Øving 3 - gruppe 140

Oppgave 1 - Sql spørringer

a) SELECT songID, name, duration, year, artistID FROM song

	songID	name	duration	year	artistID
1	1	Saved	178	2015	1
2	2	Oops!... I Did It Again	221	2000	2
3	3	Don't Start Now	183	2019	3
4	4	Strangers	233	2017	4
5	5	I Went Too Far	294	2016	5
6	6	Blasé	286	2015	1
7	7	Hot Girl Summer	199	2019	9
8	8	drivers license	242	2021	13
9	9	Black Beatles	291	2016	8
10	10	Nice for What	204	2018	12
11	11	Work	219	2016	15
12	12	I'm the One	288	2017	16
13	13	WAP	187	2020	11
14	14	Watermelon Sugar	173	2020	21
15	15	positions	172	2020	22
16	16	7 rings	178	2019	22
17	17	thank u, next	217	2018	22
18	18	Don't Judge Me	242	2017	1
19	19	Love U Better	183	2017	1
20	20	Savage Remix	242	2020	9
21	21	Cardigan	232	2020	26
22	22	Butter	265	2021	27
23	23	good 4 u	178	2021	13

b)

```
sqlite> SELECT name, year FROM song WHERE year < 2017 ;
Saved|2015
Oops!... I Did It Again|2000
I Went Too Far|2016
Blasé|2015
Black Beatles|2016
Work|2016
Love Me|2013
```

c) SELECT name, year FROM song WHERE year >= 2018 AND year <= 2020 ORDER BY year ASC

	name	year
1	Nice for What	2018
2	thank u, next	2018
3	Don't Start Now	2019
4	Hot Girl Summer	2019
5	7 rings	2019
6	WAP	2020
7	Watermelon Sugar	2020
8	positions	2020
9	Savage Remix	2020
10	Cardigan	2020

d) SELECT artist.name, song.name FROM (artist NATURAL JOIN featuredOn) inner JOIN song on song.songID = featuredOn.songID ;

	name	name
1	Future	Blasé
2	Rae Sremmurd	Blasé
3	E-40	Saved
4	Ty Dolla Sign	Hot Girl Summer
5	Nicki Minaj	Hot Girl Summer
6	Gucci Mane	Black Beatles
7	Drake	Work
8	Justin Bieber	I'm the One
9	Chance the Rapper	I'm the One
10	Quavo	I'm the One
11	Lil Wayne	I'm the One
12	Megan Thee Stallion	WAP
13	Future	Don't Judge Me
14	Swae Lee	Don't Judge Me
15	Lil Wayne	Love U Better
16	The-Dream	Love U Better
17	Beyoncé	Savage Remix
18	Future	Love Me
19	Drake	Love Me

e) SELECT song.name as songName, album.name as albumName, song.year FROM (((song NATURAL JOIN songOnAlbum) INNER JOIN album ON album.albumID = songOnAlbum.albumID) INNER JOIN artist ON artist.artistID = song.artistID) WHERE artist.name = "Ariana Grande"

	songName	albumName	year
1	positions	Positions	2020
2	7 rings	thank u, next	2019
3	thank u, next	thank u, next	2018

f)SELECT DISTINCT hovedArtist.name, song.name
FROM ((artist AS hovedArtist INNER JOIN song ON hovedArtist.artistID = song.artistID) INNER JOIN featuredOn ON featuredOn.songID = song.songID)

INNER JOIN artist AS feature ON featuredOn.artistID = feature.artistID
WHERE feature.name = 'Ty Dolla Sign' OR hovedArtist.name = 'Ty Dolla Sign' ;

	name	name
1	Ty Dolla Sign	Blasé
2	Ty Dolla Sign	Saved
3	Megan Thee Stallion	Hot Girl Summer
4	Ty Dolla Sign	Don't Judge Me
5	Ty Dolla Sign	Love U Better

g) SELECT artist.name as artistName, song.name as songName FROM song INNER JOIN artist ON song.artistID = artist.artistID WHERE song.name LIKE "%the%"

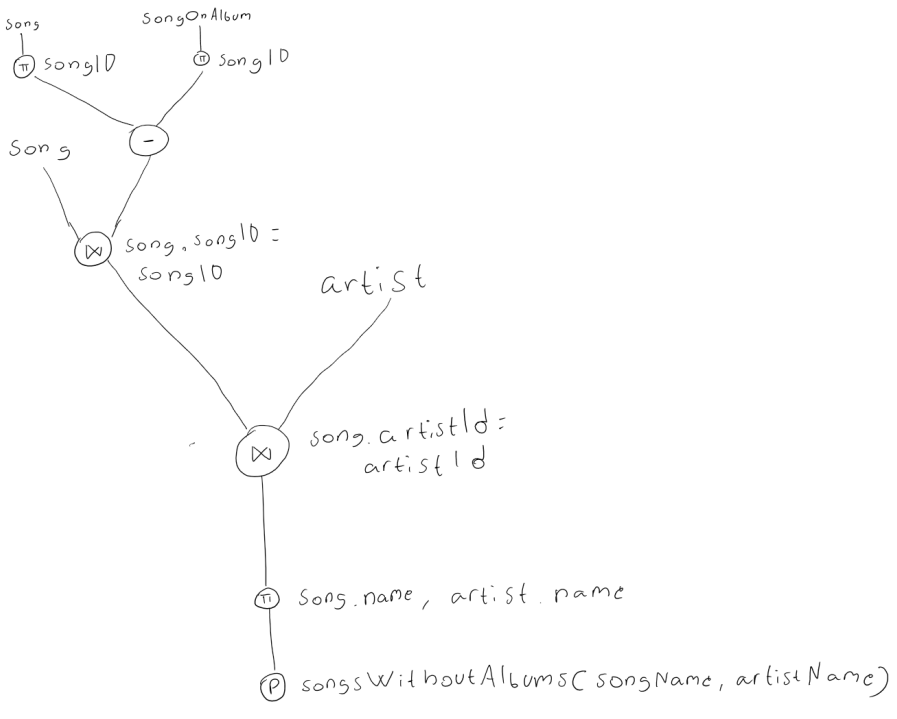
	artistName	songName
1	DJ Khaled	I'm the One

h) SELECT name, MAX(howManyFeatures) as howManyFeatures FROM ((SELECT COUNT(featuredOn.songID) AS howManyFeatures, artist.artistID
FROM artist NATURAL JOIN featuredOn
GROUP BY artist.artistID) NATURAL JOIN artist)

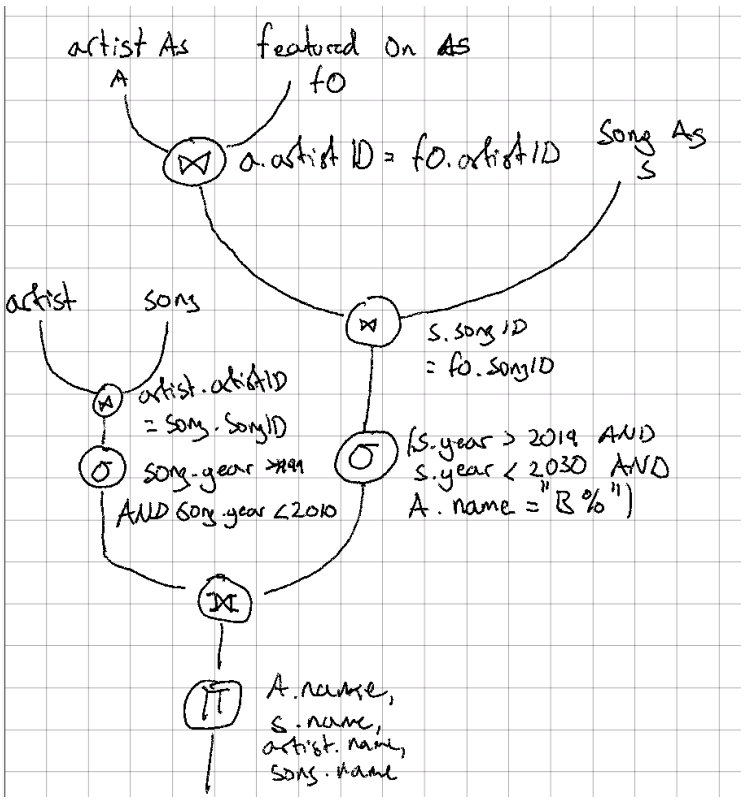
	name	howManyFeatures
1	Future	3

Oppgave 2 - relasjonsalgebra

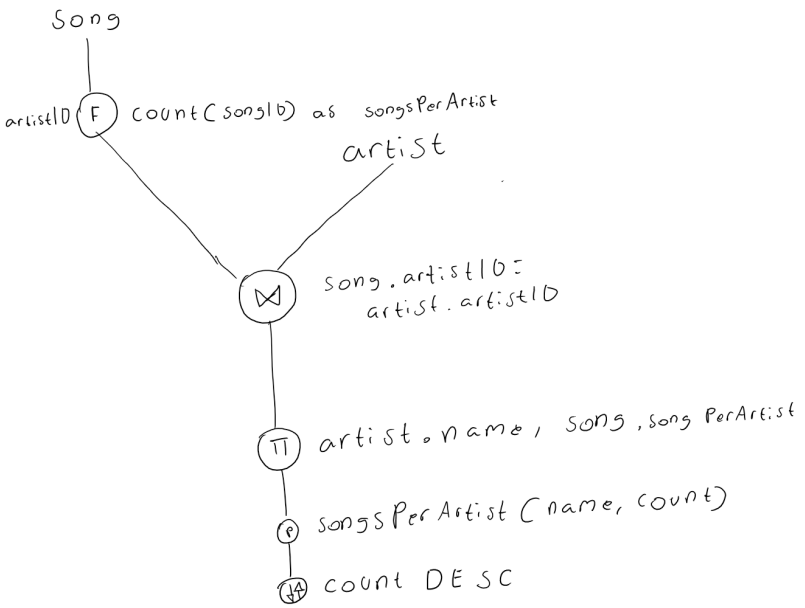
a)



b)



c)



Oppgave 3 - introduksjon til normalisering

a)

Hvis vi må endre faar og navn til Jacques Tatis må vi endre 8 celler

b)

filmID (primær)	name	year	directorID (fremmed)

directorID (primær)	directorName	directorBIrthYear

Oppgave 4 - Funksjonelle avhengigheter, nøkler og tillukning

a)
2 fordi a_3 peker på både b_4 og b_3
5 fordi c_1 peker på både d_1 og d_2
6 fordi d_2 peker på både c_1 og c_2
8 fordi a_1 b_1 c_1 forekommer to ganger
9 fordi d_2 peker på både a_1,b_1,c_1 og a_2,b_2,c_2
10 fordi du kan fjerne A fra ABD og fortsatt ha en supernøkkel

b)

D+ = AD
BC+ = ABCD
AB+ = ABCD
BD+ = ABCD

Det er tre supernøkler.