ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО» Высшая школа программной инженерии



ОТЧЕТ ПО ПРАКТИЧЕСКИМ ЗАДАНИЯМ

по дисциплине «Базы данных»

Студент 3530202/90202

Потапова А. М.

Преподаватель

Прокофьев О. В.

Санкт-Петербург 2021 г.

Hinaval

Содержание

Практическое задание №1.1 Проектирование схемы базы данных	3
Практическое задание №1.2 Создание и заполнение таблиц	4
Практическое задание №1.3 Операторы манипулирования	10
Практическое задание №1.4 Контроль целостности данных	16
Практическое задание №2.1 Управление доступом	21
Практическое задание №2.2 Функции и язык PL/pgSQL	25
Приложение (SQL dump)	29

Практическое задание №1.1 Проектирование схемы базы данных

Постановка задачи

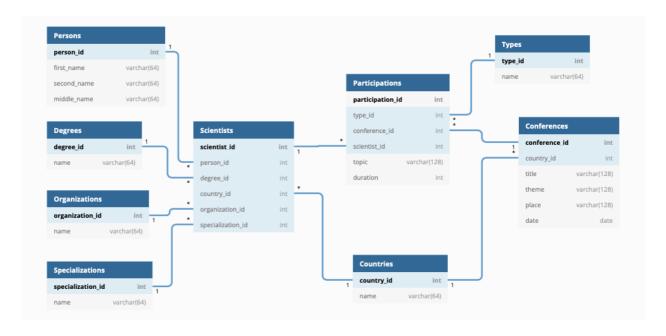
Практическое задание связано с проектированием схемы базы данных для работы приложения. Результатом данного практического задания является схема базы данных.

Вариант 22 Научная конференция



Ход работы

Следуя предметной области, была разработана следующая схема базы данных:



Практическое задание №1.2 Создание и заполнение таблиц

Постановка задачи

Практическое задание заключается в подготовке SQL-скрипта для создания таблиц согласно схеме, полученной в предыдущем задании (с уточнением типов столбцов). Необходимо определить первичные и внешние ключи, а также декларативные ограничения целостности (возможность принимать неопределенное значение, уникальные ключи, проверочные ограничения и т. д.). Таблицы следует создавать в отдельной базе данных. Кроме того, нужно подготовить данные для заполнения созданных таблиц. Объем подготовленных данных должен составлять не менее 10 экземпляров для каждой из стержневых (неподчиняющиеся основные сущности) сущностей и 15 экземпляров для каждой из ассоциативных (подчиненные многие-комногим). На основе этих данных необходимо создать SQL-скрипт для вставки соответствующих строк в таблицы БД.

Ход работы

1. Создание таблиц:

```
CREATE TABLE "Persons" (
      "person id" SERIAL PRIMARY KEY,
      "first name" varchar(64) NOT NULL,
      "second name" varchar(64) NOT NULL,
      "middle name" varchar(64)
);
CREATE TABLE "Countries" (
      "country id" SERIAL PRIMARY KEY,
      "name" varchar(64) NOT NULL,
      UNIQUE(name)
);
CREATE TABLE "Degrees" (
      "degree id" SERIAL PRIMARY KEY,
      "name" varchar(64) NOT NULL,
      UNIQUE(name)
);
CREATE TABLE "Specializations" (
      "specialization id" SERIAL PRIMARY KEY,
      "name" varchar(64) NOT NULL,
      UNIQUE(name)
);
```

```
CREATE TABLE "Organizations" (
      "organization id" SERIAL PRIMARY KEY,
      "name" varchar(128) NOT NULL,
      UNIQUE(name)
);
CREATE TABLE "Types" (
      "type_id" SERIAL PRIMARY KEY.
      "name" varchar(64) NOT NULL,
      UNIQUE(name)
);
CREATE TABLE "Scientists" (
      "scientist id" SERIAL PRIMARY KEY,
      "person id" int NOT NULL,
      "degree id" int,
      "country id" int,
      "organization id" int,
      "specialization id" int
);
CREATE TABLE "Participations" (
      "participation id" SERIAL PRIMARY KEY,
      "scientist id" int NOT NULL,
      "conference id" int NOT NULL,
      "type id" int,
      "topic" varchar(128),
      "duration" time
);
CREATE TABLE "Conferences" (
      "conference_id" SERIAL PRIMARY KEY,
      "country id" int NOT NULL,
      "title" varchar(128) NOT NULL,
      "theme" varchar(128) NOT NULL,
      "place" varchar(128),
      "date" timestamp
);
```

2. Связи таблиц:

```
ALTER TABLE "Scientists" ADD FOREIGN KEY ("person_id") REFERENCES "Persons" ("person_id") ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE "Scientists" ADD FOREIGN KEY ("degree_id") REFERENCES "Degrees" ("degree_id") ON DELETE SET NULL ON UPDATE CASCADE;

ALTER TABLE "Scientists" ADD FOREIGN KEY ("country_id") REFERENCES "Countries" ("country_id") ON DELETE SET NULL ON UPDATE CASCADE;
```

ALTER TABLE "Scientists" ADD FOREIGN KEY ("organization_id") REFERENCES "Organizations" ("organization_id") ON DELETE SET NULL ON UPDATE CASCADE;

ALTER TABLE "Scientists" ADD FOREIGN KEY ("specialization_id") REFERENCES "Specializations" ("specialization_id") ON DELETE SET NULL ON UPDATE CASCADE;

ALTER TABLE "Participations" ADD FOREIGN KEY ("scientist_id") REFERENCES "Scientists" ("scientist id") ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE "Participations" ADD FOREIGN KEY ("conference_id") REFERENCES "Conferences" ("conference_id") ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE "Participations" ADD FOREIGN KEY ("type_id") REFERENCES "Types" ("type_id") ON DELETE SET NULL ON UPDATE CASCADE;

ALTER TABLE "Conferences" ADD FOREIGN KEY ("country_id") REFERENCES "Countries" ("country_id") ON DELETE CASCADE ON UPDATE CASCADE;

3. Заполнение таблиц:

```
INSERT INTO "Countries" (name) VALUES
  ('Россия'),
  ('Франция'),
  ('Китай'),
  ('США'),
  ('Германия'),
  ('Великобритания'),
  ('Япония'),
  ('Индия'),
  ('Канада'),
  ('Бразилия'),
  ('Италия'),
  ('Испания'),
  ('Австралия'),
  ('Швейцария'),
  ('Израиль')
INSERT INTO "Persons" (first name, second name, middle name) VALUES
  ('Виктор', 'Попов', 'Сергеевич'),
  ('Джон', 'Гилберт', NULL),
  ('Кур','Дина', NULL),
  ('Луи','Де ла Круа', NULL),
  ('Владислав', 'Карташов', 'Константинович'),
  ('Виктория', 'Гукова', 'Олеговна'),
  ('Кристиан', 'Кане', NULL),
  ('Елена', 'Вальверде', NULL),
```

```
('Стефано', 'Волонте', NULL),
  ('Робберт', 'Эллис', NULL),
  ('Джуди','Далтон', NULL),
  ('Глория', 'Антонелли', NULL),
  ('Бруно', 'Перес', NULL),
  ('Рю', 'Ким', NULL),
  ('Ритик', 'Хан', NULL)
INSERT INTO "Degrees" (name) VALUES
  ('Доктор архитектурных наук'),
  ('Доктор биологических наук'),
  ('Доктор географических наук'),
  ('Доктор филологических наук'),
  ('Доктор технических наук'),
  ('Доктор медицинских наук'),
  ('Доктор физико-математических наук'),
  ('Доктор политологических наук'),
  ('Кандидат архитектурных наук'),
  ('Кандидат биологических наук'),
  ('Кандидат географических наук'),
  ('Кандидат филологических наук'),
  ('Кандидат технических наук'),
  ('Кандидат медицинских наук'),
  ('Кандидат физико-математических наук')
INSERT INTO "Specializations" (name) VALUES
  ('Градостроительство'),
  ('Политтехнолог'),
  ('Биохимия'),
  ('Системная аналитика'),
  ('Терапевт'),
  ('Космическая физика'),
  ('Зарубежная филология'),
  ('Информационная безопасность'),
  ('Реставрация архитектурных объектов'),
  ('Научная и преподавательская деятельность'),
  ('Аллергология и иммунология'),
  ('Молекулярная биология'),
  ('Биоинженерия'),
  ('Математические методы в современной физике'),
  ('Промышленная разработка')
INSERT INTO "Organizations" (name) VALUES
  ('Федерация европейских биохимических обществ'),
  ('Ученые будущего'),
  ('Яндекс'),
  ('Р-Фарм'),
  ('Группа 77'),
  ('Филологический факультет МГУ имени М.В.Ломоносова'),
  ('Международный математический союз'),
  ('Международная ассоциация академий наук'),
  ('Европейская ассоциация географов'),
```

```
('Международный исследовательский центр по сохранению и реставрации
  культурных ценностей'),
  ('Ассоциация вычислительной техники'),
  ('Институт высших научных исследований'),
  ('Международная академия астронавтики'),
  ('Международный совет по науке'),
  ('Союз социальной политики')
INSERT INTO "Types" (name) VALUES
  ('Участие с докладом'),
  ('Участие с публикацией, но без доклада'),
  ('Участие без публикации и доклада')
INSERT INTO "Scientists" (person id, degree id, country id, organization id,
specialization id) VALUES
  ('1','7','1','13','6'),
  ('2','8','4','15','2'),
  ('3','6','15','1','3'),
  ('4','1','2','9','10'),
  ('5','5','1','2','4'),
  ('6','4','1','6','10'),
  ('10','14','6','1','3'),
  ('11','10','13','1','11'),
  ('12','13','10','11','4'),
  ('14','15','3','14','14'),
  ('7','5','2','12','8'),
  ('8','10','12','1','13'),
  ('9','3','11','9','10'),
  ('13','9','14','10','1'),
  ('15','1','7','10','9')
INSERT INTO "Conferences" (country id, title, theme, place, date) VALUES
  ('1','Международная научно-практическая конференция','Advances in Science and
  Technology', 'MΓУ', '2022-05-16 14:00:00'),
  ('1','Международная междисциплинарная конференция','Развитие науки в XXI
  веке', 'Научно-издательский центр «Толмачево»', '2022-06-01 12:00:00'),
  ('6','4-я Международная конференция','Информационные технологии и
  безопасные трансакции', 'Университет Ист-Лондона', '2022-04-25 15:00:00'),
  ('4','10-я Международная научно-практическая конференция','Вопросы
  трансформации образования', 'SCOPUS', '2022-08-03 10:00:00'),
  ('4','10-я Ежегодная международная конференция','Наука и
  технологии', 'SCOPUS', '2022-05-30 13:30:00'),
  ('8','VII Международная научно-практическая конференция','Экономические и
  социальные тренды устойчивого развития современного
  общества', 'ГУАП', '2022-05-30 13:30:00'),
  ('1','10-я Ежегодная международная конференция','AGRITECH-VII – 2022:
  Агротехнологии, экологический инжиниринг и устойчивое
  развитие', 'Красноярский краевой Дом науки и техники', '2022-07-30 10:00:00')
  ('1','III Всероссийская научно-практическая конференция с международным
  участием', 'Социальные и гуманитарные науки в XXI веке: итоги, вызовы,
  перспективы', 'Научно-исследовательский центр «Пересвет»', '2022-09-30
   16:00:00'),
```

('4','I Международная научно-практическая конференция-семинар','«Актуальные вопросы современного здравоохранения. Электронная медицина»','Университет им. Д. Бен гуриона','2022-10-30 12:00:00')

INSERT INTO "Participations" (scientist_id, conference_id, type_id, topic, duration) VALUES

('8','9','1','Вакцинация населения. Значение вакцинации для жизни отдельного человека и человечества в целом','00:10:00'),

('7','9','1','Микроэлементы в человеческой жизни, их основная роль','00:10:00'), ('3','9','2','Бесплодные пары: ЭКО как решение проблемы на современном этапе жизни','00:10:00'),

('12','9','3',NULL,'00:10:00'),

('13','4','1','Современное географическое образование','00:10:00'),

('4','4','1','История региональной архитектуры','00:10:00'),

('6','4','1','Языковое сознание: формирование и функционирование','00:10:00'),

('1','2','1','Перспективы освоения космического пространства','00:10:00'),

('5','2','2','Средства защиты отчуждаемых носителей информации','00:10:00'),

('9','1','1','Методики системного анализа и бизнес-аналитики','00:10:00'),

('10','1','2','Современные физические методы применяемые в онкологии','00:10:00'),

('11','3','3',NULL,'00:10:00')

('8','9','1','Анализ частоты и причин отказа родителей от проведения вакцинации','00:12:00'),

('9','1','1','Система как объект системного анализа','00:10:00'),

('5','2','1','Источники возникновения и последствия реализации угроз информационной безопасности','00:08:00'),

('5','2','1','Защита информации от несанкционированного доступа методом криптопреобразования','00:12:00'),

('5','1','1','Защита информации от несанкционированного доступа методом криптопреобразования','00:12:00')

Практическое задание №1.3 Операторы манипулирования

Постановка задачи

Практическое задание посвящено манипулированию данными с помощью операторов SQL. Нужно подготовить 3-4 выборки, которые имеют осмысленное значение для предметной области, и также составить для них SQL-скрипты. Сформулировать 3-4 запроса на изменение и удаление из базы данных. Запросы должны быть сформулированы в терминах предметной области. Составить SQL-скрипты для выполнения этих запросов.

Ход работы

- 1. Были составлены следующие выборки с соответствующими SQLскриптами:
- Список конференций в хронологическом порядке:

```
SELECT
TO_CHAR("Conferences".date, 'DD-MM-YYYY') AS date,
"Conferences".title,
"Conferences".theme,
"Conferences".place,
"Countries".name AS "country"
FROM "Conferences", "Countries" WHERE "Countries".country_id =
"Conferences".country_id
ORDER BY "Conferences".date;
```

4	date text	title character varying (128)	theme character varying (128)	place character varying (128)	country character varying (64)
1	25-04-2	4-я Международная конференция	Информационные технологии и безопасные трансакц	Университет Ист-Лондона	Великобритания
2	30-04-2	8-я Ежегодная международная конференция	Big Data Days 2022	СПБПУ	Россия
3	16-05-2	Международная научно-практическая конференция	Advances in Science and Technology	МГУ	Россия
4	16-05-2	18-я международная научная конференция	Advances in Biology	СПБГУ	Россия
5	30-05-2	10-я Ежегодная международная конференция	Наука и технологии	SCOPUS	США
6	30-05-2	VII Международная научно-практическая конферен	Экономические и социальные тренды устойчивого ра	ГУАП	Индия
7	01-06-2	Международная междисциплинарная конференция	Развитие науки в XXI веке	Научно-издательский центр «Толмачево»	Россия
8	30-07-2	10-я Ежегодная международная конференция	AGRITECH-VII – 2022: Агротехнологии, экологический и	Красноярский краевой Дом науки и техни	Россия
9	03-08-2	10-я Международная научно-практическая конфере	Вопросы трансформации образования	SCOPUS	США
10	30-09-2	III Всероссийская научно-практическая конференци	Социальные и гуманитарные науки в XXI веке: итоги, в	Научно-исследовательский центр «Пересв	Россия
11	30-10-2	I Международная научно-практическая конференц	«Актуальные вопросы современного здравоохранени	Университет им. Д. Бен гуриона	США

• Список всех ученых:

```
SELECT

"Persons".first_name,

"Persons".second_name,

"Degrees".name AS degree,

"Specializations".name AS "specialization",

"Countries".name AS "country"

FROM "Scientists"

JOIN "Persons" ON "Persons".person_id = "Scientists".person_id

JOIN "Degrees" ON "Degrees".degree_id = "Scientists".degree_id

JOIN "Countries" ON "Countries".country_id = "Scientists".country_id

JOIN "Specializations" ON "Specializations".specialization_id =

"Scientists".specialization_id
```

Результат:

4	first_name character varying	second_name character varying	degree character varying (64)	specialization character varying (64)	country character varying (64)
1	Джон	Гилберт	Доктор политологических наук	Политтехнолог	США
2	Кур	Дина	Доктор медицинских наук	Биохимия	Израиль
3	Луи	Де ла Круа	Доктор архитектурных наук	Научная и преподавательская деятельность	Франция
4	Владислав	Карташов	Доктор технических наук	Системная аналитика	Россия
5	Виктория	Гукова	Доктор филологических наук	Научная и преподавательская деятельность	Россия
6	Робберт	Эллис	Кандидат медицинских наук	Биохимия	Великобритания
7	Джуди	Далтон	Кандидат биологических наук	Аллергология и иммунология	Австралия
8	Глория	Антонелли	Кандидат технических наук	Системная аналитика	Бразилия
9	Рю	Ким	Кандидат физико-математических наук	Математические методы в современной физике	Китай
10	Кристиан	Кане	Доктор технических наук	Информационная безопасность	Франция
11	Елена	Вальверде	Кандидат биологических наук	Биоинженерия	Испания
12	Стефано	Волонте	Доктор географических наук	Научная и преподавательская деятельность	Италия
13	Ритик	Хан	Доктор архитектурных наук	Реставрация архитектурных объектов	Япония
14	Бруно	Перес	Кандидат архитектурных наук	Градостроительство	Швейцария
15	Петр	Петров	Доктор физико-математических наук	Информационная безопасность	Россия
16	Виктор	Попов	Доктор физико-математических наук	Космическая физика	Россия

• Список участников выбранной научной конференций:

```
SELECT "Persons".first_name,

"Persons".second_name,

"Types".name AS Type,

"Participations".topic,

TO_CHAR("Participations".duration, 'MI:SS')

FROM "Participations"

JOIN "Conferences" ON "Conferences".conference_id = "Participations".conference_id

AND "Conferences".conference_id = 9

JOIN "Scientists" ON "Scientists".scientist_id = "Participations".scientist_id

JOIN "Persons" ON "Persons".person_id = "Scientists".person_id

JOIN "Types" ON "Types".type_id = "Participations".type_id
```

Результат:

4		second_name	type character varying (64)	topic character varying (128)	to_char_text
1	Джуди	Далтон	Участие с докладом	Вакцинация населения. Значение вакцинации для жизни	10:00
2	Робберт	Эллис	Участие с докладом	Микроэлементы в человеческой жизни, их основная роль	10:00
3	Кур	Дина	Участие с публикацией, но без доклада	Бесплодные пары: ЭКО как решение проблемы на соврем	10:00
4	Елена	Вальверде	Участие без публикации и доклада	[null]	10:00
5	Джуди	Далтон	Участие с докладом	Анализ частоты и причин отказа родителей от проведени	12:00

• Список научных конференций с количеством участников по убыванию:

```
(SELECT

"Conferences".title,

COUNT("Participations".scientist_id) as number_of_scientists

FROM "Participations"

JOIN "Conferences" ON "Conferences".conference_id = "Participations".conference_id

GROUP BY "Conferences".title ORDER BY COUNT("Participations".scientist_id) DESC)

UNION ALL

(SELECT

"Conferences".title, 0 AS "number_of_scientists"

FROM "Conferences"

WHERE NOT EXISTS (SELECT * FROM "Participations" WHERE

"Conferences".conference_id = "Participations".conference_id))
```

4	title character varying (128)	number_of_scientis bigint
1	I Международная научно-практическая конференция-семинар	5
2	Международная междисциплинарная конференция	4
3	Международная научно-практическая конференция	4
4	10-я Международная научно-практическая конференция	3
5	4-я Международная конференция	1
6	10-я Ежегодная международная конференция	0
7	VII Международная научно-практическая конференция	0
8	8-я Ежегодная международная конференция	0
9	18-я международная научная конференция	0
10	10-я Ежегодная международная конференция	0
11	III Всероссийская научно-практическая конференция с международным участием	0

• Ученые, которые докладывали в одной конференции более 1 раза:

```
SELECT

"Persons".second_name,
COUNT("Participations".conference_id),
"Conferences".title
FROM "Participations"

JOIN "Scientists" ON "Scientists".scientist_id = "Participations".scientist_id

JOIN "Persons" ON "Persons".person_id = "Scientists".person_id

JOIN "Conferences" ON "Conferences".conference_id = "Participations".conference_id

JOIN "Types" ON "Types".type_id = "Participations".type_id AND "Types".name =

'Участие с докладом'

GROUP BY "Persons".second_name,"Conferences".title

HAVING COUNT("Participations".conference_id) > 1
```

Результат:

4	second_name character varying (64)	count bigint	<u></u>	title character varying (128)
1	Антонелли		2	Международная научно-практическая конференция
2	Далтон		2	I Международная научно-практическая конференция-семинар
3	Карташов		2	Международная междисциплинарная конференция

• Фамилии ученых за этот год, которые приняли участие в 2 международных конференциях:

```
SELECT

"Persons".second_name,
    COUNT(DISTINCT "Conferences".conference_id)

FROM "Scientists"

JOIN "Persons" ON "Persons".person_id = "Scientists".person_id

JOIN "Participations" ON "Participations".scientist_id = "Scientists".scientist_id

JOIN "Conferences" ON "Conferences".conference_id = "Participations".conference_id

AND TO_CHAR("Conferences".date, 'YYYY') = '2022' AND "Conferences".title ~

'(международная|Международная)'

GROUP BY "Persons".second_name

HAVING COUNT(DISTINCT "Conferences".conference_id) = 2
```

4	second_name character varying (64)	count bigint	
1	Карташов		2

• Количество конференций по странам (оконная функция):

SELECT

"Conferences".title,

"Countries".name AS country,

COUNT(*) OVER (PARTITION BY "Conferences".country_id) AS

country_conference_number

FROM "Conferences"

JOIN "Countries" ON "Countries".country id = "Conferences".country id

Результат:

4	title character varying (128)	country character varying (64)	country_conference_number_bigint
1	Международная междисциплинарная конференция	Россия	6
2	8-я Ежегодная международная конференция	Россия	6
3	Международная научно-практическая конференция	Россия	6
4	18-я международная научная конференция	Россия	6
5	10-я Ежегодная международная конференция	Россия	6
6	III Всероссийская научно-практическая конференция с международным участием	Россия	6
7	10-я Международная научно-практическая конференция	США	3
8	10-я Ежегодная международная конференция	США	3
9	I Международная научно-практическая конференция-семинар	США	3
10	4-я Международная конференция	Великобритания	1
11	VII Международная научно-практическая конференция	Индия	1

• Нумерованный список конференций по странам по дате (оконная функция):

SELECT

ROW NUMBER() OVER (PARTITION BY "Conferences".country_id ORDER BY

"Conferences".date ASC) AS row number,

"Conferences".date,

"Countries".name AS country,

"Conferences".title

FROM "Conferences"

JOIN "Countries" ON "Countries".country_id = "Conferences".country_id

4	row_number bigint	date timestamp without time zone	country character varying	title character varying (128)
1	1	2022-04-30 12:00:00	Россия	8-я Ежегодная международная конференция
2	2	2022-05-16 14:00:00	Россия	18-я международная научная конференция
3	3	2022-05-16 14:00:00	Россия	Международная научно-практическая конференция
4	4	2022-06-01 12:00:00	Россия	Международная междисциплинарная конференция
5	5	2022-07-30 10:00:00	Россия	10-я Ежегодная международная конференция
6	6	2022-09-30 16:00:00	Россия	III Всероссийская научно-практическая конференция с международным участием

7	1	2022-05-30 13:30:00	США	10-я Ежегодная международная конференция
8	2	2022-08-03 10:00:00	США	10-я Международная научно-практическая конференция
9	3	2022-10-30 12:00:00	США	I Международная научно-практическая конференция-семинар
10	1	2022-04-25 15:00:00	Великобритания	4-я Международная конференция
11	1	2022-05-30 13:30:00	Индия	VII Международная научно-практическая конференция

2. Запросы на удаление/изменение с соответствующими SQL-скриптами:

• Изменение даты проведения научной конференции:

```
UPDATE "Conferences"
SET date = '2022-11-05 10:00:00'
WHERE "Conferences".title = 'I Международная научно-практическая конференциясеминар'
AND "Conferences".theme = '«Актуальные вопросы современного здравоохранения.
Электронная медицина»'
```

• Изменение научной степени выбранного ученого:

```
UPDATE "Scientists"
SET degree_id = (SELECT "Degrees".degree_id FROM "Degrees" WHERE
"Degrees".name = 'Кандидат архитектурных наук')
WHERE "Scientists".person_id = (SELECT "Persons".person_id FROM "Persons"
WHERE "Persons".first_name = 'Бруно' AND "Persons".second_name = 'Перес')
```

• Удаление конференции:

```
DELETE FROM "Conferences"
WHERE title = 'Tect'
AND theme = 'Big Data Days 2022'
```

Практическое задание №1.4 Контроль целостности данных

Постановка задачи

Практическое задание посвящено контролю целостности данных, который производится с помощью механизма транзакций и триггеров. Необходимо подготовить SQL-скрипты для проверки наличия аномалий (потерянных изменений, грязных чтений, неповторяющихся чтений, фантомов) при параллельном исполнении транзакций на различных уровнях изолированности SQL/92 (READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ, SERIALIZABLE). Для проверки наличия аномалий потребуются два параллельных сеанса, операторы в которых выполняются пошагово. Необходимо составить скрипт для создания триггера, а также подготовить несколько запросов для проверки и демонстрации его полезных свойств.

Ход работы

Уровни изоляции и аномалии

Уровень	Потерянные	Грязное	Неповторяемое	Фантомное	Аномалия
изоляции	изменения	чтение	чтение	чтение	сериализации
Read					
uncommitted					
Read					
committed					
Repeatable					
read					
Serializable					

Возможно –
Невозможно –
Лопускается, но не в PG –

Скрипты для проверки

1. Запрос на выборку. Дата конференции с id = 1:

SELECT "Conferences".date
FROM "Conferences"
WHERE conference_id = 1

2. Запрос на изменение даты конференции с id = 1:

UPDATE "Conferences" SET date = date + interval '1 hour' WHERE conference_id = 1

Проверка наличия аномалий

1. Установить в обоих сеансах уровень изоляции READ UNCOMMITTED. Выполнить сценарии проверки наличия аномалий потерянных изменений и грязных чтений.

Потерянных изменений нет		Da	ate
BEGIN TRANSACTION	BEGIN TRANSACTION ISOLATION		
ISOLATION LEVEL	LEVEL		
READ UNCOMMITTED;	READ UNCOMMITTED;		
SELECT "Conferences".date		14:00:00	
FROM "Conferences"			
WHERE conference_id = 1			
	SELECT "Conferences".date		14:00:00
	FROM "Conferences"		
	WHERE conference_id = 1		
UPDATE "Conferences" SET		15:00:00	
date = date + interval '1 hour'			
WHERE conference_id = 1			
COMMIT			
	UPDATE "Conferences" SET date =		16:00:00
	date + interval '1 hour'		
	WHERE conference_id = 1		
	COMMIT		

Грязных чтений нет		Da	ate
BEGIN TRANSACTION	BEGIN TRANSACTION ISOLATION		
ISOLATION LEVEL	LEVEL		
READ UNCOMMITTED;	READ UNCOMMITTED;		
SELECT "Conferences".date		16:00:00	
FROM "Conferences"			
WHERE conference_id = 1			
	UPDATE "Conferences" SET date =		17:00:00
	date + interval '1 hour'		
	WHERE conference_id = 1		
SELECT "Conferences".date		16:00:00	
FROM "Conferences"			
WHERE conference_id = 1			
COMMIT	COMMIT		

2. Установить в обоих сеансах уровень изоляции READ COMMITTED. Выполнить сценарии проверки наличия аномалий грязных чтений и неповторяющихся чтений.

Грязных чтений нет			ate
BEGIN TRANSACTION	BEGIN TRANSACTION ISOLATION		
ISOLATION LEVEL	LEVEL		
READ COMMITTED;	READ COMMITTED;		
	SELECT "Conferences".date		17:00:00
	FROM "Conferences"		
	WHERE conference_id = 1		
UPDATE "Conferences" SET		18:00:00	
date = date + interval '1 hour'			
WHERE conference_id = 1			
	SELECT "Conferences".date		17:00:00
	FROM "Conferences"		
	WHERE conference_id = 1		
COMMIT	COMMIT		

Неповторяющиеся чтения есть		D	ate
BEGIN TRANSACTION	BEGIN TRANSACTION ISOLATION		
ISOLATION LEVEL	LEVEL		
READ COMMITTED;	READ COMMITTED;		
	SELECT "Conferences".date		18:00:00
	FROM "Conferences"		
	WHERE conference_id = 1		
SELECT "Conferences".date		18:00:00	
FROM "Conferences"			
WHERE conference_id = 1			
UPDATE "Conferences" SET		19:00:00	
date = date + interval '1 hour'			
WHERE conference_id = 1			
COMMIT			
	SELECT "Conferences".date		19:00:00
	FROM "Conferences"		
	WHERE conference_id = 1		
	COMMIT		

3. Установить в обоих сеансах уровень изоляции REPEATABLE READ. Выполнить сценарии проверки наличия аномалий неповторяющихся чтений и фантомов.

Неповторяющихся чтений нет		Date
BEGIN TRANSACTION	BEGIN TRANSACTION ISOLATION	
ISOLATION LEVEL	LEVEL	
REPEATABLE READ;	REPEATABLE READ;	

	SELECT "Conferences".date		19:00:00
	FROM "Conferences"		
	WHERE conference_id = 1		
UPDATE "Conferences" SET		20:00:00	
date = date + interval '1 hour'			
WHERE conference_id = 1			
COMMIT			
	SELECT "Conferences".date		19:00:00
	FROM "Conferences"		
	WHERE conference_id = 1		
	COMMIT		

Фантомного чтения нет		Date
BEGIN TRANSACTION	BEGIN TRANSACTION ISOLATION	
ISOLATION LEVEL	LEVEL	
REPEATABLE READ;	REPEATABLE READ;	
	SELECT "Conferences".date	19:00:00
	FROM "Conferences"	
	WHERE conference_id = 1	
UPDATE "Conferences" SET		
date = date + interval '1 hour'		
WHERE conference_id = 1		
COMMIT		
	SELECT "Conferences".date	19:00:00
	FROM "Conferences"	
	WHERE conference_id = 1	
	COMMIT	

4. Установить в обоих сеансах уровень изоляции SERIALIZABLE. Выполнить сценарий проверки наличия фантомов.

Фантомного чтения нет		Date
BEGIN TRANSACTION	BEGIN TRANSACTION ISOLATION	
ISOLATION LEVEL	LEVEL	
SERIALIZABLE;	SERIALIZABLE;	
	SELECT "Conferences".date	19:00:00
	FROM "Conferences"	
	WHERE conference_id = 1	
UPDATE "Conferences" SET		
date = date + interval '1 hour'		
WHERE conference_id = 1		
COMMIT		
	SELECT "Conferences".date	19:00:00
	FROM "Conferences"	
	WHERE conference_id = 1	
	COMMIT	

Аномалия се	риализации
BEGIN TRANSACTION ISOLATION LEVEL	BEGIN TRANSACTION ISOLATION LEVEL
SERIALIZABLE;	SERIALIZABLE;
UPDATE "Conferences" SET date = date +	
interval '1 hour'	
WHERE conference_id = 1	
	UPDATE "Conferences" SET date = date +
	interval '1 hour'
	WHERE conference_id = 2
COMMIT	
	COMMIT
	[40001] ОШИБКА: не удалось сериализовать
	доступ из-за зависимостей чтения/записи
	между транзакциями

Триггер

Скрипт:

CREATE OR REPLACE FUNCTION check conference date()

RETURNS TRIGGER AS \$\$

BEGIN

IF NEW.date <= now()

THEN

RAISE EXCEPTION 'The date of the conference must be greater than the current';

END IF:

RETURN NEW;

END;

\$\$ LANGUAGE plpgsql;

CREATE TRIGGER check conference date trigger

AFTER INSERT ON "Conferences"

FOR EACH ROW EXECUTE PROCEDURE check conference date()

• Запросы для проверки и демонстрации полезных свойств:

-- ОШИБКА ('The date of the conference must be greater than the current')

INSERT INTO "Conferences" (country_id, title, theme, place, date) VALUES ('1','18-я международная научная конференция','Advances in Biology','СПБГУ','2022-03-16 14:00:00')

-- ВЫПОЛНИЛСЯ (в таблицу 'Conferences' добавлена новая конференция)

INSERT INTO "Conferences" (country_id, title, theme, place, date) VALUES ('1','18-я международная научная конференция','Advances in Biology','СПБГУ','2022-05-16 14:00:00')

Практическое задание №2.1 Управление доступом

Постановка задачи

Целью практического задания является освоение работы с представлениями и другими способами управления доступом.

Ход работы

1. Создать пользователя test и выдать ему доступ к базе данных.

```
CREATE USER test WITH PASSWORD '123';
GRANT ALL PRIVILEGES ON DATABASE scientific conferences TO test;
```

2. По крайней мере, для одной таблицы новому пользователю присваиваются права SELECT, INSERT, UPDATE в полном объеме.

```
GRANT SELECT, INSERT, UPDATE ON TABLE "Conferences" TO test;
```

3. По крайней мере, для одной таблицы новому пользователю присваиваются права SELECT и UPDATE только избранных столбцов.

```
GRANT SELECT(scientist_id, person_id, country_id),
UPDATE(organization_id, specialization_id, degree_id)
ON TABLE "Scientists" TO test;
```

4. По крайней мере, для одной таблицы новому пользователю присваивается только право SELECT.

```
GRANT SELECT ON TABLE "Persons" TO test;
GRANT SELECT ON TABLE "Participations" TO test;
GRANT SELECT ON TABLE "Countries" TO test;
```

- 5. Составить SQL-скрипты для создания нескольких представлений.
- Представление с учеными и количеством выступлений(участий) на конференциях

```
CREATE VIEW Scientists_Number_Of_Participations AS (
SELECT

"Persons".first_name,

"Persons".second_name,

"Countries".name AS country,

COUNT(distinct "Participations".participation_id) AS paricipation_number
```

```
FROM "Participations"

JOIN "Conferences" ON "Conferences".conference_id = "Participations".conference_id

JOIN "Scientists" ON "Scientists".scientist_id = "Participations".scientist_id

JOIN "Persons" ON "Persons".person_id = "Scientists".person_id

JOIN "Countries" ON "Countries".country_id = "Scientists".country_id

GROUP BY "Persons".first_name, "Persons".second_name, "Countries".name

ORDER BY paricipation_number desc
);
```

• Представление со странами по убыванию количества проводимых конференций

```
CREATE VIEW Countries_By_Number_Of_Conferences AS (
(SELECT
    "Countries".name AS country,
    COUNT("Conferences".country_id) AS conferences_num
FROM "Conferences"

JOIN "Countries" ON "Countries".country_id = "Conferences".country_id
GROUP BY "Countries".name ORDER BY COUNT("Conferences".country_id) DESC)
UNION ALL
(SELECT
    "Countries".name AS country, 0
FROM "Countries"
WHERE NOT EXISTS (SELECT * FROM "Conferences"
WHERE "Countries".country_id = "Conferences".country_id))
);
```

6. Присвоить новому пользователю право доступа (SELECT) к одному из представлений

```
GRANT SELECT ON Scientists Number Of Participations TO test;
```

7. Создать стандартную роль уровня базы данных.

```
CREATE ROLE test_role;
```

8. Присвоить ей право доступа (UPDATE на некоторые столбцы) к одному из представлений.

```
GRANT UPDATE(country) ON Countries_By_Number_Of_Conferences TO test_role;
```

9. Назначить новому пользователю созданную роль.

```
GRANT test_role TO test;
```

Проверка доступа

Таблица	Разрешен доступ	Запрос	Результат
Conferences	SELECT UPDATE INSERT	SELECT * FROM "Conferences" WHERE "Conferences".conference_id = 11	
		UPDATE "Conferences" SET date = date + interval '1 hour' WHERE "Conferences".conference_id = 11	
		INSERT INTO "Conferences" (country_id, title, theme, place, date) VALUES ('1','14-я Международная научная конференция','Advances in Biology','СПБГУ','2022-08-04 10:00:00')	
Scientists	SELECT (scientist_id, person_id,	SELECT scientist_id, person_id, country_id FROM "Scientists";	
	country_id) UPDATE	UPDATE "Scientists" SET organization_id = 1 WHERE scientist_id = 1;	
	(organization_id, specialization_id, degree_id)	SELECT * FROM "Scientists" UPDATE "Scientists" SET country_id = 2 WHERE scientist_id = 1;	
Persons	SELECT	SELECT * FROM "Persons"	
		UPDATE "Persons" SET first_name = 'Иван' WHERE "Persons".person_id = 1	
		INSERT INTO "Persons" (first_name, second_name, middle_name) VALUES ('Евгений','Панарин','Владимирович')	
Scientists_ Number_ Of_ Participations	SELECT	SELECT * FROM Scientists_Number_Of_Participations	
Countries_ By_ Number_ Of_ Conferences	_	SELECT * FROM Countries_By_Number_Of_Conferences	

Ошибка доступа – Запрос выполняется –

Вывод макрокоманды \dp

scientific_conferences-# \dp

	<u>-</u>		Access privileges		
Schema	Name	Туре	Access privileges	Column privileges Policie	s
public	Conferences	table	alinapotapova=arwdDxt/alinapotapova+ test=arw/alinapotapova		
public	Conference_conference_id_seq	sequence	alinapotapova=rwU/alinapotapova + test=rU/alinapotapova	į	
public	Countries	table	alinapotapova=arwdDxt/alinapotapova+ test=r/alinapotapova	į	
public	Countries_country_id_seq	sequence	alinapotapova=rwU/alinapotapova + test=rU/alinapotapova		
public	Degrees	table		j	
public	Degrees degree id seq	sequence	alinapotapova=rwU/alinapotapova +	j	
_		_	test=rU/alinapotapova	į	
public	Organizations	table		ļ	
public	Organizations_organization_id_seq	sequence	alinapotapova=rwU/alinapotapova + test=rU/alinapotapova		
public	Participations	table	alinapotapova=arwdDxt/alinapotapova+ test=r/alinapotapova		
public	Participations_participation_id_seq	sequence	alinapotapova + test=rU/alinapotapova +		
public	Persons	table	alinapotapova=arwdDxt/alinapotapova+ test=r/alinapotapova	į	
public	Persons_person_id_seq	sequence	alinapotapova=rwU/alinapotapova + test=rU/alinapotapova		
public public public public	Scientists Scientists_scientist_id_seq Specializations Specialization_id_seq	sequence table sequence	alinapotapova=rwU/alinapotapova + test=rU/alinapotapova alinapotapova=rwU/alinapotapova +	<pre>scientist_id: test=r/alinapotapova + person_id: test=r/alinapotapova degree_id: test=w/alinapotapova country_id: test=r/alinapotapova organization_id: test=w/alinapotapova specialization_id: test=w/alinapotapova</pre>	
public	Tunes	 table	test=rU/alinapotapova	!	
public	Types Types type id seq	sequence	 alinapotapova=rwU/alinapotapova +		
Public	115bes_c1be_1g_sed	sequence	test=rU/alinapotapova	İ	
public	scientists_number_of_participations	view	alinapotapova=arwdDxt/alinapotapova+ test=r/alinapotapova		
public	Countries_by_number_of_conferences	view	alinapotapova=arwdDxt/alinapotapova	country: + test role=w/alinapotapova	
(20 rows			'		

Практическое задание №2.2 Функции и язык PL/pgSQL

Постановка задачи

Практическое задание посвящено упрощению работы с помощью создания и использования функций. При выполнении задания необходимо:

- Составить SQL-скрипты для создания нескольких (2–3) функций, упрощающих работу с данными.
- Продемонстрировать полученные знания о возможностях языка PL/pgSQL. В скриптах должны использоваться:
 - о Циклы.
 - о Переменные.
 - о Курсоры.
 - о Ветвления.
 - о Исключения.
- Обосновать преимущества механизма функций перед механизмом представлений.

Ход работы

1. Скрипт на добавление конференции с использованием *ветвления* и *исключений*:

```
CREATE OR REPLACE FUNCTION add_new_conference(country_id INT, title
VARCHAR(128), theme VARCHAR(128), place VARCHAR(128), date TIMESTAMP)
RETURNS VOID AS $$
BEGIN

IF (date <= now())
THEN

RAISE EXCEPTION 'The date of the conference must be greater than the current';
ELSIF (SELECT EXISTS(SELECT * FROM "Countries" WHERE
"Countries".country_id = add_new_conference.country_id))
THEN

INSERT INTO "Conferences" (country_id, title, theme, place, date)
VALUES (add_new_conference.country_id, add_new_conference.title,
add_new_conference.theme, add_new_conference.place,
add_new_conference.date);
```

ELSE

RAISE EXCEPTION 'Country pk does not exist';

END IF;

END;

\$\$ LANGUAGE plpgsql;

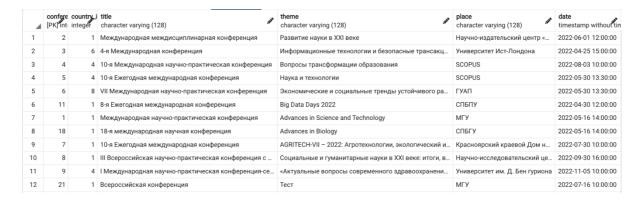
Запрос:

SELECT add_new_conference('1','Всероссийская конференция','Тест','МГУ','2022-07-16 10:00:00')

До выполнения запроса:

4	confere [PK] int	country_i integer	title character varying (128)	theme character varying (128)	place character varying (128)	date timestamp without tin
1	2	1	Международная междисциплинарная конференция	Развитие науки в XXI веке	Научно-издательский центр «	2022-06-01 12:00:00
2	3	6	4-я Международная конференция	Информационные технологии и безопасные трансакц	Университет Ист-Лондона	2022-04-25 15:00:00
3	4	4	10-я Международная научно-практическая конференция	Вопросы трансформации образования	SCOPUS	2022-08-03 10:00:00
4	5	4	10-я Ежегодная международная конференция	Наука и технологии	SCOPUS	2022-05-30 13:30:00
5	6	8	VII Международная научно-практическая конференция	Экономические и социальные тренды устойчивого ра	ГУАП	2022-05-30 13:30:00
6	11	1	8-я Ежегодная международная конференция	Big Data Days 2022	СПБПУ	2022-04-30 12:00:00
7	1	1	Международная научно-практическая конференция	Advances in Science and Technology	МГУ	2022-05-16 14:00:00
8	18	1	18-я международная научная конференция	Advances in Biology	СПБГУ	2022-05-16 14:00:00
9	7	1	10-я Ежегодная международная конференция	AGRITECH-VII – 2022: Агротехнологии, экологический и	Красноярский краевой Дом н	2022-07-30 10:00:00
10	8	1	III Всероссийская научно-практическая конференция с	Социальные и гуманитарные науки в XXI веке: итоги, в	Научно-исследовательский це	2022-09-30 16:00:00
11	9	4	I Международная научно-практическая конференция-се	«Актуальные вопросы современного здравоохранени	Университет им. Д. Бен гуриона	2022-11-05 10:00:00

После выполнения запроса:



2. Скрипт на поиск конференций по странам:

```
CREATE OR REPLACE FUNCTION search_country(country_name VARCHAR(64))

RETURNS TABLE(
    conference_title VARCHAR(128),
    conference_theme VARCHAR(128),
    conference_place VARCHAR(128),
    conference_date TIMESTAMP) AS

$$
```

```
BEGIN

RETURN QUERY(SELECT

"Conferences".title,
"Conferences".theme,
"Conferences".place,
"Conferences".date

FROM "Conferences" JOIN "Countries" ON "Countries".country_id =
"Conferences".country_id

AND "Countries".name ILIKE '%' || country_name || '%');

END;

$$ LANGUAGE plpgsql;
```

Запрос:

```
SELECT * FROM search_country('Россия');
```

Результат:

4	conference_title character varying	conference_theme character varying	conference_place character varying	conference_date timestamp without
1	Международная междисциплинарная конференция	Развитие науки в XXI веке	Научно-издательский центр «Толмачево»	2022-06-01 12:00:00
2	8-я Ежегодная международная конференция	Big Data Days 2022	СПБПУ	2022-04-30 12:00:00
3	Международная научно-практическая конференция	Advances in Science and Technology	МГУ	2022-05-16 14:00:00
4	18-я международная научная конференция	Advances in Biology	СПБГУ	2022-05-16 14:00:00
5	10-я Ежегодная международная конференция	AGRITECH-VII - 2022: Агротехноло	Красноярский краевой Дом науки и техники	2022-07-30 10:00:00
6	III Всероссийская научно-практическая конференц	Социальные и гуманитарные наук	Научно-исследовательский центр «Пересвет»	2022-09-30 16:00:00
7	Всероссийская конференция	Тест	МГУ	2022-07-16 10:00:00

3. Скрипт на вывод количества конференций за определенный период с использованием *курсоров*, *переменных и циклов*:

```
CREATE OR REPLACE FUNCTION conferences_num_for_selected_period
(start_period TIMESTAMP, end_period TIMESTAMP)
RETURNS INT AS

$$

DECLARE

cur CURSOR FOR SELECT "Conferences".date FROM "Conferences";
result_conferences_num INT;

BEGIN

result_conferences_num = 0;
FOR rec IN cur

LOOP

IF ((rec.date >= start_period) AND (rec.date <= end_period))
THEN

result_conferences_num = result_conferences_num + 1;
```

```
END IF;
END LOOP;
RETURN result_conferences_num;
END;
$$ LANGUAGE plpgsql;
```

Запрос:

SELECT * FROM conferences_num_for_selected_period('2022-03-16', '2022-05-20')



Приложение (SQL dump)

```
-- PostgreSQL database dump
-- Dumped from database version 14.2
-- Dumped by pg dump version 14.2
SET statement timeout = 0;
SET lock timeout = 0;
SET idle in transaction session timeout = 0;
SET client encoding = 'UTF8';
SET standard conforming strings = on;
SELECT pg catalog.set config('search path', ", false);
SET check function bodies = false;
SET xmloption = content;
SET client min messages = warning;
SET row security = off;
-- Name: add new conference(integer, character varying, character varying, character varying,
timestamp without time zone); Type: FUNCTION; Schema: public; Owner: alinapotapova
CREATE FUNCTION public."add new conference"(country_id integer, title character varying,
theme character varying, place character varying, date timestamp without time zone) RETURNS
void
  LANGUAGE plpgsql
  AS $$
BEGIN
      IF (date \le now())
      THEN
             RAISE EXCEPTION 'The date of the conference must be greater than the
current';
      ELSIF (SELECT EXISTS(SELECT * FROM "Countries" WHERE
"Countries".country id = add new conference.country id))
      THEN
             INSERT INTO "Conferences" (country id, title, theme, place, date)
             VALUES (add new conference.country id, add new conference.title,
                           add new conference.theme, add new conference.place,
add new conference.date);
      ELSE
             RAISE EXCEPTION 'Country pk does not exist';
      END IF;
END;
$$;
```

ALTER FUNCTION public."add_new_conference"(country_id integer, title character varying, theme character varying, place character varying, date timestamp without time zone) OWNER TO alinapotapova;

```
-- Name: check conference date(); Type: FUNCTION; Schema: public; Owner: alinapotapova
CREATE FUNCTION public.check conference date() RETURNS trigger
  LANGUAGE plpgsql
  AS $$
BEGIN
IF NEW.date <= now()
THEN
RAISE EXCEPTION 'The date of the conference must be greater than the current';
END IF:
RETURN NEW;
END;
$$;
ALTER FUNCTION public.check conference date() OWNER TO alinapotapova;
-- Name: conferences num for selected period(timestamp without time zone, timestamp
without time zone); Type: FUNCTION; Schema: public; Owner: alinapotapova
CREATE FUNCTION public.conferences num for selected period(start period timestamp
without time zone, end period timestamp without time zone) RETURNS integer
  LANGUAGE plpgsql
  AS $$
DECLARE
      cur CURSOR FOR SELECT "Conferences".date FROM "Conferences";
      result conferences num INT;
BEGIN
      result conferences num = 0;
      FOR rec IN cur
            LOOP
                   IF ((rec.date >= start period) AND (rec.date <= end period))
                          result conferences num = result conferences num + 1;
                   END IF:
            END LOOP;
      RETURN result conferences num;
END;
$$;
ALTER FUNCTION public.conferences num for selected period(start period timestamp
without time zone, end period timestamp without time zone) OWNER TO alinapotapova;
-- Name: search country(character varying); Type: FUNCTION; Schema: public; Owner:
alinapotapova
```

```
CREATE FUNCTION public.search country(country name character varying) RETURNS
TABLE(conference title character varying, conference theme character varying,
conference place character varying, conference date timestamp without time zone)
  LANGUAGE plpgsql
  AS $$
BEGIN
      RETURN QUERY(SELECT
             "Conferences".title.
             "Conferences".theme,
             "Conferences".place,
             "Conferences".date
      FROM "Conferences" JOIN "Countries" ON "Countries".country_id =
"Conferences".country id
      AND "Countries".name ILIKE '%' || country name ||'%');
END;
$$:
ALTER FUNCTION public.search country(country name character varying) OWNER TO
alinapotapova;
SET default tablespace = ";
SET default table access method = heap;
-- Name: Conferences; Type: TABLE; Schema: public; Owner: alinapotapova
CREATE TABLE public."Conferences" (
  conference id integer NOT NULL,
  country id integer NOT NULL,
  title character varying(128) NOT NULL,
  theme character varying (128) NOT NULL,
  place character varying(128),
  date timestamp without time zone
);
ALTER TABLE public. "Conferences" OWNER TO alinapotapova;
-- Name: Conferences conference id seq; Type: SEQUENCE; Schema: public; Owner:
alinapotapova
CREATE SEQUENCE public."Conferences conference id seq"
  AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
```

```
CACHE 1;
ALTER TABLE public."Conferences conference id seq" OWNER TO alinapotapova;
-- Name: Conferences conference id seq; Type: SEQUENCE OWNED BY; Schema: public;
Owner: alinapotapova
ALTER SEQUENCE public."Conferences conference id seq" OWNED BY
public."Conferences".conference id;
-- Name: Countries; Type: TABLE; Schema: public; Owner: alinapotapova
CREATE TABLE public."Countries" (
  country id integer NOT NULL,
  name character varying(64) NOT NULL
);
ALTER TABLE public."Countries" OWNER TO alinapotapova;
-- Name: Countries country id seq; Type: SEQUENCE; Schema: public; Owner: alinapotapova
CREATE SEQUENCE public."Countries country id seq"
  AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;
ALTER TABLE public. "Countries country id seq" OWNER TO alinapotapova;
-- Name: Countries country id seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:
alinapotapova
ALTER SEQUENCE public."Countries country id seq" OWNED BY
public."Countries".country id;
-- Name: Degrees; Type: TABLE; Schema: public; Owner: alinapotapova
```

```
CREATE TABLE public."Degrees" (
  degree id integer NOT NULL,
  name character varying(64) NOT NULL
);
ALTER TABLE public. "Degrees" OWNER TO alinapotapova;
-- Name: Degrees degree id seq; Type: SEQUENCE; Schema: public; Owner: alinapotapova
CREATE SEQUENCE public. "Degrees degree id seq"
  AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;
ALTER TABLE public." Degrees degree id seq" OWNER TO alinapotapova;
-- Name: Degrees degree id seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:
alinapotapova
ALTER SEQUENCE public."Degrees degree id seq" OWNED BY
public."Degrees".degree id;
-- Name: Organizations; Type: TABLE; Schema: public; Owner: alinapotapova
CREATE TABLE public. "Organizations" (
  organization id integer NOT NULL,
  name character varying(128) NOT NULL
);
ALTER TABLE public."Organizations" OWNER TO alinapotapova;
-- Name: Organizations organization id seq; Type: SEQUENCE; Schema: public; Owner:
alinapotapova
CREATE SEQUENCE public."Organizations organization id seq"
```

```
AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;
ALTER TABLE public." Organizations organization id seq" OWNER TO alinapotapova;
-- Name: Organizations organization id seq; Type: SEQUENCE OWNED BY; Schema:
public; Owner: alinapotapova
ALTER SEQUENCE public. "Organizations organization id seq" OWNED BY
public."Organizations".organization id;
-- Name: Participations; Type: TABLE; Schema: public; Owner: alinapotapova
CREATE TABLE public. "Participations" (
  participation id integer NOT NULL,
  scientist id integer NOT NULL,
  conference id integer NOT NULL,
  type id integer,
  topic character varying(128),
  duration time without time zone
);
ALTER TABLE public. "Participations" OWNER TO alinapotapova;
-- Name: Participations participation id seq; Type: SEQUENCE; Schema: public; Owner:
alinapotapova
CREATE SEQUENCE public."Participations participation id seq"
  AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;
ALTER TABLE public."Participations participation id seq" OWNER TO alinapotapova;
```

```
-- Name: Participations participation id seq; Type: SEQUENCE OWNED BY; Schema: public;
Owner: alinapotapova
ALTER SEQUENCE public. "Participations participation id seq" OWNED BY
public."Participations".participation id;
-- Name: Persons; Type: TABLE; Schema: public; Owner: alinapotapova
CREATE TABLE public. "Persons" (
  person id integer NOT NULL,
  first name character varying(64) NOT NULL,
  second name character varying(64) NOT NULL,
  middle name character varying(64)
);
ALTER TABLE public. "Persons" OWNER TO alinapotapova;
-- Name: Persons person id seq; Type: SEQUENCE; Schema: public; Owner: alinapotapova
CREATE SEQUENCE public."Persons person id seq"
  AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;
ALTER TABLE public. "Persons_person_id_seq" OWNER TO alinapotapova;
-- Name: Persons person id seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:
alinapotapova
ALTER SEQUENCE public."Persons person id seq" OWNED BY public."Persons".person id;
-- Name: Scientists; Type: TABLE; Schema: public; Owner: alinapotapova
CREATE TABLE public. "Scientists" (
  scientist id integer NOT NULL,
  person id integer NOT NULL,
```

```
degree id integer,
  country_id integer,
  organization id integer,
  specialization id integer
);
ALTER TABLE public. "Scientists" OWNER TO alinapotapova;
-- Name: Scientists scientist id seq; Type: SEQUENCE; Schema: public; Owner:
alinapotapova
CREATE SEQUENCE public. "Scientists scientist id seq"
  AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;
ALTER TABLE public." Scientists scientist id seq" OWNER TO alinapotapova;
-- Name: Scientists scientist id seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:
alinapotapova
ALTER SEQUENCE public. "Scientists scientist id seq" OWNED BY
public."Scientists".scientist id;
-- Name: Specializations; Type: TABLE; Schema: public; Owner: alinapotapova
CREATE TABLE public. "Specializations" (
  specialization id integer NOT NULL,
  name character varying(64) NOT NULL
);
ALTER TABLE public. "Specializations" OWNER TO alinapotapova;
-- Name: Specializations specialization id seq; Type: SEQUENCE; Schema: public; Owner:
alinapotapova
CREATE SEQUENCE public."Specializations specialization id seq"
```

```
AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
  CACHE 1;
ALTER TABLE public." Specializations specialization id seq." OWNER TO alinapotapova;
-- Name: Specializations specialization id seq; Type: SEQUENCE OWNED BY; Schema:
public; Owner: alinapotapova
ALTER SEQUENCE public." Specializations specialization id seq" OWNED BY
public."Specializations".specialization id;
-- Name: Types; Type: TABLE; Schema: public; Owner: alinapotapova
CREATE TABLE public."Types" (
  type id integer NOT NULL,
  name character varying(64) NOT NULL
);
ALTER TABLE public. "Types" OWNER TO alinapotapova;
-- Name: Types type id seq; Type: SEQUENCE; Schema: public; Owner: alinapotapova
CREATE SEQUENCE public."Types type id seq"
  AS integer
  START WITH 1
  INCREMENT BY 1
  NO MINVALUE
  NO MAXVALUE
 CACHE 1;
ALTER TABLE public."Types type id seq" OWNER TO alinapotapova;
-- Name: Types type id seq; Type: SEQUENCE OWNED BY; Schema: public; Owner:
alinapotapova
ALTER SEQUENCE public."Types type id seq" OWNED BY public."Types".type id;
```

```
-- Name: scientists number of participations; Type: VIEW; Schema: public; Owner:
alinapotapova
CREATE VIEW public.scientists number of participations AS
SELECT "Persons".first name,
  "Persons".second name,
  "Countries".name AS country,
  count(DISTINCT "Participations".participation id) AS paricipation number
 FROM ((((public."Participations"
  JOIN public. "Conferences" ON (("Conferences".conference id =
"Participations".conference id)))
  JOIN public. "Scientists" ON (("Scientists".scientist id = "Participations".scientist id)))
  JOIN public."Persons" ON (("Persons".person id = "Scientists".person id)))
  JOIN public. "Countries" ON (("Countries".country id = "Scientists".country id)))
 GROUP BY "Persons".first name, "Persons".second name, "Countries".name
 ORDER BY (count(DISTINCT "Participations".participation id)) DESC;
ALTER TABLE public.scientists number of participations OWNER TO alinapotapova;
-- Name: Countries by number of conferences; Type: VIEW; Schema: public; Owner:
alinapotapova
CREATE VIEW public."Countries by number of conferences" AS
(SELECT "Countries".name AS country,
  count("Conferences".country id) AS conferences num
 FROM (public."Conferences"
  JOIN public. "Countries" ON (("Countries".country id = "Conferences".country id)))
 GROUP BY "Countries".name
 ORDER BY (count("Conferences".country id)) DESC)
UNION ALL
SELECT "Countries".name AS country,
  0 AS conferences num
 FROM public."Countries"
 WHERE (NOT (EXISTS ( SELECT "Conferences".conference id,
       "Conferences".country id,
       "Conferences".title,
       "Conferences".theme,
       "Conferences".place,
      "Conferences".date
      FROM public."Conferences"
     WHERE ("Countries".country id = "Conferences".country id))));
ALTER TABLE public."Countries by number of conferences" OWNER TO alinapotapova;
-- Name: Conferences conference id; Type: DEFAULT; Schema: public; Owner: alinapotapova
```

--

ALTER TABLE ONLY public. "Conferences" ALTER COLUMN conference id SET DEFAULT nextval('public."Conferences conference id seq"::regclass); -- Name: Countries country id; Type: DEFAULT; Schema: public; Owner: alinapotapova ALTER TABLE ONLY public."Countries" ALTER COLUMN country id SET DEFAULT nextval('public."Countries country id seq"::regclass); -- Name: Degrees degree id; Type: DEFAULT; Schema: public; Owner: alinapotapova ALTER TABLE ONLY public."Degrees" ALTER COLUMN degree id SET DEFAULT nextval('public."Degrees degree id seq"::regclass); -- Name: Organizations organization id; Type: DEFAULT; Schema: public; Owner: alinapotapova ALTER TABLE ONLY public."Organizations" ALTER COLUMN organization id SET DEFAULT nextval('public."Organizations organization id seq"::regclass); -- Name: Participations participation id; Type: DEFAULT; Schema: public; Owner: alinapotapova ALTER TABLE ONLY public."Participations" ALTER COLUMN participation_id SET DEFAULT nextval('public."Participations participation id seq"::regclass); -- Name: Persons person id; Type: DEFAULT; Schema: public; Owner: alinapotapova ALTER TABLE ONLY public."Persons" ALTER COLUMN person id SET DEFAULT nextval('public."Persons person id seq"::regclass);

-- Name: Scientists scientist_id; Type: DEFAULT; Schema: public; Owner: alinapotapova

ALTER TABLE ONLY public. "Scientists" ALTER COLUMN scientist_id SET DEFAULT nextval('public. "Scientists scientist id seq"::regclass);

-- -- Name: Specializations specialization_id; Type: DEFAULT; Schema: public; Owner: alinapotapova

ALTER TABLE ONLY public."Specializations" ALTER COLUMN specialization_id SET DEFAULT nextval('public."Specializations specialization id seq"::regclass);

-- Name: Types type_id; Type: DEFAULT; Schema: public; Owner: alinapotapova

ALTER TABLE ONLY public."Types" ALTER COLUMN type_id SET DEFAULT nextval('public."Types type id seq"::regclass);

-- Data for Name: Conferences; Type: TABLE DATA; Schema: public; Owner: alinapotapova

COPY public."Conferences" (conference id, country id, title, theme, place, date) FROM stdin;

- 2 1 Международная междисциплинарная конференция Развитие науки в XXI веке Научно-издательский центр «Толмачево» 2022-06-01 12:00:00
- 3 6 4-я Международная конференция Информационные технологии и безопасные трансакции Университет Ист-Лондона 2022-04-25 15:00:00
- 4 4 10-я Международная научно-практическая конференция Вопросы трансформации образования SCOPUS 2022-08-03 10:00:00
- 5 4 10-я Ежегодная международная конференция Наука и технологии SCOPUS 2022-05-30 13:30:00
- 6 8 VII Международная научно-практическая конференция Экономические и социальные тренды устойчивого развития современного общества ГУАП 2022-05-30 13:30:00
- 11 1 8-я Ежегодная международная конференция Big Data Days 2022 СПБПУ 2022-04-30 12:00:00
- 1 1 Международная научно-практическая конференция Advances in Science and Technology МГУ 2022-05-16 14:00:00
- 18 1 18-я международная научная конференция Advances in Biology СПБГУ 2022-05-16 14:00:00
- 7 1 10-я Ежегодная международная конференция AGRITECH-VII 2022: Агротехнологии, экологический инжиниринг и устойчивое развитие Красноярский краевой Дом науки и техники 2022-07-30 10:00:00
- 8 1 III Всероссийская научно-практическая конференция с международным участием Социальные и гуманитарные науки в XXI веке: итоги, вызовы, перспективы Научно-исследовательский центр «Пересвет» 2022-09-30 16:00:00
- 9 4 I Международная научно-практическая конференция-семинар «Актуальные вопросы современного здравоохранения. Электронная медицина» Университет им. Д. Бен гуриона 2022-10-30 12:00:00

```
\.
-- Data for Name: Countries; Type: TABLE DATA; Schema: public; Owner: alinapotapova
COPY public."Countries" (country id, name) FROM stdin;
2
      Франция
3
      Китай
4
      США
5
      Германия
      Великобритания
6
7
      Япония
8
      Индия
9
      Канада
10
      Бразилия
11
      Италия
12
      Испания
13
      Австралия
14
      Швейцария
      Израиль
15
      Россия
1
17
      финляндия
-- Data for Name: Degrees; Type: TABLE DATA; Schema: public; Owner: alinapotapova
COPY public."Degrees" (degree id, name) FROM stdin;
      Доктор архитектурных наук
1
      Доктор биологических наук
2
      Доктор географических наук
3
4
      Доктор филологических наук
5
      Доктор технических наук
6
      Доктор медицинских наук
7
      Доктор физико-математических наук
      Доктор политологических наук
8
9
      Кандидат архитектурных наук
      Кандидат биологических наук
10
      Кандидат географических наук
11
      Кандидат филологических наук
12
13
      Кандидат технических наук
14
      Кандидат медицинских наук
      Кандидат физико-математических наук
15
١.
-- Data for Name: Organizations; Type: TABLE DATA; Schema: public; Owner: alinapotapova
```

41

```
Федерация европейских биохимических обществ
2
      Ученые будущего
3
      Яндекс
4
      Р-Фарм
5
      Группа 77
6
      Филологический факультет МГУ имени М.В.Ломоносова
7
      Международный математический союз
8
      Международная ассоциация академий наук
9
      Европейская ассоциация географов
10
      Международный исследовательский центр по сохранению и реставрации
культурных ценностей
11
      Ассоциация вычислительной техники
12
      Институт высших научных исследований
13
      Международная академия астронавтики
14
      Международный совет по науке
15
      Союз социальной политики
-- Data for Name: Participations; Type: TABLE DATA; Schema: public; Owner: alinapotapova
COPY public."Participations" (participation id, scientist id, conference id, type id, topic,
duration) FROM stdin;
                          Вакцинация населения. Значение вакцинации для жизни
отдельного человека и человечества в целом
                                             00:10:00
                          Микроэлементы в человеческой жизни, их основная роль
6
                   1
      7
      00:10:00
      3
                   2
                          Бесплодные пары: ЭКО как решение проблемы на
современном этапе жизни
                         00:10:00
      12
             9
                   3
                          \N
                                00:10:00
9
      13
             4
                          Современное географическое образование
                                                                       00:10:00
                   1
10
      4
             4
                   1
                          История региональной архитектуры
                                                                 00:10:00
11
      6
             4
                   1
                          Языковое сознание: формирование и функционирование
      00:10:00
12
             2
                   1
                          Перспективы освоения космического пространства
                                                                             00:10:00
      1
      5
13
             2
                   2
                          Средства защиты отчуждаемых носителей информации
      00:10:00
14
                   1
             1
                          Методики системного анализа и бизнес-аналитики
                                                                             00:10:00
15
      10
             1
                   2
                          Современные физические методы применяемые в онкологии
      00:10:00
16
      11
             3
                   3
                          \backslash N
                                00:10:00
17
                   1
                          Анализ частоты и причин отказа родителей от проведения
            00:12:00
вакцинации
18
      9
             1
                          Система как объект системного анализа 00:10:00
                   1
19
                          Источники возникновения и последствия реализации угроз
информационной безопасности
                                00:08:00
20
                          Защита информации от несанкционированного доступа
```

COPY public."Organizations" (organization id, name) FROM stdin;

методом криптопреобразования

```
21
                          Защита информации от несанкционированного доступа
методом криптопреобразования 00:12:00
\.
-- Data for Name: Persons; Type: TABLE DATA; Schema: public; Owner: alinapotapova
COPY public."Persons" (person id, first name, second name, middle name) FROM stdin;
                   Попов Сергеевич
1
      Виктор
2
      Джон Гилберт
                          \N
3
             Дина \N
      Кур
4
             Де ла Круа
      Луи
                          \N
5
                   Карташов
      Владислав
                                 Константинович
6
                   Гукова
      Виктория
                                 Олеговна
7
      Кристиан
                   Кане \N
8
      Елена Вальверде
                          \N
9
      Стефано
                   Волонте
                                 \N
10
      Робберт
                   Эллис \N
11
      ДжудиДалтон
                          \N
12
      Глория
                   Антонелли
                                \N
13
      Бруно Перес \N
14
      Рю
             Ким
                   \N
15
      Ритик Хан
17
      Петр Петров
                          Петрович
\.
-- Data for Name: Scientists; Type: TABLE DATA; Schema: public; Owner: alinapotapova
COPY public. "Scientists" (scientist id, person id, degree id, country id, organization id,
specialization id) FROM stdin;
                                 2
2
      2
             8
                   4
                          15
3
      3
             6
                   15
                                 3
                          1
4
      4
                   2
                          9
                                 10
             1
5
      5
             5
                          2
                    1
                                 4
6
      6
             4
                   1
                          6
                                 10
7
                          1
      10
             14
                   6
                                 3
8
      11
             10
                   13
                          1
                                 11
9
      12
             13
                    10
                          11
                                 4
10
      14
             15
                   3
                          14
                                 14
11
      7
             5
                   2
                          12
                                 8
12
             10
      8
                   12
                          1
                                 13
13
      9
             3
                          9
                    11
                                 10
15
      15
             1
                   7
                          10
                                 9
14
      13
             9
                    14
                          10
                                 1
19
             7
      17
                    1
                          13
                                 8
1
      1
             7
                    1
                          13
                                 6
\.
```

```
-- Data for Name: Specializations; Type: TABLE DATA; Schema: public; Owner: alinapotapova
COPY public."Specializations" (specialization id, name) FROM stdin;
      Градостроительство
2
      Политтехнолог
3
      Биохимия
4
      Системная аналитика
5
      Терапевт
6
      Космическая физика
7
      Зарубежная филология
8
      Информационная безопасность
9
      Реставрация архитектурных объектов
      Научная и преподавательская деятельность
10
      Аллергология и иммунология
11
12
      Молекулярная биология
13
      Биоинженерия
14
      Математические методы в современной физике
15
      Промышленная разработка
\.
-- Data for Name: Types; Type: TABLE DATA; Schema: public; Owner: alinapotapova
COPY public."Types" (type id, name) FROM stdin;
      Участие с докладом
1
2
      Участие с публикацией, но без доклада
3
      Участие без публикации и доклада
-- Name: Conferences conference id seq; Type: SEQUENCE SET; Schema: public; Owner:
alinapotapova
SELECT pg catalog.setval('public."Conferences conference id seq''', 19, true);
-- Name: Countries country id seq; Type: SEQUENCE SET; Schema: public; Owner:
alinapotapova
SELECT pg catalog.setval('public."Countries country id seq"', 18, true);
```

44

```
-- Name: Degrees degree id seq; Type: SEQUENCE SET; Schema: public; Owner:
alinapotapova
SELECT pg catalog.setval('public."Degrees degree id seq"', 15, true);
-- Name: Organizations organization id seq; Type: SEQUENCE SET; Schema: public; Owner:
alinapotapova
SELECT pg catalog.setval('public."Organization organization id seq", 15, true);
-- Name: Participations participation id seq; Type: SEQUENCE SET; Schema: public; Owner:
alinapotapova
SELECT pg catalog.setval('public."Participations participation id seq", 21, true);
-- Name: Persons person id seq; Type: SEQUENCE SET; Schema: public; Owner:
alinapotapova
SELECT pg catalog.setval('public."Persons person id seq'', 17, true);
-- Name: Scientists scientist id seq; Type: SEQUENCE SET; Schema: public; Owner:
alinapotapova
SELECT pg catalog.setval('public."Scientists scientist id seq", 19, true);
-- Name: Specializations specialization id seq; Type: SEQUENCE SET; Schema: public;
Owner: alinapotapova
SELECT pg catalog.setval('public."Specialization specialization id seq", 15, true);
-- Name: Types type id seq; Type: SEQUENCE SET; Schema: public; Owner: alinapotapova
SELECT pg_catalog.setval('public."Types_type_id_seq"', 3, true);
```

```
-- Name: Conferences Conferences pkey; Type: CONSTRAINT; Schema: public; Owner:
alinapotapova
ALTER TABLE ONLY public."Conferences"
  ADD CONSTRAINT "Conferences pkey" PRIMARY KEY (conference_id);
-- Name: Countries Countries name key; Type: CONSTRAINT; Schema: public; Owner:
alinapotapova
ALTER TABLE ONLY public. "Countries"
  ADD CONSTRAINT "Countries name key" UNIQUE (name);
-- Name: Countries Countries pkey; Type: CONSTRAINT; Schema: public; Owner:
alinapotapova
ALTER TABLE ONLY public. "Countries"
  ADD CONSTRAINT "Countries pkey" PRIMARY KEY (country id);
-- Name: Degrees Degrees name key; Type: CONSTRAINT; Schema: public; Owner:
alinapotapova
ALTER TABLE ONLY public. "Degrees"
  ADD CONSTRAINT "Degrees name key" UNIQUE (name);
-- Name: Degrees Degrees pkey; Type: CONSTRAINT; Schema: public; Owner: alinapotapova
ALTER TABLE ONLY public. "Degrees"
  ADD CONSTRAINT "Degrees pkey" PRIMARY KEY (degree id);
-- Name: Organizations Organizations name key; Type: CONSTRAINT; Schema: public;
Owner: alinapotapova
ALTER TABLE ONLY public. "Organizations"
  ADD CONSTRAINT "Organizations name key" UNIQUE (name);
```

```
-- Name: Organizations Organizations pkey; Type: CONSTRAINT; Schema: public; Owner:
alinapotapova
ALTER TABLE ONLY public."Organizations"
  ADD CONSTRAINT "Organizations pkey" PRIMARY KEY (organization id);
-- Name: Participations Participations pkey; Type: CONSTRAINT; Schema: public; Owner:
alinapotapova
ALTER TABLE ONLY public."Participations"
  ADD CONSTRAINT "Participations pkey" PRIMARY KEY (participation id);
-- Name: Persons Persons pkey; Type: CONSTRAINT; Schema: public; Owner: alinapotapova
ALTER TABLE ONLY public. "Persons"
  ADD CONSTRAINT "Persons pkey" PRIMARY KEY (person id);
-- Name: Scientists Scientists pkey; Type: CONSTRAINT; Schema: public; Owner:
alinapotapova
ALTER TABLE ONLY public. "Scientists"
  ADD CONSTRAINT "Scientists pkey" PRIMARY KEY (scientist id);
-- Name: Specializations Specializations name key; Type: CONSTRAINT; Schema: public;
Owner: alinapotapova
ALTER TABLE ONLY public. "Specializations"
  ADD CONSTRAINT "Specializations name key" UNIQUE (name);
-- Name: Specializations Specializations pkey; Type: CONSTRAINT; Schema: public; Owner:
alinapotapova
ALTER TABLE ONLY public. "Specializations"
  ADD CONSTRAINT "Specializations pkey" PRIMARY KEY (specialization id);
```

```
-- Name: Types Types name key; Type: CONSTRAINT; Schema: public; Owner:
alinapotapova
ALTER TABLE ONLY public. "Types"
  ADD CONSTRAINT "Types name key" UNIQUE (name);
-- Name: Types Types pkey; Type: CONSTRAINT; Schema: public; Owner: alinapotapova
ALTER TABLE ONLY public. "Types"
  ADD CONSTRAINT "Types pkey" PRIMARY KEY (type id);
-- Name: Conferences check conference date trigger; Type: TRIGGER; Schema: public;
Owner: alinapotapova
CREATE TRIGGER check conference date trigger AFTER INSERT ON public."Conferences"
FOR EACH ROW EXECUTE FUNCTION public.check conference date();
-- Name: Conferences Conferences country id fkey; Type: FK CONSTRAINT; Schema:
public; Owner: alinapotapova
ALTER TABLE ONLY public. "Conferences"
  ADD CONSTRAINT "Conferences country id fkey" FOREIGN KEY (country id)
REFERENCES public. "Countries" (country id) ON UPDATE CASCADE ON DELETE
CASCADE;
-- Name: Participations Participations conference id fkey; Type: FK CONSTRAINT; Schema:
public; Owner: alinapotapova
ALTER TABLE ONLY public. "Participations"
  ADD CONSTRAINT "Participations conference id fkey" FOREIGN KEY (conference id)
REFERENCES public. "Conferences" (conference id) ON UPDATE CASCADE ON DELETE
CASCADE;
-- Name: Participations Participations scientist id fkey; Type: FK CONSTRAINT; Schema:
public; Owner: alinapotapova
```

ALTER TABLE ONLY public. "Participations"

ADD CONSTRAINT "Participations_scientist_id_fkey" FOREIGN KEY (scientist_id) REFERENCES public. "Scientists" (scientist_id) ON UPDATE CASCADE ON DELETE CASCADE;

--

-- Name: Participations Participations_type_id_fkey; Type: FK CONSTRAINT; Schema: public; Owner: alinapotapova

--

ALTER TABLE ONLY public."Participations"

ADD CONSTRAINT "Participations_type_id_fkey" FOREIGN KEY (type_id) REFERENCES public."Types"(type_id) ON UPDATE CASCADE ON DELETE SET NULL;

--

-- Name: Scientists Scientists_country_id_fkey; Type: FK CONSTRAINT; Schema: public; Owner: alinapotapova

--

ALTER TABLE ONLY public. "Scientists"

ADD CONSTRAINT "Scientists_country_id_fkey" FOREIGN KEY (country_id) REFERENCES public."Countries"(country_id) ON UPDATE CASCADE ON DELETE SET NULL;

--

-- Name: Scientists Scientists_degree_id_fkey; Type: FK CONSTRAINT; Schema: public; Owner: alinapotapova

--

ALTER TABLE ONLY public. "Scientists"

ADD CONSTRAINT "Scientists_degree_id_fkey" FOREIGN KEY (degree_id) REFERENCES public."Degrees"(degree_id) ON UPDATE CASCADE ON DELETE SET NULL;

--

-- Name: Scientists Scientists_organization_id_fkey; Type: FK CONSTRAINT; Schema: public; Owner: alinapotapova

--

ALTER TABLE ONLY public. "Scientists"

ADD CONSTRAINT "Scientists_organization_id_fkey" FOREIGN KEY (organization_id) REFERENCES public."Organizations"(organization_id) ON UPDATE CASCADE ON DELETE SET NULL;

--

-- Name: Scientists Scientists_person_id_fkey; Type: FK CONSTRAINT; Schema: public; Owner: alinapotapova

ALTER TABLE ONLY public. "Scientists" ADD CONSTRAINT "Scientists_person_id_fkey" FOREIGN KEY (person_id) REFERENCES public."Persons"(person id) ON UPDATE CASCADE ON DELETE CASCADE; -- Name: Scientists Scientists specialization id fkey; Type: FK CONSTRAINT; Schema: public; Owner: alinapotapova ALTER TABLE ONLY public. "Scientists" ADD CONSTRAINT "Scientists specialization id fkey" FOREIGN KEY (specialization id) REFERENCES public. "Specializations" (specialization id) ON UPDATE CASCADE ON DELETE SET NULL; -- Name: TABLE "Conferences"; Type: ACL; Schema: public; Owner: alinapotapova GRANT SELECT, INSERT, UPDATE ON TABLE public. "Conferences" TO test; -- Name: SEQUENCE "Conferences conference id seq"; Type: ACL; Schema: public; Owner: alinapotapova GRANT SELECT, USAGE ON SEQUENCE public. "Conferences conference id seq" TO test; -- Name: TABLE "Countries"; Type: ACL; Schema: public; Owner: alinapotapova GRANT SELECT ON TABLE public. "Countries" TO test; -- Name: SEQUENCE "Countries country id seq"; Type: ACL; Schema: public; Owner: alinapotapova

GRANT SELECT, USAGE ON SEQUENCE public. "Countries country id seq" TO test;

-- Name: SEQUENCE "Degrees_degree_id_seq"; Type: ACL; Schema: public; Owner: alinapotapova

GRANT SELECT, USAGE ON SEQUENCE public. "Degrees degree id seq" TO test; -- Name: SEQUENCE "Organizations_organization_id_seq"; Type: ACL; Schema: public; Owner: alinapotapova GRANT SELECT, USAGE ON SEQUENCE public. "Organizations organization id seq" TO -- Name: TABLE "Participations"; Type: ACL; Schema: public; Owner: alinapotapova GRANT SELECT ON TABLE public."Participations" TO test; -- Name: SEQUENCE "Participations participation id seq"; Type: ACL; Schema: public; Owner: alinapotapova GRANT SELECT, USAGE ON SEQUENCE public. "Participations participation id seq" TO -- Name: TABLE "Persons"; Type: ACL; Schema: public; Owner: alinapotapova GRANT SELECT ON TABLE public. "Persons" TO test; -- Name: SEQUENCE "Persons person id seq"; Type: ACL; Schema: public; Owner: alinapotapova GRANT SELECT, USAGE ON SEQUENCE public. "Persons person id seq" TO test; -- Name: COLUMN "Scientists".scientist id; Type: ACL; Schema: public; Owner:

GRANT SELECT(scientist id) ON TABLE public. "Scientists" TO test;

alinapotapova

```
-- Name: COLUMN "Scientists".person id; Type: ACL; Schema: public; Owner: alinapotapova
GRANT SELECT(person id) ON TABLE public. "Scientists" TO test;
-- Name: COLUMN "Scientists".degree id; Type: ACL; Schema: public; Owner: alinapotapova
GRANT UPDATE(degree id) ON TABLE public. "Scientists" TO test;
-- Name: COLUMN "Scientists".country id; Type: ACL; Schema: public; Owner: alinapotapova
GRANT SELECT(country id) ON TABLE public. "Scientists" TO test;
-- Name: COLUMN "Scientists".organization id; Type: ACL; Schema: public; Owner:
alinapotapova
GRANT UPDATE(organization id) ON TABLE public. "Scientists" TO test;
-- Name: COLUMN "Scientists".specialization id; Type: ACL; Schema: public; Owner:
alinapotapova
GRANT UPDATE(specialization id) ON TABLE public. "Scientists" TO test;
-- Name: SEQUENCE "Scientists scientist id seq"; Type: ACL; Schema: public; Owner:
alinapotapova
GRANT SELECT, USAGE ON SEQUENCE public. "Scientists_scientist_id_seq" TO test;
-- Name: SEQUENCE "Specializations specialization id seq"; Type: ACL; Schema: public;
Owner: alinapotapova
GRANT SELECT, USAGE ON SEQUENCE public. "Specializations specialization id seq" TO
```

test;

```
-- Name: SEQUENCE "Types_type_id_seq"; Type: ACL; Schema: public; Owner: alinapotapova
-- GRANT SELECT,USAGE ON SEQUENCE public."Types_type_id_seq" TO test;

-- Name: TABLE scientists_number_of_participations; Type: ACL; Schema: public; Owner: alinapotapova
-- GRANT SELECT ON TABLE public.scientists_number_of_participations TO test;

-- Name: COLUMN "Countries_by_number_of_conferences".country; Type: ACL; Schema: public; Owner: alinapotapova
-- GRANT UPDATE(country) ON TABLE public."Countries_by_number_of_conferences" TO test_role;

-- PostgreSQL database dump complete
```