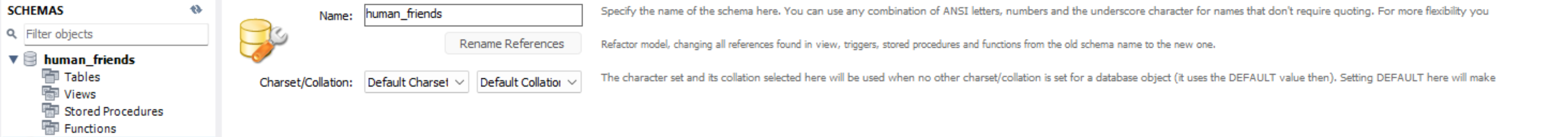
***7. Создать базу данных “Друзья человека”***

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.).

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

CREATE SCHEMA `human\_friends` ;



***8. Создать таблицы с иерархией из диаграммы в БД***

CREATE TABLE `human\_friends`.`pets` (

 `ID` INT NOT NULL AUTO\_INCREMENT,

`Name` VARCHAR(45) NOT NULL,

`Type` VARCHAR(45) NOT NULL,

`BirthDate` DATE NOT NULL,

`Commands` VARCHAR(100) NULL,

PRIMARY KEY (`ID`));

CREATE TABLE `human\_friends`.`packanimals` (

`ID` INT NOT NULL AUTO\_INCREMENT,

`Name` VARCHAR(45) NOT NULL,

`Type` VARCHAR(45) NOT NULL,

`BirthDate` DATE NOT NULL,

`Commands` VARCHAR(100) NULL,

PRIMARY KEY (`ID`));

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

INSERT INTO `human\_friends`.`pets` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Fido', 'Dog', '2020-01-01', 'Sit, Stay, Fetch');

INSERT INTO `human\_friends`.`pets` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Whiskers', 'Cat', '2019-05-15', 'Sit, Pounce');

INSERT INTO `human\_friends`.`pets` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Hammy', 'Hamster', '2021-03-10', 'Roll, Hide');

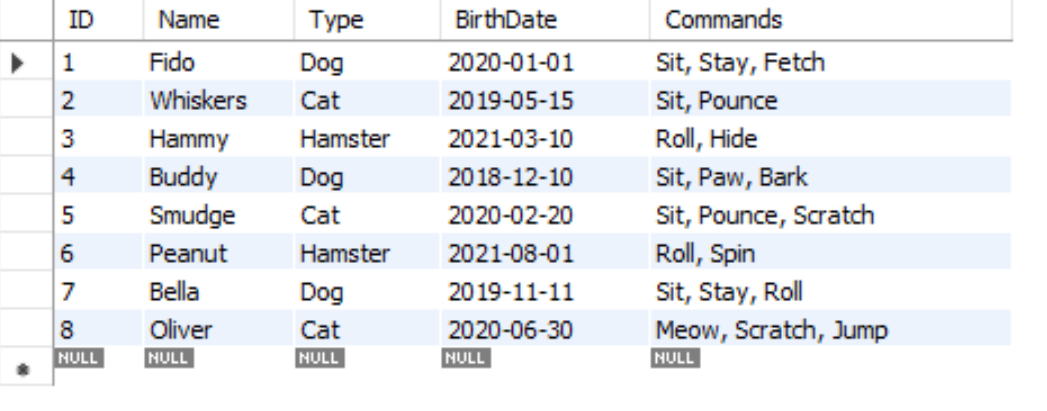
INSERT INTO `human\_friends`.`pets` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Buddy', 'Dog', '2018-12-10', 'Sit, Paw, Bark');

INSERT INTO `human\_friends`.`pets` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Smudge', 'Cat', '2020-02-20', 'Sit, Pounce, Scratch');

INSERT INTO `human\_friends`.`pets` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Peanut', 'Hamster', '2021-08-01', 'Roll, Spin');

INSERT INTO `human\_friends`.`pets` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Bella', 'Dog', '2019-11-11', 'Sit, Stay, Roll');

INSERT INTO `human\_friends`.`pets` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Oliver', 'Cat', '2020-06-30', 'Meow, Scratch, Jump');



INSERT INTO `human\_friends`.`packanimals` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Thunder', 'Horse', '2015-07-21', 'Trot, Canter, Gallop');

INSERT INTO `human\_friends`.`packanimals` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Sandy', 'Camel', '2016-11-03', 'Walk, Carry Load');

INSERT INTO `human\_friends`.`packanimals` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Eeyore', 'Donkey', '2017-09-18', 'Walk, Carry Load, Bray');

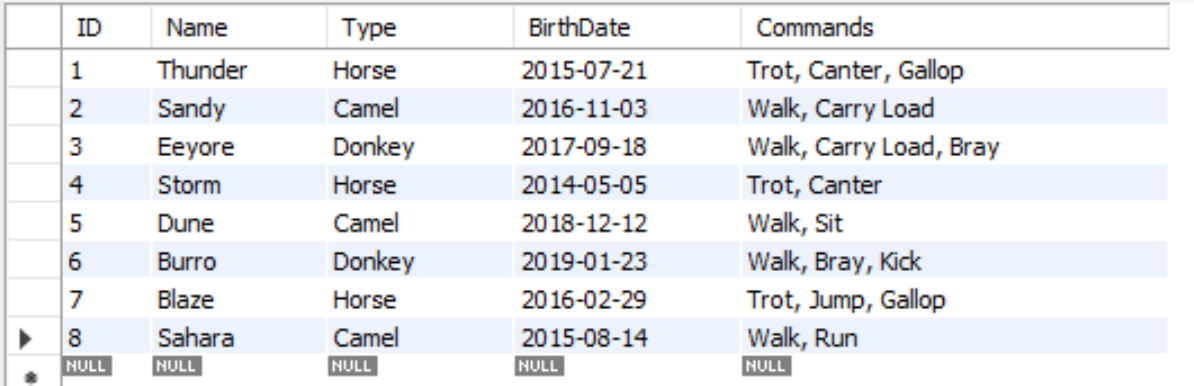
INSERT INTO `human\_friends`.`packanimals` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Storm', 'Horse', '2014-05-05', 'Trot, Canter');

INSERT INTO `human\_friends`.`packanimals` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Dune', 'Camel', '2018-12-12', 'Walk, Sit');

INSERT INTO `human\_friends`.`packanimals` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Burro', 'Donkey', '2019-01-23', 'Walk, Bray, Kick');

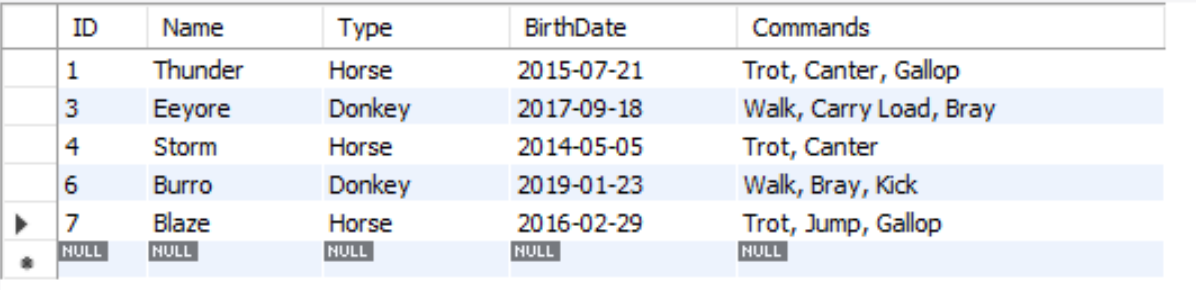
INSERT INTO `human\_friends`.`packanimals` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Blaze', 'Horse', '2016-02-29', 'Trot, Jump, Gallop');

INSERT INTO `human\_friends`.`packanimals` (`Name`, `Type`, `BirthDate`, `Commands`) VALUES ('Sahara', 'Camel', '2015-08-14', 'Walk, Run');



***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 human\_friends.packanimals WHERE `Type` = "Camel";



***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***

DROP TABLE IF EXISTS human\_friends.young\_animals;

CREATE TABLE human\_friends.young\_animals AS

SELECT \* FROM human\_friends.packanimals

UNION

SELECT \* FROM human\_friends.pets;

DROP TABLE IF EXISTS human\_friends.young\_animals\_age;

CREATE TABLE human\_friends.young\_animals\_age AS

SELECT

ID,

`Name`,

`Type`,

BirthDate,

Commands,

CONCAT(

FLOOR(DATEDIFF(CURRENT\_DATE, BirthDate) / 365), ' years ',

FLOOR((DATEDIFF(CURRENT\_DATE, BirthDate) % 365) / 30), ' months'

) AS age

FROM human\_friends.young\_animals;

DROP TABLE IF EXISTS human\_friends.young\_animals\_age\_1\_3;

CREATE TABLE human\_friends.young\_animals\_age\_1\_3 AS

SELECT \*

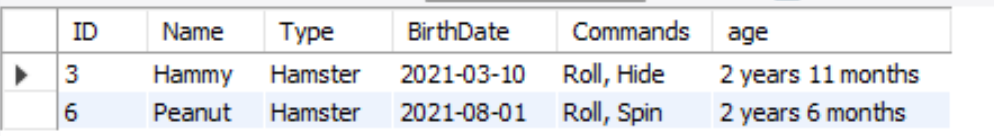
FROM human\_friends.young\_animals\_age

WHERE

(FLOOR(DATEDIFF(CURRENT\_DATE, Birthdate) / 365) > 1) AND

(FLOOR(DATEDIFF(CURRENT\_DATE, Birthdate) / 365) < 3);

SELECT \* FROM human\_friends.young\_animals\_age\_1\_3;



***12. Объединить все таблицы в одну, при этом сохраняя поля, указывающие на прошлую принадлежность к старым таблицам.***DROP TABLE IF EXISTS human\_friends.all\_animals;

CREATE TABLE human\_friends.all\_animals AS

SELECT \*, ' ' AS age , 'packanimals' AS source\_table

FROM human\_friends.packanimals

UNION ALL

SELECT \*, ' ' AS age , 'pets' AS source\_table

FROM human\_friends.pets

UNION ALL

SELECT \*, 'young\_animals\_age\_1\_3' AS source\_table

FROM human\_friends.young\_animals\_age\_1\_3;

SELECT \* FROM human\_friends.all\_animals;

