Oblig 1 Adam Aske

Oppgave 1

1. CREATE DATABASE IF NOT EXISTS Oblig1;

CREATE TABLE IF NOT EXISTS Film(

FNr INT,

Title VARCHAR(20) NOT NULL,

År Int,

Land VARCHAR(50) NOT NULL,

Sjanger VARCHAR(20) NOT NULL,

Alder INT,

Tid INT,

Pris FLOAT,

CONSTRAINT PKFilm PRIMARY KEY(FNr)

)Engine = InnoDB;

1. INSERT INTO Film Values

(1, 'Casablanca', 1942, 'USA','Drama', 15, 102, 149.00),

(2, 'Fort Apechea', 1948, 'USA','Drama', 15, 102, 149.00),

(3, 'Apocolypse Now', 1979, 'USA', 'Action', 18, 155, 123.00),

(4, 'Streets of Fire', 1984, 'USA', 'Action', 15, 93, NULL),

(5, 'High Noon', 1952, 'USA', 'Western', 15, 85, 123.00),

(6, 'Cinema Paradiso', 1988,'Italy', 'Comedy', 11, 123, NULL),

(7, 'Asterix hos britene', 1988, 'France', 'Animation', 7, 78, 149.00),

(8, 'veiviseren', 1987, 'Norway', 'Action', 15, 96, 87.00),

(9, 'Salmer Fra Kjøkkenet', 2002, 'Norway', 'Comedy', 7, 80, 149.00),

(10, 'Anastasia', 1997, 'USA', 'Animation', 7, 94, 123.00),

(11, 'LA Grande bouffe', 1973, 'France', 'Drama', 15, 129, 87.00),

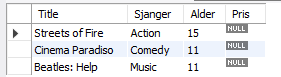
(12, 'Blues Brothers 2000', 1998, 'USA', 'Comedy', 11, 124, 135.00),

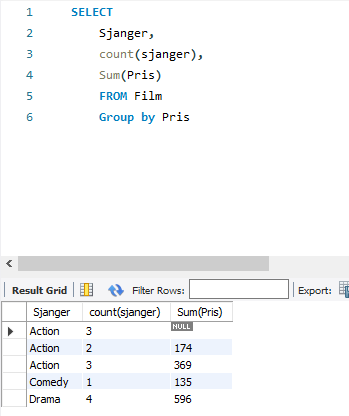
(13, 'Beatles: Help', 1965, 'Great Britain', 'Music', 11, 144, NULL);

Select

Sjanger,

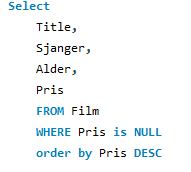
count(Sjanger),

Sum(Pris), 

Alder, 

Pris

FROM Film GROUP BY Sjanger

2. 

Select

Title,

Sjanger,

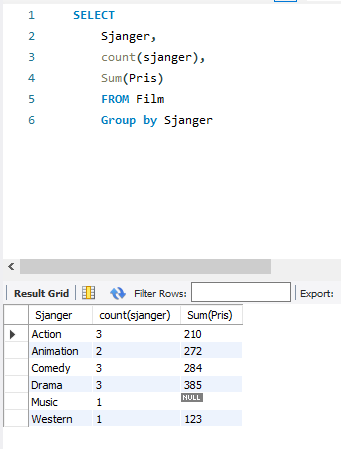
Alder,

Pris

FROM Film

WHERE Pris is NULL

order by Pris DESC



1. UPDATE Film SET Title = 'High Moon' WHERE Title = 'High Noon';
2. UPDATE Film SET Pris = Pris\*1.1 WHERE Sjanger = 'Action';
3. DELETE FROM Film WHERE Title = 'Anastasia';

Oppgave 2

1. CREATE TABLE IF NOT EXISTS Kunde(

Kundenummer INT(20) PRIMARY KEY,

Fornavn VARCHAR(20),

Etternavn VARCHAR(20),

Adresse VARCHAR(50),

Postnummer INT

)Engine = InnoDB;

1. INSERT INTO Kunde VALUES

(1, 'Jonas', 'Ghar Støre', 'Statsministerveien 1', 1312),

(2, 'Erna', 'Solberg', 'Tapergata 13', 4312),

(3, 'Kari', 'Mo', 'Gateveien 79', 1234);

1. CREATE TABLE IF NOT EXISTS Faktura (

FakturaNummer INT,

KundeNummer INT,

Pris FLOAT,

PRIMARY KEY (FakturaNummer),

FOREIGN KEY (KundeNummer)

REFERENCES Kunde (Kundenummer)

) ENGINE=INNODB; Jeg bruker fakturanummer som primær nøkkelen fordi alle fakturaer er unike og må ikke forveksles. Jeg bruker kundenummer som foreign key for å kunne referere til Kunde tabellen og jeg kan da f-eks enkelt finne alle fakturaene til visse kunder.

1. INSERT INTO Faktura VALUES

(1, 3, 100),

(13, 3, 199);

Kari har kundenummer 3.

1. SELECT

F.Fakturanummer, F.Kundenummer, Pris, K.Fornavn, K.Etternavn

FROM

Faktura AS F,

Kunde AS K

WHERE

F.Kundenummer = K.Kundenummer;

1. CREATE TRIGGER SjekkPris BEFORE INSERT ON Faktura FOR EACH ROW

BEGIN

IF NEW.Pris < 0 THEN

SET new.Pris = 0;

END IF;

IF NEW.Pris > 1000 THEN

SET new.Pris = 10000;

End if;

END;//

Oppgave 3)

1. Hvis prisen er under 0, blir den satt til 0. Hvis den er over 10 000 så blir den satt til 10 000.

Returns the substring from string *str* before count occurrences of the delimiter *delim*. If *count* is positive, everything to the left of the final delimiter (counting from the left) is returned. If *count* is negative, everything to the right of the final delimiter (counting from the right) is returned. SUBSTRING\_INDEX() performs a case-sensitive match when searching for *delim*.

If any argument is NULL, returns NULL.

CREATE TABLE IF NOT EXISTS Ansatte(

Navn VARCHAR(50),

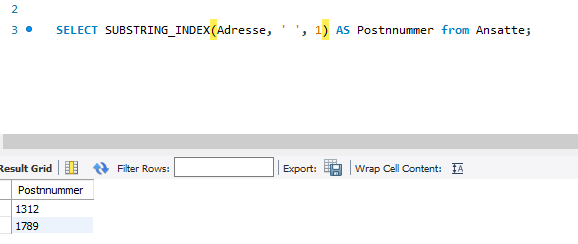
Adresse VARCHAR(50)) Engine = InnoDB;

INSERT INTO Ansatte VALUES

("Elisabeth", "1312 Krokveien"),

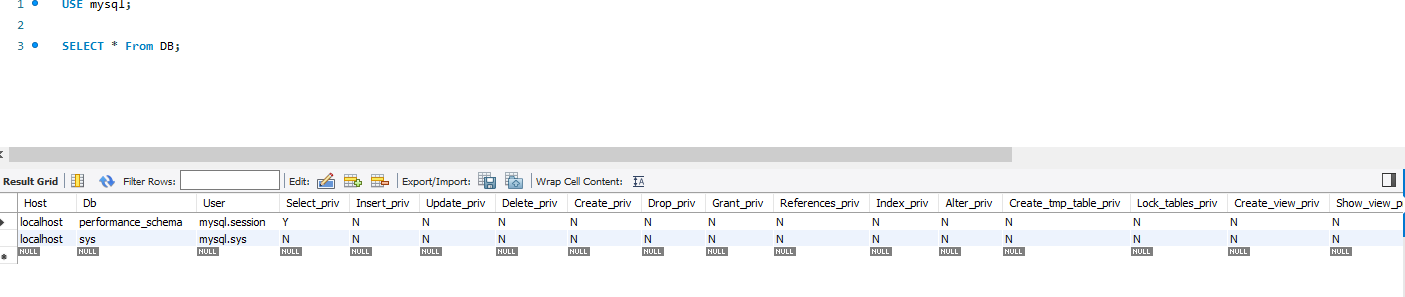
("Andrew", "1789 Dataspill");

SELECT SUBSTRING\_INDEX(Adresse, ' ', 1) AS Postnnummer from Ansatte;

1. 

Her velger den alle postnummerene til de ansatte i bedriften ved å velge alt frem til første mellomrom i Stringen.

Oppgave 4

1. 
2. 