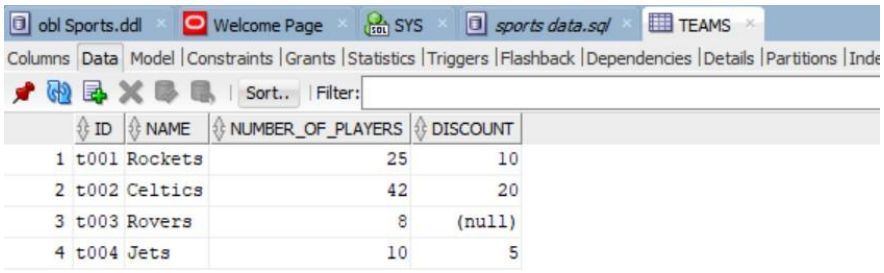
Exercise 1

1. Add a new team to the system

id	name	Number_of_players	discount
t004	Jets	10	5

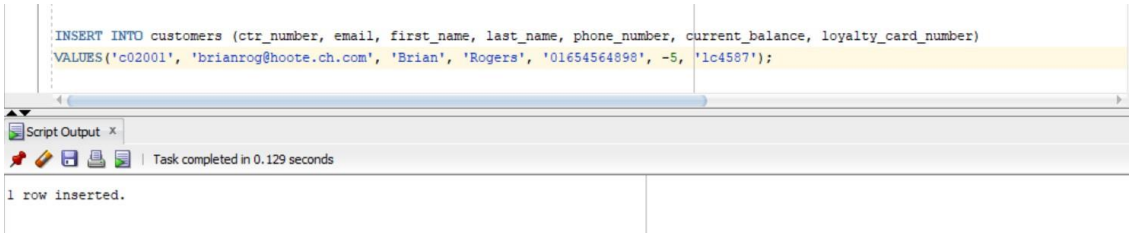
```
INSERT INTO ordered_items (quantity_ordered, quantity_shipped, odr_id, itm_number)
VALUES(1, 1, 'or01101750', 'im01101048');

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

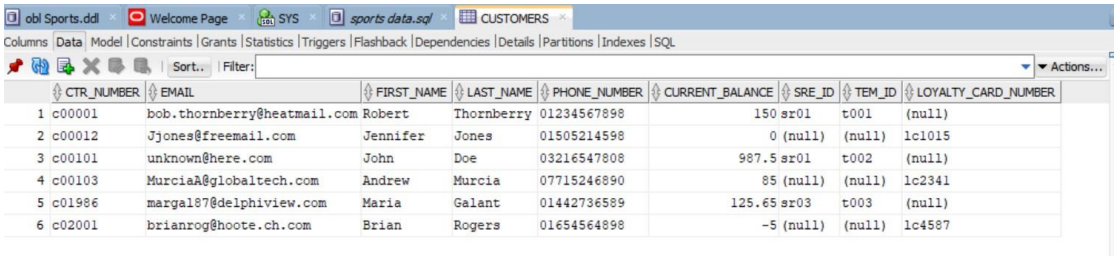


2. Add a new Customer with the following details to the system

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		

```
INSERT INTO customers (ctr_number, email, first_name, last_name, phone_number, current_balance, loyalty_card_number)
VALUES('c02001', 'brianrog@hoote.ch.com', 'Brian', 'Rogers', '01654564898', -5, 'lc4587');
```





3. This information violates the check constraint that the current balance must not be less than zero. Change the current balance to 50 and rerun the query.

```
UPDATE customers
SET current_balance = 50
WHERE current_balance < 0;
```

Script Output

Task completed in 0.112 seconds

```
1 row updated.
```

obl Sports.ddl | Welcome Page | SYS | *sports data.sql* | CUSTOMERS

Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

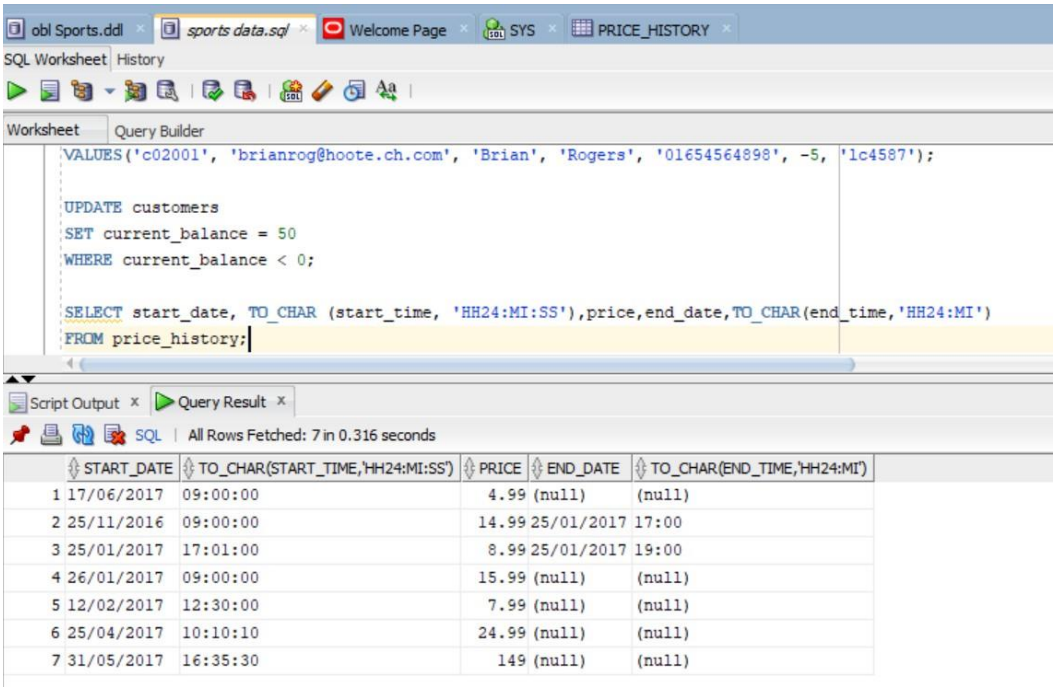
Sort.. | Filter: | Actions...

	CTR_NUMBER	EMAIL	FIRST_NAME	LAST_NAME	PHONE_NUMBER	CURRENT_BALANCE	SRE_ID	TEM_ID	LOYALTY_CARD_NUMBER
1	c00001	bob.thornberry@heatmail.com	Robert	Thornberry	01234567898	150	sr01	t001	(null)
2	c00012	Jjones@freemail.com	Jennifer	Jones	01505214598	0	(null)	(null)	lc1015
3	c00101	unknown@here.com	John	Doe	03216547808	987.5	sr01	t002	(null)
4	c00103	MurciaA@globaltech.com	Andrew	Murcia	07715246890	85	(null)	(null)	lc2341
5	c01986	margal87@delphiview.com	Maria	Galant	01442736589	125.65	sr03	t003	(null)
6	c02001	brianrog@hoote.ch.com	Brian	Rogers	01654564898	50	(null)	(null)	lc4587

Exercise 2-Part 1

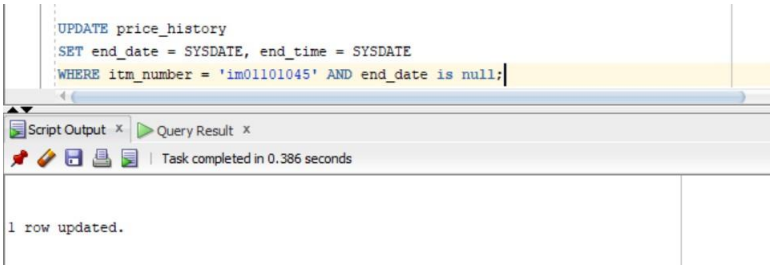
1. Run the following query to view the content of the price_history table:

```
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')
1	17/06/2017	09:00:00	4.99	(null)	(null)
2	25/11/2016	09:00:00	14.99	25/01/2017	17:00
3	25/01/2017	17:01:00	8.99	25/01/2017	19:00
4	26/01/2017	09:00:00	15.99	(null)	(null)
5	12/02/2017	12:30:00	7.99	(null)	(null)
6	25/04/2017	10:10:10	24.99	(null)	(null)
7	31/05/2017	16:35:30	149	(null)	(null)

2. Obl is going to update the price of the premium bat so you will need to write a query that will close off the current price by adding the system date values to the end_date and end_time fields. To run this query you will need to both match the item number and identify that the end date is null. This ensures that you are updating the latest price.



3. Rerun the select statement on the price_history table to ensure that the statement has been executed.

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

Script Output | Query Result

SQL | All Rows Fetched: 7 in 0.042 seconds

	START_DATE	TO_CHAR(START_TIME,'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME,'HH24:MI')
1	17/06/2017	09:00:00	4.99	(null)	(null)
2	25/11/2016	09:00:00	14.99	25/01/2017	17:00
3	25/01/2017	17:01:00	8.99	25/01/2017	19:00
4	26/01/2017	09:00:00	15.99	10/11/2023	16:57
5	12/02/2017	12:30:00	7.99	(null)	(null)
6	25/04/2017	10:10:10	24.99	(null)	(null)
7	31/05/2017	16:35:30	149	(null)	(null)

4. Insert a new row that will use the current date and time to set the new price of the premium bat to be 99.99.

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

Script Output | Query Result

Task completed in 0.123 seconds

```
1 row inserted.
```

5. Rerun the select statement on the price_history table to ensure that the statement has been executed.

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

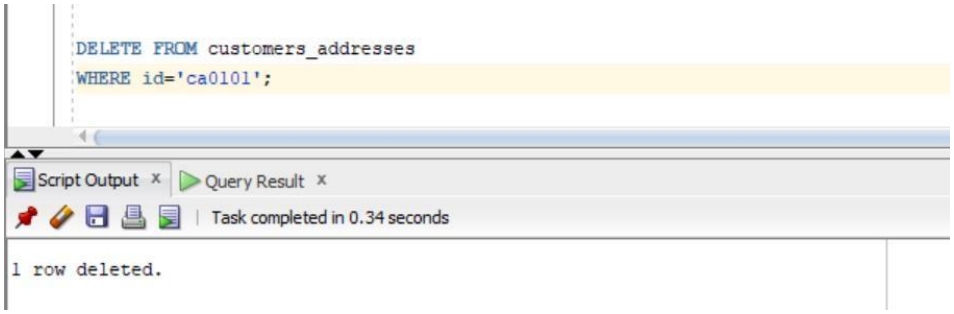
Script Output | Query Result

SQL | All Rows Fetched: 8 in 0.091 seconds

	START_DATE	TO_CHAR(START_TIME,'HH24:MI:SS')	PRICE	END_DATE	TO_CHAR(END_TIME,'HH24:MI')
1	17/06/2017	09:00:00	4.99	(null)	(null)
2	25/11/2016	09:00:00	14.99	25/01/2017	17:00
3	25/01/2017	17:01:00	8.99	25/01/2017	19:00
4	26/01/2017	09:00:00	15.99	10/11/2023	16:57
5	12/02/2017	12:30:00	7.99	(null)	(null)
6	25/04/2017	10:10:10	24.99	(null)	(null)
7	31/05/2017	16:35:30	149	(null)	(null)
8	10/11/2023	17:07:48	99.99	(null)	(null)

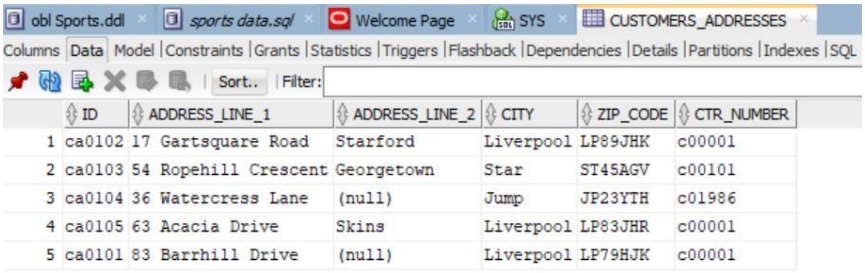
Exercise 2-Part 2

1. Bob Thornberry has contacted Obl to ask that the 83 Barrhill Drive address be removed from the system as he can longer receive parcels at this address. Write a SQL statement that will remove this address from the system.



2. Run a select statement on the customers_addresses table to ensure that the statement has been executed.

Before deletion:



After deletion: