**Отчёт по лабораторной работе №3**

Выполнил студент группы ИВТ(ВМК)-17-2:

Ковалёв Владимир Сергеевич

**Цель работы**

Реализация декларативных ограничений целостности; проектирование таблиц БД с использованием pgModeler и pgAdmin 4 (PostgreSQL)

**Содержание работы**

* Выявление декларативных и процедурных ограничений целостности предметной области, создание ограничений в pgModeler;
* Создание БД и таблиц в pgAdmin 4;
* Реализация декларативных ограничений целостности в pgAdmin 4

**Ход работы**

Предметная область:

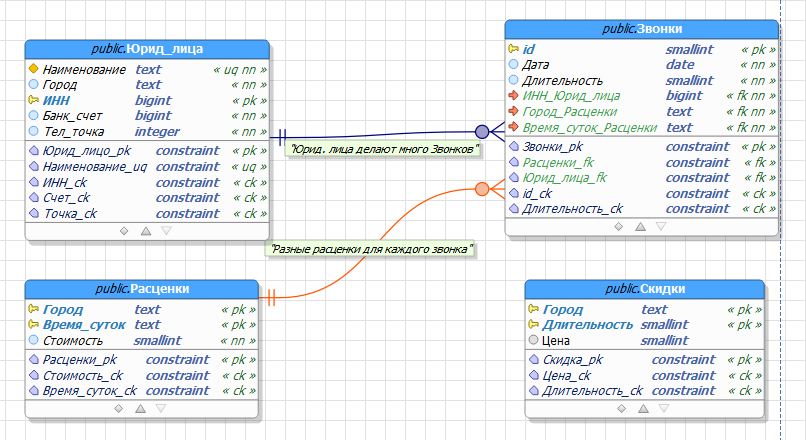
Вы работаете в коммерческой службе телефонной компании. Компания предоставляет абонентам телефонные линии для междугородних переговоров. Вашей задачей является отслеживание стоимости междугородних телефонных переговоров. Абонентами компании являются юридические лица, имеющие телефонную точку, ИНН, расчетный счет в банке. Стоимость переговоров зависит от города, в который осуществляется звонок, и времени суток (день, ночь). Каждый звонок абонента автоматически фиксируется в базе данных. При этом запоминаются город, дата, длительность разговора и время суток. Фирма вводит гибкую систему скидок. Так, стоимость минуты теперь уменьшается в зависимости от длительности разговора. Размер скидки для каждого города разный.

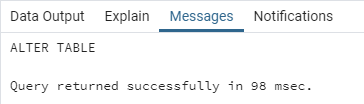
Рисунок 1. Диаграмма ER

Проанализировав данные и бизнес-правила предметной области, были выявлены следующие декларативные ограничения целостности:

|  |  |
| --- | --- |
| Описание ограничений | Код SQL |
| **«Юрид\_лица»** | |
| Все поля в таблице не должны содержать и принимать пустые значения. | "Наименование" text NOT NULL,  "Город" text NOT NULL, "ИНН" bigint NOT NULL,  "Банк\_счет" bigint NOT NULL,  "Тел\_точка" integer NOT NULL, |
| Поле «ИНН» может выступать в качестве первичного ключа, так как данный номер является уникальным и однозначно идентифицирующим. | CONSTRAINT "Юрид\_лицо\_pk" PRIMARY KEY ("ИНН") |
| Поле «Наименование» может выступать в качестве уникального ключа. | CONSTRAINT "Название\_фирмы" UNIQUE ("Наименование") |
| Значения поля «ИНН» должны быть больше нуля и содержать 10 цифр. | CONSTRAINT "ИНН\_из\_10\_цифр" CHECK (ИНН between 1000000000 and 9999999999) |
| Значения поля «Банк\_счет» должны быть больше нуля и содержать 12 цифр | CONSTRAINT "Cчет\_из\_12\_цифр" CHECK (Банк\_счет between 100000000000 and 999999999999) |
| Значения поля «Тел\_точка» должны быть больше нуля. | CONSTRAINT "Номер\_точки\_больше\_0" CHECK (Тел\_точка > 0) |
| При обновлении/удалении абонента следует обновить/удалить записи об его звонках. | ON DELETE CASCADE ON UPDATE CASCADE |
| **«Расценки»** | |
| Все поля в таблице не должны содержать и принимать пустые значения. | "Город" text NOT NULL DEFAULT 'Неизвестно',  "Время\_суток" text NOT NULL DEFAULT 'день',  "Стоимость" smallint NOT NULL DEFAULT 10 |
| Поля «Город» и «Время\_суток» вместе могут выступать в качестве составного первичного ключа, поскольку стоимость минуты зависит от города, куда совершается звонок, и от времени суток. | CONSTRAINT "Расценки\_pk" PRIMARY KEY ("Город","Время\_суток") |
| Значения поля «Стоимость» должны быть больше нуля. | CONSTRAINT "Стоимость\_больше\_0" CHECK (Стоимость > 0) |
| Значения поля «Время\_суток» должны принимать только два значения – «день» и «ночь» | CONSTRAINT "Только\_день\_или\_ночь" CHECK (Время\_суток in ('день', 'ночь')) |
| Поля «Город», «Время\_суток», «Стоимость» должны иметь значения по-умолчанию. | "Город" text NOT NULL DEFAULT 'Неизвестно', "Время\_суток" text NOT NULL DEFAULT 'день', "Стоимость" smallint NOT NULL DEFAULT 10, |
| **«Скидки»** | |
| Все поля в таблице не должны содержать и принимать пустые значения. | "Город" text NOT NULL,  "Длительность" smallint NOT NULL,  "Цена" smallint NOT NULL |
| Поля «Город» и «Длительность» вместе могут выступать в качестве составного первичного ключа, поскольку размер скидки может быть разными в зависимости от названия города и длительности разговора. | CONSTRAINT "Скидка\_pk" PRIMARY KEY ("Город","Длительность") |
| Значения поля «Цена» должны быть строго больше нуля. | CONSTRAINT "Цена\_больше\_0" CHECK (Цена > 0) |
| Значения поля «Длительность» должны быть строго больше нуля. | CONSTRAINT "Длительность\_больше\_0" CHECK (Длительность > 0) |
| **«Звонки»** | |
| Все поля в таблице не должны содержать и принимать пустые значения. | id smallint NOT NULL,  "Дата" date NOT NULL,  "Длительность" smallint NOT NULL,  "ИНН" bigint NOT NULL, "Город" text NOT NULL, "Время\_суток" text NOT NULL |
| Поле «id» может выступать в качестве первичного ключа, каждый номер должен фиксироваться в базе данных. | CONSTRAINT "Звонки\_pk" PRIMARY KEY (id) |
| Значения поля «id» должны быть больше нуля. | CONSTRAINT "id\_больше\_0" CHECK (id > 0) |
| Значения поля «Длительность» должны быть строго больше нуля. | CONSTRAINT "Длительность\_больше\_0" CHECK (Длительность > 0) |
| Так как абоненты могут делать множество звонков, то следует создать внешний ключ «ИНН» | "ИНН" bigint NOT NULL  ALTER TABLE public."Звонки" ADD CONSTRAINT "Юрид\_лица\_fk" FOREIGN KEY ("ИНН")  REFERENCES public."Юрид\_лица" ("ИНН") MATCH FULL |
| Так как для каждого звонка может быть своя расценка, то следует создать внешние ключи «Город» и «Время\_суток» | "Город" text NOT NULL, "Время\_суток" text NOT NULL  ALTER TABLE public."Звонки" ADD CONSTRAINT "Расценки\_fk" FOREIGN KEY ("Город","Время\_суток")  REFERENCES public."Расценки" ("Город","Время\_суток") MATCH FULL |
| При изменении информации он расценке следует обновить запись о звонке. Удаление записей из данной таблицы ограничено, если есть ссылки в таблице "Звонки". | ALTER TABLE public."Звонки" ADD CONSTRAINT "Расценки\_fk" FOREIGN KEY ("Город","Время\_суток")  REFERENCES public."Расценки" ("Город","Время\_суток") MATCH FULL  ON DELETE RESTRICT ON UPDATE CASCADE; |

Таблица 1. Перечень декларативных ограничений целостности.

Вносим вышеперечисленные изменения в сгенерированный SQL-скрипт (приложение 1). Далее мы должны его запустить на сервере PostgreSQL через программу pgAdmin версии 4. Для данной лабораторной работы таблицы будут храниться в базе данных «db\_telephone» (владелец postgres) в схеме «public».

С помощью «Query Tool» импортируем данный SQL-скрипт в базу данных и запускаем его. В результате должны получить следующее сообщение:

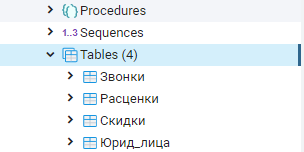
Рисунок 1. Результат работы SQL-скрипта в консоли

Рисунок 2. Таблицы в базе данных «db\_telephone»

После создания таблиц необходимо их заполнить кортежами. Это можно сделать с помощью INSERT запросов, но мы будем использовать инструмент «View/Edit Data». Далее в окне «Data Output» мы заносим различные кортежи в каждую из таблиц.

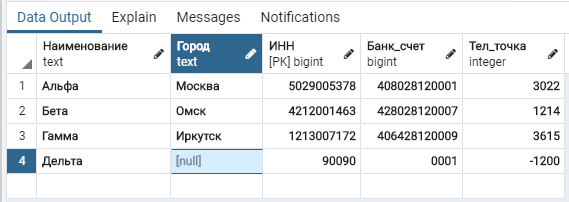
Для примера возьмём таблицу «Юрид\_лица» и заполним её следующими данными:

Рисунок 3. Таблица «Юрид\_лица» с данными

Поскольку четвертая строка содержит неверно введенные данные, программа не занесет информацию в таблицу и сообщит, чтобы были нарушены различные декларативные ограничения целостности:

* ОШИБКА: нулевое значение в столбце "Город" нарушает ограничение **NOT NULL** DETAIL: Ошибочная строка содержит (Дельта, **null**, 90090, 1, -1200).
* ОШИБКА: новая строка в отношении "Юрид\_лица" нарушает ограничение-проверку "**Cчет\_из\_12\_цифр**" DETAIL: Ошибочная строка содержит (Дельта, Сочи, 90090, **1**, -1200).
* ОШИБКА: новая строка в отношении "Юрид\_лица" нарушает ограничение-проверку "**ИНН\_из\_10\_цифр**" DETAIL: Ошибочная строка содержит (Дельта, Сочи, **90090**, 628458120001, -1200).
* ОШИБКА: новая строка в отношении "Юрид\_лица" нарушает ограничение-проверку "Номер\_точки\_больше\_0" DETAIL: Ошибочная строка содержит (Дельта, Сочи, 9009003301, 628458120001, **-1200**).

После внесения правильных данных в четвертую строку и остальной информации в таблицу, программа успешно их занесет и выведет сообщение: «Data saved successfully».

Для проверки ссылочной целостности, для примера занесём в таблицу «Расценки» и «Звонки» следующие данные:

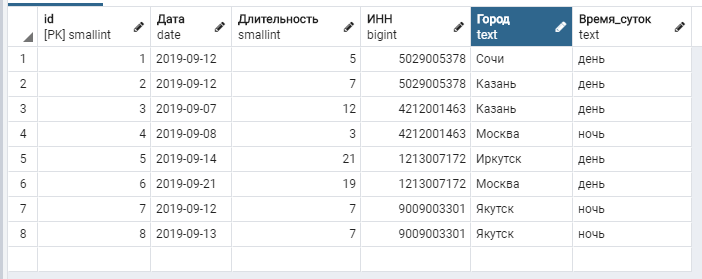


Рисунок 4. Таблица «Звонки» с данными.

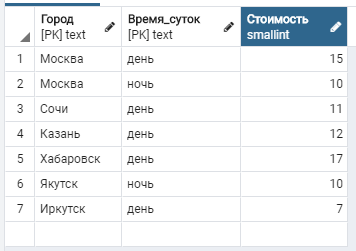


Рисунок 5. Таблица «Расценки» с данными.

Важно подчеркнуть, что сначала следует заполнять таблицу «Расценки», а затем «Звонки». Иначе, если заполнять таблицу «Звонки» первой, то программа сообщит, что было нарушено декларативное ограничение целостности внешнего ключа:

При обновлении данных в таблице «Расценки» значения соответствующих внешних ключей в таблице «Звонки» также обновляться.

При удалении данных из таблицы «Расценки» программа сообщит, что в таблице «Звонки» могут быть ссылки на первичный ключ, и при наличии этих ссылок запретит удаление записей.

|  |  |
| --- | --- |
| Таблица «Расценки» при обновлении | Таблица «Звонки» при обновлении |
|  |  |
| Таблица «Расценки» при удалении | Таблица «Звонки» при удалении |
|  | ОШИБКА: UPDATE или DELETE в таблице "Расценки" нарушает ограничение внешнего ключа "Расценки\_fk" таблицы "Звонки" DETAIL: На ключ (Город, Время\_суток)=(Чита, день) всё ещё есть ссылки в таблице "Звонки". |

**Приложение 1**

**SQL-скрипт**

-- Database generated with pgModeler (PostgreSQL Database Modeler).

-- pgModeler version: 0.9.2-beta1

-- PostgreSQL version: 11.0

-- Project Site: pgmodeler.io

-- Model Author: ---

-- Database creation must be done outside a multicommand file.

-- These commands were put in this file only as a convenience.

-- -- object: db\_telephone | type: DATABASE --

-- -- DROP DATABASE IF EXISTS db\_telephone;

-- CREATE DATABASE db\_telephone;

-- -- ddl-end --

--

-- object: public."Юрид\_лица" | type: TABLE --

-- DROP TABLE IF EXISTS public."Юрид\_лица" CASCADE;

CREATE TABLE public."Юрид\_лица" (

"Наименование" text NOT NULL,

"Город" text NOT NULL,

"ИНН" bigint NOT NULL,

"Банк\_счет" bigint NOT NULL,

"Тел\_точка" integer NOT NULL,

CONSTRAINT "Юрид\_лицо\_pk" PRIMARY KEY ("ИНН"),

CONSTRAINT "Название\_фирмы" UNIQUE ("Наименование"),

CONSTRAINT "ИНН\_из\_10\_цифр" CHECK (ИНН between 1000000000 and 9999999999),

CONSTRAINT "Cчет\_из\_12\_цифр" CHECK (Банк\_счет between 100000000000 and 999999999999),

CONSTRAINT "Номер\_точки\_больше\_0" CHECK (Тел\_точка > 0)

);

-- ddl-end --

ALTER TABLE public."Юрид\_лица" OWNER TO postgres;

-- ddl-end --

-- object: public."Звонки" | type: TABLE --

-- DROP TABLE IF EXISTS public."Звонки" CASCADE;

CREATE TABLE public."Звонки" (

id smallint NOT NULL,

"Дата" date NOT NULL,

"Длительность" smallint NOT NULL,

"ИНН" bigint NOT NULL,

"Город" text NOT NULL,

"Время\_суток" text NOT NULL,

CONSTRAINT "Звонки\_pk" PRIMARY KEY (id),

CONSTRAINT "id\_больше\_0" CHECK (id > 0),

CONSTRAINT "Длительность\_больше\_0" CHECK (Длительность > 0)

);

-- ddl-end --

ALTER TABLE public."Звонки" OWNER TO postgres;

-- ddl-end --

-- object: public."Расценки" | type: TABLE --

-- DROP TABLE IF EXISTS public."Расценки" CASCADE;

CREATE TABLE public."Расценки" (

"Город" text NOT NULL DEFAULT 'Неизвестно',

"Время\_суток" text NOT NULL DEFAULT 'день',

"Стоимость" smallint NOT NULL DEFAULT 10,

CONSTRAINT "Расценки\_pk" PRIMARY KEY ("Город","Время\_суток"),

CONSTRAINT "Стоимость\_больше\_0" CHECK (Стоимость > 0),

CONSTRAINT "Только\_день\_или\_ночь" CHECK (Время\_суток in ('день', 'ночь'))

);

-- ddl-end --

ALTER TABLE public."Расценки" OWNER TO postgres;

-- ddl-end --

-- object: "Юрид\_лица\_fk" | type: CONSTRAINT --

-- ALTER TABLE public."Звонки" DROP CONSTRAINT IF EXISTS "Юрид\_лица\_fk" CASCADE;

ALTER TABLE public."Звонки" ADD CONSTRAINT "Юрид\_лица\_fk" FOREIGN KEY ("ИНН")

REFERENCES public."Юрид\_лица" ("ИНН") MATCH FULL

ON DELETE CASCADE ON UPDATE CASCADE;

-- ddl-end --

-- object: public."Скидки" | type: TABLE --

-- DROP TABLE IF EXISTS public."Скидки" CASCADE;

CREATE TABLE public."Скидки" (

"Город" text NOT NULL,

"Длительность" smallint NOT NULL,

"Цена" smallint NOT NULL,

CONSTRAINT "Скидка\_pk" PRIMARY KEY ("Город","Длительность"),

CONSTRAINT "Цена\_больше\_0" CHECK (Цена > 0),

CONSTRAINT "Длительность\_больше\_0" CHECK (Длительность > 0)

);

-- ddl-end --

ALTER TABLE public."Скидки" OWNER TO postgres;

-- ddl-end --

-- object: "Расценки\_fk" | type: CONSTRAINT --

-- ALTER TABLE public."Звонки" DROP CONSTRAINT IF EXISTS "Расценки\_fk" CASCADE;

ALTER TABLE public."Звонки" ADD CONSTRAINT "Расценки\_fk" FOREIGN KEY ("Город","Время\_суток")

REFERENCES public."Расценки" ("Город","Время\_суток") MATCH FULL

ON DELETE RESTRICT ON UPDATE CASCADE;

-- ddl-end --