//kraji

CREATE OR REPLACE FUNCTION return\_vsi\_kraji()

RETURNS TABLE (

ime VARCHAR(100),

posta INT

) AS $$

BEGIN

RETURN QUERY

SELECT k.ime, k.postna\_st

FROM kraji AS k;

END

$$ LANGUAGE 'plpgsql';

//id kraja

CREATE OR REPLACE FUNCTION return\_id\_krajev(posta int, k\_ime varchar(100))

RETURNS INTEGER AS  
$$  
DECLARE

k\_id int;

BEGIN

SELECT INTO k\_id id FROM kraji WHERE (postna\_st = posta) AND (ime = k\_ime);

RETURN k\_id;

END

$$ LANGUAGE 'plpgsql';

//insertanje krajev

CREATE OR REPLACE FUNCTION insertanje\_krajev(k\_posta int, k\_ime varchar(100))

RETURNS void

AS $$

BEGIN

INSERT INTO kraji(postna\_st, ime) VALUES (k\_posta, k\_ime);

END

$$ LANGUAGE 'plpgsql';

//updatanje krajev

CREATE OR REPLACE FUNCTION updatanje\_krajev(k\_id int, posta int, k\_ime varchar(100))

RETURNS void AS  
$$

BEGIN

UPDATE kraji SET postna\_st = posta, ime = k\_ime WHERE (id = k\_id);  
END

$$ LANGUAGE 'plpgsql';

//deletanje kraja

CREATE OR REPLACE FUNCTION deletanje\_kraja(k\_id int)

RETURNS TABLE(

l\_ime varchar(100),

k\_ime varchar(100),

rp\_ime varchar(100)

)

AS  
$$

BEGIN

RETURN QUERY

SELECT l.ime,k.ime,rp.naslov FROM lokalni\_studiji AS l INNER JOIN kraji AS k ON l.kraj\_id = k.id INNER JOIN radijske\_postaje AS rp ON l.radijska\_postaja\_id = rp.id WHERE (l.kraj\_id = k\_id);

DELETE FROM lokalni\_studiji WHERE kraj\_id = k\_id;

END

$$ LANGUAGE 'plpgsql';

//deletanje krajev

CREATE OR REPLACE FUNCTION deletanje\_krajev(k\_id int)

RETURNS void AS  
$$

BEGIN

DELETE FROM kraji WHERE (id = k\_id);  
END

$$ LANGUAGE 'plpgsql';

//radijske postaje

CREATE OR REPLACE FUNCTION return\_vse\_postaje()

RETURNS TABLE (

rp\_ime VARCHAR(100),

rp\_frekvenca VARCHAR(100),

rp\_jezik VARCHAR(100),

z\_ime VARCHAR(100),

r\_ime VARCHAR(100)

) AS $$

BEGIN

RETURN QUERY

SELECT rp.naslov , rp.frekvenca, rp.jezik, z.ime, r.ime

FROM radijske\_postaje AS rp INNER JOIN zvrsti AS z ON rp.zvrst\_id = z.id INNER JOIN regije AS r ON rp.regija\_id = r.id;

END

$$ LANGUAGE 'plpgsql';

//id postaje

CREATE OR REPLACE FUNCTION return\_id\_postaje(rp\_ime varchar(100), rp\_frekvenca varchar(100), rp\_jezik varchar(100), zvrst varchar(100), regija varchar(100))

RETURNS INTEGER

AS $$

DECLARE

id\_r int;

BEGIN

SELECT INTO id\_r rp.id FROM radijske\_postaje AS rp INNER JOIN zvrsti AS z ON rp.zvrst\_id = z.id INNER JOIN regije AS r ON rp.regija\_id = r.id WHERE (rp.naslov = rp\_ime) AND (rp.frekvenca = rp\_frekvenca) AND (rp.jezik = rp\_jezik) AND (z.ime = zvrst) AND (r.ime = regija);

RETURN id\_r;  
END  
$$ LANGUAGE 'plpgsql';

// update logi

CREATE OR REPLACE FUNCTION logi\_radijske\_postaje(id\_r int, id\_u int)

RETURNS void AS

$$

BEGIN

INSERT INTO radijske\_postaje\_update\_logi(radijska\_postaja\_id, uporabnik\_id, datum) VALUES (id\_r, id\_u, now()::timestamp(0));  
END

$$ LANGUAGE 'plpgsql';

//deletanje update logov

CREATE OR REPLACE FUNCTION deletanje\_logov(r\_id int)

RETURNS void AS  
$$  
BEGIN

DELETE FROM radijske\_postaje\_update\_logi WHERE radijska\_postaja\_id = r\_id;  
END

$$ LANGUAGE 'plpgsql';

//opis postaje

CREATE OR REPLACE FUNCTION return\_opis\_postaj(r\_ime varchar(100), r\_frekvenca varchar(100), r\_jezik varchar(100), zvrst varchar(100), regija varchar(100))

RETURNS TEXT

AS $$

DECLARE

opis\_r text;

BEGIN

SELECT INTO opis\_r rp.opis FROM radijske\_postaje AS rp INNER JOIN zvrsti AS z ON rp.zvrst\_id = z.id INNER JOIN regije AS r ON rp.regija\_id = r.id WHERE (rp.naslov = r\_ime) AND (rp.frekvenca = r\_frekvenca) AND (rp.jezik = r\_jezik) AND (z.ime = zvrst) AND (r.ime = regija) AND (rp.opis IS NOT NULL);

IF(opis\_r IS NULL) THEN

opis\_r = '';

END IF;

RETURN opis\_r;  
END  
$$ LANGUAGE 'plpgsql';

//spletna stran postaje

CREATE OR REPLACE FUNCTION return\_stran\_postaj(r\_ime varchar(100), r\_frekvenca varchar(100), r\_jezik varchar(100), zvrst varchar(100), regija varchar(100))

RETURNS VARCHAR

AS $$

DECLARE

stran\_r varchar(100);

BEGIN

SELECT INTO stran\_r rp.spletna\_stran FROM radijske\_postaje AS rp INNER JOIN zvrsti AS z ON rp.zvrst\_id = z.id INNER JOIN regije AS r ON rp.regija\_id = r.id WHERE (rp.naslov = r\_ime) AND (rp.frekvenca = r\_frekvenca) AND (rp.jezik = r\_jezik) AND (z.ime = zvrst) AND (r.ime = regija) AND (rp.spletna\_stran IS NOT NULL);

IF(stran\_r IS NULL) THEN  
stran\_r = '';

END IF;

RETURN stran\_r;  
END  
$$ LANGUAGE 'plpgsql';

//yt postaj

CREATE OR REPLACE FUNCTION return\_yt\_postaj(r\_ime varchar(100), r\_frekvenca varchar(100), r\_jezik varchar(100), zvrst varchar(100), regija varchar(100))

RETURNS VARCHAR

AS $$

DECLARE

yt\_r varchar(100);

BEGIN

SELECT INTO yt\_r rp.yt\_link FROM radijske\_postaje AS rp INNER JOIN zvrsti AS z ON rp.zvrst\_id = z.id INNER JOIN regije AS r ON rp.regija\_id = r.id WHERE (rp.naslov = r\_ime) AND (rp.frekvenca = r\_frekvenca) AND (rp.jezik = r\_jezik) AND (z.ime = zvrst) AND (r.ime = regija) AND (rp.yt\_link IS NOT NULL);

IF(yt\_r IS NULL) THEN  
yt\_r = '';

END IF;

RETURN yt\_r;  
END  
$$ LANGUAGE 'plpgsql';

//twitter postaj

CREATE OR REPLACE FUNCTION return\_tw\_postaj(r\_ime varchar(100), r\_frekvenca varchar(100), r\_jezik varchar(100), zvrst varchar(100), regija varchar(100))

RETURNS VARCHAR

AS $$

DECLARE

tw\_r varchar(100);

BEGIN

SELECT INTO tw\_r rp.twitter\_link FROM radijske\_postaje AS rp INNER JOIN zvrsti AS z ON rp.zvrst\_id = z.id INNER JOIN regije AS r ON rp.regija\_id = r.id WHERE (rp.naslov = r\_ime) AND (rp.frekvenca = r\_frekvenca) AND (rp.jezik = r\_jezik) AND (z.ime = zvrst) AND (r.ime = regija) AND (rp.twitter\_link IS NOT NULL);

IF(tw\_r IS NULL) THEN  
tw\_r = '';  
END IF;

RETURN tw\_r;  
END  
$$ LANGUAGE 'plpgsql';

//insertanje postaj

CREATE OR REPLACE FUNCTION insertanje\_postaj(ime\_r varchar(100), r\_opis text, r\_frekvenca varchar(100), r\_jezik varchar(100), stran varchar(100), yt varchar(100), twitter varchar(100), zvrst varchar(100), regija varchar(100))

RETURNS void AS  
$$

BEGIN

IF(r\_opis = 'Opis') AND (stran = 'Spletna stran') AND (yt = 'Youtube') AND (twitter = 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, frekvenca, jezik, zvrst\_id, regija\_id) VALUES (ime\_r, r\_frekvenca, r\_jezik, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));  
END IF;

IF (stran= 'Spletna stran') AND (yt = 'Youtube') AND (twitter = 'Twitter') AND (r\_opis != 'Opis') THEN

INSERT INTO radijske\_postaje (naslov, opis, frekvenca, jezik, zvrst\_id, regija\_id) VALUES (ime\_r, r\_opis, r\_frekvenca, r\_jezik, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis = 'Opis') AND (stran != 'Spletna stran') AND (yt = 'Youtube') AND (twitter = 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, frekvenca, jezik, spletna\_stran, zvrst\_id, regija\_id) VALUES (ime\_r, r\_frekvenca, r\_jezik, stran, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis = 'Opis') AND (stran = 'Spletna stran') AND (yt != 'Youtube') AND (twitter = 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, frekvenca, jezik, yt\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_frekvenca, r\_jezik, yt, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis = 'Opis') AND (stran = 'Spletna stran') AND (yt = 'Youtube') AND (twitter != 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, frekvenca, jezik, twitter\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_frekvenca, r\_jezik, twitter, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis = 'Opis') AND (stran != 'Spletna stran') AND (yt != 'Youtube') AND (twitter = 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, frekvenca, jezik, spletna\_stran, yt\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_frekvenca, r\_jezik, stran, yt, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis = 'Opis') AND (stran = 'Spletna stran') AND (yt != 'Youtube') AND (twitter != 'Twitter') THEN  
INSERT INTO radijske\_postaje (naslov, frekvenca, jezik, yt\_link, twitter\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_frekvenca, r\_jezik, yt, twitter, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis != 'Opis') AND (stran = 'Spletna stran') AND (yt = 'Youtube') AND (twitter != 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, opis, frekvenca, jezik, twitter\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_opis, r\_frekvenca, r\_jezik, twitter,(SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis = 'Opis') AND (stran != 'Spletna stran') AND (yt = 'Youtube') AND (twitter != 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, frekvenca, jezik, spletna\_stran, twitter\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_frekvenca, r\_jezik, stran, twitter, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis != 'Opis') AND (stran = 'Spletna stran') AND (yt != 'Youtube') AND (twitter = 'Twitter') THEN  
INSERT INTO radijske\_postaje (naslov,opis, frekvenca, jezik, yt\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_opis, r\_frekvenca, r\_jezik, yt, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis != 'Opis') AND (stran != 'Spletna stran') AND (yt = 'Youtube') AND (twitter = 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, opis, frekvenca, jezik, spletna\_stran, zvrst\_id, regija\_id) VALUES (ime\_r, r\_opis, r\_frekvenca, r\_jezik, stran, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis = 'Opis') AND (stran != 'Spletna stran') AND (yt != 'Youtube') AND (twitter != 'Twitter') THEN  
INSERT INTO radijske\_postaje (naslov, frekvenca, jezik, spletna\_stran, yt\_link, twitter\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_frekvenca, r\_jezik, stran, yt, twitter, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis != 'Opis') AND (stran = 'Spletna stran') AND (yt != 'Youtube') AND (twitter != 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, opis, frekvenca, jezik, yt\_link, twitter\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_opis, r\_frekvenca, r\_jezik, yt, twitter, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis != 'Opis') AND (stran != 'Spletna stran') AND (yt = 'Youtube') AND (twitter != 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov,opis, frekvenca, jezik, spletna\_stran, twitter\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_opis, r\_frekvenca, r\_jezik, stran, twitter, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

IF (r\_opis != 'Opis') AND (stran != 'Spletna stran') AND (yt != 'Youtube') AND (twitter = 'Twitter') THEN

INSERT INTO radijske\_postaje (naslov, opis, frekvenca, jezik, spletna\_stran, yt\_link, zvrst\_id, regija\_id) VALUES (ime\_r, r\_opis, r\_frekvenca, r\_jezik, stran, twitter, (SELECT id FROM zvrsti WHERE ime = zvrst), (SELECT id FROM regije WHERE ime = regija));

END IF;

END

$$ LANGUAGE 'plpgsql';

//updatanje postaj

CREATE OR REPLACE FUNCTION updatanje\_postaj(r\_id int, r\_opis text, r\_stran varchar(100), r\_yt varchar(100), r\_twitter varchar(100), zvrst varchar(100), regija varchar(100))

RETURNS void AS  
$$  
BEGIN

IF(r\_opis != '') AND (r\_opis!='Opis') THEN  
UPDATE radijske\_postaje SET opis = r\_opis, zvrst\_id = (SELECT id FROM zvrsti WHERE ime = zvrst), regija\_id = (SELECT id FROM regije WHERE ime = regija) WHERE id = r\_id;  
END IF;

IF(r\_stran != '') AND (r\_stran !='Spletna stran') THEN  
UPDATE radijske\_postaje SET spletna\_stran = r\_stran, zvrst\_id = (SELECT id FROM zvrsti WHERE ime = zvrst), regija\_id = (SELECT id FROM regije WHERE ime = regija) WHERE id = r\_id;  
END IF;

IF(r\_yt != '') AND (r\_yt !='Youtube') THEN  
UPDATE radijske\_postaje SET yt\_link = r\_yt, zvrst\_id = (SELECT id FROM zvrsti WHERE ime = zvrst), regija\_id = (SELECT id FROM regije WHERE ime = regija) WHERE id = r\_id;  
END IF;

IF(r\_twitter != '') AND (r\_twitter !='Twitter') THEN  
UPDATE radijske\_postaje SET twitter\_link = r\_twitter, zvrst\_id = (SELECT id FROM zvrsti WHERE ime = zvrst), regija\_id = (SELECT id FROM regije WHERE ime = regija) WHERE id = r\_id;  
END IF;

END

$$ LANGUAGE 'plpgsql';

//deletanje postaj

CREATE OR REPLACE FUNCTION deletanje\_postaj(r\_id int)

RETURNS void AS  
$$  
BEGIN

DELETE FROM radijske\_postaje WHERE id = r\_id;  
END  
$$ LANGUAGE 'plpgsql';

//zvrsti

CREATE OR REPLACE FUNCTION return\_vse\_zvrsti()

RETURNS TABLE (

ime VARCHAR(100)

) AS $$

BEGIN

RETURN QUERY

SELECT z.ime

FROM zvrsti AS z;

END

$$ LANGUAGE 'plpgsql';

//id zvrsti

CREATE OR REPLACE FUNCTION return\_id\_zvrsti(z\_ime varchar(100))

RETURNS INTEGER AS  
$$  
DECLARE

z\_id int;

BEGIN

SELECT INTO z\_id id FROM zvrsti WHERE (ime = z\_ime);

RETURN z\_id;

END

$$ LANGUAGE 'plpgsql';

//insertanje zvrsti

CREATE OR REPLACE FUNCTION insertanje\_zvrsti(ime\_z varchar(100))

RETURNS void

AS $$

BEGIN

INSERT INTO zvrsti(ime) VALUES (ime\_z);

END

$$ LANGUAGE 'plpgsql';

//updatanje zvrsti

CREATE OR REPLACE FUNCTION updatanje\_zvrsti(id\_z int, ime\_z varchar(100))

RETURNS void

AS $$

BEGIN

UPDATE zvrsti SET ime = ime\_z WHERE id = id\_z;

END

$$ LANGUAGE 'plpgsql';

//deletanje vrste

CREATE OR REPLACE FUNCTION deletanje\_vrste(id\_z int)

RETURNS TABLE(

rp\_ime VARCHAR(100),

rp\_frekvenca VARCHAR(100),

rp\_jezik VARCHAR(100),

z\_ime VARCHAR(100),

r\_ime VARCHAR(100)

) AS $$

BEGIN

RETURN QUERY

SELECT rp.naslov, rp.frekvenca, rp.jezik, z.ime, r.ime FROM radijske\_postaje AS rp INNER JOIN zvrsti AS z ON rp.zvrst\_id = z.id INNER JOIN regije AS r ON rp.regija\_id = r.id WHERE (rp.zvrst\_id = id\_z);

DELETE FROM radijske\_postaje WHERE zvrst\_id = id\_z;

END

$$ LANGUAGE 'plpgsql';

//deletanje zvrsti

CREATE OR REPLACE FUNCTION deletanje\_zvrsti(id\_z int)

RETURNS void

AS $$

BEGIN

DELETE FROM zvrsti WHERE id = id\_z;

END

$$ LANGUAGE 'plpgsql';

//regije

CREATE OR REPLACE FUNCTION return\_vse\_regije()

RETURNS TABLE (

ime VARCHAR(100)

) AS $$

BEGIN

RETURN QUERY

SELECT r.ime

FROM regije AS r;

END

$$ LANGUAGE 'plpgsql';

//id regije

CREATE OR REPLACE FUNCTION return\_id\_regij(r\_ime varchar(100))

RETURNS INTEGER AS  
$$  
DECLARE

r\_id int;

BEGIN

SELECT INTO r\_id id FROM regije WHERE (ime = r\_ime);

RETURN r\_id;

END

$$ LANGUAGE 'plpgsql';

//insertanje regije

CREATE OR REPLACE FUNCTION insertanje\_regije(ime\_r varchar(100))

RETURNS void

AS $$

BEGIN

INSERT INTO regije(ime) VALUES (ime\_r);

END

$$ LANGUAGE 'plpgsql';

//updatanje regije

CREATE OR REPLACE FUNCTION updatanje\_regije(id\_r int, ime\_r varchar(100))

RETURNS void

AS $$

BEGIN

UPDATE regije SET ime = ime\_r WHERE id = id\_r;

END

$$ LANGUAGE 'plpgsql';

//deletanje regij

CREATE OR REPLACE FUNCTION deletanje\_regij(id\_r int)

RETURNS TABLE(

rp\_ime VARCHAR(100),

rp\_frekvenca VARCHAR(100),

rp\_jezik VARCHAR(100),

z\_ime VARCHAR(100),

r\_ime VARCHAR(100)

)

AS  
$$

BEGIN

RETURN QUERY

SELECT rp.naslov, rp.frekvenca, rp.jezik, z.ime, r.ime FROM radijske\_postaje AS rp INNER JOIN zvrsti AS z ON rp.zvrst\_id = z.id INNER JOIN regije AS r ON rp.regija\_id = r.id WHERE (rp.regija\_id = id\_r);

DELETE FROM radijske\_postaje WHERE regija\_id = id\_r;

END

$$ LANGUAGE 'plpgsql';

//deletanje regije

CREATE OR REPLACE FUNCTION deletanje\_regije(id\_r int)

RETURNS void

AS $$

BEGIN

DELETE FROM regije WHERE id = id\_r;

END

$$ LANGUAGE 'plpgsql';

//id studija

CREATE OR REPLACE FUNCTION return\_id\_studija(l\_ime varchar(100), kraj varchar(100), postaja varchar(100))

RETURNS INTEGER

AS $$

DECLARE

id\_l int;

BEGIN

SELECT INTO id\_l l.id FROM lokalni\_studiji AS l INNER JOIN kraji AS k ON l.kraj\_id = k.id INNER JOIN radijske\_postaje AS rp ON l.radijska\_postaja\_id = rp.id WHERE (l.ime = l\_ime) AND (l.kraj\_id = (SELECT id FROM kraji WHERE (ime = kraj))) AND (l.radijska\_postaja\_id = (SELECT id FROM radijske\_postaje WHERE (naslov = postaja)));

RETURN id\_l;  
END  
$$ LANGUAGE 'plpgsql';

//return opis studijev

CREATE OR REPLACE FUNCTION return\_opis\_studijev(l\_ime VARCHAR(100), k\_ime VARCHAR(100), rp\_ime VARCHAR(100))

RETURNS TEXT  
AS $$

DECLARE

opis\_l text;

BEGIN

SELECT INTO opis\_l l.opis FROM lokalni\_studiji AS l INNER JOIN kraji AS k ON l.kraj\_id = k.id INNER JOIN radijske\_postaje AS rp ON l.radijska\_postaja\_id = rp.id WHERE (l.ime = l\_ime) AND (l.kraj\_id = (SELECT id FROM kraji WHERE (ime = k\_ime))) AND (l.radijska\_postaja\_id = (SELECT id FROM radijske\_postaje WHERE (naslov = rp\_ime))) AND (l.opis IS NOT NULL);

IF(opis\_l IS NULL) THEN

opis\_l = '';

END IF;

RETURN opis\_l;

END

$$ LANGUAGE 'plpgsql';

//list lokalnih studijev

CREATE OR REPLACE FUNCTION list\_studijev()

RETURNS TABLE (

l\_ime VARCHAR(100),

k\_ime VARCHAR(100),

rp\_ime VARCHAR(100)

)

AS $$

BEGIN

RETURN QUERY

SELECT l.ime, k.ime, rp.naslov FROM lokalni\_studiji AS l INNER JOIN kraji AS k ON l.kraj\_id = k.id INNER JOIN radijske\_postaje AS rp ON l.radijska\_postaja\_id = rp.id;

END

$$ LANGUAGE 'plpgsql';

//insertanje studija

CREATE OR REPLACE FUNCTION insertanje\_studijev(l\_ime varchar(100), l\_opis text, kraj varchar(100), postaja varchar(100))

RETURNS void AS  
$$

BEGIN

IF(l\_opis = '') OR (l\_opis = 'Opis') THEN

INSERT INTO lokalni\_studiji(ime, kraj\_id, radijska\_postaja\_id) VALUES (l\_ime, (SELECT id FROM kraji WHERE ime = kraj), (SELECT id FROM radijske\_postaje WHERE naslov = postaja));

END IF;

IF(l\_opis != '') AND (l\_opis != 'Opis') THEN

INSERT INTO lokalni\_studiji(ime, opis, kraj\_id, radijska\_postaja\_id) VALUES (l\_ime, l\_opis, (SELECT id FROM kraji WHERE ime = kraj), (SELECT id FROM radijske\_postaje WHERE naslov = postaja));

END IF;

END

$$ LANGUAGE 'plpgsql';

//updatanje studijev

CREATE OR REPLACE FUNCTION updatanje\_studijev(l\_id int, l\_opis text, kraj varchar(100), postaja varchar(100))

RETURNS void AS  
$$

BEGIN

IF(l\_opis = '') OR (l\_opis = 'Opis') THEN

UPDATE lokalni\_studiji SET kraj\_id = (SELECT id FROM kraji WHERE (ime = kraj)), radijska\_postaja\_id = (SELECT id FROM radijske\_postaje WHERE (naslov = postaja)) WHERE id = l\_id;

END IF;

IF(l\_opis != '') AND (l\_opis != 'Opis') THEN

UPDATE lokalni\_studiji SET opis= l\_opis, kraj\_id = (SELECT id FROM kraji WHERE (ime = kraj)), radijska\_postaja\_id = (SELECT id FROM radijske\_postaje WHERE (naslov = postaja)) WHERE id = l\_id;

END IF;

END

$$ LANGUAGE 'plpgsql';

//deletanje studijev

CREATE OR REPLACE FUNCTION deletanje\_studijev(l\_id int)

RETURNS TABLE(

l\_ime varchar(100),

k\_ime varchar(100),

rp\_ime varchar(100)

)

AS  
$$

BEGIN

RETURN QUERY

SELECT l.ime,k.ime,rp.naslov FROM lokalni\_studiji AS l INNER JOIN kraji AS k ON l.kraj\_id = k.id INNER JOIN radijske\_postaje AS rp ON l.radijska\_postaja\_id = rp.id WHERE (l.radijska\_postaja\_id = l\_id);

DELETE FROM lokalni\_studiji WHERE radijska\_postaja\_id = l\_id;

END

$$ LANGUAGE 'plpgsql';

//deletanje studija

CREATE OR REPLACE FUNCTION deletanje\_studija(l\_id int)

RETURNS void

AS  
$$

BEGIN

DELETE FROM lokalni\_studiji WHERE id = l\_id;

END

$$ LANGUAGE 'plpgsql';

//barva ozadja

CREATE OR REPLACE FUNCTION return\_vse\_barve\_ozadja()

RETURNS TABLE(

bz varchar(100)

) AS $$

BEGIN

RETURN QUERY

SELECT ozadje FROM nastavitve;  
END

$$ LANGUAGE 'plpgsql';

//font style

CREATE OR REPLACE FUNCTION return\_vse\_fonte()

RETURNS TABLE(

style varchar(100)

) AS $$

BEGIN

RETURN QUERY

SELECT font FROM nastavitve;  
END

$$ LANGUAGE 'plpgsql';

//registracija

CREATE OR REPLACE FUNCTION registracija (u\_ime varchar(100), u\_priimek varchar(100), u\_spol char, u\_starost int, mail varchar(100), pass varchar(100), u\_kraj varchar(50))

RETURNS VARCHAR

AS $$

DECLARE

u\_mail varchar(100);

BEGIN

SELECT INTO u\_mail u.email FROM uporabniki AS u WHERE (u.email = mail);

IF(u\_mail IS NULL) THEN

INSERT INTO uporabniki(ime, priimek, spol, starost, email, geslo, kraj\_id)

VALUES(u\_ime, u\_priimek, u\_spol, u\_starost, mail, pass, (SELECT id FROM kraji WHERE ime = u\_kraj));

u\_mail = 'sf';

END IF;

RETURN u\_mail;

END

$$ LANGUAGE 'plpgsql';

//login

CREATE OR REPLACE FUNCTION login(mail varchar(100), pass varchar(100))

RETURNS TABLE (

u\_email varchar(100),

geslo varchar(100)

)

AS $$

BEGIN

RETURN QUERY

SELECT u.email, u.geslo

FROM uporabniki AS u WHERE (u.email = mail) AND (u.geslo = pass);

END

$$ LANGUAGE 'plpgsql';

//resetiranje gesla

CREATE OR REPLACE FUNCTION new\_pass(mail varchar(100), novo\_geslo varchar(200))

RETURNS void AS

$$

DECLARE

BEGIN

UPDATE uporabniki SET geslo=novo\_geslo WHERE email = mail;  
END  
$$ LANGUAGE 'plpgsql';

//vrne ime in priimek od maila za prijavo da se ve kdo je prijavljen

CREATE OR REPLACE FUNCTION vrni\_ime\_priimek(u\_mail varchar(100))

RETURNS TABLE(

ime varchar(100),

priimek varchar(100)

)

AS $$

BEGIN

RETURN QUERY

SELECT u.ime, u.priimek

FROM uporabniki AS u WHERE (u.email = u\_mail);

END

$$ LANGUAGE 'plpgsql';

//vrne id uporabnika

CREATE OR REPLACE FUNCTION IDuporabnika(u\_mail varchar(100))

RETURNS INT

AS $$

DECLARE

u\_id int;

BEGIN

SELECT INTO u\_id id FROM uporabniki WHERE (email = u\_mail);

RETURN u\_id;

END

$$ LANGUAGE 'plpgsql';