```
print("Hello World")
```

```
Hello World
```

This is me writing some linking commentary in *Markdown* - Bullet one - Bullet two

bbc website

```
print("Something")
```

```
import sqlite3 as sql


conn = sql.connect("db.sqlite")


c = conn.cursor()
```

This created a connection and a cursor

```
my_big_query = '''
SELECT Artist.Name,Album.Title,Track.Name
FROM Artist
INNER JOIN Album
ON Artist.ArtistId=Album.ArtistId
INNER JOIN Track
ON Album.AlbumId=Track.AlbumID
ORDER BY Artist.Name,Album.Title;
'''
```

This was constructed in the console and then I simply copied and pasted into Jupyter.

```
c.execute(my_big_query)
```

```
<sqlite3.Cursor at 0x112eb71c0>
```

This runs the query, but notice that the output is huge. So we don’t use a `.fetchall()` (I tried - it was silly)…

```
# query_response = c.fetchall()


# This is a good option - we'll just grab 10 to prove a point ;)


query_response = c.fetchmany(size=10)
```

```
for line in query_response:
    print(line)
```

```
('AC/DC', 'For Those About To Rock We Salute You', 'For Those About To Rock (We Salute You)')
('AC/DC', 'For Those About To Rock We Salute You', 'Put The Finger On You')
('AC/DC', 'For Those About To Rock We Salute You', "Let's Get It Up")
('AC/DC', 'For Those About To Rock We Salute You', 'Inject The Venom')
('AC/DC', 'For Those About To Rock We Salute You', 'Snowballed')
('AC/DC', 'For Those About To Rock We Salute You', 'Evil Walks')
('AC/DC', 'For Those About To Rock We Salute You', 'C.O.D.')
('AC/DC', 'For Those About To Rock We Salute You', 'Breaking The Rules')
('AC/DC', 'For Those About To Rock We Salute You', 'Night Of The Long Knives')
('AC/DC', 'For Those About To Rock We Salute You', 'Spellbound')
```

```
my_big_query = '''
SELECT Artist.Name,Album.Title,Track.Name
FROM Artist
INNER JOIN Album
ON Artist.ArtistId=Album.ArtistId
INNER JOIN Track
ON Album.AlbumId=Track.AlbumID
WHERE Artist.Name="Van Halen"
'''


c.execute(my_big_query)
query_response = c.fetchmany(10)
for line in query_response:
    print(line)
```

```
('Van Halen', 'Diver Down', 'Where Have All The Good Times
Gone?')
('Van Halen', 'Diver Down', "Hang 'Em High")
('Van Halen', 'Diver Down', 'Cathedral')
('Van Halen', 'Diver Down', 'Secrets')
('Van Halen', 'Diver Down', 'Intruder')
('Van Halen', 'Diver Down', '(Oh) Pretty Woman')
('Van Halen', 'Diver Down', 'Dancing In The Street')
('Van Halen', 'Diver Down', 'Little Guitars (Intro)')
('Van Halen', 'Diver Down', 'Little Guitars')
('Van Halen', 'Diver Down', 'Big Bad Bill (Is Sweet William
Now)')
```

```
genre_choice = input("Pick a genre: ")

id_query = '''
SELECT GenreId FROM Genre
WHERE Genre.Name=?

'''

c.execute(id_query, (genre_choice,))

genre_id, = c.fetchone()

for line in query_response:
    print(line)

genre_query = '''
SELECT * FROM Track
INNER JOIN Album
ON Track.AlbumId=Album.AlbumId
WHERE Track.GenreId=?
GROUP BY Album.Title
'''

c.execute(genre_query, (genre_id,))
query_response = c.fetchmany(10)

for line in query_response:
    print(line)
```

```
('Van Halen', 'Diver Down', 'Where Have All The Good Times
Gone?')
('Van Halen', 'Diver Down', "Hang 'Em High")
```

```
('Van Halen', 'Diver Down', 'Cathedral')
('Van Halen', 'Diver Down', 'Secrets')
('Van Halen', 'Diver Down', 'Intruder')
('Van Halen', 'Diver Down', '(Oh) Pretty Woman')
('Van Halen', 'Diver Down', 'Dancing In The Street')
('Van Halen', 'Diver Down', 'Little Guitars (Intro)')
('Van Halen', 'Diver Down', 'Little Guitars')
('Van Halen', 'Diver Down', 'Big Bad Bill (Is Sweet William
Now)')
(3288, 'Rock You Like a Hurricane', 257, 2, 1, None, 255766,
4300973, 0.99, 257, '20th Century Masters - The Millennium
Collection: The Best of Scorpions', 179)
(1201, 'Different World', 94, 2, 1, None, 258692, 4383764, 0.99,
94, 'A Matter of Life and Death', 90)
(2506, 'Nothing To Say', 203, 1, 1, 'Chris Cornell/Kim Thayil',
238027, 7744833, 0.99, 203, 'A-Sides', 132)
(2926, 'Zoo Station', 232, 1, 1, 'U2', 276349, 9056902, 0.99,
232, 'Achtung Baby', 150)
(2938, 'Beautiful Day', 233, 1, 1, 'Adam Clayton, Bono, Larry
Mullen, The Edge', 248163, 8056723, 0.99, 233, "All That You
Can't Leave Behind", 150)
(1146, 'Welcome to the Jungle', 90, 2, 1, None, 273552, 4538451,
0.99, 90, 'Appetite for Destruction', 88)
(1479, 'Foxy Lady', 120, 1, 1, 'Jimi Hendrix', 199340, 6480896,
0.99, 120, 'Are You Experienced?', 94)
(85, 'Cochise', 10, 1, 1, 'Audioslave/Chris Cornell', 222380,
5339931, 0.99, 10, 'Audioslave', 8)
(2949, 'The Three Sunrises', 234, 1, 1, 'U2', 234788, 7717990,
0.99, 234, 'B-Sides 1980-1990', 150)
(337, 'You Shook Me', 30, 1, 1, 'J B Lenoir/Willie Dixon',
315951, 10249958, 0.99, 30, 'BBC Sessions [Disc 1] [Live]', 22)
```

```
drop_view = '''
```

```
DROP VIEW IF EXISTS AlbumGenre;
'''
c.execute(drop_view)



album_genre_view = '''
CREATE VIEW AlbumGenre AS
SELECT Album.Title, Genre.GenreID
FROM Album
INNER JOIN Track
ON Album.AlbumId=Track.AlbumId
INNER JOIN Genre
ON Track.GenreId = Genre.GenreId
'''

c.execute(album_genre_view)

print('\nPause\n')

genre_choice = "Rock"

id_query = '''
SELECT GenreId FROM Genre
WHERE Genre.Name=?

'''

c.execute(id_query, (genre_choice,))

genre_id, = c.fetchone()

for line in query_response:
    print(line)
print('\nPause\n')

genre_query = '''
SELECT Title FROM AlbumGenre
WHERE GenreId=?
GROUP BY Title
'''

c.execute(genre_query, (genre_id,))
query_response = c.fetchmany(20)

for line in query_response:
    print(line)
```

```
Pause

(3288, 'Rock You Like a Hurricane', 257, 2, 1, None, 255766,
4300973, 0.99, 257, '20th Century Masters - The Millennium
Collection: The Best of Scorpions', 179)
(1201, 'Different World', 94, 2, 1, None, 258692, 4383764, 0.99,
94, 'A Matter of Life and Death', 90)
(2506, 'Nothing To Say', 203, 1, 1, 'Chris Cornell/Kim Thayil',
238027, 7744833, 0.99, 203, 'A-Sides', 132)
(2926, 'Zoo Station', 232, 1, 1, 'U2', 276349, 9056902, 0.99,
232, 'Achtung Baby', 150)
(2938, 'Beautiful Day', 233, 1, 1, 'Adam Clayton, Bono, Larry
Mullen, The Edge', 248163, 8056723, 0.99, 233, "All That You
Can't Leave Behind", 150)
(1146, 'Welcome to the Jungle', 90, 2, 1, None, 273552, 4538451,
0.99, 90, 'Appetite for Destruction', 88)
(1479, 'Foxy Lady', 120, 1, 1, 'Jimi Hendrix', 199340, 6480896,
0.99, 120, 'Are You Experienced?', 94)
```

```
(85, 'Cochise', 10, 1, 1, 'Audioslave/Chris Cornell', 222380,
5339931, 0.99, 10, 'Audioslave', 8)
(2949, 'The Three Sunrises', 234, 1, 1, 'U2', 234788, 7717990,
0.99, 234, 'B-Sides 1980-1990', 150)
(337, 'You Shook Me', 30, 1, 1, 'J B Lenoir/Willie Dixon',
315951, 10249958, 0.99, 30, 'BBC Sessions [Disc 1] [Live]', 22)

Pause

('20th Century Masters - The Millennium Collection: The Best of
Scorpions',)
('A Matter of Life and Death',)
('A-Sides',)
('Achtung Baby',)
("All That You Can't Leave Behind",)
('Appetite for Destruction',)
('Are You Experienced?',)
('Audioslave',)
('B-Sides 1980-1990',)
('BBC Sessions [Disc 1] [Live]',)
('BBC Sessions [Disc 2] [Live]',)
('Balls to the Wall',)
('Bark at the Moon (Remastered)',)
('Beyond Good And Evil',)
('Big Ones',)
('Blizzard of Ozz',)
('Bongo Fury',)
('Brave New World',)
('By The Way',)
('Californication',)
```