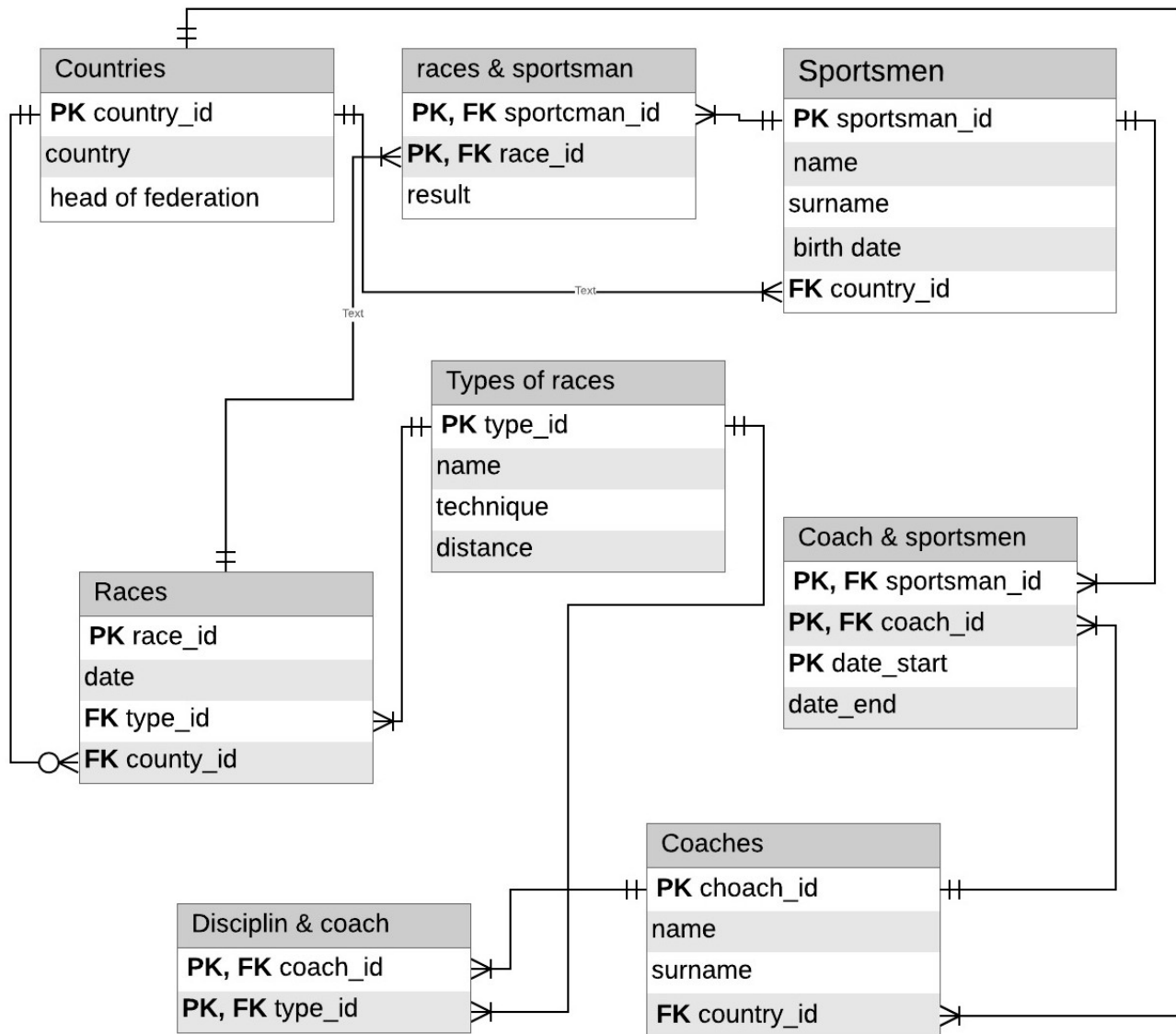


1 ER-диаграмма

Cross-country skiing Men 2018/2019

Виктория Трофимовна Рудых | May 11, 2019



2 Описание базы данных

Cross-country skiing Men 2018/2019: База данных состоит из 8 таблиц, в третьей нормальной форме. Описывает лыжные гонки 2018/2019 сезона. Содержит информацию о спортсменах, гонках, странах, тренерах и дисциплинах.

Sportsmen: Таблица, описывающая спортсменов. **PK** - sportsman_id. Описание спортсмена - его имя, фамилия, дата рождения, страна.

Counties: Таблица, описывающая страны. **PK** - countries_id. Описание страны - название страны, ФИ главы федерации.

Races: Таблица, описывающая гонки. **PK** - race_id. Описание гонки - дата проведения гонки, тип гонки, страна проведения.

Coaches: Таблица, описывающая тренеров. **PK** - coach_id. Описание тренера - его имя, фамилия, страна.

Races & sportsman: Таблица, описывающая связь спортсменов с гонками, т. е. в какой гонке кто участвовал, и сколько баллов при этом набрал. **PK** - sportsman_id, race_id.

Types of races: Таблица, описывающая виды гонок. **PK** - type_id. Описание типа гонок - название дисциплины, техника передвижения (классический стиль, коньковый стиль), дистанция.

Disciplin & coach: Таблица, показывающая связь тренеров и дисциплин, т. е. какой тренер готовит к каким видам гонок. **PK** - type_id, coach_id.

Coach & sportsmen: Таблица, описывающая связь тренеров и спортсменов, т. е. какой тренер тренирует какого спортсмена. Таблица является версионной SCD2, т. е. спортсмен может менять тренера за время карьеры. **PK** - coach_id, sportsman_id, date_start.

3 Представления

Задание: Написать 3-4 осмысленных представления, которые должны скрывать технические поля, или собирать и агрегировать данные из нескольких таблиц.

Разница между представлениями и таблицами базы данных в том, что во всех таблицах связей используются суррогатные ключи, что неинформативно для обычного пользователя.

Sportsmen: Представление таблицы спортсменов, в котором выводится фамилия, имя, страна.

Races: Представление гонок со странами и дистанцией, в котором выводится дата гонки, тип гонки, дистанция, страна проведения.

Races_Sportsman: Представление гонок со спортсменами и результатами, в котором выводится фамилия спортсмена, дата гонки, результат.

4 Триггеры и хранимые функции

Задание: Написать не менее 2 хранимых процедур и не менее 2 триггеров.

age_sp: Функция определяет возраст спортсмена. Отдаем в нее дату рождения спортсмена, и функция вычисляет сколько лет прошло с этого момента до текущего момента.

before_delete: При удалении спортсмена проверяет, не остался ли он последним в своей стране. Если да, то бросает исключение.

Триггер - monitor_sportsmen: Используется для наблюдения за изменениями в таблице спортсменов. При удалении/добавлении/обновлении вызывает вспомогательную функцию, которая во вспомогательную таблицу логов вносит строку add/delete/update и sportsman_id.

5 Роли доступа

Задание: Создать роли доступа и выдать права.

Обычный пользователь - `ordinary_user`: Может делать `select`, чтобы просматривать все таблицы.

Редактор - `editor`: Может работать с таблицами гонок, гонок и спортсменов, тренеров, дисциплин и тренеров, т. к. гонки и тренера - часто меняющиеся таблицью. Может делать `delete/update/insert` в эти таблицы.

Модератор - `moderator`: Может работать со всеми таблицами, т. е. делать `delete/update/insert` во все таблицы.

Администратор - `administrator`: Может работать со всеми таблицами, а также выдавать права на `delete/update/insert` другим пользователям.

6 Приложение

```
create schema cross_country_skiing;

create type _tech as enum ('C', 'K', 'Both');
create type _result as enum ('1', '2', '3');

CREATE TABLE cross_country_skiing.countries(
    country_id SERIAL PRIMARY KEY,
    country VARCHAR(30) NOT NULL UNIQUE,
    head_of_federation varchar(30)
);

create table cross_country_skiing.types_of_races(
    type_id serial primary key,
    name varchar(30) not null,
    technique _tech,
    distance integer
);

create table cross_country_skiing.races (
    race_id serial primary key,
    date date,
    type_id serial not null,
    country_id serial not null,
    foreign key (country_id)
        references cross_country_skiing.countries(country_id),
    foreign key (type_id)
        references cross_country_skiing.types_of_races(type_id)
);

create table cross_country_skiing.sportsmen (
    sportsman_id serial primary key,
    name varchar(30) not null,
    surname varchar(30) not null unique,
    birth_date date,
    country_id serial not null,
    foreign key (country_id)
        references cross_country_skiing.countries(country_id)
);

create table cross_country_skiing.sportsman_race(
    sportsman_id serial not null,
    race_id serial not null,
    result _result,
    primary key(sportsman_id,race_id),
    foreign key(sportsman_id)
        references cross_country_skiing.sportsmen(sportsman_id),
    foreign key(race_id)
        references cross_country_skiing.races(race_id)
```

```

);

create table cross_country_skiing.coaches(
    coach_id serial primary key,
    name varchar(30) not null,
    surname varchar(30) not null unique,
    country_id serial not null,
    foreign key(country_id)
        references cross_country_skiing.countries(country_id)
);

create table cross_country_skiing.coach_sportsman(
    sportsman_id SERIAL NOT NULL,
    coach_id SERIAL NOT NULL,
    date_start smallint not null,
    date_end smallint not null,
    PRIMARY KEY (sportsman_id, coach_id, date_start),
    FOREIGN KEY (sportsman_id)
        REFERENCES cross_country_skiing.sportsmen(sportsman_id),
    FOREIGN KEY (coach_id)
        REFERENCES cross_country_skiing.coaches(coach_id)
);

create table cross_country_skiing.discipline_coach(
    coach_id SERIAL NOT NULL,
    type_id SERIAL NOT NULL,
    PRIMARY KEY (coach_id, type_id),
    FOREIGN KEY (coach_id)
        REFERENCES cross_country_skiing.coaches(coach_id),
    FOREIGN KEY (type_id)
        REFERENCES cross_country_skiing.types_of_races(type_id)
);

```

– Функция определяющая возраст спортсмена, в конце есть пример использования

```

create or replace function age_sp(birth_date date)

```

```

returns int
as $$
declare i int := 0;
begin
    while birth_date < now()
    loop
        birth_date := birth_date + interval '1 year';
        i := i + 1;
    end loop;
    return i;
end;
$$ language plpgsql;

```

– Таблица для тригера

```

CREATE TABLE cross_country_skiing.logs

```

```
(
    operation_id serial not null,
    "info"varchar(100),
    "time"timestamp(0) without time zone
);
```

– Функция для тригера

```
CREATE OR REPLACE FUNCTION add_to_log() RETURNS TRIGGER AS $$
DECLARE
```

```
    mess varchar(100);
```

```
    sp_id varchar(100);
```

```
    final varchar(254);
```

```
BEGIN
```

```
    IF TG_OP = 'INSERT' THEN
```

```
        sp_id = NEW.sportsman_id;
```

```
        mess := 'Added new sportsmen with sportsmen_id = ';
```

```
        final := mess || sp_id;
```

```
        INSERT INTO cross_country_skiing.logs values (default, final, NOW());
```

```
        RETURN NEW;
```

```
    ELSIF TG_OP = 'UPDATE' THEN
```

```
        sp_id = NEW.sportsman_id;
```

```
        mess := 'Updated sportsmen with sportsmen_id ';
```

```
        final := mess || sp_id;
```

```
        INSERT INTO cross_country_skiing.logs values (default, final, NOW());
```

```
        RETURN NEW;
```

```
    ELSIF TG_OP = 'DELETE' THEN
```

```
        sp_id = OLD.sportsman_id;
```

```
        mess := 'Removed user with sportsmen_id';
```

```
        final := mess || sp_id;
```

```
        INSERT INTO cross_country_skiing.logs values (default, final, NOW());
```

```
        RETURN OLD;
```

```
    END IF;
```

```
END;
```

```
$$ LANGUAGE plpgsql;
```

```
CREATE TRIGGER monitor_sportsmen
```

```
AFTER INSERT OR UPDATE or delete ON cross_country_skiing.sportsmen FOR EACH
ROW EXECUTE PROCEDURE add_to_log ();
```

– Функция ограничение на удаление последнего спортсмена, тоже с тригером, чтобы она автоматически запускалась перед попыткой удаления

```
CREATE OR REPLACE FUNCTION before_delete() RETURNS TRIGGER AS $$
```

```
BEGIN
```

```
    if exists (select count(cross_country_skiing.sportsmen.sportsman_id) from cross_country_skiing.spor
```

```
        where cross_country_skiing.sportsmen.country_id = new.country_id
```

```
        group by cross_country_skiing.sportsmen.country_id
```

```
        having count(cross_country_skiing.sportsmen.sportsman_id) > 1)
```

```
    then
```

```
        begin
```

```
            return old;
```

```

        end;
    else
        raise exception 'Sportsman who you are trying to delete is the last from his country';
    end if;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER check_remove
BEFORE delete ON cross_country_skiing.sportsmen FOR EACH ROW EXECUTE PROCEDURE
before_delete ();

insert into cross_country_skiing.countries values(default, 'Russia', 'Elena Vialbe');
insert into cross_country_skiing.countries values(default, 'Norway', 'Eric Roeste');
insert into cross_country_skiing.countries values(default, 'Finland', 'Markku Haapasalmi');
insert into cross_country_skiing.countries values(default, 'Italy', 'Loretta Piroia');
insert into cross_country_skiing.countries values(default, 'Swedish', 'Karin Weijber');
insert into cross_country_skiing.countries values(default, 'German', 'Alfons Hormann');
insert into cross_country_skiing.countries values(default, 'Khazakstan', 'Sorokin Vladimir');
insert into cross_country_skiing.countries values(default, 'USA', 'Simi Hamilton');
insert into cross_country_skiing.countries values(default, 'Canada', 'Tyler King');
insert into cross_country_skiing.countries values(default, 'Austria', 'Peter Schrocksnadel');
insert into cross_country_skiing.countries values(default, 'Switzerland', 'Gian-Franco Kasper');
insert into cross_country_skiing.countries values(default, 'France', 'Jan Kasper');

insert into cross_country_skiing.types_of_races values (default, 'Skiatlon', 'Both', '30');
insert into cross_country_skiing.types_of_races values (default, 'Sprint', 'C', '4');
insert into cross_country_skiing.types_of_races values (default, 'Sprint', 'K', '3');
insert into cross_country_skiing.types_of_races values (default, 'Individual race', 'C', '15');
insert into cross_country_skiing.types_of_races values (default, 'Individual race', 'K', '10');
insert into cross_country_skiing.types_of_races values (default, 'Maraphon', 'C', '50');
insert into cross_country_skiing.types_of_races values (default, 'Maraphon', 'K', '40');
insert into cross_country_skiing.types_of_races values (default, 'Pursuit', 'C', '10');
insert into cross_country_skiing.types_of_races values (default, 'Pursuit', 'K', '15');
insert into cross_country_skiing.types_of_races values (default, 'Relay', 'C', '10');
insert into cross_country_skiing.types_of_races values (default, 'Relay', 'K', '10');
insert into cross_country_skiing.types_of_races values (default, 'Mass start', 'C', '15');
insert into cross_country_skiing.types_of_races values (default, 'Mass start', 'K', '10');

insert into cross_country_skiing.races values(default, '24.11.2018', '2', '3');
insert into cross_country_skiing.races values(default, '25.11.2018', '4', '3');
insert into cross_country_skiing.races values(default, '30.11.2018', '3', '2');
insert into cross_country_skiing.races values(default, '01.12.2018', '5', '2');
insert into cross_country_skiing.races values(default, '02.12.2018', '8', '2');
insert into cross_country_skiing.races values(default, '15.12.2018', '3', '4');
insert into cross_country_skiing.races values(default, '16.12.2018', '5', '4');
insert into cross_country_skiing.races values(default, '05.01.2019', '13', '4');
insert into cross_country_skiing.races values(default, '12.01.2019', '2', '6');
insert into cross_country_skiing.races values(default, '13.01.2019', '6', '6');

```

```

insert into cross_country_skiing.races values(default, '09.02.2019', '1', '3');
insert into cross_country_skiing.races values(default, '10.02.2019', '7', '3');
insert into cross_country_skiing.races values(default, '27.02.2019', '4', '8');
insert into cross_country_skiing.races values(default, '01.03.2019', '10', '8');
insert into cross_country_skiing.races values(default, '03.03.2019', '7', '8');
insert into cross_country_skiing.races values(default, '16.03.2019', '11', '5');
insert into cross_country_skiing.races values(default, '22.03.2019', '12', '9');
insert into cross_country_skiing.races values(default, '23.03.2019', '9', '9');

insert into cross_country_skiing.sportsmen values
    (default, 'Alexander', 'Bolshunov', '31.12.1996', '1');
insert into cross_country_skiing.sportsmen values
    (default, 'Denis', 'Spitsov', '16.08.1996', '1');
insert into cross_country_skiing.sportsmen values
    (default, 'Evgeniy', 'Belov', '07.08.1990', '1');
insert into cross_country_skiing.sportsmen values
    (default, 'Sergey', 'Ustygov', '08.04.1992', '1');
insert into cross_country_skiing.sportsmen values
    (default, 'Johanes', 'Klaebo', '22.10.1996', '2');
insert into cross_country_skiing.sportsmen values
    (default, 'Emil', 'Iversen', '12.08.1991', '2');
insert into cross_country_skiing.sportsmen values
    (default, 'Sjir', 'Rothe', '02.07.1988', '2');
insert into cross_country_skiing.sportsmen values
    (default, 'Simen', 'Kruger', '13.03.1993', '2');
insert into cross_country_skiing.sportsmen values
    (default, 'Martin', 'Sundby', '26.09.1984', '2');
insert into cross_country_skiing.sportsmen values
    (default, 'Janosh', 'Brugger', '01.06.1997', '6');
insert into cross_country_skiing.sportsmen values
    (default, 'Federico', 'Pellegrino', '01.09.1990', '4');
insert into cross_country_skiing.sportsmen values
    (default, 'Francesco', 'De Fabiani', '21.04.1993', '4');
insert into cross_country_skiing.sportsmen values
    (default, 'Adrien', 'Baksheider', '07.08.1992', '10');
insert into cross_country_skiing.sportsmen values
    (default, 'Moris', 'Manificat', '04.04.1986', '10');
insert into cross_country_skiing.sportsmen values
    (default, 'Richard', 'Jouve', '25.10.1994', '10');
insert into cross_country_skiing.sportsmen values
    (default, 'Jean', 'Gailard', '07.10.1980', '10');
insert into cross_country_skiing.sportsmen values
    (default, 'Iivo', 'Niskanen', '12.01.1992', '3');
insert into cross_country_skiing.sportsmen values
    (default, 'Alex', 'Harvey', '07.09.1988', '8');
insert into cross_country_skiing.sportsmen values
    (default, 'Erik', 'Bjorsnen', '14.07.1991', '7');
insert into cross_country_skiing.sportsmen values
    (default, 'Calle', 'Halvarsson', '17.03.1989', '5');

```



```

insert into cross_country_skiing.sportsmen values
      (default, 'Dario', 'Cologna', '11.03.1986', '9');

insert into cross_country_skiing.sportsman_race values('1', '1', '1');
insert into cross_country_skiing.sportsman_race values('5', '1', '2');
insert into cross_country_skiing.sportsman_race values('7', '1', '3');
insert into cross_country_skiing.sportsman_race values('1', '2', '1');
insert into cross_country_skiing.sportsman_race values('6', '2', '2');
insert into cross_country_skiing.sportsman_race values('20', '2', '3');
insert into cross_country_skiing.sportsman_race values('11', '3', '1');
insert into cross_country_skiing.sportsman_race values('6', '3', '2');
insert into cross_country_skiing.sportsman_race values('18', '3', '3');
insert into cross_country_skiing.sportsman_race values('7', '4', '1');
insert into cross_country_skiing.sportsman_race values('8', '4', '2');
insert into cross_country_skiing.sportsman_race values('2', '4', '3');
insert into cross_country_skiing.sportsman_race values('10', '5', '1');
insert into cross_country_skiing.sportsman_race values('16', '5', '2');
insert into cross_country_skiing.sportsman_race values('19', '5', '3');
insert into cross_country_skiing.sportsman_race values('5', '6', '1');
insert into cross_country_skiing.sportsman_race values('11', '6', '2');
insert into cross_country_skiing.sportsman_race values('13', '6', '3');
insert into cross_country_skiing.sportsman_race values('3', '7', '1');
insert into cross_country_skiing.sportsman_race values('14', '7', '2');
insert into cross_country_skiing.sportsman_race values('9', '7', '3');
insert into cross_country_skiing.sportsman_race values('5', '8', '1');
insert into cross_country_skiing.sportsman_race values('12', '8', '2');
insert into cross_country_skiing.sportsman_race values('1', '8', '3');
insert into cross_country_skiing.sportsman_race values('5', '9', '1');
insert into cross_country_skiing.sportsman_race values('4', '9', '2');
insert into cross_country_skiing.sportsman_race values('8', '9', '3');
insert into cross_country_skiing.sportsman_race values('21', '10', '1');
insert into cross_country_skiing.sportsman_race values('2', '10', '2');
insert into cross_country_skiing.sportsman_race values('6', '10', '3');
insert into cross_country_skiing.sportsman_race values('17', '11', '1');
insert into cross_country_skiing.sportsman_race values('10', '11', '2');
insert into cross_country_skiing.sportsman_race values('7', '11', '3');
insert into cross_country_skiing.sportsman_race values('9', '12', '1');
insert into cross_country_skiing.sportsman_race values('7', '12', '2');
insert into cross_country_skiing.sportsman_race values('5', '12', '3');
insert into cross_country_skiing.sportsman_race values('20', '13', '1');
insert into cross_country_skiing.sportsman_race values('1', '13', '2');
insert into cross_country_skiing.sportsman_race values('6', '13', '3');
insert into cross_country_skiing.sportsman_race values('5', '15', '1');
insert into cross_country_skiing.sportsman_race values('11', '15', '2');
insert into cross_country_skiing.sportsman_race values('1', '15', '3');
insert into cross_country_skiing.sportsman_race values('3', '17', '1');
insert into cross_country_skiing.sportsman_race values('8', '17', '2');
insert into cross_country_skiing.sportsman_race values('7', '17', '3');
insert into cross_country_skiing.sportsman_race values('8', '18', '1');
insert into cross_country_skiing.sportsman_race values('9', '18', '2');

```

```

insert into cross_country_skiing.sportsman_race values('21', '18', '3');
insert into cross_country_skiing.sportsman_race values('5', '14', '1');
insert into cross_country_skiing.sportsman_race values('6', '14', '1');
insert into cross_country_skiing.sportsman_race values('7', '14', '1');
insert into cross_country_skiing.sportsman_race values('9', '14', '1');
insert into cross_country_skiing.sportsman_race values('1', '14', '2');
insert into cross_country_skiing.sportsman_race values('2', '14', '2');
insert into cross_country_skiing.sportsman_race values('3', '14', '2');
insert into cross_country_skiing.sportsman_race values('4', '14', '2');
insert into cross_country_skiing.sportsman_race values('13', '14', '3');
insert into cross_country_skiing.sportsman_race values('14', '14', '3');
insert into cross_country_skiing.sportsman_race values('15', '14', '3');
insert into cross_country_skiing.sportsman_race values('16', '14', '3');
insert into cross_country_skiing.sportsman_race values('1', '16', '1');
insert into cross_country_skiing.sportsman_race values('2', '16', '1');
insert into cross_country_skiing.sportsman_race values('3', '16', '1');
insert into cross_country_skiing.sportsman_race values('4', '16', '1');
insert into cross_country_skiing.sportsman_race values('5', '16', '2');
insert into cross_country_skiing.sportsman_race values('6', '16', '2');
insert into cross_country_skiing.sportsman_race values('7', '16', '2');
insert into cross_country_skiing.sportsman_race values('8', '16', '2');
insert into cross_country_skiing.sportsman_race values('13', '16', '3');
insert into cross_country_skiing.sportsman_race values('14', '16', '3');
insert into cross_country_skiing.sportsman_race values('15', '16', '3');
insert into cross_country_skiing.sportsman_race values('16', '16', '3');

insert into cross_country_skiing.coaches values (default, 'Marcus', 'Cramer', '6');
insert into cross_country_skiing.coaches values (default, 'Uriy', 'Borodavko', '1');
insert into cross_country_skiing.coaches values (default, 'Arlid', 'Monser', '2');
insert into cross_country_skiing.coaches values (default, 'Petter', 'Brandsdal', '2');
insert into cross_country_skiing.coaches values (default, 'Gury', 'Hetland', '2');
insert into cross_country_skiing.coaches values (default, 'Alonzo', 'Burgmaster', '9');
insert into cross_country_skiing.coaches values (default, 'Doren', 'Aber', '10');
insert into cross_country_skiing.coaches values (default, 'Kerty', 'Niskanen', '3');
insert into cross_country_skiing.coaches values (default, 'Zibert', 'Clause', '6');
insert into cross_country_skiing.coaches values (default, 'Martin', 'Weng', '2');
insert into cross_country_skiing.coaches values (default, 'Sam', 'Wilson', '8');
insert into cross_country_skiing.coaches values (default, 'Ricard', 'Grenborg', '5');
insert into cross_country_skiing.coaches values (default, 'Yanne', 'Anderson', '5');

insert into cross_country_skiing.coach_sportsman values ('4', '1', '2002', '3000');
insert into cross_country_skiing.coach_sportsman values ('1', '2', '2006', '3000');
insert into cross_country_skiing.coach_sportsman values ('2', '1', '2002', '3000');
insert into cross_country_skiing.coach_sportsman values ('3', '2', '2002', '3000');
insert into cross_country_skiing.coach_sportsman values ('5', '3', '2002', '3000');
insert into cross_country_skiing.coach_sportsman values ('6', '4', '2003', '3000');
insert into cross_country_skiing.coach_sportsman values ('7', '4', '2004', '3000');
insert into cross_country_skiing.coach_sportsman values ('8', '3', '2008', '2016');
insert into cross_country_skiing.coach_sportsman values ('7', '3', '2017', '3000');
insert into cross_country_skiing.coach_sportsman values ('9', '4', '2007', '3000');

```



```

insert into cross_country_skiing.discipline_coach values ('9', '5');
insert into cross_country_skiing.discipline_coach values ('9', '13');
insert into cross_country_skiing.discipline_coach values ('10', '1');
insert into cross_country_skiing.discipline_coach values ('10', '2');
insert into cross_country_skiing.discipline_coach values ('10', '3');
insert into cross_country_skiing.discipline_coach values ('10', '4');
insert into cross_country_skiing.discipline_coach values ('10', '5');
insert into cross_country_skiing.discipline_coach values ('10', '6');
insert into cross_country_skiing.discipline_coach values ('10', '7');
insert into cross_country_skiing.discipline_coach values ('10', '8');
insert into cross_country_skiing.discipline_coach values ('10', '9');
insert into cross_country_skiing.discipline_coach values ('10', '10');
insert into cross_country_skiing.discipline_coach values ('10', '11');
insert into cross_country_skiing.discipline_coach values ('10', '12');
insert into cross_country_skiing.discipline_coach values ('10', '13');
insert into cross_country_skiing.discipline_coach values ('11', '2');
insert into cross_country_skiing.discipline_coach values ('11', '3');
insert into cross_country_skiing.discipline_coach values ('12', '4');
insert into cross_country_skiing.discipline_coach values ('12', '6');
insert into cross_country_skiing.discipline_coach values ('13', '8');
insert into cross_country_skiing.discipline_coach values ('13', '5');

```

– Выбираем всех победителей гонок, сортируем по возрасту

```

select t.name, t.surname
  from (
    select distinct cross_country_skiing.sportsmen.name, cross_country_skiing.sportsmen.surname,
           cross_country_skiing.sportsmen.birth_date
    from cross_country_skiing.sportsmen, cross_country_skiing.sportsman_race
    where cross_country_skiing.sportsmen.sportsman_id = cross_country_skiing.sportsman_race.sportsman_id
    and cross_country_skiing.sportsman_race.result = '1'
  ) as t
 order by t.birth_date desc;

```

– Считаем кол-во спортсменов по странам

```

select distinct cross_country_skiing.countries.country,
       count(cross_country_skiing.sportsmen.surname) as t
  from cross_country_skiing.countries, cross_country_skiing.sportsmen
 where cross_country_skiing.sportsmen.country_id = cross_country_skiing.countries.country_id
 group by cross_country_skiing.countries.country
 order by t desc;

```

– Считаем кол-во подиумов у спортсменов

```

select cross_country_skiing.sportsmen.surname,
       count(cross_country_skiing.sportsman_race.result) as t
  from cross_country_skiing.sportsmen, cross_country_skiing.sportsman_race
 where cross_country_skiing.sportsmen.sportsman_id = cross_country_skiing.sportsman_race.sportsman_id
 group by cross_country_skiing.sportsmen.surname
 order by t desc;

```

– Считаем кол-во гонок каждого типа

```
select cross_country_skiing.types_of_races.name,  
       count(cross_country_skiing.races.race_id) as t  
from cross_country_skiing.types_of_races, cross_country_skiing.races  
where cross_country_skiing.types_of_races.type_id = cross_country_skiing.races.type_id  
group by cross_country_skiing.types_of_races.name  
order by t desc;
```

– Выводим дистанции, на которых российские спортсмены были в призах хотя бы 2 раза

```
select cross_country_skiing.types_of_races.distance, count(cross_country_skiing.sportsmen.surname)  
from cross_country_skiing.races, cross_country_skiing.types_of_races, cross_country_skiing.sportsmen,  
cross_country_skiing.sportsmen,  
cross_country_skiing.countries  
where cross_country_skiing.races.type_id = cross_country_skiing.types_of_races.type_id  
and cross_country_skiing.sportsman_race.race_id = cross_country_skiing.races.race_id  
and cross_country_skiing.sportsman_race.sportsman_id = cross_country_skiing.sportsmen.sportsman_id  
and cross_country_skiing.sportsmen.country_id = cross_country_skiing.countries.country_id  
and cross_country_skiing.countries.country = 'Russia'  
group by cross_country_skiing.types_of_races.distance  
having count(cross_country_skiing.sportsmen.surname) > 1;
```

– Представление таблицы спортсменов

```
create view Sportsmen  
as select cross_country_skiing.sportsmen.surname, cross_country_skiing.sportsmen.name, cross_country_skiing.countries.country  
from cross_country_skiing.sportsmen, cross_country_skiing.countries  
where cross_country_skiing.sportsmen.country_id = cross_country_skiing.countries.country_id;
```

– Представление гонок со странами и дистанцией

```
create view Races  
as select cross_country_skiing.races.date, cross_country_skiing.types_of_races.name as type,  
       cross_country_skiing.types_of_races.distance, cross_country_skiing.countries.country  
from cross_country_skiing.countries, cross_country_skiing.types_of_races, cross_country_skiing.races  
where cross_country_skiing.countries.country_id = cross_country_skiing.races.country_id  
and cross_country_skiing.races.type_id = cross_country_skiing.types_of_races.type_id;
```

– Представление гонок со спортсменами и результатами

```
create view Races_Sportsman  
as select cross_country_skiing.sportsmen.surname as sportsman, cross_country_skiing.races.date,  
cross_country_skiing.sportsman_race.result  
from cross_country_skiing.races, cross_country_skiing.sportsman_race, cross_country_skiing.sportsmen  
where cross_country_skiing.sportsmen.sportsman_id = cross_country_skiing.sportsman_race.sportsman_id  
and cross_country_skiing.sportsman_race.race_id = cross_country_skiing.races.race_id;
```

– Пример для использования age_sp()

```
select cross_country_skiing.sportsmen.name, cross_country_skiing.sportsmen.surname, age_sp(cross_country_skiing.sportsmen.date_of_birth,  
as age  
from cross_country_skiing.sportsmen  
order by age;
```

```

create role ordinary_user;
create role editor with password '1234';
create role moderator with password '12345678';
create role administrator with password 'qwerty1234';

grant select on cross_country_skiing.sportsmen, cross_country_skiing.countries, cross_country_skiing.ra
    cross_country_skiing.sportsman_race, cross_country_skiing.discipline_coach,
    cross_country_skiing.types_of_races, cross_country_skiing.coach_sportsman, cross_country_skiing
to ordinary_user;

grant select on cross_country_skiing.sportsmen, cross_country_skiing.countries, cross_country_skiing.ra
    cross_country_skiing.sportsman_race, cross_country_skiing.discipline_coach,
    cross_country_skiing.types_of_races, cross_country_skiing.coach_sportsman, cross_country_skiing
to editor;

grant update, insert, delete on cross_country_skiing.races, cross_country_skiing.sportsman_race,
    cross_country_skiing.coaches, cross_country_skiing.coach_sportsman to editor;

grant select, update, insert, delete on cross_country_skiing.sportsmen, cross_country_skiing.countries,
cross_country_skiing.races,
    cross_country_skiing.sportsman_race, cross_country_skiing.discipline_coach,
    cross_country_skiing.types_of_races, cross_country_skiing.coach_sportsman, cross_country_skiing
to moderator;

grant update, insert, delete on cross_country_skiing.sportsmen, cross_country_skiing.countries,
cross_country_skiing.races,
    cross_country_skiing.sportsman_race, cross_country_skiing.discipline_coach,
    cross_country_skiing.types_of_races, cross_country_skiing.coach_sportsman, cross_country_skiing
to administrator with grant option;

```