**Задание**

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).

cat > pets

cat > packAnimals

cat pets packAnimals >> animals

mv animals humanFriends

1. Создать директорию, переместить файл туда.

mkdir newFolder

mv humanFriends newFolder/

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

sudo apt install mysql-server

sudo apt install mysql-client

4. Установить и удалить deb-пакет с помощью dpkg.

wget -c <https://download.virtualbox.org/virtualbox/6.1.38/virtualbox-6.1_6.1.38-153438~Ubuntu~jammy_amd64.deb>

sudo dpkg -i virtualbox-6.1\_6.1.38-153438~Ubuntu~jammy\_amd64.deb

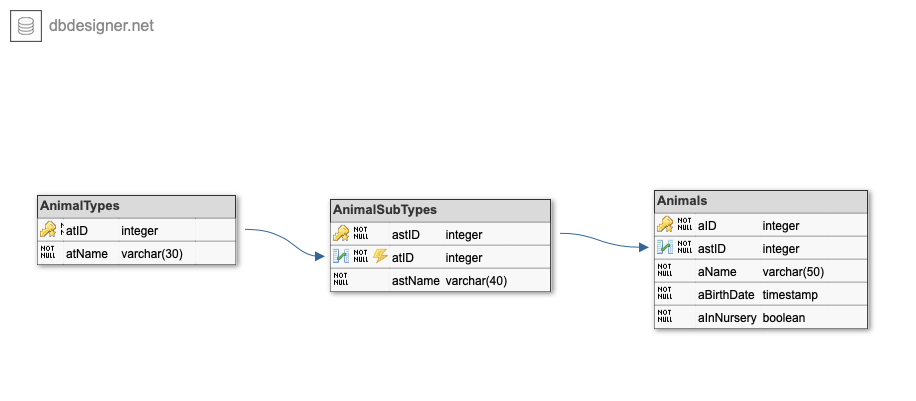
sudo apt -f install

sudo dpkg -P virtualbox-6.1

sudo dpkg -P virtualbox-guest-utils

5. Выложить историю команд в терминале ubuntu

6. 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).



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

8. Создать таблицы с иерархией из диаграммы в БД

CREATE TABLE `AnimalTypes` (

`atID` INT NOT NULL AUTO\_INCREMENT,

`atName` varchar(30) NOT NULL UNIQUE,

PRIMARY KEY (`atID`)

);

CREATE TABLE `AnimalSubTypes` (

`astID` INT NOT NULL AUTO\_INCREMENT,

`atID` INT NOT NULL,

`astName` varchar(40) NOT NULL UNIQUE,

PRIMARY KEY (`astID`)

);

CREATE TABLE `Animals` (

`aID` INT NOT NULL AUTO\_INCREMENT,

`astID` INT NOT NULL,

`aName` varchar(50) NOT NULL,

`aBirthDate` DATE NOT NULL,

`aInNursery` BOOLEAN NOT NULL,

PRIMARY KEY (`aID`)

);

ALTER TABLE `AnimalSubTypes` ADD CONSTRAINT `AnimalSubTypes\_AniiimalTypes` FOREIGN KEY (`atID`) REFERENCES `AnimalTypes`(`atID`);

ALTER TABLE `Animals` ADD CONSTRAINT `Animals\_AnimalSubTypes` FOREIGN KEY (`astID`) REFERENCES `AnimalSubTypes`(`astID`);

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

use `AnimalNursery`;

insert into AnimalTypes(atName) values ('Домашние');

insert into AnimalTypes(atName) values ('Вьючные');

INSERT into AnimalSubTypes (atID, astName) values (1, 'Собака');

INSERT into AnimalSubTypes (atID, astName) values (1, 'Кошка');

INSERT into AnimalSubTypes (atID, astName) values (1, 'Хомяк');

INSERT into AnimalSubTypes (atID, astName) values (2, 'Лошадь');

INSERT into AnimalSubTypes (atID, astName) values (2, 'Верблюд');

INSERT into AnimalSubTypes (atID, astName) values (2, 'Осел');

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (1, 'Шарик', '2021-01-01', true);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (1, 'Тузик', '2020-02-01', false);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (1, 'Трезор', '2019-03-01', true);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (1, 'Алый', '2015-04-01', true);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (1, 'Бобик', '2022-05-01', false);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (2, 'Мурка', '2021-01-18', true);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (2, 'Баксик', '2020-02-23', true);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (2, 'Кнопа', '2017-03-13', false);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (2, 'Мис', '2014-04-11', true);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (4, 'Быстрый', '2015-01-18', true);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (4, 'Резвый', '2020-02-23', true);

INSERT into Animals (astID, aName, aBirthDate, aInNursery) values (4, 'Гнедой', '2017-03-13', false);

10. Удалив из таблицы верблюдов, т.к. верблюдов решили перевезти в другой питомник на зимовку. Объединить таблицы лошади, и ослы в одну таблицу.

delete from Animals

where astID in (select astID from AnimalSubTypes where astName = 'Верблюд')

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 table yangAnimals

as

select aID, astID, aName, aBirthDate, aInNursery, TIMESTAMPDIFF(MONTH , a.aBirthDate, CURDATE()) as q from Animals a

where TIMESTAMPDIFF(YEAR, a.aBirthDate, CURDATE()) < 3 and TIMESTAMPDIFF(YEAR, a.aBirthDate, CURDATE()) >= 1

12. Объединить все таблицы в одну, при этом сохраняя поля, указывающие на прошлую принадлежность к старым таблицам.

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

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.