7. В подключенном MySQL репозитории создать базу данных “Друзья

человека”

CREATE DATABASE Human\_friends;

USE Human\_friends;

8. Создать таблицы с иерархией из диаграммы в БД

CREATE TABLE animal\_classes

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Class\_name VARCHAR(20)

);

INSERT INTO animal\_classes (Class\_name)

VALUES ('pack\_animals'),

('home\_animals');

CREATE TABLE pack\_animals

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Nickname VARCHAR (20),

Class\_id INT,

FOREIGN KEY (Class\_id) REFERENCES animal\_classes (Id) ON DELETE CASCADE ON UPDATE CASCADE

);

INSERT INTO pack\_animals (Nickname, Class\_id)

VALUES ('horses', 1),

('donkeys', 1),

('camels', 1);

CREATE TABLE home\_animals

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Nickname VARCHAR (20),

Class\_id INT,

FOREIGN KEY (Class\_id) REFERENCES animal\_classes (Id) ON DELETE CASCADE ON UPDATE CASCADE

);

INSERT INTO home\_animals (Nickname, Class\_id)

VALUES ('cats', 1),

('dogs', 1),

('hamsters', 1);

9. Заполнить низкоуровневые таблицы именами(животных), командами

которые они выполняют и датами рождения

CREATE TABLE cats

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Name VARCHAR(20),

Birthday DATE,

Commands VARCHAR(50),

Genus\_id int,

Foreign KEY (Genus\_id) REFERENCES home\_animals (Id) ON DELETE CASCADE ON UPDATE CASCADE

);

INSERT INTO cats (Name, Birthday, Commands, Genus\_id)

VALUES ('Tom', '2021-11-27', 'Voice', 1),

('Shrek', '2020-01-22', "Sit", 1),

('Mila', '2018-08-11', "Lie", 1);

CREATE TABLE dogs

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Name VARCHAR(20),

Birthday DATE,

Commands VARCHAR(50),

Genus\_id int,

Foreign KEY (Genus\_id) REFERENCES home\_animals (Id) ON DELETE CASCADE ON UPDATE CASCADE

);

INSERT INTO dogs (Name, Birthday, Commands, Genus\_id)

VALUES ('Reks', '2021-12-21', 'Voice', 2),

('Houp', '2020-09-21', "Lie", 2),

('Morok', '2018-09-21', "Lie", 2);

CREATE TABLE hamsters

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Name VARCHAR(20),

Birthday DATE,

Commands VARCHAR(50),

Genus\_id int,

Foreign KEY (Genus\_id) REFERENCES home\_animals (Id) ON DELETE CASCADE ON UPDATE CASCADE

);

INSERT INTO hamsters (Name, Birthday, Commands, Genus\_id)

VALUES ('Oliver', '2024-01-21', 'Voice', 3),

('Honi', '2023-08-12', 'Sit', 3),

('Kit', '2023-12-14', 'Lie', 3);

CREATE TABLE horses

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Name VARCHAR(20),

Birthday DATE,

Commands VARCHAR(50),

Genus\_id int,

Foreign KEY (Genus\_id) REFERENCES packed\_animals (Id) ON DELETE CASCADE ON UPDATE CASCADE

);

INSERT INTO horses (Name, Birthday, Commands, Genus\_id)

VALUES ('Fang', '2007-01-22', 'Run', 4),

('Spirit', '2021-03-21', "Voice", 4),

('Wind', '2011-01-23', "Jump", 4);

CREATE TABLE donkeys

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Name VARCHAR(20),

Birthday DATE,

Commands VARCHAR(50),

Genus\_id int,

Foreign KEY (Genus\_id) REFERENCES packed\_animals (Id) ON DELETE CASCADE ON UPDATE CASCADE

);

INSERT INTO donkeys (Name, Birthday, Commands, Genus\_id)

VALUES ('Lazy', '2013-01-12', 'Lie', 5),

('Split', '2022-06-11', 'Voice', 5),

('Conan', '2023-11-27', 'Jump', 5);

CREATE TABLE camels

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Name VARCHAR(20),

Birthday DATE,

Commands VARCHAR(50),

Genus\_id int,

Foreign KEY (Genus\_id) REFERENCES packed\_animals (Id) ON DELETE CASCADE ON UPDATE CASCADE

);

INSERT INTO camels (Name, Birthday, Commands, Genus\_id)

VALUES ('Sun', '2023-11-22', 'Lie', 6),

('Nick', '2012-02-21', 'Voice', 6),

('Frog', '2022-12-17', 'Jump', 6);

10. Удалив из таблицы верблюдов, т.к. верблюдов решили перевезти в другой

питомник на зимовку. Объединить таблицы лошади, и ослы в одну таблицу

SET SQL\_SAFE\_UPDATES = 0;

DELETE FROM camels;

SELECT Name, Birthday, Commands FROM horses

UNION SELECT Name, Birthday, Commands FROM donkeys;

11.Создать новую таблицу “молодые животные” в которую попадут все

животные старше 1 года, но младше 3 лет и в отдельном столбце с точностью

до месяца подсчитать возраст животных в новой таблице

CREATE TEMPORARY TABLE animals AS

SELECT \*, 'horses' as genus FROM horses

UNION SELECT \*, 'donkeys' AS genus FROM donkeys

UNION SELECT \*, 'dogs' AS genus FROM dogs

UNION SELECT \*, 'cats' AS genus FROM cats

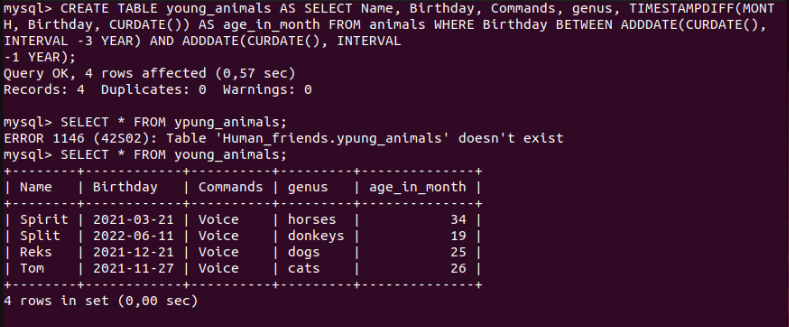
UNION SELECT \*, 'hamsters' AS genus FROM hamsters;

CREATE TABLE young\_animal AS

SELECT Name, Birthday, Commands, genus, TIMESTAMPDIFF(MONTH, Birthday, CURDATE()) AS Age\_in\_month

FROM animals WHERE Birthday BETWEEN ADDDATE(curdate(), INTERVAL -3 YEAR) AND ADDDATE(CURDATE(), INTERVAL -1 YEAR);

SELECT \* FROM young\_animal;



12. Объединить все таблицы в одну, при этом сохраняя поля, указывающие на

прошлую принадлежность к старым таблицам.

SELECT h.Name, h.Birthday, h.Commands, pa.Nickname, ya.age\_in\_month

FROM horses h

LEFT JOIN young\_animals ya ON ya.Name = h.Name

LEFT JOIN packed\_animals pa ON pa.Id = h.genus\_id

UNION

SELECT d.Name, d.Birthday, d.Commands, pa.Nickname, ya.age\_in\_month

FROM donkeys d

LEFT JOIN young\_animals ya ON ya.Name = d.Name

LEFT JOIN packed\_animals pa ON pa.Id = d.genus\_id

UNION

SELECT c.Name, c.Birthday, c.Commands, ha.Nickname, ya.age\_in\_month

FROM cats c

LEFT JOIN young\_animals ya ON ya.Name = c.Name

LEFT JOIN home\_animals ha ON ha.Id = c.genus\_id

UNION

SELECT d.Name, d.Birthday, d.Commands, ha.Nickname, ya.age\_in\_month

FROM dogs d

LEFT JOIN young\_animals ya ON ya.Name = d.Name

LEFT JOIN home\_animals ha ON ha.Id = d.genus\_id

UNION

SELECT hm.Name, hm.Birthday, hm.Commands, ha.Nickname, ya.age\_in\_month

FROM hamsters hm

LEFT JOIN young\_animals ya ON ya.Name = hm.Name

LEFT JOIN home\_animals ha ON ha.Id = hm.genus\_id;

