Part 1- Updating rows to the system

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;
```

Statement 95

```
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')
17-JUN-17	09:00:00	4.99	-	-
25-NOV-16	09:00:00	14.99	25-JAN-17	17:00
25-JAN-17	17:01:00	8.99	25-JAN-17	19:00
26-JAN-17	09:00:00	15.99	-	-
12-FEB-17	12:30:00	7.99	-	-
25-APR-17	10:10:10	24.99	-	-
31-MAY-17	16:35:30	149	-	-

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.

Statement 96

```
UPDATE price_history
SET end_date = SYSDATE , end_time = SYSDATE
WHERE itm_number ='im01101048' AND end_date IS NULL
```

1 row(s) updated.

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

Statement 98

```
SELECT *from price_history
```

START_DATE	START_TIME	PRICE	END_DATE	END_TIME	ITM_NUMBER
17-JUN-17	17-JUN-16	4.99	-	-	im01101044
25-NOV-16	25-NOV-16	14.99	25-JAN-17	25-JAN-17	im01101045
25-JAN-17	25-JAN-17	8.99	25-JAN-17	25-JAN-17	im01101045
26-JAN-17	26-JAN-17	15.99	-	-	im01101045
12-FEB-17	12-FEB-17	7.99	-	-	im01101046
25-APR-17	25-APR-17	24.99	-	-	im01101047
31-MAY-17	31-MAY-17	149	09-NOV-23	09-NOV-23	im01101048

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.

Statement 99

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

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

Statement 100

```
SELECT* from price_history
```

START_DATE	START_TIME	PRICE	END_DATE	END_TIME	ITM_NUMBER
17-JUN-17	17-JUN-16	4.99	-	-	im01101044
25-NOV-16	25-NOV-16	14.99	25-JAN-17	25-JAN-17	im01101045
25-JAN-17	25-JAN-17	8.99	25-JAN-17	25-JAN-17	im01101045
26-JAN-17	26-JAN-17	15.99	-	-	im01101045
12-FEB-17	12-FEB-17	7.99	-	-	im01101046
25-APR-17	25-APR-17	24.99	-	-	im01101047
31-MAY-17	31-MAY-17	149	09-NOV-23	09-NOV-23	im01101048
09-NOV-23	09-NOV-23	99.99	-	-	im01101048

Part 2: Deleting rows from the system

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.

Statement 103

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

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

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