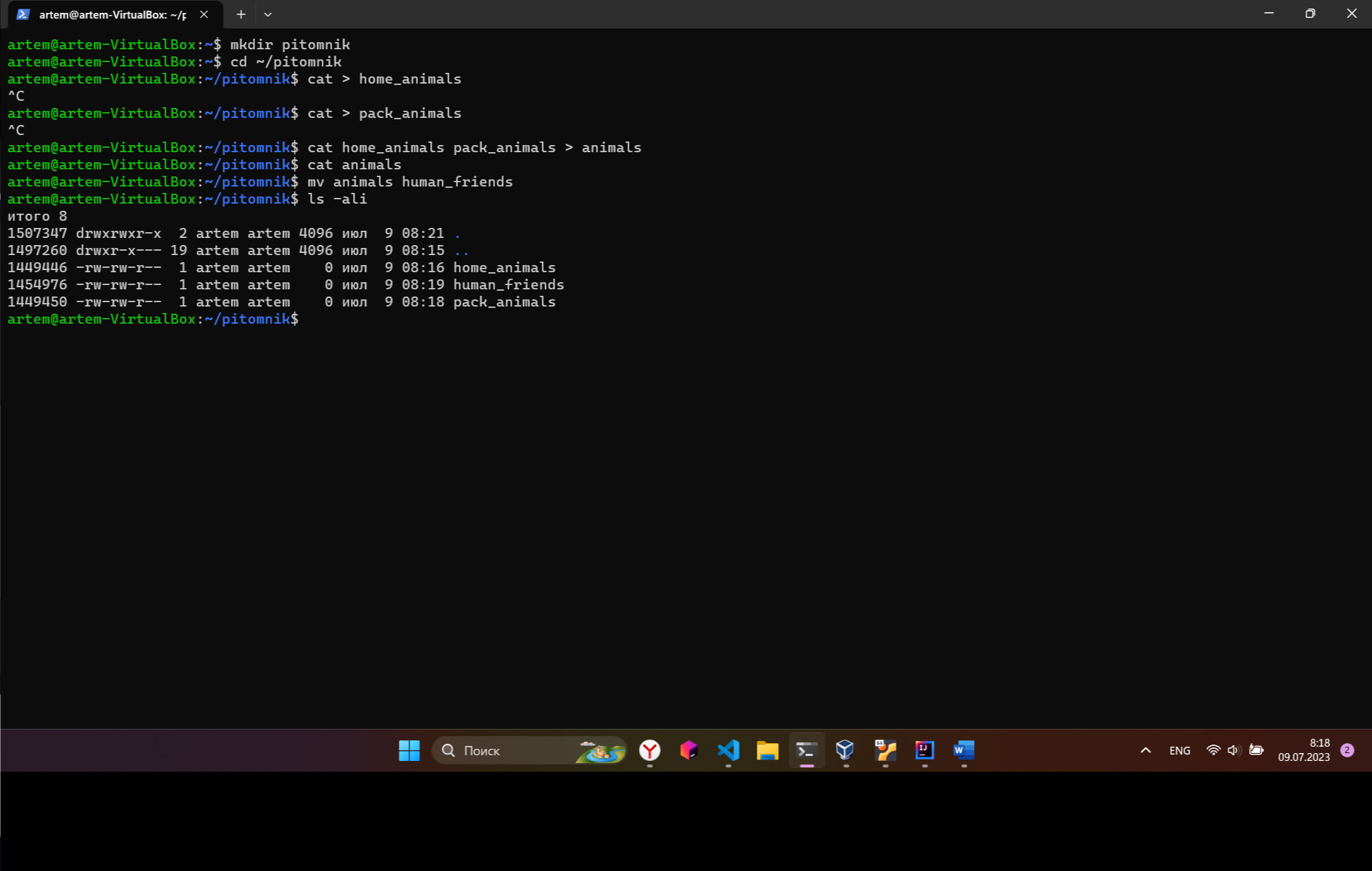
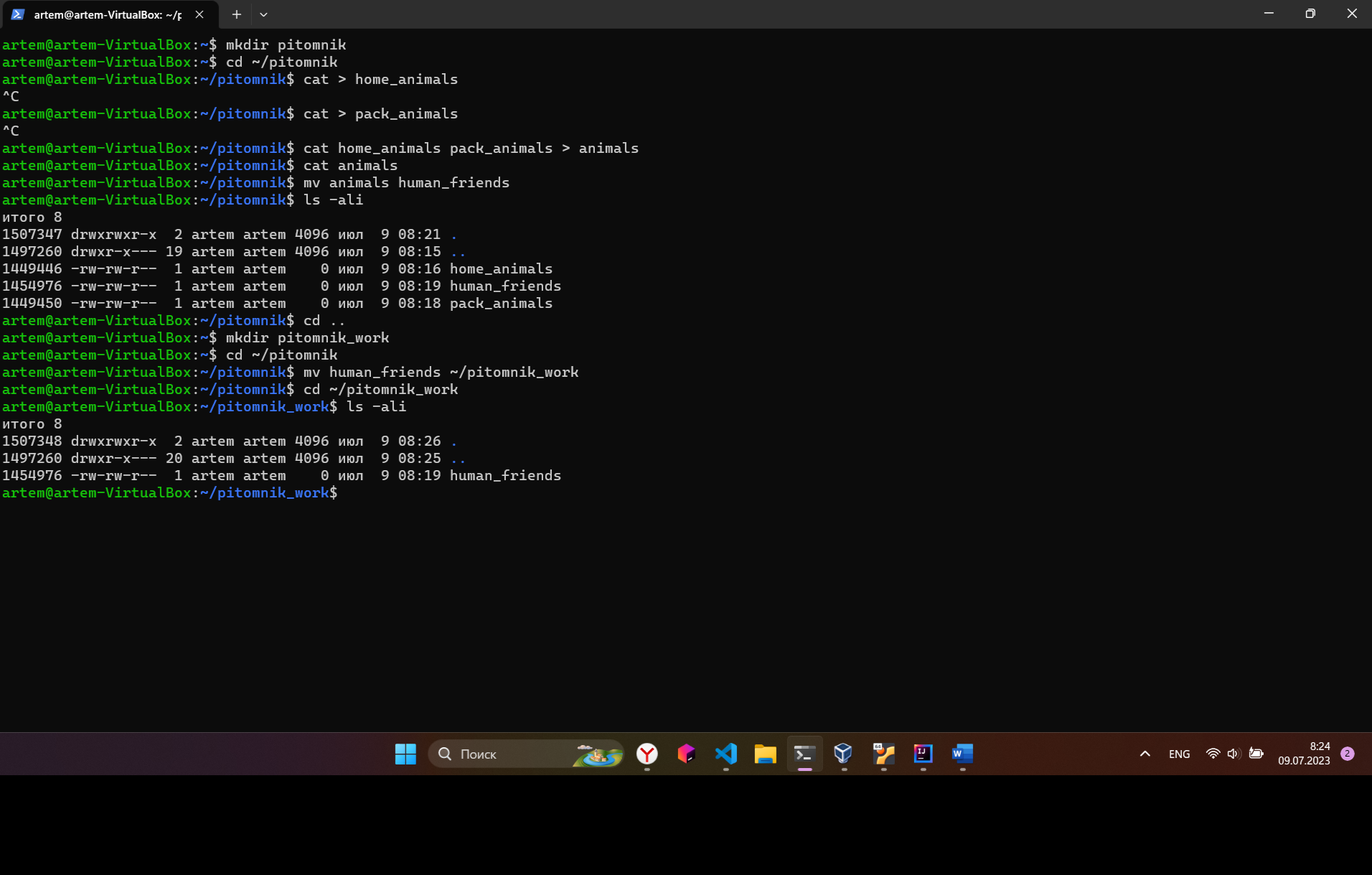
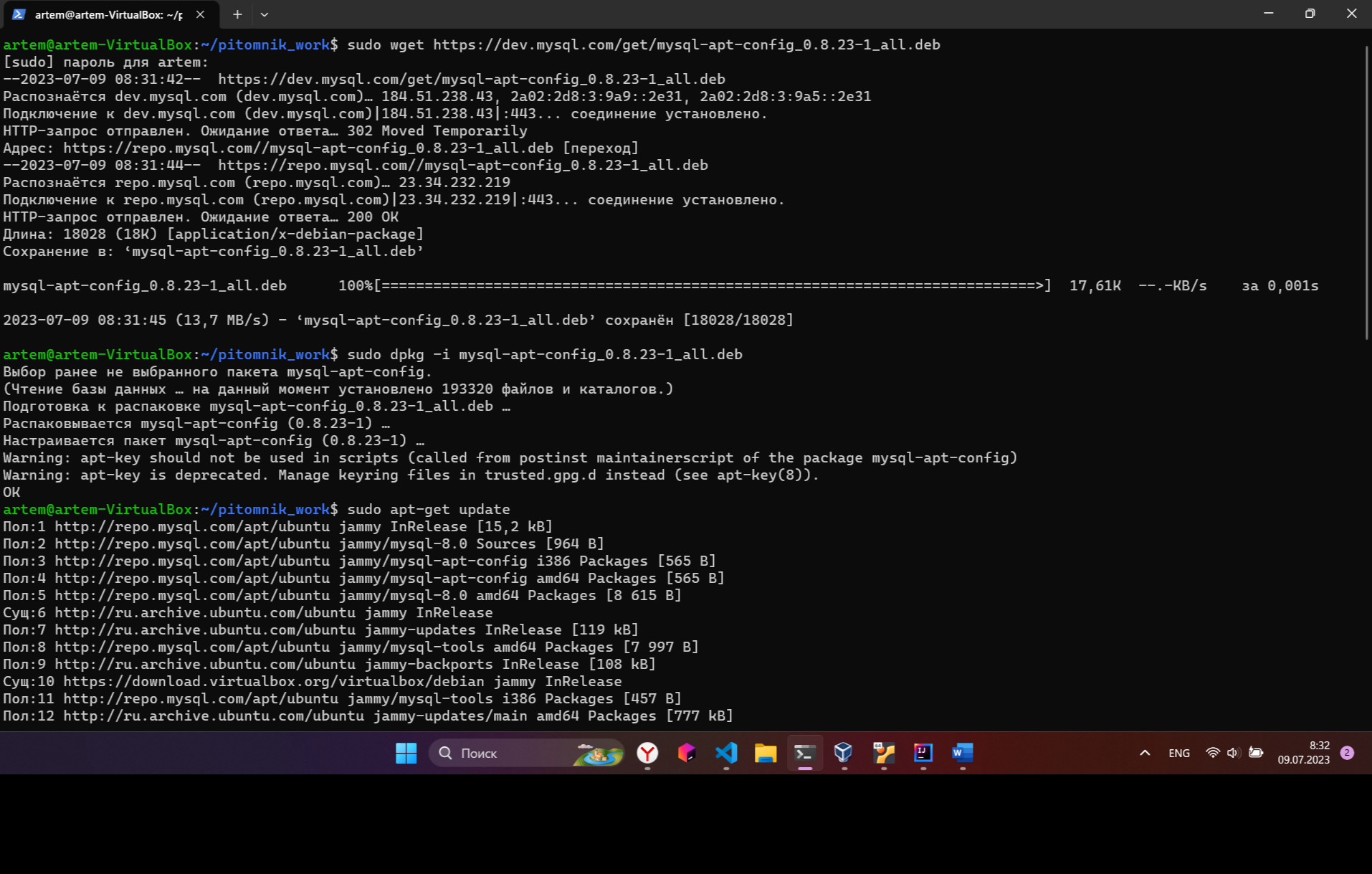
1. Используя команду cat в терминале операционной системы Linux, создать два файла Домашние животные (заполнив файл собаками, кошками, хомяками) и Вьючные животными заполнив файл Лошадьми, верблюдами и ослы), а затем объединить их. Просмотреть содержимое созданного файла. Переименовать файл, дав ему новое имя (Друзья человека).

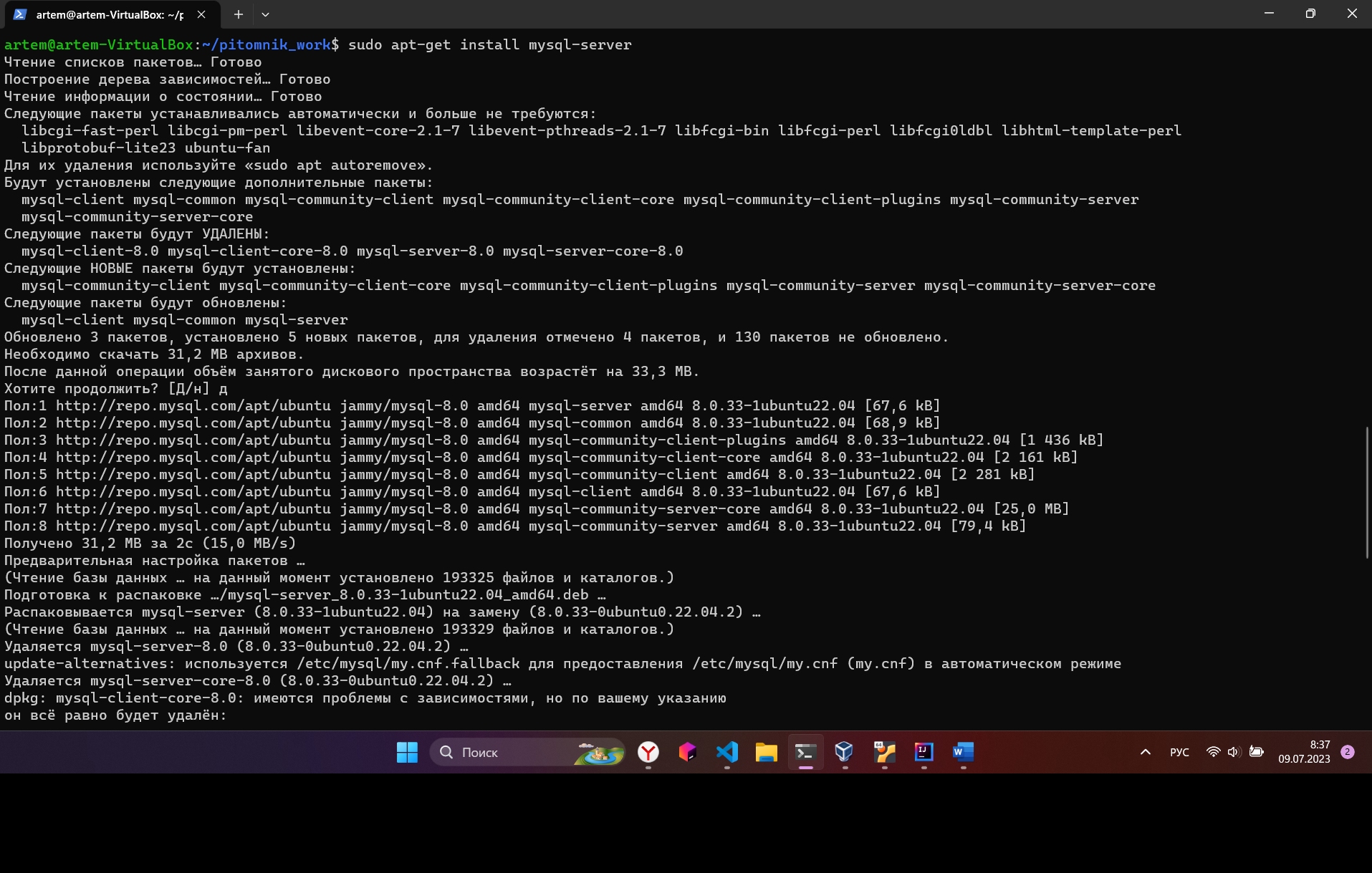


1. Создать директорию, переместить файл туда.

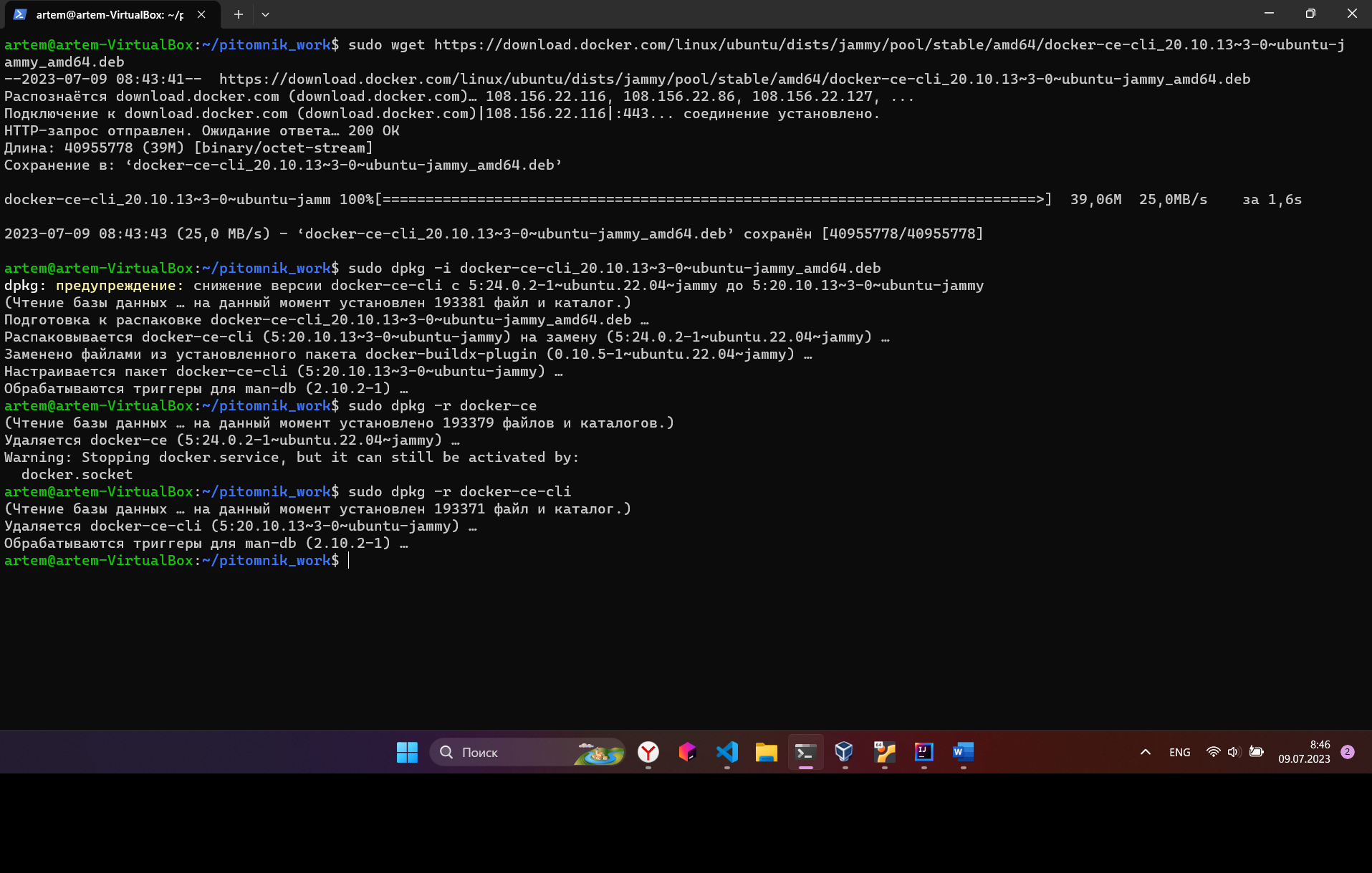


1. Подключить дополнительный репозиторий MySQL. Установить любой пакет из этого репозитория.





1. Установить и удалить deb-пакет с помощью dpkg



1. Выложить историю команд в терминале ubuntu

Задание 1

mkdir pitomnik

cd ~/pitomnik

cat > home\_animals

cat > pack\_animals

cat home\_animals pack\_animals > animals

cat animals

mv animals human\_friends

ls -ali

Задание 2

cd ..

mkdir pitomnik\_work

cd ~/pitomnik

mv human\_friends ~/pitomnik\_work

cd ~/pitomnik\_work

ls -ali

Задание 3

sudo wget https://dev.mysql.com/get/mysql-apt-config\_0.8.23-1\_all.deb

sudo dpkg -i mysql-apt-config\_0.8.23-1\_all.deb

sudo apt-get update

sudo apt-get install mysql-server

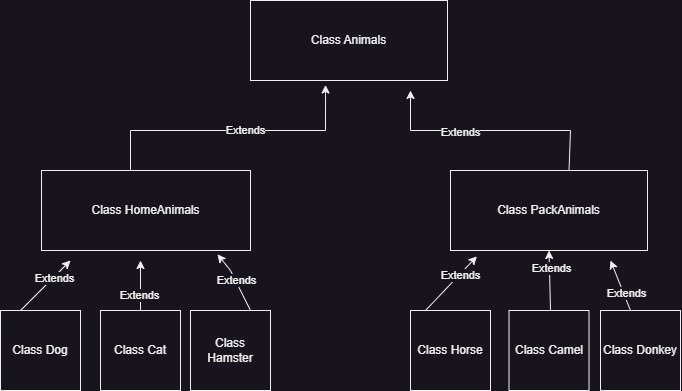
Задание 4

sudo wget https://download.docker.com/linux/ubuntu/dists/jammy/pool/stable/amd64/docker-ce-cli\_20.10.13~3-0~ubuntu-jammy\_amd64.deb

sudo dpkg -i docker-ce-cli\_20.10.13~3-0~ubuntu-jammy\_amd64.deb

sudo dpkg -r docker-ce-cli

1. Нарисовать диаграмму, в которой есть класс родительский класс, домашние животные и вьючные животные, в составы которых в случае домашних животных войдут классы: собаки, кошки, хомяки, а в класс вьючные животные войдут: Лошади, верблюды и ослы).



1. В подключенном MySQL репозитории создать базу данных “Друзья человека”

CREATE DATABASE Human\_friends;

1. Создать таблицы с иерархией из диаграммы в БД

USE Human\_friends;

CREATE TABLE animal\_classes

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Class\_name VARCHAR(20)

);

INSERT INTO animal\_classes (Class\_name)

VALUES ('вьючные'),

('домашние');

CREATE TABLE packed\_animals

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Genus\_name VARCHAR (20),

Class\_id INT,

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

);

INSERT INTO packed\_animals (Genus\_name, Class\_id)

VALUES ('Лошади', 1),

('Ослы', 1),

('Верблюды', 1);

CREATE TABLE home\_animals

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Genus\_name VARCHAR (20),

Class\_id INT,

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

);

INSERT INTO home\_animals (Genus\_name, Class\_id)

VALUES ('Кошки', 2),

('Собаки', 2),

('Хомяки', 2);

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

);

1. Заполнить низкоуровневые таблицы именами(животных), командами которые они выполняют и датами рождения

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

VALUES ('Барсик', '2012-01-01', 'кс-кс-кс', 1),

('Сажик', '2013-01-01', "фу", 1),

('Гоня', '2014-01-01', "иди сюда", 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 ('Шарик', '2021-01-01', 'ко мне, лежать, лапу, голос', 2),

('Бобик', '2019-06-02', "сидеть, лежать, лапу", 2),

('Рекс', '2019-05-05', "сидеть, лежать, лапу, след, фас", 2),

('Босс', '2020-10-10', "сидеть, лежать, фу, место", 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 ('Малыш', '2020-10-12', "фу", 3),

('Хома', '2021-03-12', "нельзя", 3),

('Васька', '2022-07-11', NULL, 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 ('Лютик', '2020-01-12', 'бегом, шагом', 1),

('Дуня', '2017-03-12', "бегом, шагом, хоп", 1),

('Молния', '2020-11-10', "бегом, шагом, хоп", 1);

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 ('Ослик1', '2019-07-09', NULL, 2),

('Ослик2', '2021-05-14', "", 2),

('Ослик3', '2018-07-12', "", 2),

('Ослик4', '2016-12-10', NULL, 2);

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 ('Верблюд1', '2021-05-12', 'вернись', 3),

('Верблюд2', '2020-06-11', "остановись", 3),

('Верблюд3', '2017-03-09', "повернись", 3),

('Верблюд4', '2018-10-04', "стой", 3);

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 \*, 'Лошади' as genus FROM horses

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

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

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

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

CREATE TABLE yang\_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 yang\_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.Genus\_name, ya.Age\_in\_month

FROM horses h

LEFT JOIN yang\_animal 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.Genus\_name, ya.Age\_in\_month

FROM donkeys d

LEFT JOIN yang\_animal 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.Genus\_name, ya.Age\_in\_month

FROM cats c

LEFT JOIN yang\_animal 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.Genus\_name, ya.Age\_in\_month

FROM dogs d

LEFT JOIN yang\_animal 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.Genus\_name, ya.Age\_in\_month

FROM hamsters hm

LEFT JOIN yang\_animal ya ON ya.Name = hm.Name

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

13. Создать класс с Инкапсуляцией методов и наследованием по диаграмме.

14. Написать программу, имитирующую работу реестра домашних животных. В программе должен быть реализован следующий функционал:

14.1 Завести новое животное

14.2 определять животное в правильный класс

14.3 увидеть список команд, которое выполняет животное

14.4 обучить животное новым командам

14.5 Реализовать навигацию по меню

CREATE TABLE pet\_list

(

Id INT AUTO\_INCREMENT PRIMARY KEY,

Name VARCHAR(20),

Birthday DATE,

Command VARCHAR(50),

Genus\_id int

);

INSERT INTO pet\_list (Name, Birthday, Command, Genus\_id)

VALUES ('Барсик', '2012-01-01', 'кс-кс-кс', 1),

('Сажик', '2013-01-01', "фу", 1),

('Гоня', '2014-01-01', "иди сюда", 1),

('Шарик', '2021-01-01', 'ко мне, лежать, лапу, голос', 2),

('Бобик', '2019-06-02', "сидеть, лежать, лапу", 2),

('Рекс', '2019-05-05', "сидеть, лежать, лапу, след, фас", 2),

('Босс', '2020-10-10', "сидеть, лежать, фу, место", 2),

('Малыш', '2020-10-12', "фу", 3),

('Хома', '2021-03-12', "нельзя", 3),

('Васька', '2022-07-11', NULL, 3);

15. Создайте класс Счетчик, у которого есть метод add(), увеличивающий̆ значение внутренней̆ int переменной̆ на 1 при нажатии “Завести новое животное” Сделайте так, чтобы с объектом такого типа можно было работать в блоке try-with-resources. Нужно бросить исключение, если работа с объектом типа счетчик была не в ресурсном try и/или ресурс остался открыт. Значение считать в ресурсе try, если при заведении животного заполнены все поля.