# Advanced SQL Techniques

Submitted on February 16, 2022

Shareable Link

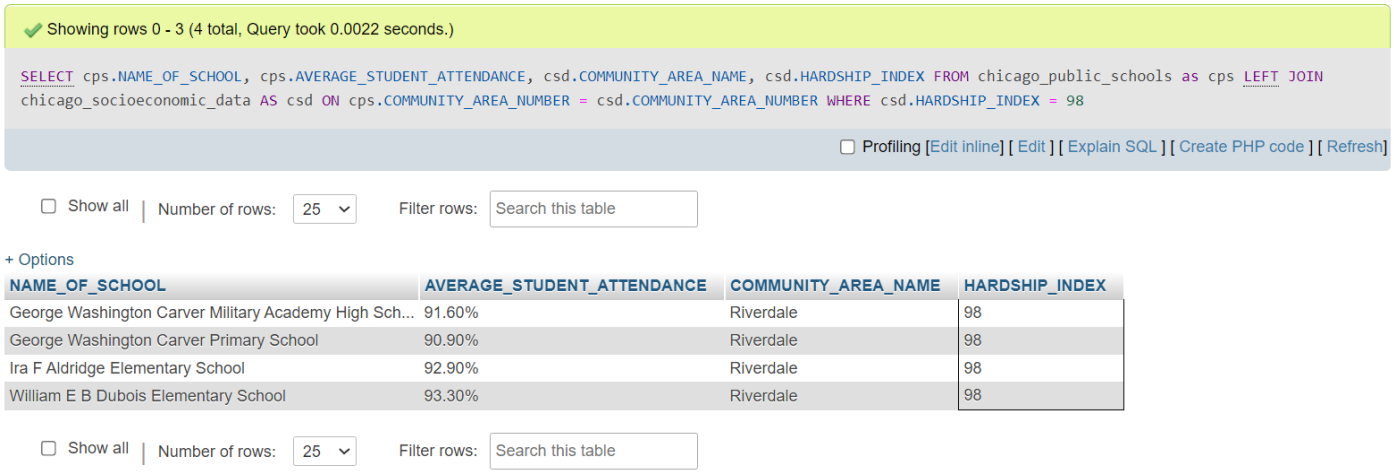
**PROMPT**

### Exercise 1, Question 1:

Write and execute a SQL query to list the school names, community names and average attendance for communities with a hardship index of 98.

Take a screenshot showing the SQL query and its results. Upload the JPEG (.jpg) file below for your peers to review.

the school names, community names and average attendance for communities with a hardship index of 98




SELECT cps.NAME_OF_SCHOOL, cps.AVERAGE_STUDENT_ATTENDANCE, csd.COMMUNITY_AREA_NAME, csd.HARDSHIP_INDEX FROM chicago_public_schools as cps LEFT JOIN chicago_socioeconomic_data AS csd ON cps.COMMUNITY_AREA_NUMBER = csd.COMMUNITY_AREA_NUMBER WHERE csd.HARDSHIP_INDEX = 98

**RUBRIC**

Did the learner find the correct 4 rows from the two tables?

TIP: If the screenshot appears small and is hard to read try zooming in by pressing "Ctrl" and "+" keys together (Mac: "Command" and "+"), or Right-click on the image and "View Image" (Firefox) or "Open Image in new Tab" (Chrome).

- 0 points
  Did not attempt the problem or response is incorrect.

- 1 point
  SQL query is incorrect, but does include a syntactically correct join. A syntactically correct left join should look like:

FROM <table name 1> <table alias 1> LEFT JOIN <table name 2> <table alias 2> ON <table alias 1>.<column name> = <table alias 2>.<column name>

**or** a right join like:

FROM <table name 1> <table alias 1> RIGHT JOIN <table name 2> <table alias 2> ON <table alias 1>.<column name> = <table alias 2>.<column name>

2 points
SQL query is correct; it uses a join, but returns more than 4 rows. The SQL query should either use a left join like:

FROM census_data a LEFT JOIN chicago_public_schools b

ON a.community_area_number = b.community_area_number

**or** a right join like:

FROM chicago_public_schools a RIGHT JOIN census_data b

ON a.community_area_number = b.community_area_number

**3 points**
**SQL query is correct; it uses a join and returns 4 rows. The SQL query should either use a left join like:**

**FROM census_data a LEFT JOIN chicago_public_schools b**

**ON a.community_area_number = b.community_area_number**

**or a right join like:**

**FROM chicago_public_schools a**
**RIGHT JOIN census_data b**

**ON a.community_area_number =**
**b.community_area_number**

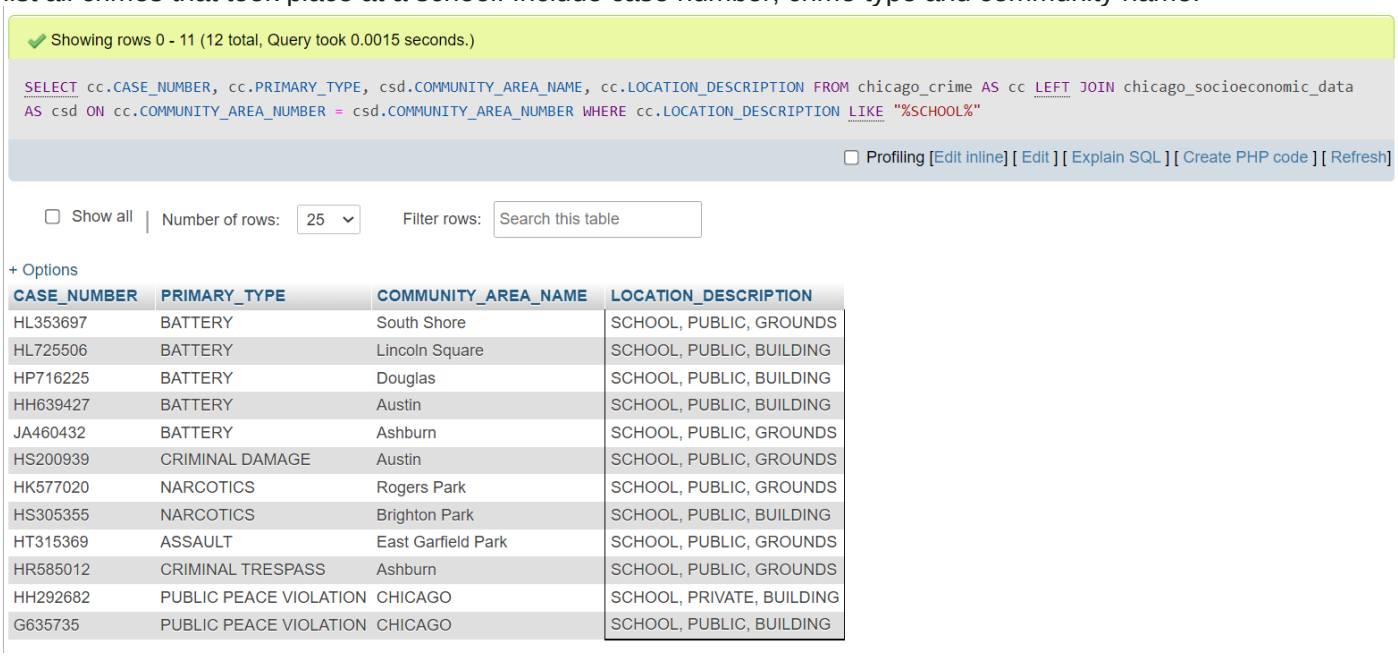
**PROMPT**

**Exercise 1, Question 2:**

Write and execute a SQL query to list all crimes that took place at a school. Include case number, crime type and community name.

Take a screenshot showing the SQL query and its results. Upload the JPEG (.jpg) file below for your peers to review.

list all crimes that took place at a school. Include case number, crime type and community name.




| CASE_NUMBER | PRIMARY_TYPE | COMMUNITY_AREA_NAME | LOCATION_DESCRIPTION |
|---|---|---|---|
| HL353697 | BATTERY | South Shore | SCHOOL, PUBLIC, GROUNDS |
| HL725506 | BATTERY | Lincoln Square | SCHOOL, PUBLIC, BUILDING |
| HP716225 | BATTERY | Douglas | SCHOOL, PUBLIC, BUILDING |
| HH639427 | BATTERY | Austin | SCHOOL, PUBLIC, BUILDING |
| JA460432 | BATTERY | Ashburn | SCHOOL, PUBLIC, GROUNDS |
| HS200939 | CRIMINAL DAMAGE | Austin | SCHOOL, PUBLIC, GROUNDS |
| HK577020 | NARCOTICS | Rogers Park | SCHOOL, PUBLIC, GROUNDS |
| HS305355 | NARCOTICS | Brighton Park | SCHOOL, PUBLIC, BUILDING |
| HT315369 | ASSAULT | East Garfield Park | SCHOOL, PUBLIC, GROUNDS |
| HR585012 | CRIMINAL TRESPASS | Ashburn | SCHOOL, PUBLIC, GROUNDS |
| HH292682 | PUBLIC PEACE VIOLATION | CHICAGO | SCHOOL, PRIVATE, BUILDING |
| G635735 | PUBLIC PEACE VIOLATION | CHICAGO | SCHOOL, PUBLIC, BUILDING |

SELECT cc.CASE_NUMBER, cc.PRIMARY_TYPE, csd.COMMUNITY_AREA_NAME, cc.LOCATION_DESCRIPTION FROM chicago_crime AS cc LEFT JOIN chicago_socioeconomic_data AS csd ON cc.COMMUNITY_AREA_NUMBER = csd.COMMUNITY_AREA_NUMBER WHERE cc.LOCATION_DESCRIPTION LIKE "%SCHOOL%"

**RUBRIC**

Did the learner find the correct 4 rows from the two tables?

TIP: If the screenshot appears small and is hard to read try zooming in by pressing "Ctrl" and "+" keys together (Mac: "Command" and "+"), or Right-click on the image and "View Image" (Firefox) or "Open Image in new Tab" (Chrome).

0 points
Did not attempt the problem or response is incorrect.

1 point

SQL query is incorrect but does include a syntactically correct join. A syntactically correct left join should look like:

FROM <table name 1> <table alias 1> LEFT JOIN <table name 2> <table alias 2> ON <table alias 1>.<column name> = <table alias 2>.<column name>

**or** a right join like:

FROM <table name 1> <table alias 1> RIGHT JOIN <table name 2> <table alias 2> ON <table alias 1>.<column name> = <table alias 2>.<column name>

2 points

SQL query is correct; it uses a join but returns more than 12 rows. The SQL query should either use a left join like:

FROM chicago_crime_data a LEFT JOIN census_data b ON a.community_area_number = b.community_area_number

**or** a right join like:

FROM census_data a RIGHT JOIN chicago_crime_data b ON a.community_area_number = b.community_area_number

**3 points**

**SQL query is correct; it uses a join and returns 12 rows. The SQL query should either use a left join like:**

**FROM chicago_crime_data a LEFT JOIN census_data b ON a.community_area_number = b.community_area_number**

**or a right join like:**

**FROM census_data a RIGHT JOIN chicago_crime_data b ON a.community_area_number = b.community_area_number**

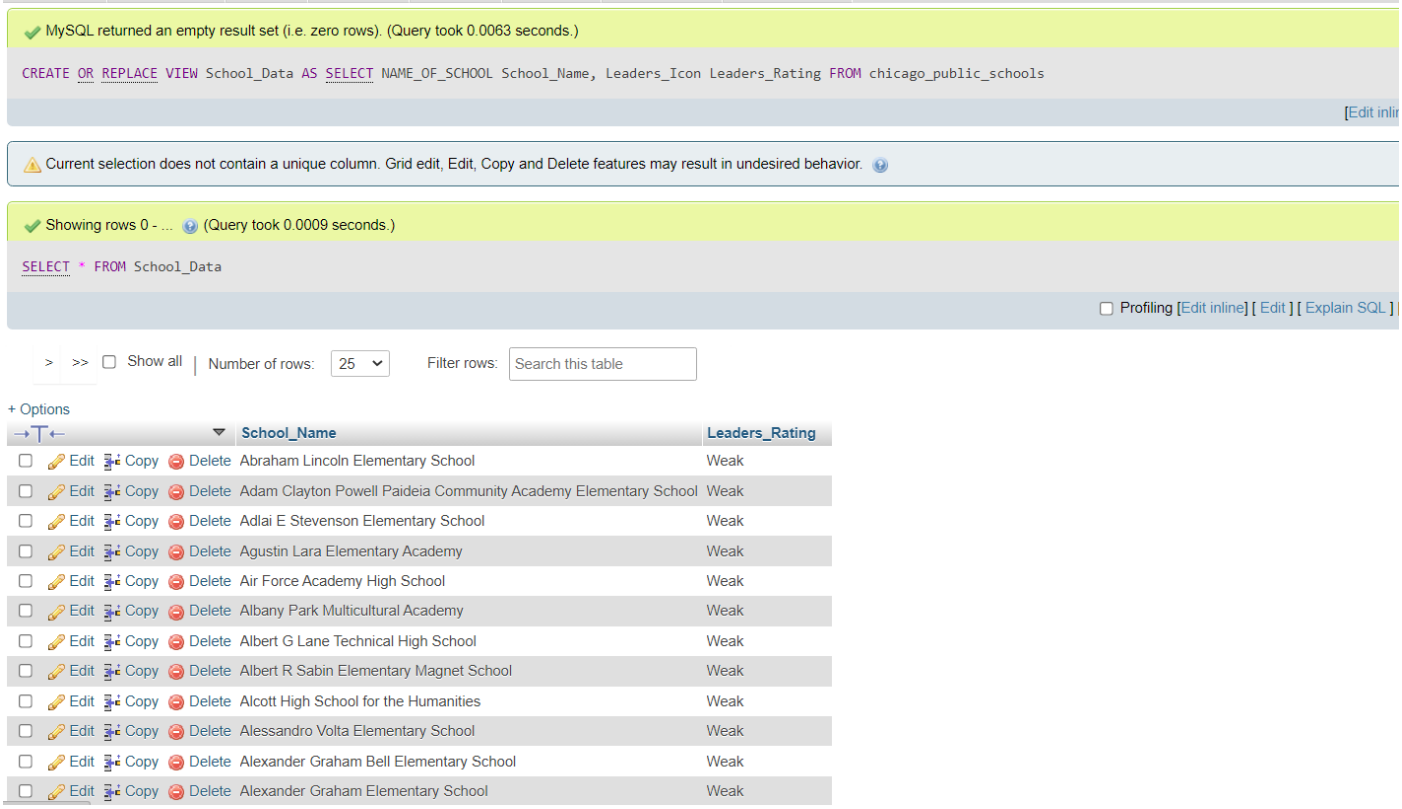
**PROMPT**

**Exercise 2, Question 1:**

Write and execute a SQL statement that returns just the school name and leaders' icon from the view.

Take a screenshot showing the SQL query and its results. Upload the JPEG (.jpg) file below for your peers to review.

the school name and leaders' icon from the view




CREATE OR REPLACE VIEW School_Data AS SELECT NAME_OF_SCHOOL School_Name, Leaders_Icon Leaders_Rating FROM chicago_public_schools; SELECT * FROM School_Data;

**RUBRIC**

Did the learner successfully use the view to return the two columns in their SELECT statement?

TIP: If the screenshot appears small and is hard to read try zooming in by pressing "Ctrl" and "+" keys together (Mac: "Command" and "+"), or Right-click on the image and "View Image" (Firefox) or "Open Image in new Tab" (Chrome).

0 points

Did not attempt the problem or response is incorrect.

1 point
SQL query returns some data from the view, but not necessarily the specified columns or using the correct names.

**2 points**
**SQL query correctly returns two columns, named SCHOOL_NAME and LEADERS_RATING from the view that they created.**

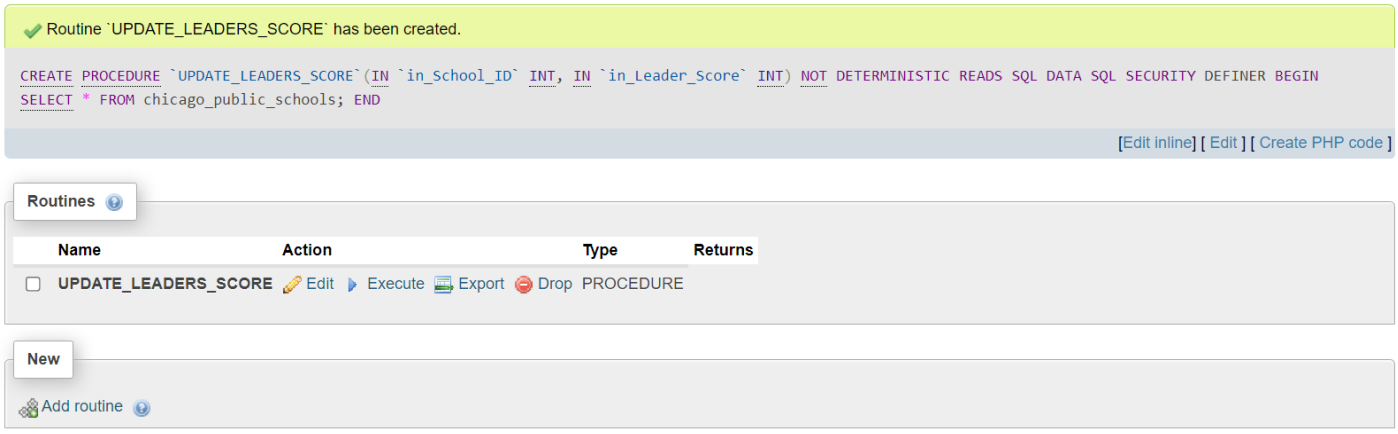
**PROMPT**

**Exercise 3, Question 1:**

Write the structure of a query to create or replace a stored procedure called UPDATE_LEADERS_SCORE that takes a in_School_ID parameter as an integer and a in_Leader_Score parameter as an integer. Don't forget to use the #SET TERMINATOR statement to use the @ for the CREATE statement terminator.

Take a screenshot showing the SQL query. Upload the JPEG (.jpg) file below for your peers to review.

create or replace a stored procedure called UPDATE_LEADERS_SCORE




CREATE PROCEDURE `UPDATE_LEADERS_SCORE`(IN `in_School_ID` INT, IN `in_Leader_Score` INT) NOT DETERMINISTIC READS SQL DATA SQL SECURITY DEFINER BEGIN SELECT * FROM chicago_public_schools; END

**RUBRIC**

Did the learner successfully write the shell of the SQL statement?

TIP: If the screenshot appears small and is hard to read try zooming in by pressing "Ctrl" and "+" keys together (Mac: "Command" and "+"), or Right-click on the image and "View Image" (Firefox) or "Open Image in new Tab" (Chrome).

0 points
Did not attempt the problem or response is incorrect.

1 point
SQL query contains most of the following, but has one line missing. --#SET TERMINATOR @ CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE (IN in_School_ID INTEGER, IN in_Leader_Score INTEGER) LANGUAGE SQL BEGIN END@

**2 points**
**SQL query is similar to the following: --#SET TERMINATOR @ CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE (IN in_School_ID INTEGER, IN in_Leader_Score INTEGER) LANGUAGE SQL BEGIN END@**

**PROMPT**

**Exercise 3, Question 2:**

Inside your stored procedure, write a SQL statement to update the Leaders_Score field in the CHICAGO_PUBLIC_SCHOOLS table for the school identified by in_School_ID to the value in the in_Leader_Score parameter.

Take a screenshot showing the SQL query. Upload the JPEG (.jpg) file below for your peers to review.

update the Leaders_Score field in the CHICAGO_PUBLIC_SCHOOLS table

```
1  --#SET TERMINATOR @
2  CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE(
3  IN in_School_ID INTEGER,
4  IN in_Leader_Score INTEGER)
5
6  LANGUAGE SQL
7  MODIFIES SQL DATA
8
9  BEGIN
10
11    UPDATE CHICAGO_PUBLIC_SCHOOLS
12    SET "Leaders_Score" = in_Leader_Score
13    WHERE "School_ID" = in_School_ID;
14
15 END
16
17 @
```

```
CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCOR...   Run time: 0.054 s
Status: Success | Affected Rows: 0
```

--#SET TERMINATOR @ CREATE OR REPLACE PROCEDURE UPDATE_LEADERS SCORE( IN in SchooLID INTEGER, IN in Leader Score INTEGER) LANGUAGE SQL MODIFIES SQL DATA BEGIN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders_Score" = in Leader Score WHERE "School ID" = in School ID; END @

**RUBRIC**

Did the learner successfully write the update statement inside the stored procedure definition? TIP: If the screenshot appears small and is hard to read try zooming in by pressing "Ctrl" and "+" keys together (Mac: "Command" and "+"), or Right-click on the image and "View Image" (Firefox) or "Open Image in new Tab" (Chrome).

- 0 points
  Did not attempt the problem or response is incorrect.

- 1 point
  SQL query contains most of the following, but has one line missing. --#SET TERMINATOR @ CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE (IN in_School_ID INTEGER, IN in_Leader_Score INTEGER) LANGUAGE SQL BEGIN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Score" = in_Leader_Score WHERE "School_ID" = in_School_ID; END@

- **2 points**
  
  **SQL query is similar to the following: --#SET TERMINATOR @ CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE (IN in_School_ID INTEGER, IN in_Leader_Score INTEGER) LANGUAGE SQL BEGIN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Score" = in_Leader_Score WHERE "School_ID" = in_School_ID; END@**

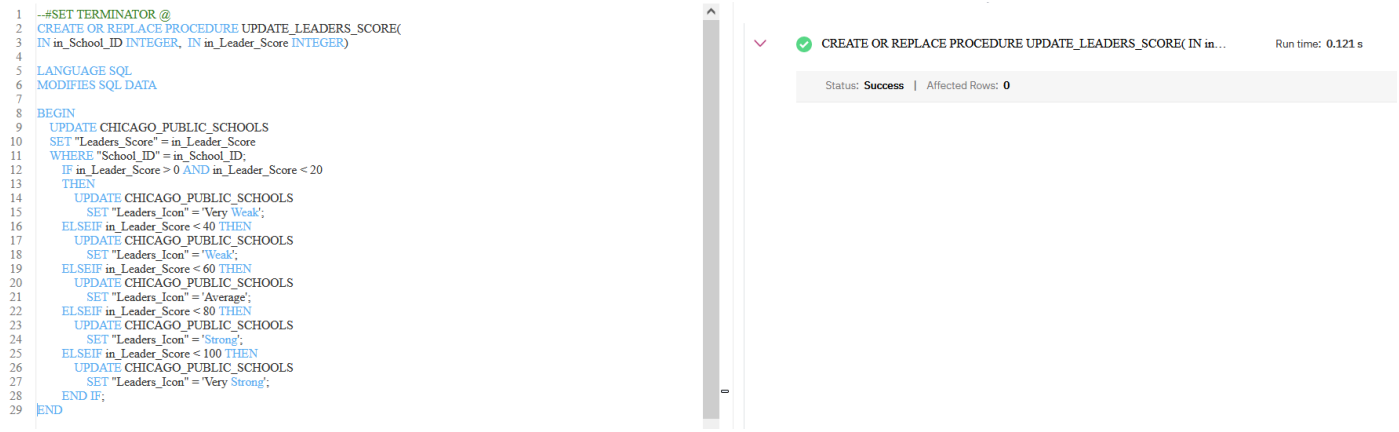
**PROMPT**

**Exercise 3, Question 3:**

Inside your stored procedure, write a SQL IF statement to update the Leaders_Icon field in the CHICAGO_PUBLIC_SCHOOLS table for the school identified by in_School_ID using the following information.

Take a screenshot showing the SQL query. Upload the JPEG (.jpg) file below for your peers to review.

SQL IF statement to update the Leaders_Icon field in the CHICAGO_PUBLIC_SCHOOLS table




--#SET TERMINATOR @ CREATE OR REPLACE PROCEDURE UPDATE LEADERS_SCORE( IN in School ID INTEGER, IN in Leader Score INTEGER) LANGUAGE SQL MODIHES SQL DATA BEGIN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders_Score" = in Leader Score WHERE "School ID" = in School ID; IF in Leader Score > 0 AND in Leader Score < 20 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Very Weak; ELSEIF in Leader Score < 40 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders_kon" =Weak; ELSEIF in Leader Score < 60 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders_kon" ='Average'; ELSEIF in Leader Score < 80 THEN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders_kon" = 'Strong'; ELSEIF in Leader Score < 100 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders_kon" = 'Very Strong'; END IF; END

**RUBRIC**

Did the learner successfully write the IF statement inside the stored procedure definition? TIP: If the screenshot appears small and is hard to read try zooming in by pressing "Ctrl" and "+" keys together (Mac: "Command" and "+"), or Right-click on the image and "View Image" (Firefox) or "Open Image in new Tab" (Chrome).

0 points
Did not attempt the problem or response is incorrect.

1 point
SQL query contains most of the following, but has one line missing.
CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE (IN in_School_ID INTEGER, IN in_Leader_Score INTEGER) LANGUAGE SQL BEGIN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Score" = in_Leader_Score WHERE "School_ID" = in_School_ID; IF in_Leader_Score > 0 AND in_Leader_Score < 20 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Very Weak';

WHERE in_School_ID = "School_ID"
AND in_Leader_Score =
"Leaders_Score";

ELSEIF in_Leader_Score < 40 THEN
UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Icon" = 'Weak';

WHERE in_School_ID = "School_ID"
AND in_Leader_Score =
"Leaders_Score"; ELSEIF
in_Leader_Score < 60 THEN UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Icon" = 'Average';

WHERE in_School_ID = "School_ID"
AND in_Leader_Score =
"Leaders_Score"; ELSEIF
in_Leader_Score < 80 THEN UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Icon" = 'Strong';

WHERE in_School_ID = "School_ID"
AND in_Leader_Score =
"Leaders_Score"; ELSEIF
in_Leader_Score < 100 THEN
UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Icon" = 'Very Strong'

WHERE in_School_ID = "School_ID"
AND in_Leader_Score =
"Leaders_Score"; END IF; END@

**2 points**

**SQL query is similar to the following:**
**CREATE OR REPLACE PROCEDURE**
**UPDATE_LEADERS_SCORE (IN**
**in_School_ID INTEGER, IN**
**in_Leader_Score INTEGER)**
**LANGUAGE SQL BEGIN UPDATE**
**CHICAGO_PUBLIC_SCHOOLS SET**
**"Leaders_Score" = in_Leader_Score**
**WHERE "School_ID" = in_School_ID;**
**IF in_Leader_Score > 0 AND**
**in_Leader_Score < 20 THEN UPDATE**

**CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Very Weak'; ELSEIF in_Leader_Score < 40 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Weak'; ELSEIF in_Leader_Score < 60 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Average'; ELSEIF in_Leader_Score < 80 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Strong'; ELSEIF in_Leader_Score < 100 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Very Strong'; END IF; END@**

**PROMPT**

**Exercise 3, Question 4:**

Run your code to create the stored procedure.

Take a screenshot showing the SQL query and its results. Upload the JPEG (.jpg) file below for your peers to review.

create the stored procedure

```
--#SET TERMINATOR @
CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE(
IN in_School_ID INTEGER,  IN in_Leader_Score INTEGER)

LANGUAGE SQL
MODIFIES SQL DATA

BEGIN
    UPDATE CHICAGO_PUBLIC_SCHOOLS
    SET "Leaders_Score" = in_Leader_Score
    WHERE "School_ID" = in_School_ID;
        IF in_Leader_Score > 0 AND in_Leader_Score < 20
        THEN
            UPDATE CHICAGO_PUBLIC_SCHOOLS
                SET "Leaders_Icon" = 'Very Weak';
        ELSEIF in_Leader_Score < 40 THEN
            UPDATE CHICAGO_PUBLIC_SCHOOLS
                SET "Leaders_Icon" = 'Weak';
        ELSEIF in_Leader_Score < 60 THEN
            UPDATE CHICAGO_PUBLIC_SCHOOLS
                SET "Leaders_Icon" = 'Average';
        ELSEIF in_Leader_Score < 80 THEN
            UPDATE CHICAGO_PUBLIC_SCHOOLS
                SET "Leaders_Icon" = 'Strong';
        ELSEIF in_Leader_Score < 100 THEN
            UPDATE CHICAGO_PUBLIC_SCHOOLS
                SET "Leaders_Icon" = 'Very Strong';
        END IF;
END
```

```
CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE( IN in...   Run time: 0.121 s
Status: Success | Affected Rows: 0
```

--#SET TERMINATOR @ CREATE OR REPLACE PROCEDURE UPDATE LEADERS_SCORE( IN in School ID INTEGER, IN in Leader Score INTEGER) LANGUAGE SQL MODIFIES SQL DATA BEGIN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders_Score" = in Leader Score WHERE "School ID" = in School ID; IF in Leader Score > 0 AND in Leader Score < 20 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Very Weak; ELSEIF in Leader Score < 40 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders_kon" = 'Weak% ELSEIF in Leader Score < 60 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders_kon" ='Average'; ELSEIF in Leader Score < 80 THEN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders_kon" ='Strong'; ELSEIF in Leader Score < 100 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders_kon" = 'Very Strong; END IF; END

**RUBRIC**

Did the learner successfully create the stored procedure? TIP: If the screenshot appears small and is hard to read try zooming in by pressing "Ctrl" and "+" keys together (Mac: "Command" and "+"), or Right-click on the image and "View Image" (Firefox) or "Open Image in new Tab" (Chrome).

0 points
Did not attempt the problem or response is incorrect.

**1 point**
**SQL query contains most of the following and the results pane states that the stored procedure was successfully created. CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE (IN in_School_ID INTEGER, IN in_Leader_Score INTEGER) LANGUAGE SQL BEGIN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Score" = in_Leader_Score WHERE "School_ID" = in_School_ID; IF in_Leader_Score > 0 AND in_Leader_Score < 20 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Very Weak'; ELSEIF in_Leader_Score < 40 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Weak'; ELSEIF in_Leader_Score < 60 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Average'; ELSEIF in_Leader_Score < 80 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Strong'; ELSEIF in_Leader_Score < 100 THEN UPDATE CHICAGO_PUBLIC_SCHOOLS SET "Leaders_Icon" = 'Very Strong'; END IF; END@**

2 points
SQL query is correct as follows and the results pane states that the stored procedure was successfully created.

```
CREATE OR REPLACE PROCEDURE
UPDATE_LEADERS_SCORE (IN
in_School_ID INTEGER, IN
in_Leader_Score INTEGER)
LANGUAGE SQL BEGIN UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Score" = in_Leader_Score
WHERE "School_ID" = in_School_ID;
IF in_Leader_Score > 0 AND
in_Leader_Score < 20 THEN UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Icon" = 'Very Weak'; ELSEIF
in_Leader_Score < 40 THEN UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Icon" = 'Weak'; ELSEIF
in_Leader_Score < 60 THEN UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Icon" = 'Average'; ELSEIF
in_Leader_Score < 80 THEN UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Icon" = 'Strong'; ELSEIF
in_Leader_Score < 100 THEN
UPDATE
CHICAGO_PUBLIC_SCHOOLS SET
"Leaders_Icon" = 'Very Strong'; END IF;
END@
```

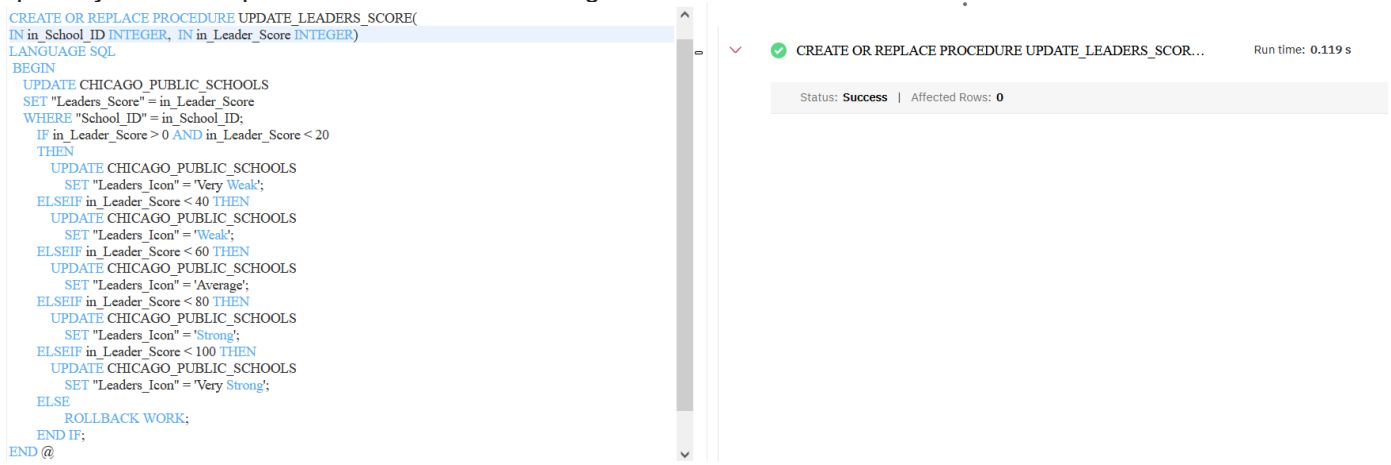
**PROMPT**

### Exercise 4, Question 1:

Update your stored procedure definition. Add a generic ELSE clause to the IF statement that rolls back the current work if the score did not fit any of the preceding categories.

Take a screenshot showing the SQL query. Upload the JPEG (.jpg) file below for your peers to review.

Update your stored procedure definition. Add a generic ELSE clause to the IF statement



```
CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE(
IN in_School_ID INTEGER,  IN in_Leader_Score INTEGER)
LANGUAGE SQL
 BEGIN
   UPDATE CHICAGO_PUBLIC_SCHOOLS
   SET "Leaders_Score" = in_Leader_Score
   WHERE "School_ID" = in_School_ID;
     IF in_Leader_Score > 0 AND in_Leader_Score < 20
     THEN
        UPDATE CHICAGO_PUBLIC_SCHOOLS
           SET "Leaders_Icon" = 'Very Weak';
     ELSEIF in_Leader_Score < 40 THEN
        UPDATE CHICAGO_PUBLIC_SCHOOLS
           SET "Leaders_Icon" = 'Weak';
     ELSEIF in_Leader_Score < 60 THEN
        UPDATE CHICAGO_PUBLIC_SCHOOLS
           SET "Leaders_Icon" = 'Average';
     ELSEIF in_Leader_Score < 80 THEN
        UPDATE CHICAGO_PUBLIC_SCHOOLS
           SET "Leaders_Icon" = 'Strong';
     ELSEIF in_Leader_Score < 100 THEN
        UPDATE CHICAGO_PUBLIC_SCHOOLS
           SET "Leaders_Icon" = 'Very Strong';
     ELSE
          ROLLBACK WORK;
     END IF;
END @
```

```
CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCOR...   Run time: 0.119 s
Status: Success | Affected Rows: 0
```

CREATE OR REPLACE PROCEDURE UPDATE LEADERS_SCORE( IN in School ID INTEGER, IN in Leader Score INTEGER) SQL BEGIN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders Score" = in Leader _Score WHERE "School ID" = in School IF in Leader Score > 0 AND in Leader_Score < 20 THEN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders Icon" = 'Very Weak'; ELSEIF in Leader—Score < 40 THEN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders Icon" = 'Weak ELSEIF in Leader—Score < 60 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders Icon" = 'Average% ELSEIF in Leader—Score < 80 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders Icon" = 'Strong% ELSEIF in Leader—Score < 100 THEN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders_Icon" = 'Very Strong'; ELSE ROLLBACK WORK; END IF; END @

**RUBRIC**

Did the learner correctly add the ROLLBACK WORK statement? TIP: If the screenshot appears small and is hard to read try zooming in by pressing "Ctrl" and "+" keys together (Mac: "Command" and "+"), or Right-click on the image and "View Image" (Firefox) or "Open Image in new Tab" (Chrome).

- 0 points
  Did not attempt the problem or response is incorrect.

- **1 point**
  **Stored procedure definition now contains a ROLLBACK WORK statement, but not in the correct place.**

- 2 points
  Stored procedure definition now contains an ELSE clause and ROLLBACK WORK before the END IF statement.

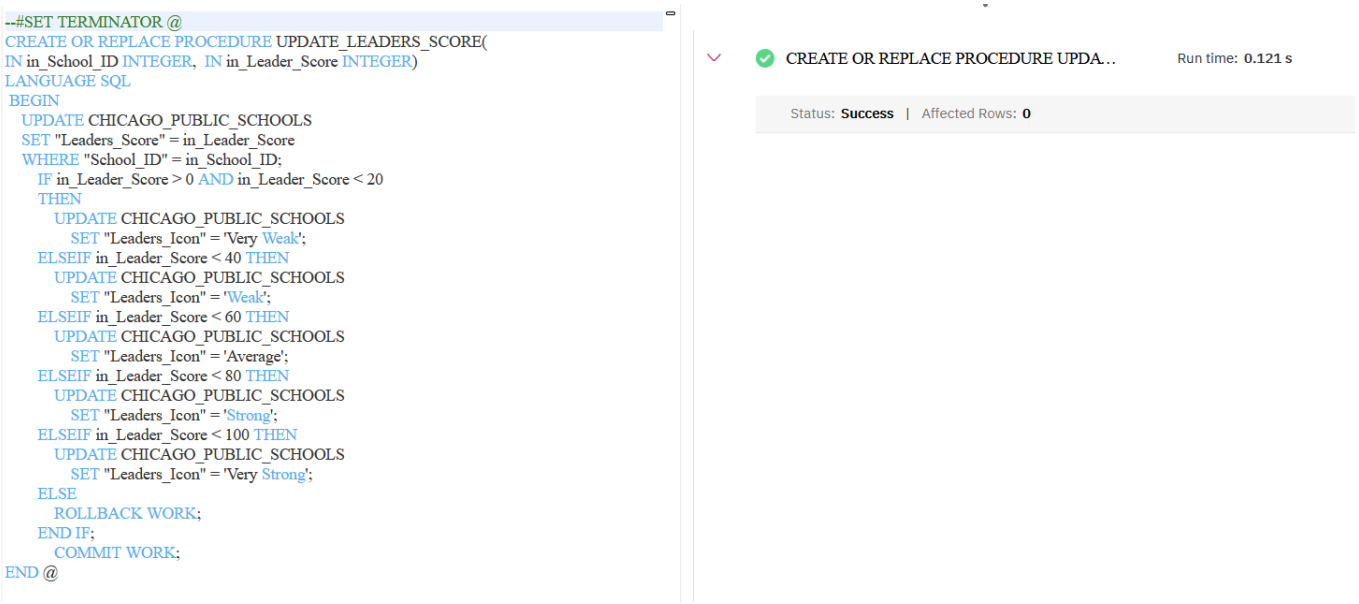
**PROMPT**

**Exercise 4, Question 2:**

Update your stored procedure definition again. Add a statement to commit the current unit of work at the end of the procedure.

Take a screenshot showing the SQL query. Upload the JPEG (.jpg) file below for your peers to review.

Updating Stored Procedure



```
--#SET TERMINATOR @
CREATE OR REPLACE PROCEDURE UPDATE_LEADERS_SCORE(
IN in_School_ID INTEGER,  IN in_Leader_Score INTEGER)
LANGUAGE SQL
 BEGIN
   UPDATE CHICAGO_PUBLIC_SCHOOLS
   SET "Leaders_Score" = in_Leader_Score
   WHERE "School_ID" = in_School_ID;
     IF in_Leader_Score > 0 AND in_Leader_Score < 20
     THEN
       UPDATE CHICAGO_PUBLIC_SCHOOLS
         SET "Leaders_Icon" = 'Very Weak';
     ELSEIF in_Leader_Score < 40 THEN
       UPDATE CHICAGO_PUBLIC_SCHOOLS
         SET "Leaders_Icon" = 'Weak';
     ELSEIF in_Leader_Score < 60 THEN
       UPDATE CHICAGO_PUBLIC_SCHOOLS
         SET "Leaders_Icon" = 'Average';
     ELSEIF in_Leader_Score < 80 THEN
       UPDATE CHICAGO_PUBLIC_SCHOOLS
         SET "Leaders_Icon" = 'Strong';
     ELSEIF in_Leader_Score < 100 THEN
       UPDATE CHICAGO_PUBLIC_SCHOOLS
         SET "Leaders_Icon" = 'Very Strong';
     ELSE
       ROLLBACK WORK;
     END IF;
       COMMIT WORK;
END @
```



--#SET TERMINATOR @ CREATE OR REPLACE PROCEDURE UPDATE LEADERS_SCORE( IN in School ID INTEGER, N in Leader Score INTEGER)LANGUAGE SQLBEGIN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders Score" = in Leader Score WHERE "School _ID" = in School ID; IF in Leader Score > 0 AND in Leader Score < 20 THEN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders Icon" = 'Very Weak% ELSEIF in LeaderScore < 40 THEN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders Icon" = 'Weak ELSEIF in Leader—Score < 60 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders Icon" = 'Average% ELSEIF in Leader—Score < 80 THEN UPDATE CHICAGO PUBLIC_SCHOOLS SET "Leaders Icon" = 'Strong% ELSEIF in Leader—Score < 100 THEN UPDATE CHICAGO PUBLIC SCHOOLS SET "Leaders_Icon" = 'Very Strong'; ELSE ROLLBACK WORK; END IF; COMMIT WORK; END @

**RUBRIC**

Did the learner correctly add the COMMIT WORK statement? TIP: If the screenshot appears small and is hard to read try zooming in by pressing "Ctrl" and "+" keys together (Mac: "Command" and "+"), or Right-click on the image and "View Image" (Firefox) or "Open Image in new Tab" (Chrome).

- 0 points
  Did not attempt the problem or response is incorrect.

- 1 point
  Stored procedure definition now contains a COMMIT WORK statement, but not in the correct place.

- **2 points**
  **Stored procedure definition now contains a COMMIT WORK statement after the END IF statement.**