**Итоговая контрольная работа**

**Задание**

**Операционные системы и виртуализация (Linux)**

1. Использование команды cat в Linux

- Создать два текстовых файла: "Pets"(Домашние животные)

В первом файле перечислить собак, кошек и хомяков

***cat > Pets.txtcatsdogshamster*** *Ctrl+D*

и "Pack animals"(вьючные животные), используя команду `cat` в терминале Linux.

. Во втором — лошадей, верблюдов и ослов.

***cat > Pack\_animals.txthorses camels donkeys Ctrl+D***

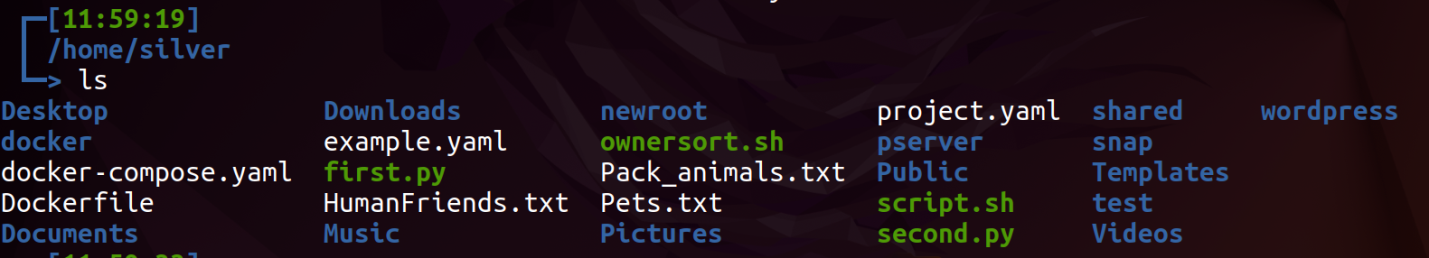
- Объединить содержимое этих двух файлов в один и просмотреть его содержимое.

- Переименовать получившийся файл в "Human Friends"

***cat Pets.txt Pack\_animals.txt > HumanFriends.txt***

***cat HumanFriends.txtcatsdogshamsterhorsescamelsdonkeys***

Пример конечного вывода после команды “ls” :



2. Работа с директориями в Linux

- Создать новую директорию и переместить туда файл "Human Friends".

***mkdir Animalsmv HumanFriends.txt Animals/***

3. Работа с MySQL в Linux. “Установить MySQL на вашу вычислительную машину ”

- Подключить дополнительный репозиторий MySQL и установить один из пакетов из этого репозитория.

***sudo add-apt-repository ppa:mysql-ubuntu/debian-mysql***

***sudo apt update***

4. Управление deb-пакетами

- Установить и затем удалить deb-пакет, используя команду `dpkg`.

***sudo dpkg -i google-chrome-stable\_current\_amd64.deb***

***sudo apt satisfy google-chrome-stable\_current\_amd64.deb***

***dpkg -l google\****

***sudo dpkg -r google-chrome-stable***

5. История команд в терминале Ubuntu

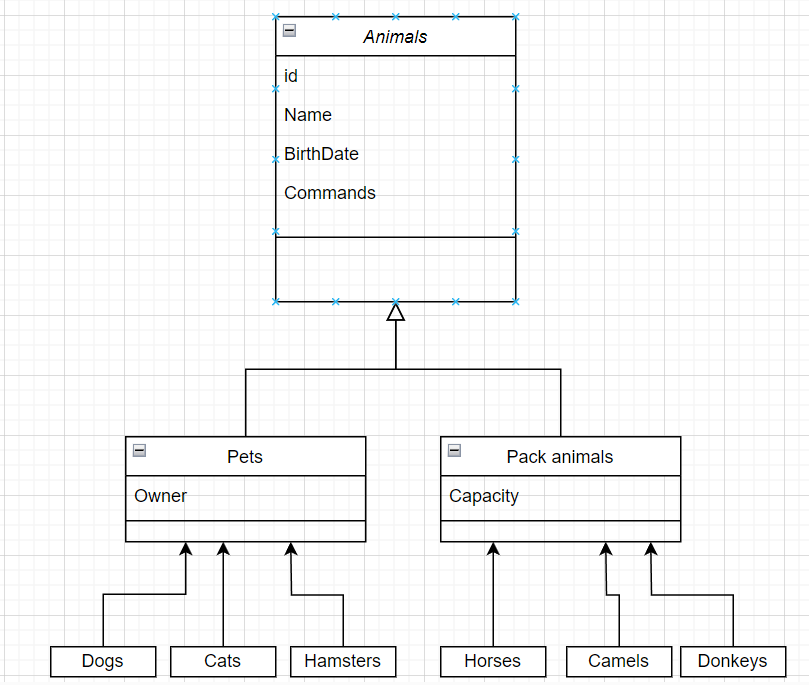
- Сохранить и выложить историю ваших терминальных команд в Ubuntu.

Файл .bash\_history

**Объектно-ориентированное программирование**

6. Диаграмма классов

- Создать диаграмму классов с родительским классом "Животные", и двумя подклассами: "Pets" и "Pack animals".



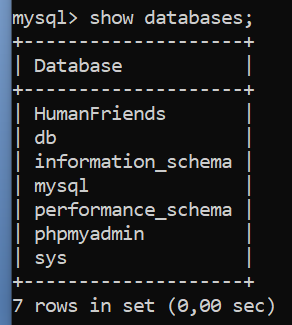
7. Работа с MySQL

7.1. После создания диаграммы классов в 6 пункте, в 7 пункте база данных "Human Friends" должна быть структурирована в соответствии с этой диаграммой. Например, можно создать таблицы, которые будут соответствовать классам "Pets" и "Pack animals", и в этих таблицах будут поля, которые характеризуют каждый тип животных (например, имена, даты рождения, выполняемые команды и т.д.).

7.2 - В ранее подключенном MySQL создать базу данных с названием "Human Friends".

***sudo mysql***

***mysql> create database HumanFriends;***



***mysql> use HumanFriends;***

***Database changed***

- Создать таблицы, соответствующие иерархии из вашей диаграммы классов.

***mysql> CREATE TABLE Dogs (***

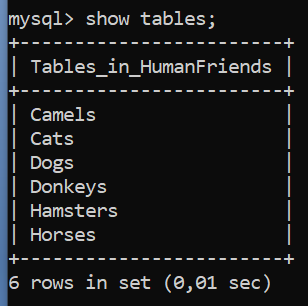
***-> id int,***

***-> Name varchar(255),***

***-> BirthDate date,***

***-> Commands varchar(225)***

***-> );***

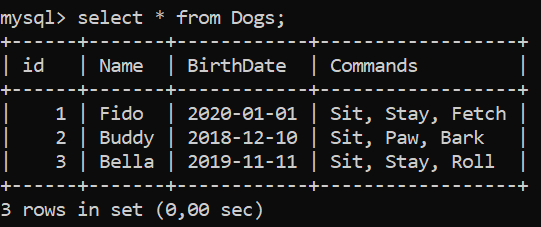


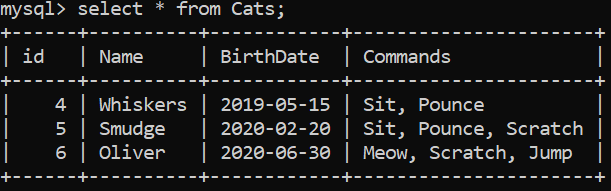
- Заполнить таблицы данными о животных, их командах и датами рождения.

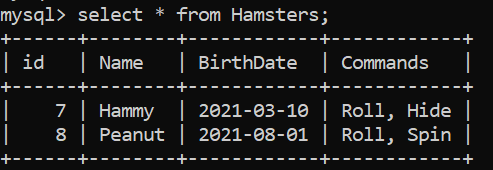
SELECT table\_name, table\_rows

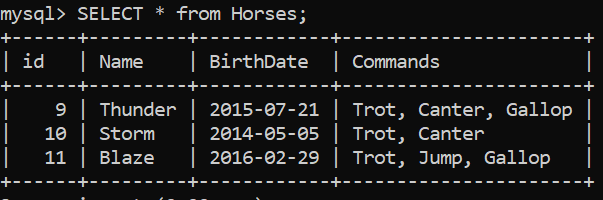
FROM INFORMATION\_SCHEMA.TABLES

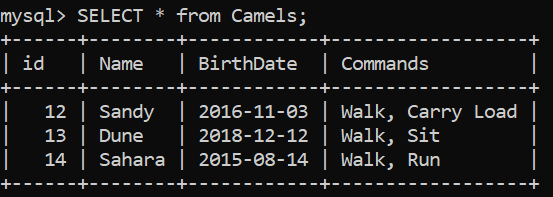
WHERE TABLE\_SCHEMA = 'HumanFriends'

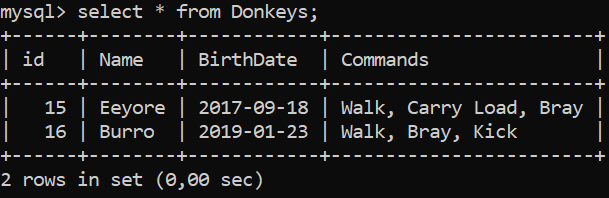












- Объединить все созданные таблицы в одну, сохраняя информацию о принадлежности к исходным таблицам.

***create table Animals as Select \*, "Horse" as Type from Horses***

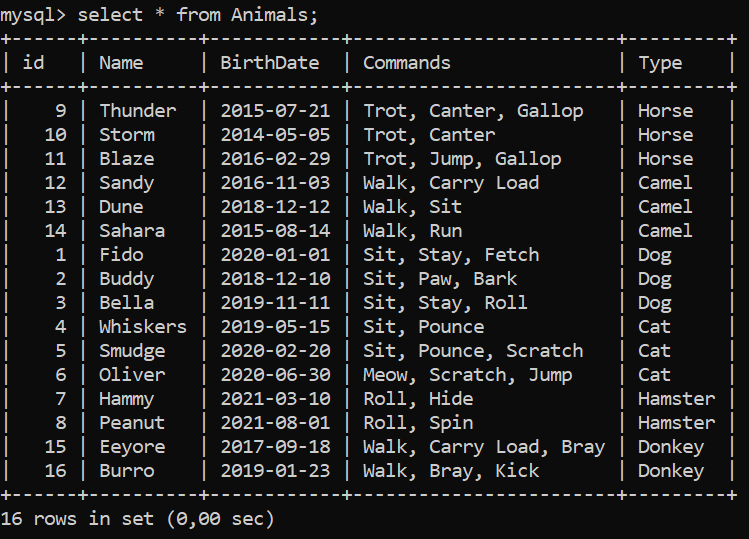
***UNION select \*, "Camel" as Type from Camels***

***UNION select \*, "Dog" as Type from Dogs***

***UNION select \*, "Cat" as Type from Cats***

***UNION select \*, "Hamster" as Type from Hamsters***

***UNION select \*, "Donkey" as Type from Donkeys;***



- Создать новую таблицу для животных в возрасте от 4 до 8 лет и вычислить их возраст с точностью до месяца.

***create table AnimalsAge as Select \*, "Horse" as Type, DATEDIFF(CURDATE(),BirthDate)/30 as Age from Horses***

***where DATEDIFF(CURDATE(),BirthDate)/30 between 48 and 96***

***UNION select \*, "Camel" as Type, DATEDIFF(CURDATE(),BirthDate)/30 as Age from Camels***

***where DATEDIFF(CURDATE(),BirthDate)/30 between 48 and 96***

***UNION select \*, "Dog" as Type, DATEDIFF(CURDATE(),BirthDate)/30 as Age from Dogs***

***where DATEDIFF(CURDATE(),BirthDate)/30 between 48 and 96***

***UNION select \*, "Cat" as Type, DATEDIFF(CURDATE(),BirthDate)/30 as Age from Cats***

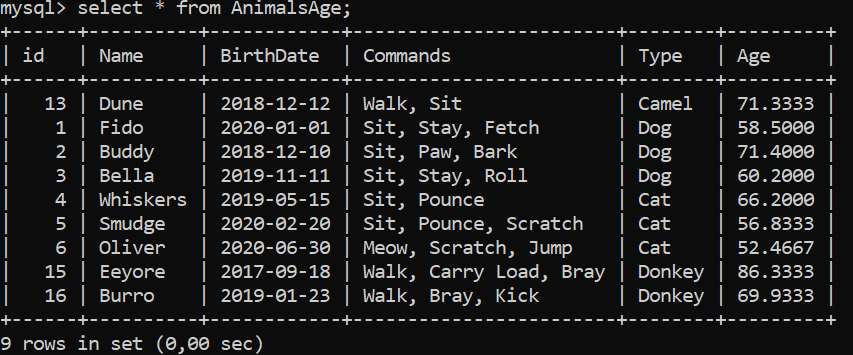
***where DATEDIFF(CURDATE(),BirthDate)/30 between 48 and 96***

***UNION select \*, "Hamster" as Type, DATEDIFF(CURDATE(),BirthDate)/30 as Age from Hamsters***

***where DATEDIFF(CURDATE(),BirthDate)/30 between 48 and 96***

***UNION select \*, "Donkey" as Type, DATEDIFF(CURDATE(),BirthDate)/30 as Age from Donkeys***

***where DATEDIFF(CURDATE(),BirthDate)/30 between 48 and 96;***



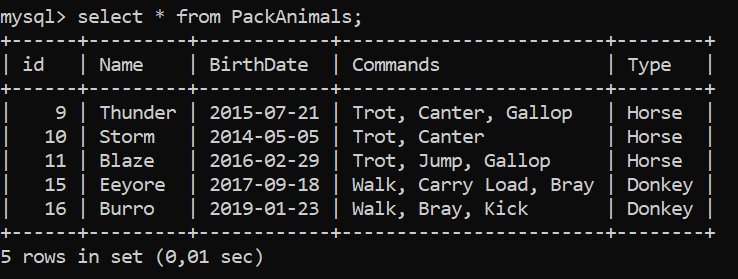
- Удалить записи о верблюдах и объединить таблицы лошадей и ослов.

***mysql> delete from Camels;***



***create table PackAnimals as Select \*, "Horse" as Type from Horses***

***UNION select \*, "Donkey" as Type from Donkeys;***

******

***Drop table Horses;***

***Drop table Donkeys;***

8. ООП и Java

- Создать иерархию классов в Java, который будет повторять диаграмму классов созданную в задаче 6(Диаграмма классов) .

9. Программа-реестр домашних животных

- Написать программу на Java, которая будет имитировать реестр домашних животных.

Должен быть реализован следующий функционал:

9.1. Добавление нового животного

- Реализовать функциональность для добавления новых животных в реестр.

Животное должно определяться в правильный класс (например, "собака", "кошка", "хомяк" и т.д.)

9.2. Список команд животного

- Вывести список команд, которые может выполнять добавленное животное (например, "сидеть", "лежать").

9.3. Обучение новым командам

- Добавить возможность обучать животных новым командам.

9.4 Вывести список животных по дате рождения

9.5. Навигация по меню

- Реализовать консольный пользовательский интерфейс с меню для навигации между вышеуказанными функциями.

10. Счетчик животных

Создать механизм, который позволяет вывести на экран общее количество созданных животных любого типа (Как домашних, так и вьючных), то есть при создании каждого нового животного счетчик увеличивается на “1”.

Решение ссылка на GitHub: https://github.com/SilverKir/FinalTestFirstSemestr/tree/main/PetRegistry