**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

###### ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

###### КЕМЕРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

**ИНСТИТУТ ЦИФРЫ**

**ОТЧЁТ**

**О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

«Словарь базы данных. Создание таблиц. Изменение таблиц и ограничений. Управление транзакциями»

Студентки 2 курса, ФИТ-211 группы

**Колесник Полины Олеговны**

Направление 02.03.02 – «Фундаментальная информатика и информационные технологии»

Руководитель:

Кандидат технических наук,

доцент Завозкин С. Ю.

Работа защищена

« »

“ ” 2023 г.

Кемерово 2023 г.

**ОТЧЁТ О ПРОДЕЛАННОЙ РАБОТЕ**

1. Создайте две таблицы на основе следующих бланков экземпляра таблицы. Убедитесь в том, что таблицы созданы.

CREATE TABLE DEPARTMENT\_2

( ID NUMBER(7,0) NOT NULL PRIMARY KEY,

NAME VARCHAR(25)

);

CREATE TABLE EMPLOYEE\_2

( ID NUMBER(7,0) NOT NULL PRIMARY KEY,

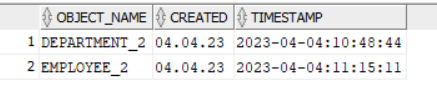
LAST\_NAME VARCHAR(25),

FIRST\_NAME VARCHAR(25),

DEPT\_ID NUMBER(7,0) NOT NULL,

FOREIGN KEY(DEPT\_ID) REFERENCES DEPARTMENT\_2 (ID)

);

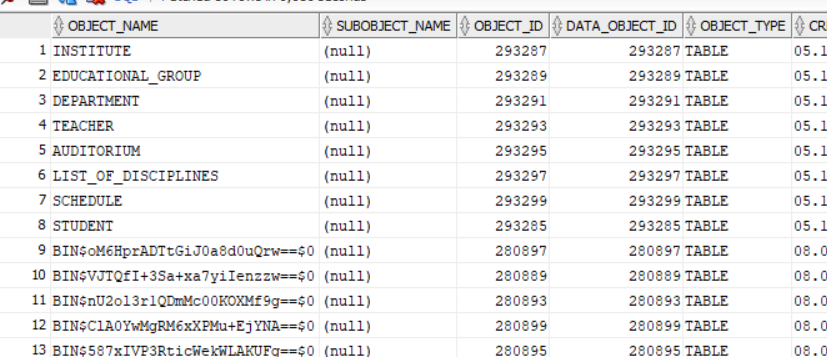


1. Из словаря данных СУБД выберите все представления, имеющие отношение к таблице.

select \*

from user\_objects

where upper(object\_type) like 'TABLE';



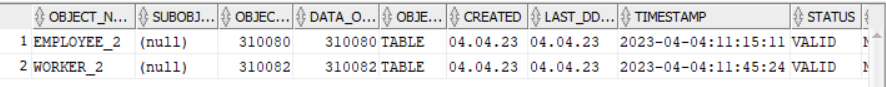
1. Создайте запрос к таблице USER\_OBJECTS словаря данных для просмотра информации о таблицах, созданных вами ранее (DEPARTMENT и EMPLOYEE).

select \*

from user\_objects

where lower(object\_name) = 'department\_2'

or lower(object\_name) = 'employee\_2';



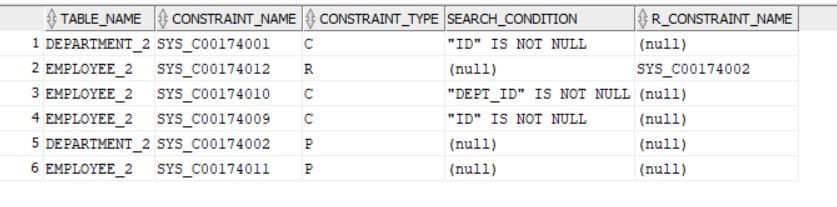
1. Создайте командный файл с универсальным запросом для проверки ограничений на созданные вами таблицы. Сохраните запрос в командном файле *\*.sql.* Выполните его для проверки ограничений на таблицы, созданные ранее Вами ранее (DEPARTMENT и EMPLOYEE).

select table\_name, constraint\_name, constraint\_type, search\_condition, r\_constraint\_name

from user\_constraints

where lower(table\_name) = 'department\_2'

or lower(table\_name) = 'employee\_2';



1. Создайте таблицу WORKER как копию таблицы EMPLOYEE. Просмотрите описание таблицы для проверки ее структуры. Просмотрите ограничения для этой таблицы. Сравните эти ограничения с ограничениями для таблицы EMPLOYEE. Запишите типы и имена ограничений.

Create table WORKER\_2 --создаётся копия таблицы EMPLOYEE

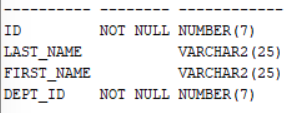
AS

select \*

from employee\_2;

-------------------------------------

describe worker\_2; --описание таблицы

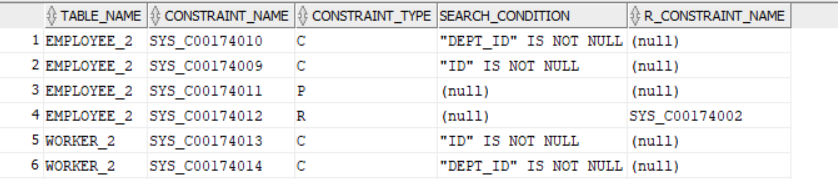


select table\_name, constraint\_name, constraint\_type, search\_condition, r\_constraint\_name --для сравнения

from user\_constraints

where lower(table\_name) = 'worker\_2'

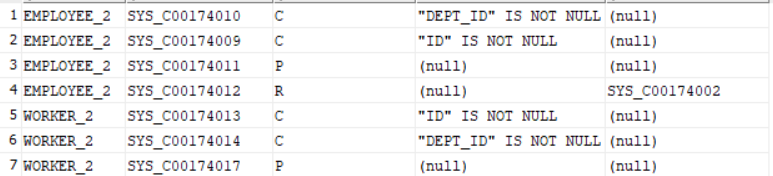
or lower(table\_name) = 'employee\_2';



1. Добавьте ограничение PRIMARY KEY из таблицы WORKER, используя столбец ID. Ограничение должно вступить в силу немедленно

alter table worker\_2

add primary key (id)

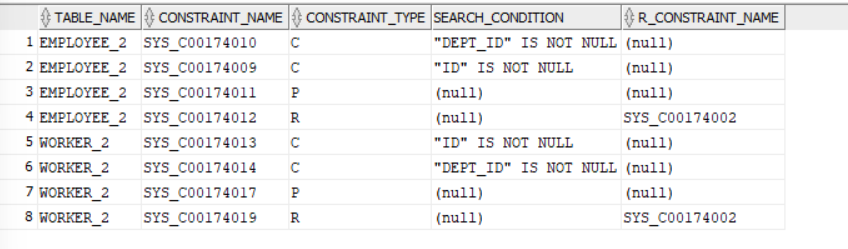


1. Добавьте ограничение типа FOREIGN KEY из таблицы DEPARTMENT на столбец DEPT\_ID таблицы WORKER. Убедитесь в том, что ограничения добавлены.

alter table worker\_2

add foreign key (dept\_id)

references DEPARTMENT\_2 (id);



1. Просмотрите имена и типы объектов с помощью представления USER\_OBJECTS словаря данных. Для удобства просмотра данных может потребоваться изменение формата столбцов. Обратите внимание на то, что создана новая таблица и новые индексы.

select uo.object\_name, uo.object\_type

from user\_objects uo

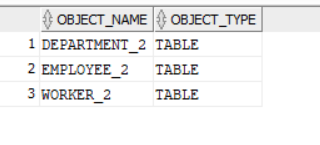
where lower(uo.object\_type) LIKE 'table'

and (lower(uo.object\_name) = 'employee\_2'

or lower(uo.object\_name) = 'worker\_2'

or lower(uo.object\_name) = 'department\_2'

);



1. **(2)** Выполните задания по управлению транзакциями с данными на примере таблиц DEPARTMENT и EMPLOYEE.
2. Создайте командный файл “test103.sql”, добавляющий в таблицу DEPARTMENT следующие строки: отдел “Marketing” (номер 37), отдел “Sales” (номер 54) и отдел “Personnel” (номер 75)). Выполните файл “test103.sql”. Проверьте правильность внесенных изменений.

insert into department\_2

values (37, 'Marketing');

insert into department\_2

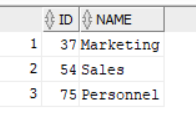
values (54, 'Sales');

insert into department\_2

values (75, 'Personnel');

select \*

from department\_2;



1. Создайте точку сохранения в транзакции.

savepoint update\_done;

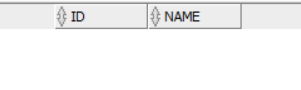


1. Удалите все данные из таблицы DEPARTMENT. Убедитесь в том, что таблица DEPARTMENT действительно пуста.

delete department\_2;

select \*

from department\_2;





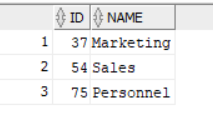
1. Отмените результаты последней операции DELETE, сохранив результаты последней операции INSERT. Убедитесь в том, что новая строка в таблице DEPARTMENT цела. Проверьте, содержит ли таблица DEPARTMENT все три строки.

rollback to update\_done;

select \*

from department\_2;





1. Сделайте эти добавления данных постоянными.

commit;

