

# SQL Data Manipulation 3

Erin Goeke

March 14, 2017

Hello again! I omitted some redundant expressions such as ASC because thats the default! I kept them in some places because the computer still spits out the same thing! If you would prefer me to explicitly write out everything I'd be happy to!

## 1

```
SELECT AVG(price) FROM albumitem
WHERE album_id IN
  (SELECT album_id FROM album
   WHERE album_artist = 'The Beatles');
```

## 2

```
SELECT * FROM albumitem
WHERE price > (SELECT AVG(price) FROM albumitem);
```

## 3

```
UPDATE albumitem
SET price = .8*price
WHERE album_id =
  (SELECT album_id FROM album
   WHERE album_title = 'The Blueprint');
```

## 4

```
DELETE FROM albumitem
WHERE album_id IN
(SELECT album_id FROM album
WHERE album_artist = 'The Smiths');
```

## 5

```
SELECT album.album_artist, album.album_year, album.album_title, albumitem.price
FROM album, albumitem
WHERE albumitem.album_id = album.album_id AND (album.album_year
BETWEEN 1980 AND 1990);
```

## 6

```
SELECT album.album_title, album.album_artist, album.album_year, AVG(albumitem.price)
WHERE albumitem.album_id = album.album_id AND (album.album_year BETWEEN 1980 AND 1990)
GROUP BY album.album_title, album.album_artist, album.album_year
ORDER BY album.album_artist ASC;
```

## 7

```
DELETE FROM albumitem
WHERE album_id IN (SELECT album_id FROM album WHERE album_year BETWEEN 1990 AND 2000);

SELECT album.album_artist, album.album_title, album.album_year, albumitem.price, albumitem.album_id
FROM album
LEFT OUTER JOIN albumitem ON albumitem.album_id = album.album_id
GROUP BY album.album_artist, album.album_title, album.album_year, albumitem.price;
```