```
guild(estate_worth > 500)[name] --select names of guilds that have more than 500 golds

(
        character(id = 1)
        *
        payment_details
)[email] --retrieve email address for user which own character with id 1


character<name -> characterName>[character-Name = name]guild
--retrieves all characters and guilds that have same names (test if there is some guild that have same name as character)
--we have to rename character.name because there would be naming conflict with guild.name

guild<id -> firstGuildId>[firstGuildId]
X
guild<id -> secondGuildId>[secondGuildId]
--create all posibilities for guilds to be enemies (select only tuple of ids) -> this way we can randomly create enemies for guilds

(
        user<id -> user_id>[user_id]
        *L
        character
)[user_id name]
--select all usernames from users with their character names, this will produce table with user_id and name where name can be null
--since we use outer joins - we require an extended relational model with null values

character[id]
/
character_in_fight[character_id] --selects all character ids that were never in the fight


weapon
÷
character<id -> character_id>[character_id]
--every name of the weapon that is owned by every character is selected
```