**Add teacher to database:**

INSERT INTO dbo.profiles (userID, username, password, email, profileType, class, mathLevel, englishLevel, historyLevel, scienceLevel)

VALUES((SELECT MAX(userID) FROM dbo.profiles)+1, '***<teacher\_name>***', '***<password>***', '***<email\_address>***', 1, NULL, 0, 0, 0, 0);

INSERT INTO dbo.classroom (classID, teacherName)

VALUES ((SELECT MAX(classID) FROM dbo.classroom)+1, (SELECT MAX(userID) FROM dbo.profiles));

UPDATE dbo.profiles

SET class = (SELECT MAX(classID) FROM dbo.classroom)

WHERE userID = (SELECT MAX(userID) FROM dbo.profiles);

**Remove teacher from database:**

UPDATE dbo.profiles

SET class = NULL

WHERE class = (SELECT classID from dbo.classroom

WHERE teacherName = (SELECT userID FROM dbo.profiles WHERE username = '***<teacher\_name>***'));

DELETE FROM dbo.classroom

WHERE teacherName = (SELECT userID FROM dbo.profiles WHERE username = '***<teacher\_name>***');

DELETE FROM dbo.profiles

WHERE username = '***<teacher\_name>***';

**Add student to database:**

**A) when teacher/classroom is NOT known:**

INSERT INTO dbo.profiles (userID, username, password, email, profileType, class, mathLevel, englishLevel, historyLevel, scienceLevel)

VALUES((SELECT MAX(userID) FROM dbo.profiles)+1, '***<student\_name>***', '***<password>***', '***<email\_address>***', 0, NULL, 0, 0, 0, 0);

**B) when teacher/classroom IS known**

INSERT INTO dbo.profiles (userID, username, password, email, profileType, class, mathLevel, englishLevel, historyLevel, scienceLevel)

VALUES((SELECT MAX(userID) FROM dbo.profiles)+1, '***<student\_name>***', '***<password>***', '***<email\_address>***', 0,

(SELECT classID FROM dbo.classroom WHERE dbo.classroom.teacherName = (SELECT userID from dbo.profiles WHERE username = '***<teacher\_name>***')), 0, 0, 0, 0);

**Delete student from database:**

DELETE FROM dbo.profiles WHERE username = '***<student\_name>***';

**Change password (either teacher or student):**

UPDATE dbo.profiles

SET password = '***<new\_password>***'

WHERE username = '***<user\_name>***';

**Add student to classroom (after they are already in database):**

UPDATE dbo.profiles SET class = (SELECT classID FROM dbo.classroom WHERE dbo.classroom.teacherName =

(SELECT userID from dbo.profiles WHERE username = '***<teacher\_name>***'))

WHERE username = '***<student\_name>***';

**Increase Math Score**

UPDATE dbo.profiles

SET mathLevel = (SELECT mathLevel FROM dbo.profiles

WHERE username = '***<student\_name>***')+1 WHERE username = '***<student\_name>***';

**Increase English Score**

UPDATE dbo.profiles

SET englishLevel = (SELECT englishLevel FROM dbo.profiles

WHERE username = '***<student\_name>***')+1 WHERE username = '***<student\_name>***';

**Increase History Score**

UPDATE dbo.profiles

SET historyLevel = (SELECT historyLevel FROM dbo.profiles

WHERE username = '***<student\_name>***')+1 WHERE username = '***<student\_name>***';

**Increase Science Score**

UPDATE dbo.profiles

SET scienceLevel = (SELECT scienceLevel FROM dbo.profiles

WHERE username = '***<student\_name>***')+1 WHERE username = '***<student\_name>***';

**List all Students in Classroom (by Teacher)**

SELECT username FROM dbo.profiles WHERE class =

(SELECT classID from dbo.classroom WHERE teacherName =

(SELECT userID FROM dbo.profiles WHERE username = '***<teacher\_name>***'))

AND profileType = 0;

**Add all points for a student**

SELECT (mathLevel + englishLevel + historyLevel + scienceLevel)

FROM dbo.profiles WHERE username = '***<student\_name>***';

**Rank ALL Students by total points**

SELECT username,

(mathLevel + englishLevel + historyLevel + scienceLevel) AS total

FROM dbo.profiles ORDER BY total DESC;

**Rank Students in class by total points**

SELECT username,

(mathLevel + englishLevel + historyLevel + scienceLevel) AS total

FROM dbo.profiles WHERE class =

(SELECT classID from dbo.classroom WHERE teacherName =

(SELECT userID FROM dbo.profiles WHERE username = '***<teacher\_name>***'))

AND profileType = 0 ORDER BY total DESC;