

Евстафеева Алена Олеговна 3530902 70201

Не поняла, как ставить комментарии в код, поэтому просто делаю скрины с моим логином
RU_A404_SQL_S04

Проект 6 4 1

Часть 1. Выполнение сценария для заполнения таблиц.

Необходимо изучить порядок заполнения таблиц. Таблицу, имеющую поле внешнего ключа, нельзя заполнить раньше, чем соответствующую таблицу с первичным ключом.

- Используя документ сопоставления таблиц, составьте список, чтобы указать порядок заполнения таблиц.
- Откройте файл «sports data.sql» и изучите порядок добавления данных. Совпадает ли он с вашим списком? Чтобы получить этот файл, найдите в Разделе 6, урок 4 файл «sports data.zip» и распакуйте его.
- Выполните сценарий «sports data.sql» в APEX, чтобы заполнить таблицы.
- Проверьте, что при выполнении этого сценария не возникли ошибки.

Number	Elapsed	Statement	Feedback	Rows
1	0.04	INSERT INTO inventory_list (id, cost, units) VALUES('i01023')	1 row(s) inserted.	1
2	0.00	INSERT INTO inventory_list (id, cost, units) VALUES('i01023')	1 row(s) inserted.	1
3	0.00	INSERT INTO inventory_list (id, cost, units) VALUES('i01023')	1 row(s) inserted.	1
4	0.00	INSERT INTO inventory_list (id, cost, units) VALUES('i01023')	1 row(s) inserted.	1
5	0.00	INSERT INTO inventory_list (id, cost, units) VALUES('i01023')	1 row(s) inserted.	1
6	0.02	INSERT INTO items (itm_number, name, description, category,	1 row(s) inserted.	1
7	0.00	INSERT INTO items (itm_number, name, description, category,	1 row(s) inserted.	1
8	0.00	INSERT INTO items (itm_number, name, description, category,	1 row(s) inserted.	1
9	0.00	INSERT INTO items (itm_number, name, description, category,	1 row(s) inserted.	1
10	0.00	INSERT INTO items (itm_number, name, description, category,	1 row(s) inserted.	1
11	0.01	INSERT INTO price_history (start_date, start_time, price, it	1 row(s) inserted.	1
12	0.00	INSERT INTO price_history (start_date, start_time, price, en	1 row(s) inserted.	1
13	0.00	INSERT INTO price_history (start_date, start_time, price, en	1 row(s) inserted.	1
14	0.00	INSERT INTO price_history (start_date, start_time, price, it	1 row(s) inserted.	1

Вставка была выполнена успешно, но так как в прошлом задании необходимо было сделать ограничение, на только положительный баланс, одна вставка не удалась.

28 0.01 INSERT INTO customers (ctr_number, email, first_name, last_n ORA-02290: check constraint (RU_A404_SQL_S04.CSTM_CBL_MIN) violated

Пришлось удалять одно ограничение. И снова его добавлять скриптом:

ALTER TABLE customers ADD CONSTRAINT cstm_cbl_min CHECK (current_balance >= 0);

Потом повторить вставку. Все прошло успешно.

Часть 2. Вставка строк в системе

- Добавьте новую команду в систему.

id	name	Number_of_players	discount
t004	Jets	10	5

Проводим вставку команды:

The screenshot shows the Oracle SQL Workshop interface. In the top navigation bar, 'SQL Workshop' is selected. On the right, a user profile is shown with the workspace 'RU_A404_SQL_S04' and role 'Developer'. The main area displays the following SQL command:

```
INSERT INTO teams (id, name, number_of_players, discount)
VALUES('t004', 'Jets', 10, 5);
```

Below the command, the results show: '1 row(s) inserted.' and '0.02 seconds'. At the bottom, there are tabs for 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'.

2. Добавьте нового клиента с соответствующими сведениями в систему.

ctr number	email	First name	Last name	Phone number	Current balance	Loyalty card number	tem id	sre id
c02001	brianrog@hootech.com	Brian	Rogers	01654564898	-5	lc4587		

The screenshot shows the Oracle SQL Workshop interface. In the top navigation bar, 'SQL Workshop' is selected. On the right, a user profile is shown with the workspace 'RU_A404_SQL_S04' and role 'Developer'. The main area displays the following SQL command:

```
INSERT INTO customers (ctr_number, email, first_name, l
sre_id, tem_id, loyalty_card_number)
VALUES('c02001', 'brianrog@hootech.com', 'Brain', 'Rogers', '01654564898', -5, null, null, 'lc4587');
```

At the bottom, the results tab is selected. A yellow box highlights the error message: 'ORA-02290: check constraint (RU_A404_SQL_S04.CSTM_CBL_MIN) violated'.

3. Эта информация нарушает условие ограничения проверки значения, указывающего, что текущий баланс не может быть меньше нуля. Измените текущий баланс на 50 и выполните запрос еще раз.

SQL Commands Schema RU_A404_SQL_S04

Rows 10 Save Run

```
INSERT INTO customers (ctr_number, email, first_name, last_name, phone_number, current_balance, sre_id, tem_id, loyalty_card_number)
VALUES('c02001', 'brianrog@hootech.com', 'Brain', 'Rogers', '01654564898', 50, null, null, 'lc4587');
```

Results Explain Describe Saved SQL History

1 row(s) inserted.

Проект 6 4 2

Часть 1. Обновление строк в системе

1. Выполните следующий запрос, чтобы просмотреть содержимое таблицы price_history:

```
SELECT start_date, TO_CHAR (start_time, 'HH24:MI:SS'), price,
end_date, TO_CHAR (end_time, 'HH24:MI')
FROM price_history;
```

SQL Commands Schema RU_A404_SQL_S04

Rows 10 Save Run

```
SELECT start_date, TO_CHAR (start_time, 'HH24:MI:SS'), price,
end_date, TO_CHAR (end_time, 'HH24:MI')
FROM price_history;
```

Results Explain Describe Saved SQL History

START_DATE	TO_CHAR(START_TIME,'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME,'HH24:MI')
17-Jun-2017	09:00:00	4.99	-	-
25-Nov-2016	09:00:00	14.99	25-Jan-2017	17:00
25-Jan-2017	17:01:00	8.99	25-Jan-2017	19:00
26-Jan-2017	09:00:00	15.99	-	-
12-Feb-2017	12:30:00	7.99	-	-
25-Apr-2017	10:10:10	24.99	-	-
31-May-2017	16:35:30	149	-	-

7 rows returned in 0.01 seconds Download

2. OBL планирует обновить цену на товар, поэтому необходимо написать запрос, который заблокирует текущую цену, добавив значения системной даты в поля end_date и end_time. Для выполнения запроса необходимо выполнить два условия: совпадение номера товара и значение end_date, равное null. Так обеспечивается обновление последней цены.

SQL Commands Schema RU_A404_SQL_S04

Rows 10 Clear Command Find Tables Save

```
UPDATE price_history
SET end_date = SYSDATE
WHERE end_date is null;
```

Results Explain Describe Saved SQL History

5 row(s) updated.

SQL Commands Schema RU_A404_SQL_S04

Rows 10 Clear Command Find Tables Save Run

```
UPDATE price_history
SET end_time = SYSDATE
WHERE end_time is NULL;
```

Results Explain Describe Saved SQL History

5 row(s) updated.

3. Выполните еще раз инструкцию select для таблицы price_history, чтобы убедиться, что инструкция выполнена.

SQL Commands Schema RU_A404_SQL_S04

Rows 10 Clear Command Find Tables Save Run

```
SELECT start_date, TO_CHAR (start_time, 'HH24:MI:SS'), price,
end_date, TO_CHAR (end_time, 'HH24:MI')
FROM price_history;
```

Results Explain Describe Saved SQL History

START_DATE	TO_CHAR(START_TIME,'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME,'HH24:MI')
17-Jun-2017	09:00:00	4.99	15-Apr-2020	02:47
25-Nov-2016	09:00:00	14.99	25-Jan-2017	17:00
25-Jan-2017	17:01:00	8.99	25-Jan-2017	19:00
26-Jan-2017	09:00:00	15.99	15-Apr-2020	02:47
12-Feb-2017	12:30:00	7.99	15-Apr-2020	02:47
25-Apr-2017	10:10:10	24.99	15-Apr-2020	02:47
31-May-2017	16:35:30	149	15-Apr-2020	02:47

4. Вставьте новую строку, которая будет использовать текущую дату и время для установки новой цены на товар, равной 99,99.

SQL Commands

Schema RU_A404_SQL_S04

Rows 10

Clear Command Find Tables Save Run

```
INSERT INTO price_history (start_date, start_time, price, item_number)
VALUES(SYSDATE, SYSDATE, 99.99, 'im01101048');
```

START_DATE	TO_CHAR(START_TIME,'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME,'HH24:MI')
17-Jun-2017	09:00:00	4.99	15-Apr-2020	02:47
25-Nov-2016	09:00:00	14.99	25-Jan-2017	17:00
25-Jan-2017	17:01:00	8.99	25-Jan-2017	19:00
26-Jan-2017	09:00:00	15.99	15-Apr-2020	02:47
12-Feb-2017	12:30:00	7.99	15-Apr-2020	02:47
25-Apr-2017	10:10:10	24.99	15-Apr-2020	02:47
31-May-2017	16:35:30	149	15-Apr-2020	02:47
15-Apr-2020	04:51:27	99.99	-	-

Часть 2. Удаление строк из системы

- Боб Торнберри (Bob Thornberry) обратился в OBL и попросил удалить адрес 83 Barrhill Drive из системы, так как он больше не может получать посылки по этому адресу. Напишите инструкцию SQL для удаления этого адреса из системы.

SQL Commands

Schema RU_A404_SQL_S04

Rows 10

Clear Command Find Tables Save Run

```
DELETE FROM customers_addresses
WHERE address_line_1 = '83 Barrhill Drive';
```

Results Explain Describe Saved SQL History

1 row(s) deleted.

- Выполните инструкцию select для таблицы customers_addresses, чтобы убедиться, что инструкция выполнена.

The screenshot shows a SQL command window with the following details:

- SQL Commands** tab is selected.
- Schema: RU_A404_SQL_S04
- Rows: 10
- Buttons: Clear Command, Find Tables, Save, Run
- Query: `SELECT * FROM customers_addresses;`
- Results pane:

ID	ADDRESS_LINE_1	ADDRESS_LINE_2	CITY	ZIP_CODE	CTR_NUMBER
ca0102	17 Gartsquare Road	Starford	Liverpool	LP89JHK	c00001
ca0103	54 Ropehill Crescent	Georgetown	Star	ST45AGV	c00101
ca0104	36 Watercress Lane	-	Jump	JP23YTH	c01986
ca0105	63 Acacia Drive	Skins	Liverpool	LP83JHR	c00001
- Message: 4 rows returned in 0.00 seconds
- Link: Download

Упражнения 6.4

Задачи

- Вставьте строки в таблицы, созданные для академической базы данных, используя следующие таблицы:

Запросы(тк же продублированы в файле ЗАПРОСЫ_INSERT):

```

insert into ACADEMIC_SESSIONS (ID, NAME) values(100, 'SPRING SESSION');
insert into ACADEMIC_SESSIONS (ID, NAME) values(200, 'FALL SESSION');
insert into ACADEMIC_SESSIONS (ID, NAME) values(300, 'SUMMER SESSION');

insert into DEPARTMENTS_FOR_AB values(10, 'ACCOUNTING', 'MARK SMITH');
insert into DEPARTMENTS_FOR_AB values(20, 'BIOLOGY', 'DAVE GOLD');
insert into DEPARTMENTS_FOR_AB values(30, 'COMPUTER SCIENCE', 'LINDA BROWN');
insert into DEPARTMENTS_FOR_AB values(40, 'LITERATURE', 'ANITA TAYLOR');

alter table PNT_INFORMATIONS READ WRITE;
insert into PNT_INFORMATIONS values(600, 'NEIL', 'SMITH', 'DORIS', 'SMITH');
insert into PNT_INFORMATIONS values(610, 'WILLIAM', 'BEN', 'NITA', 'BEN');
insert into PNT_INFORMATIONS values(620, 'SEAN', 'TAYLOR', 'RHEA', 'TAYLOR');
insert into PNT_INFORMATIONS values(630, 'DAVE', 'CARMEN', 'CATHERINE', 'CARMEN');
insert into PNT_INFORMATIONS values(640, 'JOHN', 'AUDRY', 'JANE', 'AUDRY');

insert into STUDENTS values(720, 'JACK', 'SMITH', TO_DATE('01-Jan-2012', 'DD-MON-YYYY'),
'JSMITH@SCHOOL.EDU', 600);
insert into STUDENTS values(730, 'NOAH', 'AUDRY', TO_DATE('01-Jan-2012', 'DD-MON-YYYY'),
'NAUDRY@SCHOOL.EDU', 640);
insert into STUDENTS values(740, 'RHONDA', 'TAYLOR', TO_DATE('01-Sep-2012', 'DD-MON-YYYY'),
'RTAYLOR@SCHOOL.EDU', 620);
insert into STUDENTS values(750, 'ROBERT', 'BEN', TO_DATE('01-Mar-2012', 'DD-MON-YYYY'),
'RBEN@SCHOOL.EDU', 610);

```

```

insert into STUDENTS values(760, 'JEANNE', 'BEN', TO_DATE('01-Mar-2012', 'DD-MON-YYYY'), 'JBEN@SCHOOL.EDU', 610);
insert into STUDENTS values(770, 'MILLS', 'CARMEN', TO_DATE('01-Apr-2013', 'DD-MON-YYYY'), 'MCARMEN@SCHOOL.EDU', 630);

INSERT INTO SEATEDS VALUES('BUILDING D', 401, 'MWF 9-10');
INSERT INTO SEATEDS VALUES('BUILDING A', 101, 'MWF 12-1');
INSERT INTO SEATEDS VALUES('BUILDING B', 201, 'THUR 2-4');
INSERT INTO SEATEDS VALUES('BUILDING C', 301, 'TUES 5-7');

INSERT INTO ONLINES VALUES('TAX123', 'PASSWORD');
INSERT INTO ONLINES VALUES('BIO123', 'PASSWORD');

insert into COURSES (ID, NAME, ACADEMIC_SESSIONS_ID, DPT_ID, ONLINES_LGN_ID, STD_BLD, STD_ROOM, DATE_TIME)
values (195, 'CELL BIOLOGY', 200, 20, null, 'BUILDING D', 401, 'MWF 9-10');
insert into COURSES (ID, NAME, ACADEMIC_SESSIONS_ID, DPT_ID, ONLINES_LGN_ID, STD_BLD, STD_ROOM, DATE_TIME)
values (190, 'PRINCIPLES OF ACCOUNTING', 100, 10, null, 'BUILDING A', 101, 'MWF 12-1');
insert into COURSES (ID, NAME, ACADEMIC_SESSIONS_ID, DPT_ID, ONLINES_LGN_ID, STD_BLD, STD_ROOM, DATE_TIME)
values (191, 'INTRODUCTION TO BUSINESS LAW', 100, 10, null, 'BUILDING B', 201, 'THUR 2-4');
insert into COURSES (ID, NAME, ACADEMIC_SESSIONS_ID, DPT_ID, ONLINES_LGN_ID, STD_BLD, STD_ROOM, DATE_TIME)
values (192, 'COST ACCOUNTING', 100, 10, null, 'BUILDING C', 301, 'TUES 5-7');
insert into COURSES (ID, NAME, ACADEMIC_SESSIONS_ID, DPT_ID, ONLINES_LGN_ID, STD_BLD, STD_ROOM, DATE_TIME)
values (193, 'STRATEGIC TAX PLANNING FOR BU', 100, 10, 'TAX123', null, null, null);
insert into COURSES (ID, NAME, ACADEMIC_SESSIONS_ID, DPT_ID, ONLINES_LGN_ID, STD_BLD, STD_ROOM, DATE_TIME)
values (194, 'GENERAL BIOLOGY', 200, 20, 'BIO123', null, null, null);

insert into FACULTIES values (800, 'JILL', 'MILLER', 'JMILL@SCHOOL.EDU', 10000, 'HEALTH', null, 20);
insert into FACULTIES values (810, 'JAMES', 'BORG', 'JBORG@SCHOOL.EDU', 30000, 'HEALTH,DENTAL', null, 10);
insert into FACULTIES values (820, 'LYNN', 'BROWN', 'LBROWN@SCHOOL.EDU', null, null, 50, 30);
insert into FACULTIES values (830, 'ARTHUR', 'SMITH', 'ASMITH@SCHOOL.EDU', null, null, 40, 10);
insert into FACULTIES values (840, 'SALLY', 'JONES', 'SJONES@SCHOOL.EDU', 50000, 'HEALTH,DENTAL,VISION', null, 40);

insert into XAM_TYPES values ('MCE', 'Multiple Choice Exams', 'CHOOSE MORE THAN ONE ANSWER');
insert into XAM_TYPES values ('TF', 'TRUE AND FALSE Exams', 'CHOOSE EITHER TRUE OR FALSE');
insert into XAM_TYPES values ('ESS', 'ESSAY Exams', 'WRITE PARAGRAPHS');
insert into XAM_TYPES values ('SA', 'SHORT ANSWER Exams', 'WRITE SHORT ANSWERS');
insert into XAM_TYPES values ('FIB', 'FILL IN THE BLANKS Exams', 'TYPE IN THE CORRECT ANSWER');

insert into EXAMS values (500, TO_DATE('12-Sep-2013', 'DD-MON-YYYY'), 'MCE', 190);
insert into EXAMS values (510, TO_DATE('15-Sep-2013', 'DD-MON-YYYY'), 'SA', 191);
insert into EXAMS values (520, TO_DATE('18-Sep-2013', 'DD-MON-YYYY'), 'FIB', 192);

```

```

insert into EXAMS values (530, TO_DATE('21-Mar-2014', 'DD-MON-YYYY'), 'ESS', 193);
insert into EXAMS values (540, TO_DATE('02-Apr-2014', 'DD-MON-YYYY'), 'TF', 194);

insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (190,720,500,91);
insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (195,730,540,87);
insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (194,730,530,85);
insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (195,750,510,97);
insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (191,750,520,78);
insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (192,760,510,70);
insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (193,720,520,97);
insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (192,750,500,60);
insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (192,760,540,65);
insert into XAM_RESULTS (CRS_ID, STU_ID, XAM_ID, GRADE) values (191,760,530,60);

insert into STUDENT_ATTENDANCES (STU_ID, NUMBER_OF_WRK_DAYS, NUMBER_OF_DAYS_OFF, ELIGIBILITY_FOR_XM, ACADEMIC_SESSIONS_ID) values (730,180,11,'Y', 200);
insert into STUDENT_ATTENDANCES (STU_ID, NUMBER_OF_WRK_DAYS, NUMBER_OF_DAYS_OFF, ELIGIBILITY_FOR_XM, ACADEMIC_SESSIONS_ID) values (740,180,12,'Y', 300);
insert into STUDENT_ATTENDANCES (STU_ID, NUMBER_OF_WRK_DAYS, NUMBER_OF_DAYS_OFF, ELIGIBILITY_FOR_XM, ACADEMIC_SESSIONS_ID) values (770,180,13,'Y', 300);
insert into STUDENT_ATTENDANCES (STU_ID, NUMBER_OF_WRK_DAYS, NUMBER_OF_DAYS_OFF, ELIGIBILITY_FOR_XM, ACADEMIC_SESSIONS_ID) values (720,180,21,'Y', 100);
insert into STUDENT_ATTENDANCES (STU_ID, NUMBER_OF_WRK_DAYS, NUMBER_OF_DAYS_OFF, ELIGIBILITY_FOR_XM, ACADEMIC_SESSIONS_ID) values (750,180,14,'Y', 200);
insert into STUDENT_ATTENDANCES (STU_ID, NUMBER_OF_WRK_DAYS, NUMBER_OF_DAYS_OFF, ELIGIBILITY_FOR_XM, ACADEMIC_SESSIONS_ID) values (760,180,15,'Y', 200);

insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (720,190,'A');
insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (750,192,'A');
insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (760,190,'B');
insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (770,194,'A');
insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (720,193,'B');
insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (730,191,'C');
insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (740,195,'F');
insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (760,192,'C');
insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (770,192,'D');
insert into STUDENT_COURSE_DETAILS (STU_ID, CRS_ID, GRADE) values (770,193,'F');

insert into FACULTY_COURSE_DETAILS (CRS_ID, FCLT_ID, CNT_HOURS) values (192,800,3);
insert into FACULTY_COURSE_DETAILS (CRS_ID, FCLT_ID, CNT_HOURS) values (193,800,4);
insert into FACULTY_COURSE_DETAILS (CRS_ID, FCLT_ID, CNT_HOURS) values (190,800,5);
insert into FACULTY_COURSE_DETAILS (CRS_ID, FCLT_ID, CNT_HOURS) values (191,800,3);
insert into FACULTY_COURSE_DETAILS (CRS_ID, FCLT_ID, CNT_HOURS) values (194,810,4);
insert into FACULTY_COURSE_DETAILS (CRS_ID, FCLT_ID, CNT_HOURS) values (195,810,5);

insert into FACULTY_LOGIN_DETAILS (FCLT_ID, LOGIN_DTE_TME) values (800,TO_TIMESTAMP('01-JUN-17 05.10.39.000000 PM', 'DD-MM-YYYY hh.mi.ss.ff6 pm'));
insert into FACULTY_LOGIN_DETAILS (FCLT_ID, LOGIN_DTE_TME) values (800,TO_TIMESTAMP('01-JUN-17 05.13.15.000000 PM', 'DD-MM-YYYY hh.mi.ss.ff6 pm'));

```

```

insert into FACULTY_LOGIN_DETAILS (FCLT_ID, LOGIN_DTE_TME) values (810,TO_TIMESTAMP('01-JUN-17 05.13.21.000000 PM', 'DD-MM-YYYY hh.mi.ss.ff6 pm'));
insert into FACULTY_LOGIN_DETAILS (FCLT_ID, LOGIN_DTE_TME) values (840,TO_TIMESTAMP('01-JUN-17 05.13.26.000000 PM', 'DD-MM-YYYY hh.mi.ss.ff6 pm'));
insert into FACULTY_LOGIN_DETAILS (FCLT_ID, LOGIN_DTE_TME) values (820,TO_TIMESTAMP('01-JUN-17 05.13.31.000000 PM', 'DD-MM-YYYY hh.mi.ss.ff6 pm'));
insert into FACULTY_LOGIN_DETAILS (FCLT_ID, LOGIN_DTE_TME) values (830,TO_TIMESTAMP('01-JUN-17 05.13.36.000000 PM', 'DD-MM-YYYY hh.mi.ss.ff6 pm'));

```

Вставка проходила с большим трудом, так как многие внешние ключи были с проблемами. Пришлось дорабатывать базу данных. Запросы отлаживала потихоньку, поэтому нету скрина, со всеми выполненными запросами за раз. Прикреплю только скрин (файл results_of_insert), где видно, что все запросы выполнены с ошибкой(данная запись уже добавлена)

Так же выполнение запросов можно проверить просто зайдя в каждую таблицу на вкладку data

The screenshot shows two database tables in Oracle SQL Developer:

- ACADEMIC_SESSIONS** table:

EDIT	ID	NAME
	100	SPRING SESSION
	300	SUMMER SESSION
	200	FALL SESSION
- DEPARTMENTS_FOR_AB** table:

EDIT	ID	NAME	HEAD
	10	ACCOUNTING	MARK SMITH
	20	BIOLOGY	DAVE GOLD
	30	COMPUTER SCIENCE	LINDA BROWN
	40	LITERATURE	ANITA TAYLOR

Schema RU_A404_SQL_S04

PNT_INFORMATION

Table Data Indexes Model Constraints Grants Statistics UI Defaults

Query Count Rows Insert Row

Data

EDIT	ID	PNT_1_FRST_NAME	PNT_1_LST_NAME	PNT_2_FRST_NAME	PNT_2_LST_NAME
	600	NEIL	SMITH	DORIS	SMITH
	610	WILLIAM	BEN	NITA	BEN
	620	SEAN	TAYLOR	RHEA	TAYLOF
	630	DAVE	CARMEN	CATHY	CARME
	640	JOHN	AUDRY	JANE	AUDRY

Object Browser

Schema RU_A404_SQL_S04

STUDENTS

Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies

Query Count Rows Insert Row

Data

EDIT	ID	FRST_NAME	LST_NAME	RGS_YEA	EMAIL	PNT_INF_ID
	720	JACK	SMITH	01-Jan-2012	JSMITH@SCHOOL.EDU	600
	730	NOAH	AUDRY	01-Jan-2012	NAUDRY@SCHOOL.EDU	640
	740	RHONDA	TAYLOR	01-Sep-2012	RTAYLOR@SCHOOL.EDU	620
	750	ROBERT	BEN	01-Mar-2012	RBEN@SCHOOL.EDU	610
	760	JEANNE	BEN	01-Mar-2012	JBEN@SCHOOL.EDU	610
	770	MILLS	CARMEN	01-Apr-2013	MCARMEN@SCHOOL.EDU	630

Object Browser

Schema RU_A404_SQL_S04

COURSES

Table Data Indexes Model Constraints Grants Statistics UI Defaults

EDIT ID NAME DPT_ID ONLINES_LGN_ID STD_BLD STD_ROOM DATE_TIME ACADEMIC

	194	GENERAL BIOLOGY	20	BIO123	-	-	-	200
	193	STRATEGIC TAX PLANNING FOR BU	10	TAX123	-	-	-	100
	195	CELL BIOLOGY	20	-	BUILDING D	401	MWF 9-10	200
	190	PRINCIPLES OF ACCOUNTING	10	-	BUILDING A	101	MWF 12-1	100
	191	INTRODUCTION TO BUSINESS LAW	10	-	BUILDING B	201	THUR 2-4	100
	192	COST ACCOUNTING	10	-	BUILDING C	301	TUES 5-7	100

Object Browser

Schema RU_A404_SQL_S04

FACULTIES

ID	FRST_NAME	LST_NAME	EMAIL	SLY	ISU	HOURLY_RAT	DPT_ID
800	JILL	MILLER	JMILL@SCHOOLEDU	10000	HEALTH	-	20
810	JAMES	BORG	JBORG@SCHOOLEDU	30000	HEALTH,DENTAL	-	10
820	LYNN	BROWN	LBROWN@SCHOOLEDU	-	-	50	30
830	ARTHUR	SMITH	ASMITH@SCHOOLEDU	-	-	40	10
840	SALLY	JONES	SJONES@SCHOOLEDU	50000	HEALTH,DENTAL,VISION	-	40

Schema RU_A404_SQL_S04

EXAMS

Table	Data	Indexes	Model	Constraints	Grants	Statistics	UI Default
Query	Count Rows	Insert Row					

Data

EDIT	ID	STR_DATE	XAM_TYPES_TYP	COURSE_ID
	540	02-Apr-2014	TF	194
	500	12-Sep-2013	MCE	190
	510	15-Sep-2013	SA	191
	520	18-Sep-2013	FIB	192
	530	21-Mar-2014	ESS	193

Object Browser

Schema RU_A404_SQL_S04

XAM_TYPES

Table	Data	Indexes	Model	Constraints	Grants	Statistics	UI Defaults	Triggers	Dependencies	Se
Query	Count Rows	Insert Row								

Data

EDIT	TYP	NAME	DSC
	MCE	Multiple Choice Exams	CHOOSE MORE THAN ONE ANSWER
	TF	TRUE AND FALSE Exams	CHOOSE EITHER TRUE OR FALSE
	ESS	ESSAY Exams	WRITE PARAGRAPHS
	SA	SHORT ANSWER Exams	WRITE SHORT ANSWERS
	FIB	FILL IN THE BLANKS Exams	TYPE IN THE CORRECT ANSWER

Schema RU_A404_SQL_S04

XAM_RESULTS

EDIT	GRADE	STU_ID	CRS_ID	XAM_ID
	91	720	190	500
	87	730	195	540
	85	730	194	530
	97	750	195	510
	78	750	191	520
	70	760	192	510
	97	720	193	520
	60	750	192	500
	65	760	192	540
	60	760	191	530

Object Browser

Schema RU_A404_SQL_S04

STUDENT_ATTENDANCES

Data

EDIT	NUMBER_OF_WRK_DAYS	NUMBER_OF_DAYS_OFF	ELIGIBILITY_FOR_XM	STU_ID	ACADEMIC_SESSIONS_ID
	180	11	Y	730	200
	180	12	Y	740	300
	180	13	Y	770	300
	180	21	Y	720	100
	180	14	Y	750	200
	180	15	Y	760	200

Download

Object Browser

Schema RU_A404_SQL_S04

STUDENT_COURSE_DETAILS

EDIT	GRADE	STU_ID	CRS_ID
	A	720	190
	A	750	192
	B	760	190
	A	770	194
	B	720	193
	C	730	191
	F	740	195
	C	760	192
	D	770	192
	F	770	193

Schema RU_A404_SQL_S04

FACULTY_COURSE_DETAILS

Table Data Indexes Model Constraints Grants Statistics UI Defaults

Query Count Rows Insert Row

Data

EDIT	CNT_HOURS	CRS_ID	FCLT_ID
	4	193	800
	3	192	800
	5	190	800
	3	191	800
	4	194	810
	5	195	810

Schema RU_A404_SQL_S04

FACULTY_LOGIN_DETAILS

Table Data Indexes Model Constraints Grants Statistics UI Defaults

Query Count Rows Insert Row

Data

EDIT	LOGIN_DTE_TME	FCLT_ID
	01-Jun-0017	800
	01-Jun-0017	800
	01-Jun-0017	810
	01-Jun-0017	820
	01-Jun-0017	830
	01-Jun-0017	840

Упражнение 2. Обновление строк в таблицах

Обзор

Вам необходимо обновить записи в таблице FACULTY_LOGIN_DETAILS и включить в эту таблицу поле DETAILS.

Задачи

- Измените таблицу AD_FACULTY_LOGIN_DETAILS, чтобы добавить поле DETAILS. Это поле должно поддерживать символы VARCHAR2(50) и может содержать значения null.
- Обновите хотя бы 2 записи в столбце DETAILS в таблице с данными о входах преподавателей.
**Примечание. Вам потребуется найти значения LOGIN_DATE_TIME для обновляемых записей, так как это часть первичного ключа.

Убедитесь, что столбец DETAILS обновлен и в нем появились значения.

SQL Commands Schema RU_A404_SQL_S04

Rows 10 Clear Command Find Tables Save Run

```
ALTER TABLE faculty_login_details ADD details VARCHAR(50) NULL;
```

Results Explain Describe Saved SQL History

Table altered.

FACULTY_LOGIN_DETAILS					
Table	Data	Indexes	Model	Constraints	Grants
	Add Column	Modify Column	Rename Column	Drop Column	Rename
	Copy	Drop	Truncate	Create Lookup Table	
Column Name	Data Type	Nullable	Default	Primary Key	
FCLT_ID	VARCHAR2(10)	No	-	1	
LOGIN_DTE_TME	DATE	No	-	2	
DETAILS	VARCHAR2(50)	Yes	-	-	

Задание 2

EDIT	LOGIN_DTE_TME	FCLT_ID	DETAILS
✎	01-Jun-0017	800	-
✎	01-Jun-0017	800	-
✎	01-Jun-0017	810	-
✎	01-Jun-0017	840	-
✎	01-Jun-0017	820	-
✎	01-Jun-0017	830	-

Первое обновление

SQL Commands Schema: RU_A404_SQL_S04 Rows: 10 Save Run

```
UPDATE faculty_login_details
SET details = 'details one'
WHERE (fcld_id = '800' AND login_dte_tme = TO_TIMESTAMP('01-JUN-17 05.10.39.000000 PM', 'DD-MM-YYYY hh.mi.ss.ff6 pm'));
```

Results Explain Describe Saved SQL History

1 row(s) updated.

Втрое обновление

SQL Commands Schema: RU_A404_SQL_S04 Rows: 10 Save Run

```
UPDATE faculty_login_details
SET details = 'details twd'
WHERE (fcld_id = '830' AND login_dte_tme = TO_TIMESTAMP('01-JUN-17 05.13.36.000000 PM', 'DD-MM-YYYY hh.mi.ss.ff6 pm'));
```

Results Explain Describe Saved SQL History

1 row(s) updated.

Проверим на наличие:

Schema: RU_A404_SQL_S04

FACULTY_LOGIN_DETAILS								
Table	Data	Indexes	Model	Constraints	Grants	Statistics	UI Defaults	Triggers
	Query	Count Rows	Insert Row					
Data								
EDIT	LOGIN_DTE_TME	FCLT_ID	DETAILS					
	01-Jun-0017	800	-					
	01-Jun-0017	800	details one					
	01-Jun-0017	810	-					
	01-Jun-0017	840	-					
	01-Jun-0017	820	-					
	01-Jun-0017	830	details two					

Упражнения 6-5

- Предположим, что создана таблица с такой структурой:

```
CREATE TABLE AD_STUDENT_TEST_DETAILS
(
    STUDENT_ID          NUMBER NOT NULL ,
    FIRST_NAME          VARCHAR2(50) ,
    STUDENT_REG_YEAR   DATE
);
```

Затем таблица была изменена, и в нее был добавлен столбец email_addr:

```
ALTER TABLE AD_STUDENT_TEST_DETAILS ADD ( EMAIL_ADDR VARCHAR2(100) UNIQUE );
```

После выполнения команды ALTER создается точка сохранения с именем ALTER_DONE.

После создания точки сохранения ALTER_DONE выполняется команда ROLLBACK. Останется ли новое поле электронной почты в таблице?

НЕВЕРНО

Нет, так как RollBack вернет данные к неявной точке сохранения в самом начале транзакции. То есть новое поле электронной почты в таблице создано не будет.

Для того, чтобы оно осталось созданным, необходимо сделать ROLLBACK TO ALTER_DONE.

Исправлено

Если внимательно перечитать лекцию, то можно понять, что

‣ Транзакция начинается при выполнении первой SQL-инструкции DML.

DML инструкции: «select» («выбрать»), «insert» («вставить»), «update» («обновить»), и «delete» («удалить»)

Так же можно заметить, что ALTER TABLE относится к DDL (Data Definition Language) командам.

Команды управления транзакциями TCL управляют изменениями в базе данных, которые осуществляются командами манипулирования данными DML.

ROLLBACK

Завершает текущую транзакцию, отменяя все отложенные изменения данных.

Это значит, что если сделать ROLLBACK, то команда отменена не будет, то есть новое поле электронной почты **останется** в таблице!

- Если в тестовую таблицу добавлены строки с помощью команды INSERT, создается точка сохранения с именем INSERT_DONE.

Затем выполняется команда UPDATE для строки в тестовой таблице, и создается точка сохранения с именем UPDATE_DONE.

После этого строка тестовой таблицы удаляется с помощью команды DELETE, и создается точка сохранения DELETE_DONE. Какие записи будут в таблице на этом этапе?

Затем выполняется команда ROLLBACK для точки сохранения UPDATE_DONE. Что изменится в транзакциях и остальных записях таблицы?

```
INSERT INTO AD_STUDENT_TEST_DETAILS VALUES(920, 'MAC', TO_DATE('01-JAN-2012','DD-MON-YYYY'),NULL);
INSERT INTO AD_STUDENT_TEST_DETAILS VALUES(940, 'RUTH', TO_DATE('01-SEP-2012','DD-MON-YYYY'),NULL);
INSERT INTO AD_STUDENT_TEST_DETAILS VALUES(950, 'ROBERT', TO_DATE('01-MAR-2012','DD-MON-YYYY'),NULL);
INSERT INTO AD_STUDENT_TEST_DETAILS VALUES(960, 'JEANNE', TO_DATE('01-MAR-2012','DD-MON-YYYY'),NULL);

SAVEPOINT CREATE_DONE;

UPDATE AD_STUDENT_TEST_DETAILS
SET EMAIL_ADDR = 'Mac@abc.com'
WHERE STUDENT_ID = 940;

SAVEPOINT UPDATE_DONE;

DELETE FROM AD_STUDENT_TEST_DETAILS WHERE STUDENT_ID = 950;

SAVEPOINT DELETE_DONE;

ROLLBACK TO UPDATE_DONE;
-
```

Если сделать ROLLBACK TO UPDATE_DONE, то мы откатимся к точке сохранения после update-а таблицы, то есть почта студента 950 изменится, но удаление студента с ID = 950 не произойдет.