Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Институт информатики и вычислительной техники

09.03.01 "Информатика и вычислительная техника"

профиль "Программное обеспечение средств вычислительной техники и автоматизированных систем"

Кафедра прикладной математики и кибернетики

**Расчетно-графическая работа по дисциплине  
 Сетевые базы данных**

Вариант 4

Выполнил:

Cтудент гр. ИП-813 \_\_\_\_\_\_\_\_\_\_\_\_\_/Бурдуковский И.А./

ФИО студента

«\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Проверил:

Ассистент кафедры ПМиК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Грязнов Н.Г./

ФИО преподавателя

«\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г. Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Новосибирск 2022 г.

Задание

Создать две таблицы, каждая из которых должна иметь первичный ключ и, по крайней мере, один столбец с ограничением NOT NULL. Таблицы должны быть связаны внешним ключом; тип связи - "один-ко-многим". Создать пакет, содержащий процедуру начального заполнения таблиц данными (по 7-10 записей в таблице) и процедуру очистки таблиц (удаления записей).

Для одной из таблиц разработать триггер для обеспечения дополнительных ограничений на изменение данных таблицы (см. свой вариант задания).

Создать представление, которое позволяет запрашивать данные из обеих (связанных) таблиц. Представление должно ограничивать доступ к данным по столбцам и строкам.

Написать второй пакет, в состав которого включить вызовы процедур из первого пакета. В пакет также поместить процедуру изменения данных в таблицах (см. свой вариант задания). Значения изменяемых данных должны передаваться в процедуру как параметры. В процедурах предусмотреть обработку исключений. Обеспечить подтверждение транзакций при их успешном выполнении и откат - в случае возникновения исключительной ситуации.

Предоставить привилегии всем пользователям базы данных Oracle на использование представления для просмотра данных и привилегию на выполнение процедуры изменения данных.

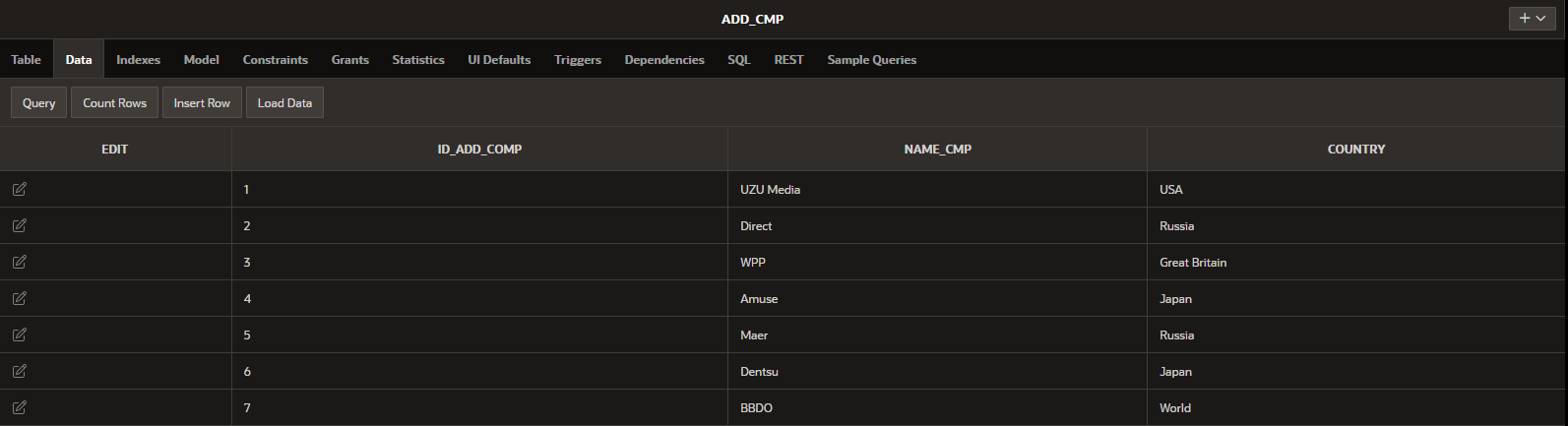
Поместить в таблицы информацию о Рекламных агентствах и Видах услуг. Каждое агентство оказывает услуги нескольких видов. Процедура должна удалять вид услуг из таблицы. Триггер должен регистрировать удаление, сохраняя удаленную запись и добавляя пользователя, выполнившего удаление. Включить в пакет еще одну процедуру, которая динамически создает ваш триггер.

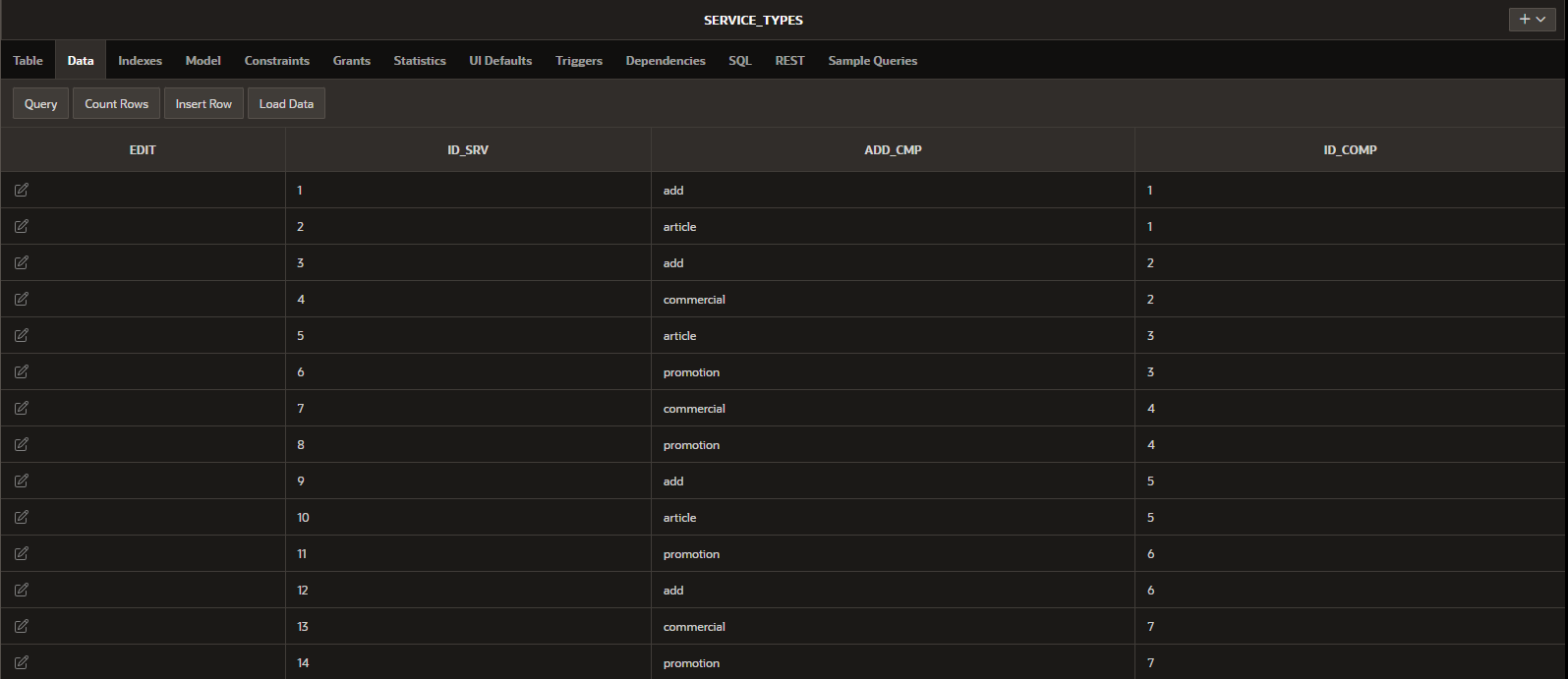
Описание работы

В результате работы было создано 2 таблицы service\_types и add\_cmp. Первая таблица содержит информацию о видах услуг, выполняющихся агенствами, где определены следующие поля: id\_srv – первичный ключ, add\_cmp, id\_comp. Поле id\_comp – это внешний ключ, который ссылается на id агенства из второй таблицы. Вторая содержит информацию о агенствах, поля таблицы: id\_add\_comp – первичный ключ, name\_cmp, country. Для корректного распределения идентификаторов было создано 2 последовательности: seq\_service – для таблицы услуг, seq\_addcmp – для таблицы агенств.

Первый пакет WorkTablePack содержит процедуры для заполнения и очистки таблиц.

Заполнение таблиц:

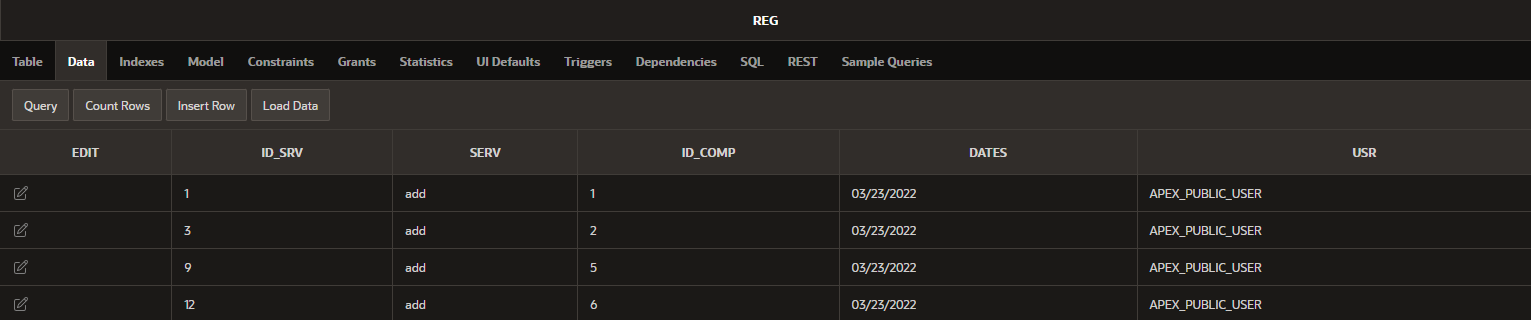




Второй пакет WorkPack содержит:

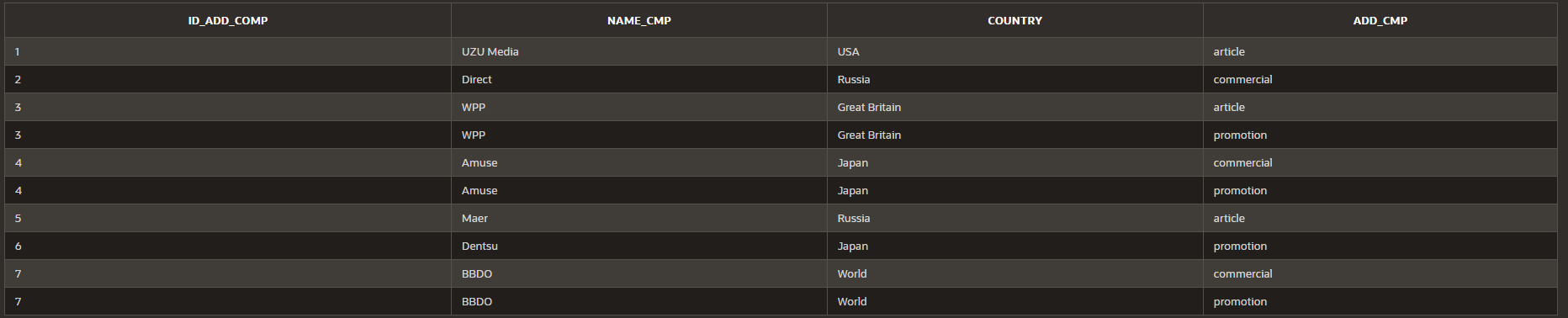
1. Процедуру work\_data, которая включает в себя вызов процедур и первого пакета;
2. Процедуру delete\_building с параметром – название услуги. Если произойдет нештатная ситуация – выдаётся исключение.
3. Процедуру dyna, которая динамически создаёт мой триггер, с параметром имени пользователя

Удаление из таблицы при помощи процедуры из задания:



Для сохранения удалённых записей и информации о пользователе был создан триггер, который сохраняет запись и регистрирует её в специальной таблице reg с полями id\_srv, serv, id\_comp, dates, usr:

Представление view\_table отображает названия компаний и связанных с ними сооружений:



Листинг

DROP SEQUENCE seq\_service;

DROP SEQUENCE seq\_addcmp;

CREATE SEQUENCE seq\_service;

CREATE SEQUENCE seq\_addcmp;

DROP TABLE service\_types;

DROP TABLE add\_cmp;

DROP TABLE reg;

CREATE TABLE add\_cmp (

id\_add\_comp NUMBER(4, 0) NOT NULL,

name\_cmp VARCHAR2(20) NOT NULL,

country VARCHAR2(20) NOT NULL,

CONSTRAINT id\_add\_comp PRIMARY KEY(id\_add\_comp)

);

CREATE TABLE service\_types (

id\_srv NUMBER(4, 0) NOT NULL,

add\_cmp VARCHAR2(20) NOT NULL,

id\_comp NUMBER(4, 0) NOT NULL,

CONSTRAINT id\_srv PRIMARY KEY(id\_srv),

CONSTRAINT id\_comp FOREIGN KEY(id\_comp) REFERENCES add\_cmp(id\_add\_comp)

);

CREATE TABLE reg (

id\_srv NUMBER(4, 0) NOT NULL,

serv VARCHAR2(20) NOT NULL,

id\_comp NUMBER(4, 0) NOT NULL,

dates DATE,

usr VARCHAR2(20) NOT NULL

);

COMMIT;

CREATE OR REPLACE PACKAGE WorkTablePack IS

PROCEDURE add\_data;

PROCEDURE delete\_data;

END;

/

CREATE OR REPLACE PACKAGE BODY WorkTablePack IS

PROCEDURE add\_data IS

BEGIN

INSERT INTO add\_cmp VALUES (seq\_service.NEXTVAL, 'TACC', 'Russia');

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'add', seq\_service.CURRVAL);

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'article', seq\_service.CURRVAL);

INSERT INTO add\_cmp VALUES (seq\_service.NEXTVAL, 'RT', 'Russia');

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'add', seq\_service.CURRVAL);

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'commercial', seq\_service.CURRVAL);

INSERT INTO add\_cmp VALUES (seq\_service.NEXTVAL, 'WPP', 'Great Britain');

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'article', seq\_service.CURRVAL);

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'promotion', seq\_service.CURRVAL);

INSERT INTO add\_cmp VALUES (seq\_service.NEXTVAL, 'Amuse', 'Japap');

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'commercial', seq\_service.CURRVAL);

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'promotion', seq\_service.CURRVAL);

INSERT INTO add\_cmp VALUES (seq\_service.NEXTVAL, 'Maer', 'Russia');

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'add', seq\_service.CURRVAL);

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'article', seq\_service.CURRVAL);

INSERT INTO add\_cmp VALUES (seq\_service.NEXTVAL, 'Dentsu', 'Japan');

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'promotion', seq\_service.CURRVAL);

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'add', seq\_service.CURRVAL);

INSERT INTO add\_cmp VALUES (seq\_service.NEXTVAL, 'BBDO', 'World');

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'commercial', seq\_service.CURRVAL);

INSERT INTO service\_types VALUES (seq\_addcmp.NEXTVAL, 'promotion', seq\_service.CURRVAL);

END add\_data;

PROCEDURE delete\_data IS

BEGIN

DELETE FROM service\_types;

DELETE FROM add\_cmp;

EXECUTE IMMEDIATE 'DROP SEQUENCE seq\_service';

EXECUTE IMMEDIATE 'DROP SEQUENCE seq\_addcmp';

EXECUTE IMMEDIATE 'CREATE SEQUENCE seq\_service';

EXECUTE IMMEDIATE 'CREATE SEQUENCE seq\_addcmp';

END delete\_data;

END;

/

CREATE OR REPLACE TRIGGER del

BEFORE DELETE ON service\_types

FOR EACH ROW

DECLARE

usr VARCHAR2(30);

BEGIN

usr := USER;

INSERT INTO reg (id\_srv, serv, id\_comp, dates, usr)

VALUES (:old.id\_srv, :old.add\_cmp, :old.id\_comp, sysdate, usr);

END;

/

CREATE OR REPLACE PACKAGE WorkPack IS

PROCEDURE work\_data;

PROCEDURE delete\_service (servs IN VARCHAR2);

PROCEDURE dyna (usr IN VARCHAR2);

END;

/

CREATE OR REPLACE PACKAGE BODY WorkPack IS

PROCEDURE work\_data IS

BEGIN

WorkTablePack.delete\_data;

WorkTablePack.add\_data;

END work\_data;

PROCEDURE delete\_service (servs IN VARCHAR2) AS

BEGIN

DELETE FROM service\_types WHERE add\_cmp = servs;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

DBMS\_OUTPUT.PUT\_LINE('Данные о ' || servs || ' не найдены в таблице "add\_cmp"');

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('Ошибка при удалении');

END delete\_service;

PROCEDURE dyna (usr in VARCHAR2) AS

BEGIN

EXECUTE IMMEDIATE 'CREATE OR REPLACE TRIGGER dele BEFORE DELETE ON service\_types FOR EACH ROW BEGIN INSERT INTO reg (id\_srv, serv, id\_comp, dates, usr) VALUES (:old.id\_srv, :old.add\_cmp, :old.id\_comp, sysdate, '|| usr ||'); END;';

END dyna;

END;

/

BEGIN

WorkTablePack.add\_data;

WorkPack.dyna('USER');

END;

/

SELECT \* FROM add\_cmp;

SELECT \* FROM service\_types;

BEGIN

WorkPack.delete\_service('add');

END;

/

SELECT \* FROM reg;

CREATE OR REPLACE VIEW view\_table AS

SELECT add\_cmp.id\_add\_comp, add\_cmp.name\_cmp, add\_cmp.country, service\_types.add\_cmp

FROM add\_cmp, service\_types

WHERE add\_cmp.id\_add\_comp = service\_types.id\_comp;

SELECT \* FROM view\_table;

GRANT SELECT ON view\_table TO public;

GRANT EXECUTE ON WorkPack TO public;