**drop** **database** **if** **exists** vk;

**create** **database** vk;

**use** vk;

**drop** **table** **if** **exists** users;

**create** **table** users(

id serial **primary** **key**,

firstname **varchar**(100),

lastname **varchar**(100),

email **varchar**(100) **unique**,

`password\_hash` **varchar**(100),

phone **varchar**(12),

**index** users\_phone\_idx(phone),

**index**(firstname, lastname)

);

**drop** **table** **if** **exists** profiles;

**create** **table** profiles(

user\_id serial **primary** **key**,

gender **char**(1),

birthday **date**,

photo\_id **bigint** **unsigned** **null**,

hometown **varchar**(100),

created\_at **datetime** **default** **now**()

);

**alter** **table** profiles

**add** **constraint** fk\_user\_id

**foreign** **key**(user\_id) **references** users(id)

**on** **delete** **restrict**

**on** **update** **cascade**;

**drop** **table** **if** **exists** messages;

**create** **table** messages(

id serial **primary** **key**,

from\_user\_id **bigint** **unsigned** **not** **null**,

to\_user\_id **bigint** **unsigned** **not** **null**,

body **text**,

created\_at **datetime** **default** **now**(),

**foreign** **key**(from\_user\_id) **references** users(id),

**foreign** **key**(to\_user\_id) **references** users(id),

**index**(from\_user\_id),

**index**(to\_user\_id)

);

**drop** **table** **if** **exists** friend\_requests;

**create** **table** friend\_requests(

initiator\_user\_id **bigint** **unsigned** **not** **null**,

target\_user\_id **bigint** **unsigned** **not** **null**,

`status` **ENUM**('requested', 'approved', 'unfriended', 'declined'),

created\_at **datetime** **default** **now**(),

updated\_at **datetime**,

**primary** **key**(initiator\_user\_id, target\_user\_id),

**index**(initiator\_user\_id),

**index**(target\_user\_id),

**foreign** **key**(initiator\_user\_id) **references** users(id),

**foreign** **key**(target\_user\_id) **references** users(id)

);

**drop** **table** **if** **exists** communities;

**create** **table** communities(

id serial **primary** **key**,

name **varchar**(200),

**index**(name)

);

**drop** **table** **if** **exists** users\_communitites;

**create** **table** users\_communities(

user\_id **bigint** **unsigned** **not** **null**,

community\_id **bigint** **unsigned** **not** **null**,

**primary** **key**(user\_id, community\_id),

**foreign** **key**(user\_id) **references** users(id),

**foreign** **key**(community\_id) **references** communities(id)

);

**drop** **table** **if** **exists** media\_types;

**create** **table** media\_types(

id serial **primary** **key**,

name **varchar**(200)

);

**drop** **table** **if** **exists** media;

**create** **table** media(

id serial **primary** **key**,

media\_type\_id **bigint** **unsigned** **not** **null**,

user\_id **bigint** **unsigned** **not** **null**,

body **text**,

filename **varchar**(250),

`size` **int**,

metadata JSON,

created\_at **datetime** **default** **now**(),

updated\_at **datetime**,

**foreign** **key**(user\_id) **references** users(id),

**foreign** **key**(media\_type\_id) **references** media\_types(id),

**index**(user\_id)

);

**drop** **table** **if** **exists** photo\_albums;

**create** **table** photo\_albums(

id serial,

name **varchar**(200),

user\_id **bigint** **unsigned** **not** **null**,

**foreign** **key**(user\_id) **references** users(id),

**primary** **key**(id)

);

**drop** **table** **if** **exists** photos;

**create** **table** photos(

id serial **primary** **key**,

album\_id **bigint** **unsigned** **null**,

media\_id **bigint** **unsigned** **null**,

**foreign** **key**(album\_id) **references** photo\_albums(id),

**foreign** **key**(media\_id) **references** media(id)

);

**drop** **table** **if** **exists** likes;

**create** **table** likes(

id serial **primary** **key**,

media\_id **bigint** **unsigned** **null**,

user\_id **bigint** **unsigned** **not** **null**,

**foreign** **key**(user\_id) **references** users(id),

**foreign** **key**(media\_id) **references** media(id)

);