Pa	ort 1: Running a script to populate the tables	
1. 0	inventory_list	
	items	
(3	price _history	
	sales_representatives	
6		
6		
0	customers	
8	customers_address	
@	orders	
	ordered_items (C	
2. Y	es, my list matches with the order the data being added in "sports data.sql".	

Number 1=	Elapsed	Statement	Feedback	Rows
析	0.05	INSERT INTO Inventory_list (tid, cost, units) VALUES(til01025	1 row(s) inserted.	
2	0.00	INSERT INTO Inventory_list (lid, cost, units) VALUES(101023	1 row(s) inserted.	- 1
5,	0,00	INSERT INTO inventory_list (id, cost, units) VALUES(1101023	1 row(s) inserted.	1
4.	0.01	INSERT INTO inventory_list (id, cost, units) VALUES(iI01023	1 row(s) inserted.	3
5)	0,00	INSERT INTO inventory_list (id, cost, units) VALUES(iI01023	1 row(s) inserted.	of .
6	0.02	INSERT INTO items (itm_number, name, description, category,	1 row(s) inserted.	51
2	0.00	INSERT INTO items (itm_number, name, description, category,	1 row(s) inserted.	
В	0.00	INSERT INTO Items (itm_number, name, description, category,	1 row(s) inserted.	
9	0.00	INSERT INTO Items (Itm_number, name, description, category,	1 row(s) inserted.	31
10	0,00	INSERT INTO Items (itm_number, name, description, category,	1 row(s) inserted.	3
ii	0.01	INSERT INTO price_history (start_date, start_time, price, it	1 row(s) inserted.	3
12	0.01	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.	51
14	0.00	INSERT INTO price_history (start_date, start_time, price, it	1 row(s) inserted.	
15	0.00	INSERT INTO price_history (start_date, start_time, price, it	1 row(s) inserted.	

Download

w(s) 1 - 15 of 47 Next ▶

		19491- BOLD NEAL P
47	47	0
Statements Processed	Successful	With Errors



4.	Check that no errors occurred when you ran the script.
	No errors occured when running the script "sports data sql".
	Part 2: Inserting rows to the system
1.	INSERT INTO teams (id, name, number of players, discount) VALUES : C't004', 'Jets', 10,51')



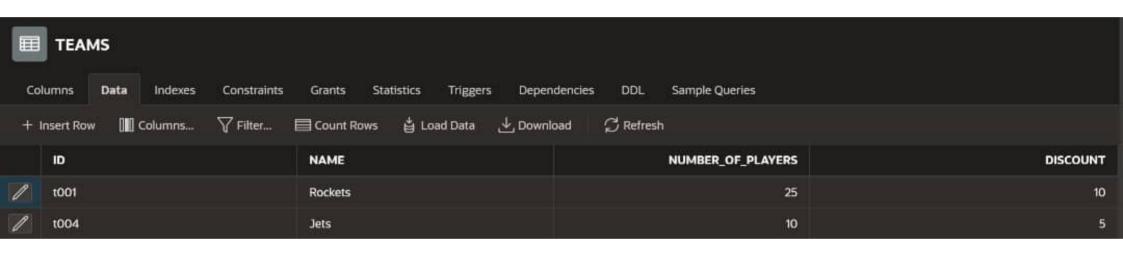
1 INSERT INTO teams (id, name, number_of_players, discount) VALUES ('t004', 'Jets', 10, 5);

Results Explain Describe Saved SQL History

1 row(s) inserted.

0.02 seconds







2. INSERT INTO customers (ctr_number, email, first_name, last_name, phone_number, current_balance, sme_id, tem_id, loyalty_cord_number) VALUES C'c 02001', 'brianrog@hovech.com', 'Brian', 'Rogers', '01654564898', -5, NULL, NULL, '1c4587');

```
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', 'Brian', 'Rogers', '01654564898', -5, NULL, NULL, '1c4587');

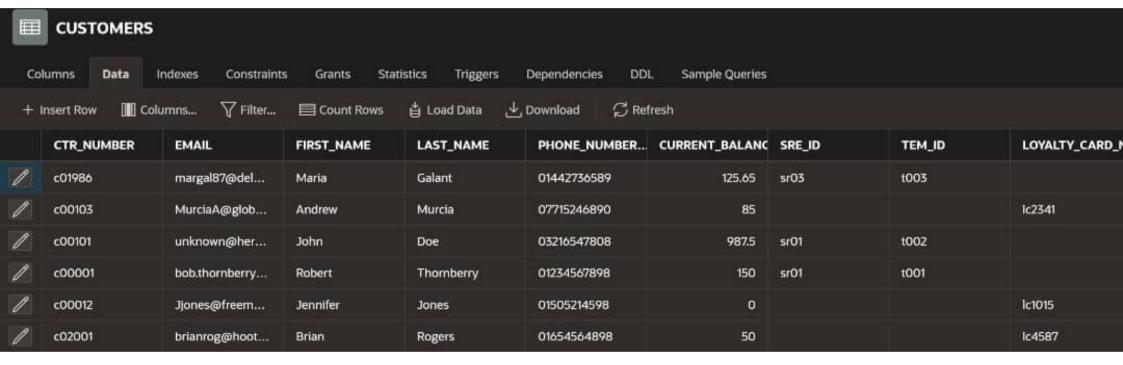
Results Explain Describe Saved SQL History

ORA-02290: check constraint (WKSP_DBLABPZY.CUSTOMER_BALANCE_CHK) violated
```



3. INSERT DUTO customers (ctr_number, email, first_name, last-name, phone-number, current_balance, sre_id, tem_id, loyalty_card_number) VALUES ('co2001', 'brianny@homech.com', 'Brian', 'Rogers', 'O1674564898, 50, NULL, NULL, '(c4587');





	No:
Database Lab 2 Part 2	Date:
Part 1: Updating rows to the system	
1. SELECT start-date, TO_CHAR & start-time, 4H124: MI'S	s'), price, end-date, To-CHAR (end-time, "HH24:MI")
From price-history;	

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

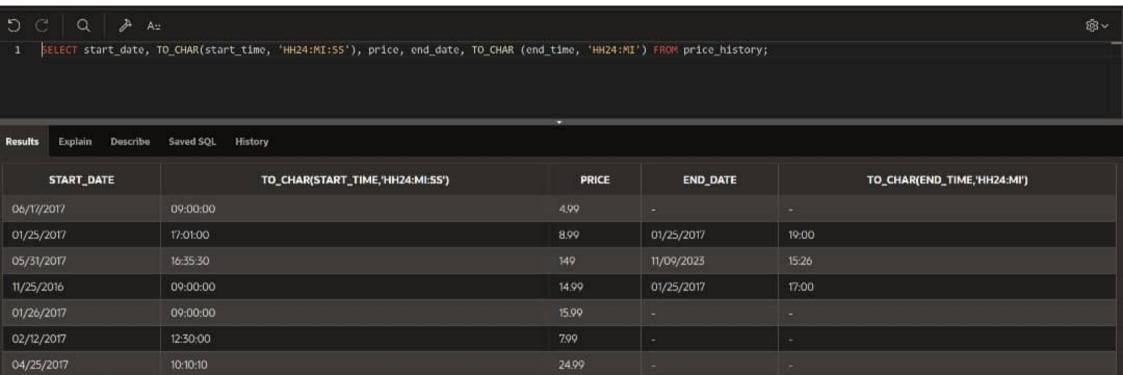
START_DATE	TO_CHAR(START_TIME, HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME,'HH24:MI')
06/17/2017	09:00:00	4.99		
01/25/2017	17:01:00	8.99	01/25/2017	19:00
05/31/2017	16:35:30	149		
11/25/2016	09:00:00	14.99	01/25/2017	17:00
01/26/2017	09:00:00	15.99		
02/12/2017	12:30:00	7.99		
04/25/2017	10:10:10	24.99		
roug returned to 0.02 coconde	- Hamilton			



2. UPPATE price_history SET and_date = SYSPATE, and_time = SYSTIMESTAMP WHERE itn_number = 'imolio1048' AND and_date IS NULL;



3. Rerun the scled statement on the price_history table to ensure that the statement has been excuted.

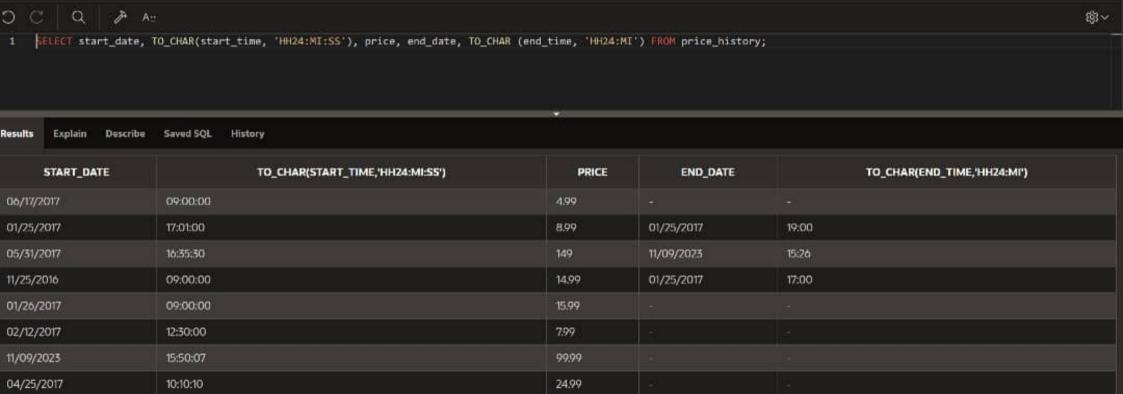




4. INSERT_INTO price history C start_date, start_time, price, itm_number) VALUES CSYSDATE, SYSTIMESTAMP, 99.99, imolio1048);



5. Rerun the select statement on the price-history table to ensure that the statement has been executed.





Part 2: Deleting rows from the system

1. DELETE FROM austomors_addresses WHERE address_line_1 = 83 Barhill Prive!;



DELETE FROM customers_addresses WHERE address_line_1 = '83 Barrhill Drive'; Explain Describe Saved SQL History Results 1 row(s) deleted. 0.01 seconds



2. SELECT * FROM austomors - addresses;



