**/\* Naredba za napraviti novu bazu podataka\*/**

CREATE DATABASE multipleks\_kino;

**/\* Naredba za napraviti tablicu \*/**

CREATE TABLE dvorana (

id SERIAL,

ime VARCHAR(256) NOT NULL UNIQUE,

broj\_sjedala INT NOT NULL

);

CREATE TABLE tehnologija (

id SERIAL,

ime VARCHAR(256) NOT NULL UNIQUE

);

CREATE TABLE dvorana\_tehnologija (

id\_dvorana BIGINT NOT NULL,

id\_tehnologija BIGINT NOT NULL

);

CREATE TABLE film (

id SERIAL,

ime VARCHAR(256) NOT NULL UNIQUE,

trajanje INT NOT NULL

);

CREATE TABLE termin (

id SERIAL,

datum TIMESTAMP,

id\_dvorane BIGINT NOT NULL,

id\_filma BIGINT NOT NULL

);

CREATE TABLE ulaznica (

id SERIAL,

cijena INT NOT NULL,

oib\_zaposlenika VARCHAR(11) NOT NULL,

id\_termina BIGINT NOT NULL,

izdano TIMESTAMP NOT NULL

);

CREATE TABLE zaposlenik (

oib VARCHAR(11) NOT NULL,

ime VARCHAR(256) NOT NULL,

prezime VARCHAR(256) NOT NULL,

adresa VARCHAR(256) NOT NULL

);

CREATE TABLE smjena (

id SERIAL,

vrijeme\_pocetka TIME NOT NULL,

vrijeme\_kraja TIME NOT NULL,

dan VARCHAR(256) NOT NULL

);

CREATE TABLE smjena\_zaposlenik (

oib\_zaposlenika VARCHAR(11) NOT NULL,

id\_smjene BIGINT NOT NULL

);

**/\* Naredba za ubacit u table/relacije \*/**

INSERT INTO dvorana (ime, broj\_sjedala) VALUES (’A1’, 40);

INSERT INTO dvorana (ime, broj\_sjedala) VALUES (’A2’, 40);

INSERT INTO dvorana (ime, broj\_sjedala) VALUES (’A3’, 40);

INSERT INTO tehnologija (ime) VALUES (’2D’);

INSERT INTO tehnologija (ime) VALUES (’3D’);

INSERT INTO tehnologija (ime) VALUES (’Surround sound’);

INSERT INTO dvorana\_tehnologija (id\_dvorana, id\_tehnologija) VALUES (1, 1);

INSERT INTO dvorana\_tehnologija (id\_dvorana, id\_tehnologija) VALUES (1, 3);

INSERT INTO dvorana\_tehnologija (id\_dvorana, id\_tehnologija) VALUES (2, 2);

INSERT INTO dvorana\_tehnologija (id\_dvorana, id\_tehnologija) VALUES (2, 3);

INSERT INTO film (ime, trajanje) VALUES (’Rick and Morty’, 60);

INSERT INTO film (ime, trajanje) VALUES (’Parasite’, 120);

INSERT INTO film (ime, trajanje) VALUES (’Joker’, 180);

INSERT INTO film (ime, trajanje) VALUES (’Jurrasic Park’, 180);

INSERT INTO termin (datum, id\_dvorane, id\_filma) VALUES (’1.5.2020 20:00:00.00 GMT+1’, 1, 1);

INSERT INTO zaposlenik (oib, ime, prezime, adresa) VALUES (’0915372038’, ’Kristijan’, ’Knezevic’, ’10000 Split, Split BB’);

INSERT INTO zaposlenik (oib, ime, prezime, adresa) VALUES (’0955402828’, ’Domagoj’, ’Buljan’, ’12345 Sinj, Picadily BB’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’08:00’, ’16:00’, ’PON’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’16:00’, ’24:00’, ’PON’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’08:00’, ’16:00’, ’UTO’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’16:00’, ’24:00’, ’UTO’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’08:00’, ’16:00’, ’SRI’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’16:00’, ’24:00’, ’SRI’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’08:00’, ’16:00’, ’CET’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’16:00’, ’24:00’, ’CET’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’08:00’, ’16:00’, ’PET’);

INSERT INTO smjena (vrijeme\_pocetka, vrijeme\_kraja, dan) VALUES (’16:00’, ’24:00’, ’PET’);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0915372038’, 1);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0915372038’, 3);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0915372038’, 5);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0915372038’, 7);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0915372038’, 9);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0955402828’, 1);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0955402828’, 3);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0955402828’, 5);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0955402828’, 7);

INSERT INTO smjena\_zaposlenik (oib\_zaposlenika, id\_smjene) VALUES (’0955402828’, 9);

INSERT INTO ulaznica (cijena, oib\_zaposlenika, id\_termina, izdano) VALUES (50, ’0915372038’, 4, now());

**/\* Naredba za listat table/relaciju \*/**

SELECT \* FROM dvorana;

SELECT \* FROM tehnologija;

SELECT \* FROM dvorana\_tehnologija;

SELECT \* FROM dvorana WHERE ime=’A1’;

/\* Koje tehnologije ime dvorana A1? \*/

SELECT tehnologija.ime FROM dvorana

JOIN dvorana\_tehnologija ON dvorana.id = dvorana\_tehnologija.id\_dvorana

JOIN tehnologija ON tehnologija.id = dvorana\_tehnologija.id\_tehnologija

WHERE dvorana.ime = ’A1’;

/\* Koje tehnologije ime dvorana A2? \*/

SELECT tehnologija.ime FROM dvorana

JOIN dvorana\_tehnologija ON dvorana.id = dvorana\_tehnologija.id\_dvorana

JOIN tehnologija ON tehnologija.id = dvorana\_tehnologija.id\_tehnologija

WHERE dvorana.ime = ’A2’;

/\*Promjeniti ime dvorane\*/

UPDATE dvorana SET ime=’B1’ WHERE ime=’A1’;

/\* Jako dobra stvar je to sto mijenjanje imena dvorane nece poremetiti relaciju zato jer se ime dvorane ne koristi za relaciju vec id dvorane :-) \*/

Trazene naredbe:

* raspored svih projekcija u zadanom vremenskom periodu.

SELECT termin.datum, dvorana.ime AS ime\_dvorane, film.ime as ime\_filma, film.trajanje

FROM termin

JOIN dvorana ON termin.id\_dvorane = dvorana.id

JOIN film ON termin.id\_filma = film.id

**WHERE datum BETWEEN '2020-01-5' AND '2020-01-7'**;

* raspored svih projekcija određenog filma u zadanom vremenskom periodu.

SELECT termin.datum, dvorana.ime AS ime\_dvorane, film.ime as ime\_filma, film.trajanje

FROM termin

JOIN dvorana ON termin.id\_dvorane = dvorana.id

JOIN film ON termin.id\_filma = film.id

**WHERE datum BETWEEN '2020-01-5' AND '2020-01-7' AND film.ime = 'Joker';**

* izvještaj o popunjenosti kino-dvorana i brojevima slobodnih mjesta u zadanom vremenskom periodu.

**Koliko koji film ima izdanih ulaznica?**

SELECT t.id\_filma, f.ime, COUNT(u)

FROM ulaznica AS u

JOIN termin AS t ON t.id = u.id\_termina

JOIN film AS f ON f.id = t.id\_filma

GROUP BY f.id\_filma, f.ime;

**Koliko izdanih ulaznica ima termin?**

SELECT t.id, COUNT(u.id) FROM termin AS t JOIN ulaznica AS u ON u.id\_termina = t.id GROUP BY t.id;

**Broj slobodnih mjesta za termin?**

SELECT t.id, (d.broj\_sjedala - COUNT(u.id)) AS slobodnih\_mijesta

FROM termin AS t

JOIN ulaznica AS u ON u.id\_termina = t.id

JOIN dvorana AS d ON d.id = t.id\_dvorane

GROUP BY t.id, d.broj\_sjedala;

* raspored svih današnjih projekcija.

SELECT termin.datum, dvorana.ime AS ime\_dvorane, film.ime as ime\_filma, film.trajanje

FROM termin

JOIN dvorana ON termin.id\_dvorane = dvorana.id

JOIN film ON termin.id\_filma = film.id

WHERE DATE(datum) = now()::date;

* raspored rada zaposlenih kino-djelatnika

**Kada odredeni zaposlenik radi?**

SELECT dan, vrijeme\_pocetka, vrijeme\_kraja

FROM zaposlenik

JOIN smjena\_zaposlenik AS sz ON sz.oib\_zaposlenika = zaposlenik.oib

JOIN smjena as s ON sz.id\_smjene = s.id

WHERE zaposlenik.oib = '0915372038';

**Sve smijene i koji zaposlenici rade u njima?**

SELECT s.id, s.vrijeme\_pocetka, s.vrijeme\_kraja, s.dan, z.oib, z.ime, z.prezime

FROM smjena AS s

JOIN smjena\_zaposlenik AS sz ON sz.id\_smjene = s.id

JOIN zaposlenik AS z ON z.oib = sz.oib\_zaposlenika

ORDER BY s.id;

* Broj izdanih ulaznica za svaki film u odredenom intervalu

SELECT

f.id,

COUNT(u.id) AS broj\_izdanih\_ulaznica

FROM film AS f

FULL JOIN (

SELECT

\*

FROM termin

WHERE termin.datum BETWEEN '2020-01-5' AND '2020-01-7'

) AS t ON t.id\_filma = f.id

FULL JOIN ulaznica AS u on u.id\_termina = t.id

WHERE f.id != 0

GROUP BY f.id

* Prihod za svaki film unutar odredenog datuma

SELECT

f.id,

CASE

WHEN SUM(u.cijena) IS null THEN 0 ELSE SUM(u.cijena)

END AS prihod

FROM film AS f

FULL JOIN (

SELECT

\*

FROM termin

WHERE termin.datum BETWEEN '2020-01-5' AND '2020-01-7'

) AS t ON t.id\_filma = f.id

FULL JOIN ulaznica AS u ON u.id\_termina = t.id

WHERE f.id != 0

GROUP BY f.id

* Ukupan broj sjedala svakog termina odredenih filmova

SELECT

f.id,

CASE

WHEN SUM(d.broj\_sjedala) IS null THEN 0 ELSE SUM(d.broj\_sjedala)

END AS ukupan\_broj\_sjedala

FROM film AS f

FULL JOIN (

SELECT

\*

FROM termin

WHERE termin.datum BETWEEN '2020-01-5' AND '2020-01-7'

) AS t ON t.id\_filma = f.id

LEFT JOIN dvorana AS d on d.id = t.id\_dvorane

GROUP BY f.id

* izvještaj o posjećenosti, popunjenosti dvorana i prihodu ostvarenom na pojedinom filmu u zadanom vremenskom periodu. (navedeni query se sastoji od zadnja tri navedena)

SELECT

f.id,

f.ime,

t1.broj\_izdanih\_ulaznica,

t2.ukupan\_broj\_sjedala,

CASE

WHEN t2.ukupan\_broj\_sjedala = 0 THEN 0 ELSE t1.broj\_izdanih\_ulaznica / CAST(t2.ukupan\_broj\_sjedala AS float) \* 100

END AS popunjenost,

t3.prihod

FROM film AS f

JOIN (

SELECT

f.id,

COUNT(u.id) AS broj\_izdanih\_ulaznica

FROM film AS f

FULL JOIN (

SELECT

\*

FROM termin

WHERE termin.datum BETWEEN '2020-01-5' AND '2020-01-7'

) AS t ON t.id\_filma = f.id

FULL JOIN ulaznica AS u on u.id\_termina = t.id

WHERE f.id != 0

GROUP BY f.id

) AS t1 ON t1.id = f.id

JOIN (

SELECT

f.id,

CASE

WHEN SUM(d.broj\_sjedala) IS null THEN 0 ELSE SUM(d.broj\_sjedala)

END AS ukupan\_broj\_sjedala

FROM film AS f

FULL JOIN (

SELECT

\*

FROM termin

WHERE termin.datum BETWEEN '2020-01-5' AND '2020-01-7'

) AS t ON t.id\_filma = f.id

LEFT JOIN dvorana AS d on d.id = t.id\_dvorane

GROUP BY f.id

) AS t2 ON t2.id = f.id

JOIN (

SELECT

f.id,

CASE

WHEN SUM(u.cijena) IS null THEN 0 ELSE SUM(u.cijena)

END AS prihod

FROM film AS f

FULL JOIN (

SELECT

\*

FROM termin

WHERE termin.datum BETWEEN '2020-01-5' AND '2020-01-7'

) AS t ON t.id\_filma = f.id

FULL JOIN ulaznica AS u ON u.id\_termina = t.id

WHERE f.id != 0

GROUP BY f.id

) AS t3 ON t3.id = f.id

* Broj izdanih ulaznica za svaki termin unutar intervala datuma

SELECT

t.id,

COUNT(u.id) AS broj\_izdanih\_ulaznica

FROM termin AS t

FULL JOIN ulaznica AS u ON u.id\_termina = t.id

WHERE t.datum BETWEEN '2020-01-5' AND '2020-01-7'

GROUP BY t.id

* Broj mjesta za svaki termin unutar intervala

SELECT

t.id,

d.broj\_sjedala

FROM termin AS t

LEFT JOIN dvorana AS d ON d.id = t.id\_dvorane

WHERE t.datum BETWEEN '2020-01-5' AND '2020-01-7'

* Ukupan prihod za svaki termin unutar intervala datuma

SELECT

t.id,

CASE

WHEN SUM(u.cijena) IS null THEN 0 ELSE SUM(u.cijena)

END AS prihod

FROM termin AS t

FULL JOIN ulaznica AS u ON u.id\_termina = t.id

WHERE t.datum BETWEEN '2020-01-5' AND '2020-01-7'

GROUP BY t.id

izvještaj o posjećenosti, popunjenosti dvorana i prihodima ostvarenim u pojedinim terminima prikazivanja u zadanom vremenskom periodu. (Sastoji se od tri zadnje navedena querya)

SELECT

t.id,

CASE

WHEN t2.broj\_sjedala = 0 THEN 0 ELSE t1.broj\_izdanih\_ulaznica / CAST(t2.broj\_sjedala AS float) \* 100

END AS popunjenost,

t3.prihod

FROM termin AS t

LEFT JOIN (

SELECT

t.id,

COUNT(u.id) AS broj\_izdanih\_ulaznica

FROM termin AS t

FULL JOIN ulaznica AS u ON u.id\_termina = t.id

GROUP BY t.id

) AS t1 ON t.id = t1.id

LEFT JOIN (

SELECT

t.id,

d.broj\_sjedala

FROM termin AS t

LEFT JOIN dvorana AS d ON d.id = t.id\_dvorane

) AS t2 ON t.id = t2.id

LEFT JOIN (

SELECT

t.id,

CASE

WHEN SUM(u.cijena) IS null THEN 0 ELSE SUM(u.cijena)

END AS prihod

FROM termin AS t

FULL JOIN ulaznica AS u ON u.id\_termina = t.id

GROUP BY t.id

) AS t3 ON t.id = t3.id

WHERE t.datum BETWEEN '2020-01-5' AND '2020-01-7'

* View za danasnje projekcije.

CREATE VIEW danasnji\_termini AS

SELECT termin.datum, dvorana.ime AS ime\_dvorane, film.ime as ime\_filma, film.trajanje

FROM termin

JOIN dvorana ON termin.id\_dvorane = dvorana.id

JOIN film ON termin.id\_filma = film.id

WHERE DATE(datum) = now()::date;