================================\*/

create table Classes

(

ClassID INTEGER not null,

ClassName VARCHAR(32),

primary key (ClassID)

);

CREATE GENERATOR GENERATOR\_Classes;

SET TERM ^ ;

CREATE TRIGGER trigger\_Classes FOR Classes ACTIVE

BEFORE INSERT POSITION 1

AS

BEGIN

if (new.ClassID is null ) then

new.ClassID = gen\_id (GENERATOR\_Classes, 1);

END^

SET TERM ; ^

/\*==============================================================\*/

/\* Table: Coaches \*/

/\*==============================================================\*/

create table Coaches

(

Country VARCHAR(32),

nameCoach VARCHAR(32) not null,

surnameCoach VARCHAR(32) not null,

patronymicCoaches VARCHAR(32),

codeCoach INTEGER not null,

primary key (surnameCoach, nameCoach, codeCoach)

);

CREATE GENERATOR GENERATOR\_Coaches;

SET TERM ^ ;

CREATE TRIGGER trigger\_Coaches FOR Coaches ACTIVE

BEFORE INSERT POSITION 1

AS

BEGIN

if (new.codeCoach is null ) then

new.codeCoach = gen\_id (GENERATOR\_Coaches, 1);

END^

SET TERM ; ^

/\*==============================================================\*/

/\* Table: Competitions \*/

/\*==============================================================\*/

create table Competitions

(

IdCompetition INTEGER not null,

Title VARCHAR(32) not null,

DateCompetition VARCHAR(32) not null,

Place VARCHAR(32) not null,

Rules VARCHAR(32),

Organizers VARCHAR(32),

Country VARCHAR(32),

primary key (IdCompetition)

);

CREATE GENERATOR GENERATOR\_Competitions;

SET TERM ^ ;

CREATE TRIGGER trigger\_Competitions FOR Competitions ACTIVE

BEFORE INSERT POSITION 1

AS

BEGIN

if (new.IdCompetition is null ) then

new.IdCompetition = gen\_id (GENERATOR\_Competitions, 1);

END^

SET TERM ; ^

/\*==============================================================\*/

/\* Table: Couples \*/

/\*==============================================================\*/

create table Couples

(

ClassID INTEGER not null,

IdCompetition INTEGER not null,

IdProgram INTEGER not null,

CategoryID INTEGER not null,

PairNumber INTEGER not null,

codePartner INTEGER not null,

namePartner VARCHAR(32) not null,

surnamePartner VARCHAR(32) not null,

surnameShepartner VARCHAR(32) not null,

nameShepartner VARCHAR(32) not null,

codeShePartner INTEGER not null,

surnameCoach VARCHAR(32),

nameCoach VARCHAR(32),

codeCoach INTEGER,

primary key (ClassID, IdCompetition, IdProgram, CategoryID, PairNumber)

);

/\*==============================================================\*/

/\* Table: Partners \*/

/\*==============================================================\*/

create table Partners

(

Country VARCHAR(32),

namePartner VARCHAR(32) not null,

surnamePartner VARCHAR(32) not null,

patronymic VARCHAR(32),

codePartner INTEGER not null,

primary key (codePartner, namePartner, surnamePartner)

);

CREATE GENERATOR GENERATOR\_Partners;

SET TERM ^ ;

CREATE TRIGGER trigger\_Partners FOR Partners ACTIVE

BEFORE INSERT POSITION 1

AS

BEGIN

if (new.codePartner is null ) then

new.codePartner = gen\_id (GENERATOR\_Partners, 1);

END^

SET TERM ; ^

/\*==============================================================\*/

/\* Table: Shepartners \*/

/\*==============================================================\*/

create table Shepartners

(

nameShepartner VARCHAR(32) not null,

surnameShepartner VARCHAR(32) not null,

patronymic VARCHAR(32),

codeShePartner INTEGER not null,

Country VARCHAR(32),

primary key (surnameShepartner, nameShepartner, codeShePartner)

);

CREATE GENERATOR GENERATOR\_Shepartners;

SET TERM ^ ;

CREATE TRIGGER trigger\_Shepartners FOR Shepartners ACTIVE

BEFORE INSERT POSITION 1

AS

BEGIN

if (new.codeShePartner is null ) then

new.codeShePartner = gen\_id (GENERATOR\_Shepartners, 1);

END^

SET TERM ; ^

/\*==============================================================\*/

/\* Table: consistClass \*/

/\*==============================================================\*/

create table consistClass

(

IdCompetition INTEGER not null,

IdProgram INTEGER not null,

CategoryID INTEGER not null,

ClassID INTEGER not null,

primary key (IdCompetition, IdProgram, CategoryID, ClassID)

);

/\*==============================================================\*/

/\* Table: judge \*/

/\*==============================================================\*/

create table judge

(

IdCompetition INTEGER not null,

surnameAdjudicator VARCHAR(32) not null,

nameAdjudicator VARCHAR(32) not null,

codeAdjudicator INTEGER not null,

primary key (IdCompetition, surnameAdjudicator, nameAdjudicator, codeAdjudicator)

);

alter table BallroomPrograms add constraint FK\_consist foreign key (IdCompetition)

references Competitions (IdCompetition);

alter table Categories add constraint FK\_consistCategories foreign key (IdCompetition, IdProgram)

references BallroomPrograms (IdCompetition, IdProgram);

alter table Couples add constraint FK\_consistPartnerIn foreign key (codePartner, namePartner, surnamePartner)

references Partners (codePartner, namePartner, surnamePartner);

alter table Couples add constraint FK\_consistShepartnerIn foreign key (surnameShepartner, nameShepartner, codeShePartner)

references Shepartners (surnameShepartner, nameShepartner, codeShePartner);

alter table Couples add constraint FK\_haveClass foreign key (ClassID)

references Classes (ClassID);

alter table Couples add constraint FK\_haveCoache foreign key (surnameCoach, nameCoach, codeCoach)

references Coaches (surnameCoach, nameCoach, codeCoach);

alter table Couples add constraint FK\_partIn foreign key (IdCompetition, IdProgram, CategoryID)

references Categories (IdCompetition, IdProgram, CategoryID);

alter table consistClass add constraint FK\_consistClass foreign key (IdCompetition, IdProgram, CategoryID)

references Categories (IdCompetition, IdProgram, CategoryID);

alter table consistClass add constraint FK\_consistClass2 foreign key (ClassID)

references Classes (ClassID);

alter table judge add constraint FK\_judge foreign key (IdCompetition)

references Competitions (IdCompetition);

alter table judge add constraint FK\_judge2 foreign key (surnameAdjudicator, nameAdjudicator, codeAdjudicator)

references Adjudicators (surnameAdjudicator, nameAdjudicator, codeAdjudicator);

/\*==============================================================\*/

/\* Insert: First competition \*/

/\*==============================================================\*/

INSERT INTO Competitions (Country, Title, DateCompetition, Place, Rules, Organizers)

VALUES ('Ukraine', 'Kyiv Open', '11.04.2017', 'Kyiv', 'WDS', 'Sergiy Gritsak')

RETURNING idCompetition, Country, Title, DateCompetition, Place, Rules, Organizers;

INSERT INTO BallroomPrograms (IdCompetition, IdProgram, typeOfProgram)

VALUES (1, 1, 'Total') RETURNING IdCompetition, idProgram, typeOfProgram;

INSERT INTO BallroomPrograms (IdCompetition, IdProgram, typeOfProgram)

VALUES (1, 2, 'Standart') RETURNING IdCompetition, idProgram, typeOfProgram;

INSERT INTO BallroomPrograms (IdCompetition, IdProgram, typeOfProgram)

VALUES(1, 3, 'Latin') RETURNING IdCompetition, idProgram, typeOfProgram;

-- --total

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (1, 1, 1, 'Junior 1');

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (1, 1, 2, 'Junior 2');

--standart

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (1, 2, 3, 'Youth 1');

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (1, 2, 4, 'Youth 2');

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (1, 2, 5, 'Adult');

--latin

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (1, 3, 3, 'Youth 1');

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (1, 3, 4, 'Youth 2');

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (1, 3, 5, 'Adult');

INSERT INTO Classes (ClassName)

VALUES ('C');

INSERT INTO Classes (ClassName)

VALUES ('B');

INSERT INTO Classes (ClassName)

VALUES ('A');

-- idcompetition, total, junior1, C

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 1, 1, 1);

-- idcompetition, total, junior2, C

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 1, 2, 1);

-- idcompetition, standart, youth1, C,B,A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 2, 3, 1);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 2, 3, 2);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 2, 3, 3);

-- idcompetition, latin, youth1, C,B,A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 3, 3, 1);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 3, 3, 2);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 3, 3, 3);

-- idcompetition, standart, youth2, C,B,A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 2, 4, 1);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 2, 4, 2);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 2, 4, 3);

-- idcompetition, latin, youth2, C,B,A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 3, 4, 1);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 3, 4, 2);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 3, 4, 3);

-- idcompetition, standart, adult, A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 2, 5, 3);

-- idcompetition, latin, adult, A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (1, 3, 5, 3);

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Dikyi', 'Igor', null, 'Ukraine');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Pasazkaya', 'Polina', null, 'Ukraine');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Frolova', 'Vera', null, 'Ukraine');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Frolov', 'Valentin', null, 'Ukraine');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Boev', 'Igor', null, 'Hong Kong');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Prohorenko', 'Yura', null, 'Ukraine');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Golovashenko', 'Igor', null, 'Ukraine');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Tkachenko', 'Artem', null, 'Ukraine');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Tkachenko', 'Anna', null, 'Ukraine');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Fujii', 'Masaaki', null, 'Japan');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Horacek', 'Petr', null, 'Slovakia');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Juul', 'Kevin', null, 'Republic of South Africa');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Sandor', 'Zoltan', null, 'Hungary');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Rusu', 'Alexandru', null, 'Romania');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Bodini', 'Luigi', null, 'Itali');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Milicija', 'Sergej', null, 'Bosna i Hercegovina');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Novak', 'Daniela', null, 'Slovenia');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Ashrafov', 'Azar', null, 'Azerbaijan');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Smirnova', 'Galina', null, 'Poland');

INSERT INTO Coaches (surnameCoach, nameCoach, patronymicCoaches, Country)

VALUES ('Kavzinadze', 'Nino', null, 'Georgia');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Dikyi', 'Igor', null, 'Ukraine');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Pasazkaya', 'Polina', null, 'Ukraine');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Frolova', 'Vera', null, 'Ukraine');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Frolov', 'Valentin', null, 'Ukraine');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Boev', 'Igor', null, 'Hong Kong');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Prohorenko', 'Yura', null, 'Ukraine');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Golovashenko', 'Igor', null, 'Ukraine');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Tkachenko', 'Artem', null, 'Ukraine');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Tkachenko', 'Anna', null, 'Ukraine');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Fujii', 'Masaaki', null, 'Japan');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Horacek', 'Petr', null, 'Slovakia');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Juul', 'Kevin', null, 'Republic of South Africa');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Sandor', 'Zoltan', null, 'Hungary');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Rusu', 'Alexandru', null, 'Romania');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Bodini', 'Luigi', null, 'Itali');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Milicija', 'Sergej', null, 'Bosna i Hercegovina');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Novak', 'Daniela', null, 'Slovenia');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Ashrafov', 'Azar', null, 'Azerbaijan');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Smirnova', 'Galina', null, 'Poland');

INSERT INTO Adjudicators (surnameAdjudicator, nameAdjudicator, patronymicAdjudicator, Country)

VALUES ('Kavzinadze', 'Nino', null, 'Georgia');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 1, 'Dikyi', 'Igor');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 2, 'Pasazkaya', 'Polina');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 3, 'Frolova', 'Vera');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 4, 'Frolov', 'Valentin');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 5, 'Boev', 'Igor');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 6, 'Prohorenko', 'Yura');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 7, 'Golovashenko', 'Igor');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 8, 'Tkachenko', 'Artem');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 9, 'Tkachenko', 'Anna');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 10, 'Fujii', 'Masaaki');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 11, 'Horacek', 'Petr');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 12, 'Juul', 'Kevin');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 13, 'Sandor', 'Zoltan');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 14, 'Rusu', 'Alexandru');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 15, 'Bodini', 'Luigi');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 16, 'Milicija', 'Sergej');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 17, 'Novak', 'Daniela');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 18, 'Ashrafov', 'Azar');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 19, 'Smirnova', 'Galina');

INSERT INTO judge (IdCompetition, codeAdjudicator, surnameAdjudicator, nameAdjudicator)

VALUES (1, 20, 'Kavzinadze', 'Nino');

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Apukhtin', 'Vladislav', 'Ukraine', 'Sergiyovich');

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Poddubny', 'Denis', 'Ukraine', 'Stanislavovich');

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Khandus', 'Vladislav', 'Ukraine', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Ishchenko', 'Artem', 'Ukraine', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Bitetto', 'Alessio', 'Itali', null);

INSERT INTO Partners (namePartner, surnamePartner, Country, patronymic)

VALUES ('Bures', 'Marek', 'Czech Republic', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Cecinati', 'Mario', 'Italy', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Choma', 'Karol', 'Poland', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Cicchitti', 'Matteo', 'Slovakia', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Cseke', 'Zsolt-Sandor', 'Germany', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Denes', 'Ferenc-Szalai', 'Budapest', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Doga', 'Dumitru', 'Germany', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Esposito', 'Francesco', 'Itali', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Fabik', 'Anton', 'Slovakia', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Kasprzak', 'Konstanty', 'Poland', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Khodenko', 'Alexandr', 'Russia', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Maidaniuk', 'Mihail', 'Belorus', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Mugerman', 'Alon', 'Israel', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Rednic', 'Paul', 'Romania', null);

INSERT INTO Partners (surnamePartner, namePartner, Country, patronymic)

VALUES ('Szkutnik', 'Bartlomiej', 'Poland', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Teterina', 'Anastasiia', 'Ukraine', 'Alexeyevna');

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Pack', 'Anastasiia', 'Ukraine', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Velikaya', 'Anna', 'Ukraine', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Glushchenko', 'Lena', 'Ukraine', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Dabramo', 'Annamaria', 'Itali', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Iermolenko', 'Anastasiia', 'Czech Republic', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Denaro', 'Rosaria\_Messina', 'Italy', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Dabek', 'Paulina', 'Poland', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Brecikova', 'Simona', 'Slovakia', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Dzumaev', 'Malika', 'Germany', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Petra', 'Balla', 'Budapest', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Shkyria', 'Valeriia', 'Germany', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Ertmer', 'Sarah', 'Itali', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Kostenko', 'Svetlana', 'Slovakia', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Vargova', 'Iveta', 'Poland', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Zanlorenzi', 'Giulia', 'Russia', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Karcagi', 'Laura', 'Belorus', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Vasickova', 'Gabriela', 'Israel', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Shalaeva', 'Ekaterina', 'Romania', null);

INSERT INTO Shepartners (surnameShepartner, nameShepartner, Country, patronymic)

VALUES ('Ekaterina', 'Mishina', 'Poland', null);

-- --Junior1 total C

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 1, 1, 1, 6, 'Prohorenko', 'Yura', 6, 'Cecinati', 'Mario', 7, 'Iermolenko', 'Anastasiia', 6);

-- --total Junior1 C

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 1, 2, 1, 6, 'Prohorenko', 'Yura', 6, 'Cecinati', 'Mario', 7, 'Iermolenko', 'Anastasiia', 6);

--st Youth2 B

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 2, 4, 2, 1, 'Dikyi', 'Igor', 1, 'Apukhtin', 'Vladislav', 1, 'Teterina', 'Anastasiia', 1);

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 2, 4, 2, 2, 'Pasazkaya', 'Polina', 2, 'Poddubny', 'Denis', 2, 'Pack', 'Anastasiia', 2);

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 2, 4, 2, 3, 'Frolova', 'Vera', 3, 'Ishchenko', 'Artem', 4, 'Velikaya', 'Anna', 3);

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 2, 4, 2, 4, 'Frolov', 'Valentin', 4, 'Khandus', 'Vladislav', 3, 'Glushchenko', 'Lena', 4);

--st Youth2 A

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 2, 4, 3, 5, 'Boev', 'Igor', 5, 'Bitetto', 'Alessio', 5, 'Dabramo', 'Annamaria', 5);

--lat Youth2 A

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 3, 4, 3, 5, 'Boev', 'Igor', 5, 'Bitetto', 'Alessio', 5, 'Dabramo', 'Annamaria', 5);

--lat Youth2 B

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 3, 4, 2, 1, 'Dikyi', 'Igor', 1, 'Apukhtin', 'Vladislav', 1, 'Teterina', 'Anastasiia', 1);

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 3, 4, 2, 2, 'Pasazkaya', 'Polina', 2, 'Poddubny', 'Denis', 2, 'Pack', 'Anastasiia', 2);

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 3, 4, 2, 3, 'Frolova', 'Vera', 3, 'Ishchenko', 'Artem', 4, 'Velikaya', 'Anna', 3);

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 3, 4, 2, 4, 'Frolov', 'Valentin', 4, 'Khandus', 'Vladislav', 3, 'Glushchenko', 'Lena', 4);

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 3, 4, 2, 5, 'Boev', 'Igor', 5, 'Bitetto', 'Alessio', 5, 'Dabramo', 'Annamaria', 5);

--st Adult A

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 2, 5, 3, 5, 'Boev', 'Igor', 5, 'Bitetto', 'Alessio', 5, 'Dabramo', 'Annamaria', 5);

--lat Adult A

INSERT INTO Couples (IdCompetition, IdProgram, CategoryID, ClassID, PairNumber, surnameCoach,

nameCoach, codeCoach, surnamePartner, namePartner, codePartner, surnameShepartner,

nameShepartner, codeShePartner)

VALUES (1, 3, 5, 3, 5, 'Boev', 'Igor', 5, 'Bitetto', 'Alessio', 5, 'Dabramo', 'Annamaria', 5);

/\*==============================================================\*/

/\* Insert: Second competition \*/

/\*==============================================================\*/

INSERT INTO Competitions (Country, Title, DateCompetition, Place, Rules, Organizers)

VALUES ('Russia', 'Kremlin Cup', '18.04.2017', 'Moscow', 'WDSF', 'Stanislav Popov')

RETURNING idCompetition, Country, Title, DateCompetition, Place, Rules, Organizers;

INSERT INTO BallroomPrograms (IdCompetition, IdProgram, typeOfProgram)

VALUES (2, 1, 'Total') RETURNING IdCompetition, idProgram, typeOfProgram;

INSERT INTO BallroomPrograms (IdCompetition, IdProgram, typeOfProgram)

VALUES (2, 2, 'Standart') RETURNING IdCompetition, idProgram, typeOfProgram;

INSERT INTO BallroomPrograms (IdCompetition, IdProgram, typeOfProgram)

VALUES(2, 3, 'Latin') RETURNING IdCompetition, idProgram, typeOfProgram;

-- --total

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (2, 1, 1, 'Junior 1');

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (2, 1, 2, 'Junior 2');

--standart

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (2, 2, 3, 'Youth 1');

--latin

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (2, 3, 3, 'Youth 1');

-- idcompetition, total, junior1, C

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (2, 1, 1, 1);

-- idcompetition, total, junior2, C

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (2, 1, 2, 1);

-- idcompetition, standart, youth1, C,B,A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (2, 2, 3, 1);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (2, 2, 3, 2);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (2, 2, 3, 3);

-- idcompetition, latin, youth1, C,B,A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (2, 3, 3, 1);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (2, 3, 3, 2);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (2, 3, 3, 3);

/\*==============================================================\*/

/\* Insert: Third competition \*/

/\*==============================================================\*/

INSERT INTO Competitions (IdCompetition, Country, Title, DateCompetition, Place, Rules, Organizers)

VALUES (3, 'Belarus', 'Open Championship', '18.04.2017', 'Minsk', 'IDSF', 'Belarusian Dance Federation')

RETURNING idCompetition, Country, Title, DateCompetition, Place, Rules, Organizers;

INSERT INTO BallroomPrograms (IdCompetition, IdProgram, typeOfProgram)

VALUES (3, 1, 'Total') RETURNING IdCompetition, idProgram, typeOfProgram;

INSERT INTO BallroomPrograms (IdCompetition, IdProgram, typeOfProgram)

VALUES (3, 2, 'Standart') RETURNING IdCompetition, idProgram, typeOfProgram;

INSERT INTO BallroomPrograms (IdCompetition, IdProgram, typeOfProgram)

VALUES(3, 3, 'Latin') RETURNING IdCompetition, idProgram, typeOfProgram;

--standart

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (3, 2, 3, 'Youth 1');

--latin

INSERT INTO Categories (IdCompetition, IdProgram, CategoryID, CategoryName)

VALUES (3, 3, 3, 'Youth 1');

-- idcompetition, standart, youth1, C,B,A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (3, 2, 3, 1);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (3, 2, 3, 2);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (3, 2, 3, 3);

-- idcompetition, latin, youth1, C,B,A

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (3, 3, 3, 1);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (3, 3, 3, 2);

INSERT INTO consistClass (IdCompetition, IdProgram, CategoryID, ClassID)

VALUES (3, 3, 3, 3);

# 

# ДОДАТОК А

**Концептуальна модель:**

****

# 

# ДОДАТОК Б

**Фізична модель бази:** 

# ДОДАТОК B

**Логічна модель:**

# 