SIBD Project - part 2

(Group 13) Diogo Moura nº87748, Filipa Rente nº81324, Filipe Inês nº78775

November 2018

1 Database Schema

```
drop table if exists produced_indicator;
drop table if exists test_procedure;
drop table if exists radiography;
drop table if exists performed;
drop table if exists procedures;
drop table if exists indicator;
drop table if exists prescription;
drop table if exists medication;
drop table if exists consult_diagnosis;
drop table if exists diagnosis_code;
drop table if exists participation;
drop table if exists consult;
drop table if exists animal;
drop table if exists generalization_species;
drop table if exists species;
drop table if exists assistant;
drop table if exists veterinary;
drop table if exists client;
drop table if exists phone_number;
drop table if exists person;

create table person (
   VAT int primary key,
   name varchar(255),
   address_street varchar(255),
   address_city varchar(255),
   address_zip varchar(255)
);

create table phone_number (
   VAT int,
   phone int primary key,
   foreign key (VAT) references person(VAT) on delete cascade
);
```

```
create table client (
    VAT int primary key,
    foreign key (VAT) references person(VAT) on delete cascade
);

create table veterinary (
    VAT int primary key,
    specialization varchar(255),
    bio varchar(255),
    foreign key (VAT) references person(VAT) on delete cascade
);


create table assistant (
    VAT int primary key,
    foreign key (VAT) references person(VAT) on delete cascade)
       ;
    create table species (
    name varchar(255) primary key,
    description varchar(255)
);

create table generalization_species (
    name1 varchar(255),
    name2 varchar(255),
    primary key(name1,name2),
    foreign key(name1) references species(name),
    foreign key(name2) references species(name)
);

create table animal (
    name varchar(255),
    VAT int,
    species_name varchar(255),
    colour varchar(255),
    gender varchar(255),
    birth_year int,
    age int,
    primary key(name, VAT),
    foreign key(VAT) references client(VAT) on delete cascade,
    foreign key(species_name) references species(name)
);

create table consult (
    name varchar(255),
    VAT_owner int,
    date_timestamp datetime,
    s text,
    o text,
    a text,
    p text,
    VAT_client int,
    VAT_vet int,
    weight numeric(4,2),
    primary key (name,VAT_owner,date_timestamp),
    foreign key (name,VAT_owner) references animal(name,VAT) on
         delete cascade,
    foreign key(VAT_client) references client(VAT) on delete
       cascade,
    foreign key(VAT_vet) references veterinary(VAT) on delete
       cascade
);

create table participation (
    name varchar(255),
    VAT_owner int,
    date_timestamp datetime,
    VAT_assistant int,
    primary key (name, VAT_owner, date_timestamp,VAT_assistant)
       ,
    foreign key(name,VAT_owner,date_timestamp) references
       consult(name,VAT_owner,date_timestamp) on delete cascade
       ,
    foreign key(VAT_assistant) references assistant(VAT) on
       delete cascade
);
```

```
create table diagnosis_code (
    code int primary key,
    name varchar (255)
);

create table consult_diagnosis (
    code int,
    name varchar(255),
    VAT_owner int,
    date_timestamp datetime,
    primary key (code, name, VAT_owner,date_timestamp) ,
    foreign key(code) references diagnosis_code(code),
    foreign key(name,VAT_owner,date_timestamp) references
       consult(name,VAT_owner,date_timestamp) ON DELETE CASCADE
);

create table medication (
    name varchar(255),
    lab varchar(255),
    dosage varchar(255),
    primary key(name,lab,dosage)
);

create table prescription (
    code int,
    name varchar(255),
    VAT_owner int,
    date_timestamp datetime,
    name_med varchar(255),
    lab varchar(255),
    dosage varchar(255),
    regime varchar(255),
    primary key (code,name,VAT_owner,date_timestamp,name_med,
       lab,dosage),
    foreign key (code,name,VAT_owner,date_timestamp) references
        consult_diagnosis(code,name,VAT_owner,date_timestamp)
       ON DELETE CASCADE on update cascade,
    foreign key (name_med,lab,dosage) references medication(
       name,lab,dosage)
 );

 create table indicator (
    name varchar(255) primary key,
    reference_value numeric(6,2),
    unit varchar(255),
    description text
 );

 create table procedures (
    name varchar(255),
    VAT_owner int,
    date_timestamp datetime,
    num int,
    description text,
    primary key (name,VAT_owner,date_timestamp,num),
    foreign key (name,VAT_owner,date_timestamp) references
       consult(name,VAT_owner,date_timestamp) ON DELETE CASCADE
 );

 create table performed (
    name varchar(255),
    VAT_owner int,
    date_timestamp datetime,
    num int,
    VAT_assistant int,
    foreign key (name,VAT_owner,date_timestamp,num) references
       procedures(name,VAT_owner,date_timestamp,num) on delete
       cascade,
    foreign key (VAT_assistant) references assistant(VAT) on
       delete cascade
 );

 create table radiography (
    name varchar(255),
    VAT_owner int,
    date_timestamp datetime,
    num int,
    file varchar(255),
    foreign key(name,VAT_owner,date_timestamp,num) references
       procedures(name,VAT_owner,date_timestamp,num) on delete
       cascade
);

create table test_procedure (
    name varchar(255),
    VAT_owner int,
    date_timestamp datetime,
    num int,
    type varchar(255),
    foreign key(name,VAT_owner,date_timestamp,num) references
       procedures(name,VAT_owner,date_timestamp,num) on delete
       cascade
);

create table produced_indicator (
    name varchar(255),
    VAT_owner int,
    date_timestamp datetime,
    num int,
    indicator_name varchar(255),
    value numeric(6,2),
    foreign key(name,VAT_owner,date_timestamp,num) references
       procedures(name,VAT_owner,date_timestamp,num) on delete
       cascade,
    foreign key(indicator_name) references indicator(name)
);
```

2 Populating the Database

```
insert into person (VAT,name,address_street,address_city,
   address_zip)
```

```
values
    (101,"Meredith Grey","North Middle River Lane","Laurel
       ","2000-001"),
    (102,"Cristina Yang","SW. Pennington Drive","Altoona
       ","2000-002"),
    (103,"Izzie Stevens","Myrtle St.","Hopewell","2000-003"),
    (104,"Alex Karev","Military Drive","Rolling Meadows
       ","2000-004"),
    (105,"George Malley","S. Amerige Street","Durham
       ","2000-005"),
    (106,"Miranda Bailey","Richardson Street","Dover
       ","2001-006"),
    (107,"Richard Webber","Myrtle Street","Melbourne
       ","2000-007"),
    (108,"Preston Burke","Jockey Hollow Court", "Palm City
       ","2000-008"),
    (109,"Derek Shepherd","E. Tunnel Ave.","Christiansburg
       ","2000-009"),
    (110,"Addison Montgomery","Eagle Drive","Fairfax
       ","2000-010"),
    (111,"Callie Torres","West Pleasant Drive","Elkridge
       ","2000-011"),
    (112,"Mark Sloan","Pilgrim St.","Bristol","2000-012"),
    (113,"Lexie Grey","Gonzales Dr.","Newton","2000-013"),
    (114,"Erica Hahn","Strawberry Ave.","Chevy Chase
       ","2000-014"),
    (115,"Owen Hunt", "W. Longfellow Road","Beachwood
       ","2000-015"),
    (116,"Arizona Robbins","Brewery Street","West Springfield
       ","2000-016"),
    (117,"Teddy Altman","Bridle St.","Johnston","2000-017"),
    (118,"April Kepner","Vernon Street","North Kingstown
       ","2000-018"),
    (119,"Jackson Avery","Inverness Road","Irvington
       ","2000-019"),
    (120,"Jo Karev","Academy Drive","Bloomington","2000-020"),
    (121,"Shane Ross","Rockville St.","SW. Peachtree Road
       ","2000-021"),
    (122,"Stephanie Edwards","SW. Peachtree Road","Trenton
       ","2000-022"),
    (123,"Leah Murphy","Kingston Drive","Brockton","2000-023"),
    (124,"Amelia Shepherd","Heather Road","Wilmette
       ","2000-024"),
    (125,"Maggie Pierce","Leatherwood Road","Peabody
       ","2000-025"),
    (126,"Benjamin Warren","Amerige Street","Newport News
       ","2000-026"),
    (127,"Andrew DeLuca","Academy Lane","Hickory","2000-027"),
    (128,"Nathan Riggs","Edgefield Lane","Rahway","2000-028"),
    (129, "John Smith", "Koxford Street", "Mumford",
       "2000-029"),
    (130, "Dexter", "Blood Street", "Murder city", "2000-030");

insert into phone_number (VAT,phone)
values
    (101,900000101),
    (102,900000102),
    (103,900000103),
    (104,900000104),
    (105,900000105),
    (106,900000106),
    (107,900000107),
    (108,900000108),
    (109,900000109),
    (110,900000110),
    (111,900000111),
    (112,900000112),
    (113,900000113),
    (114,900000114),
    (115,900000115),
    (116,900000116),
    (117,900000117),
    (118,900000118),
    (119,900000119),
    (120,900000120),
    (121,900000121),
    (122,900000122),
    (123,900000123),
    (124,900000124),
    (125,900000125),
    (126,900000126),
    (127,900000127),
    (128,900000128),
    (129, 900000129),
    (130, 900000130);

insert into client (VAT)
values
    (101),
    (102),
    (103),
    (104),
    (105),
    (106),
    (107),
    (108),
    (109),
    (110),
    (111),
    (112),
    (113),
    (114),
    (115),
    (116),
    (117),
    (118),
    (119),
    (120),
    (129),
    (130);

insert into veterinary (VAT,specialization,bio)
values
```

```
    (121,"Anaesthesiologist","9 years of experience, University of Calgary"),
    (122,"Clinical pathologist","12 years of experience, Royal College"),
    (123,"Birds","5 years of experience, Royal College"),
    (124,"Animal welfare","15 years of experience, University of Guelph"),
    (129, "Animal nutrition", "7 years of experience");

insert into assistant (VAT)
values
    (125),
    (126),
    (127),
    (128),
    (130);

insert into species (name, description)
Values
    ("bulldog","breed of dog"),
    ("poodle","breed of dog"),
    ("chihuahua","breed of dog"),
    ("rottweiler","breed of dog"),
    ("beagle","breed of dog"),
    ("boxer","breed of dog"),
    ("weimaraner","breed of dog"),
    ("dog","dog"),
    ("siamese","breed of cat"),
    ("persian","breed of cat"),
    ("maine coon","breed of cat"),
    ("ragdoll","breed of cat"),
    ("bengal","breed of cat"),
    ("abyssinian","breed of cat"),
    ("birman","breed of cat"),
    ("duroc","breed of pig"),
    ("berkshire","breed of pig"),
    ("yorkshire","breed of pig"),
    ("red-eared slider turtle","breed of turtle"),
    ("eastern box turtle","breed of turtle"),
    ("western painted turtle","breed of turtle"),
    ("map turtle","breed of turtle"),
    ("wood turtle","breed of turtle"),
    ("Canis","dog’s Genus"),
    ("Canidae","dog’s Family"),
    ("Carnivora","dog’s Order"),
    ("Mammalia","dog’s Class"),
    ("bird","bird"),
    ("Aves","bird’s Class"),
    ("Ornithurae","bird’s Clade"),
    ("cat", "cat"),
    ("Felis silvestris", "cat’s Species"),
    ("Felis", "cat’s Genus"),
    ("Felidae", "cat’s Family"),
    ("pig", "pig"),
    ("Sus", "pig’s Genus"),
    ("Suidae", "pig’s Family"),
    ("Artiodactyla", "pig’s Order"),
    ("turtle","turtle"),
    ("Testudines","turtle’s Order"),
    ("Reptilia","turtle’s Class");

insert into generalization_species (name1, name2)
values
    ("rottweiler","dog"),
    ("poodle","dog"),
    ("chihuahua","dog"),
    ("bulldog","dog"),
    ("beagle","dog"),
    ("boxer","dog"),
    ("weimaraner","dog"),
    ("siamese","cat"),
    ("persian","cat"),
    ("maine coon","cat"),
    ("ragdoll","cat"),
    ("bengal","cat"),
    ("abyssinian","cat"),
    ("birman","cat"),
    ("duroc","pig"),
    ("berkshire","pig"),
    ("yorkshire","pig"),
    ("red-eared slider turtle","turtle"),
    ("eastern box turtle","turtle"),
    ("western painted turtle","turtle"),
    ("map turtle","turtle"),
    ("wood turtle","turtle"),
    ("dog","Canis"),
    ("Canis","Canidae"),
    ("Canidae","Carnivora"),
    ("Carnivora","Mammalia"),
    ("bird","Aves"),
    ("Aves","Ornithurae"),
    ("cat","Felis silvestris"),
    ("Felis silvestris","Felis"),
    ("Felis","Felidae"),
    ("Felidae","Carnivora"),
    ("pig","Sus"),
    ("Sus","Suidae"),
    ("Suidae","Artiodactyla"),
    ("Artiodactyla","Mammalia"),
    ("turtle","Testudines"),
    ("Testudines","Reptilia");

insert into animal (name,VAT,species_name,colour,gender,
   birth_year,age)
values
    ("Bella",101,"boxer","black","female",2005,13),
    ("Lucy",102,"weimaraner","white","female",2010,8),
    ("Ash",103,"beagle","brown","male",2011,7),
    ("Benji",104,"poodle","black","male",2010,8),
    ("Twit",105,"bird","yellow","female",2012,4),
    ("Kika", 101, "bulldog", "brown", "female", 2015, 3),
    ("Leo", 117, "persian", "yellow", "male", 2014, 4),
    ("Deutsch", 109, "bird", "grey", "male", 2002, 16),
    ("Eva", 104, "red-eared slider turtle", "brown", "female",
       2013, 5),
    ("Vitria", 129,"bird","brown","female",2013,5),
    ("Willy", 101, "chihuahua","brown", "male",2008,10),
    ("Bochita", 129, "chihuahua","grey", "male",2010,8),
    ("Kashey", 129, "chihuahua","black", "female",2011,7),
    ("Brownie", 114, "rottweiler","grey", "male",2015,3),
    ("Caju", 130, "beagle","white", "male",2012,6),
    ("Caju", 120, "boxer","brown", "male",2016,2),
    ("Cloud", 106, "siamese", "white", "female", 2012, 6),
    ("Star", 106, "birman", "black", "male", 2014, 4);

insert into consult (name,VAT_owner,date_timestamp,s,o,a,p,
   VAT_client,VAT_vet,weight)
values
    ("Bella",101,"2018-02-17 13:12:20","s","o","a","p
       ",101,121,30),
    ("Lucy", 102, "2018-02-20 09:20:18", "s", "Animal presents
       symptoms of obesity and high cholesterol.", "a", "p",
       103, 122, 40.00),
    ("Twit",105,"2018-03-02 10:14:19","s","o","a","p
       ",110,123,4),
    ("Kika", 101, "2017-03-02 10:14:19", "s", "may be obese.",
       "a", "p", 101, 129, 33.20),
    ("Leo", 117, "2017-04-12 14:47:10", "s", "o", "a", "p",
       117, 129, 20.50),
    ("Eva", 104, "2017-07-15 11:30:54", "s", "o", "a", "p",
       104, 129, 30.70),
    ("Deutsch", 109, "2018-05-04 16:12:04", "s", "o", "a", "p",
        109, 129, 3.42),
    ("Vitria", 129,"2018-08-07 13:20:04", "s", "may have broken
        a wing.", "a", "p", 129, 123, 8.80),
    ("Willy", 101, "2017-03-02 13:45:57", "s", "o", "a", "p",
       101, 129, 25.20),
    ("Willy", 101, "2017-03-04 10:20:06", "s", "o", "a", "p",
       101, 129, 25.15),
    ("Bochita", 129, "2017-02-17 10:10:21", "s", "o", "a", "p",
        101, 129, 20.65),
    ("Kashey", 129, "2017-02-17 15:10:21", "s", "o", "a", "p",
       101, 129, 15.37),
    ("Ash", 103, "2018-04-12 14:34:32", "s", "o", "a", "p",
       102, 124, 24.50),
    ("Ash", 103, "2018-06-18 16:05:12", "s", "o", "a", "p",
       103, 121, 26.10),
    ("Benji", 104, "2017-09-16 14:01:45", "s", "o", "a", "p",
       104, 122, 18.20),
    ("Brownie", 114, "2017-12-27 12:20:33", "s", "o", "a", "p",
        114, 123, 28.10),
    ("Brownie", 114, "2018-01-17 13:35:20", "s", "o", "a", "p",
        114, 123, 25.40),
    ("Caju", 130, "2018-02-23 17:15:20", "s", "o", "a", "p",
       130, 124, 15.84),
    ("Caju", 130, "2018-05-23 15:15:20", "s", "o", "a", "p",
       130, 129, 16.24),
    ("Caju", 120, "2018-06-13 16:49:11", "s", "o", "a", "p",
       120, 122, 30.11),
    ("Cloud", 106, "2018-07-11 12:12:49", "s", "o", "a", "p",
       106, 121, 5.20),
    ("Star", 106, "2017-07-05 10:01:53", "s", "o", "a", "p",
       106, 122, 6.04),
    ("Star", 106, "2018-07-05 11:10:43", "s", "o", "a", "p",
       106, 123, 7.13);




insert into participation (name,VAT_owner,date_timestamp,
   VAT_assistant)
values
    ("Bella",101,"2018-02-17 13:12:20",128),
    ("Twit",105,"2018-03-02 10:14:19",127),
    ("Leo", 117, "2017-04-12 14:47:10",125),
    ("Leo", 117, "2017-04-12 14:47:10",126),
    ("Leo", 117, "2017-04-12 14:47:10",127),
    ("Leo", 117, "2017-04-12 14:47:10",128),
    ("Eva", 104, "2017-07-15 11:30:54",125),
    ("Eva", 104, "2017-07-15 11:30:54",126),
    ("Kika",101,"2017-03-02 10:14:19",125),
    ("Kika",101,"2017-03-02 10:14:19",126),
    ("Vitria", 129,"2018-08-07 13:20:04", 125),
    ("Deutsch", 109, "2018-05-04 16:12:04",130),
    ("Bochita", 129, "2017-02-17 10:10:21",126),
    ("Bochita", 129, "2017-02-17 10:10:21",128),
    ("Ash", 103, "2018-04-12 14:34:32", 127),
    ("Ash", 103, "2018-06-18 16:05:12",125),
    ("Caju", 130, "2018-02-23 17:15:20",128),
    ("Caju", 120, "2018-06-13 16:49:11", 130),
    ("Cloud", 106, "2018-07-11 12:12:49",125),
    ("Cloud", 106, "2018-07-11 12:12:49",128);





insert into diagnosis_code(code, name)
values
    (1, "Fever, slow metabolism"),
    (2, "Sinusitis"),
    (3, "Infection"),
    (4, "Kidney failure"),
    (5, "Blindness"),
    (6, "Parvo"),
    (7, "Lyme disease"),
    (8, "Poisoning"),
    (9, "Heartworm disease"),
    (10, "Cancer"),
    (11, "Leishmaniose"),
    (12, "Gastric Torsion"),
    (13, "Diabetes"),
    (14, "Immunodeficiency Virus"),
    (15, "Leukemia"),
    (16, "Rabies"),
    (17, "Obesity");
```

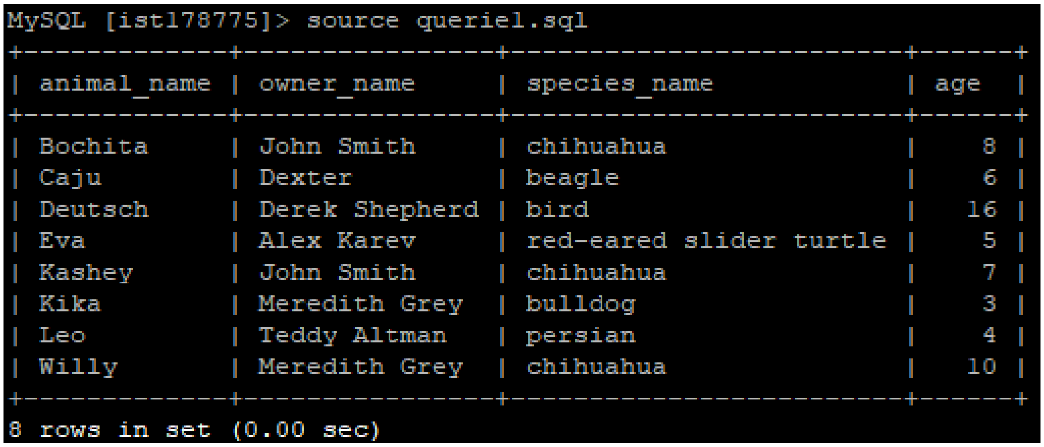
```
insert into consult_diagnosis(code, name, VAT_owner,
   date_timestamp)
Values
    (17, "Lucy", 102, "2018-02-20 09:20:18"),
    (12,"Deutsch", 109,"2018-05-04 16:12:04"),
    (13,"Eva", 104, "2017-07-15 11:30:54"),
    (2,"Eva", 104, "2017-07-15 11:30:54"),
    (3,"Eva", 104, "2017-07-15 11:30:54"),
    (14,"Leo", 117, "2017-04-12 14:47:10"),
    (2,"Leo", 117, "2017-04-12 14:47:10"),
    (2,"Kika", 101, "2017-03-02 10:14:19"),
    (15,"Vitria", 129,"2018-08-07 13:20:04"),
    (4,"Deutsch", 109,"2018-05-04 16:12:04"),
    (3,"Willy", 101, "2017-03-02 13:45:57"),
    (1,"Willy", 101, "2017-03-02 13:45:57"),
    (1,"Willy", 101, "2017-03-04 10:20:06"),
    (3,"Bochita", 129, "2017-02-17 10:10:21"),
    (3,"Kashey", 129, "2017-02-17 15:10:21"),
    (6,"Ash", 103, "2018-04-12 14:34:32"),
    (16,"Ash", 103, "2018-06-18 16:05:12"),
    (14, "Benji", 104, "2017-09-16 14:01:45"),
    (5, "Brownie", 114, "2017-12-27 12:20:33"),
    (7, "Brownie", 114, "2018-01-17 13:35:20"),
    (6,"Caju", 130, "2018-02-23 17:15:20"),
    (8,"Caju", 130, "2018-05-23 15:15:20"),
    (4, "Caju", 120, "2018-06-13 16:49:11"),
    (10, "Cloud", 106, "2018-07-11 12:12:49"),
    (9,"Star", 106, "2017-07-05 10:01:53"),
    (11,"Star", 106, "2018-07-05 11:10:43");


insert into medication (name, lab, dosage)
values
    ("Doxilax", "Montfield Lab", "400mg"),
    ("Papilem", "Montfield Lab", "200mg"),
    ("Loxifell", "Manhattan Lab&Co", "100mg"),
    ("Catnuciti", "Manhattan Lab&Co", "100ml"),
    ("Catxim", "Manhattan Lab&Co", "50ml");


insert into procedures (name,VAT_owner,date_timestamp,num,
   description)
values
    ("Bella",101,"2018-02-17 13:12:20",1,"radiography to torax
       "),
    ("Lucy",102,"2018-02-20 09:20:18",1,"radiography to right
       leg"),
    ("Kika", 101, "2017-03-02 10:14:19",1,"blood test"),
    ("Lucy",102,"2018-02-20 09:20:18",2,"blood test"),
    ("Vitria",129,"2018-08-07 13:20:04",1,"radiography to wing
       "),
    ("Deutsch", 109, "2018-05-04 16:12:04",1,"blood test"),
    ("Kika", 101,"2017-03-02 10:14:19", 2, "radiography to
       torax"),
    ("Leo", 117, "2017-04-12 14:47:10", 1, "blood test"),
    ("Willy", 101, "2017-03-02 13:45:57",1,"biopsy test"),
    ("Willy", 101, "2017-03-02 13:45:57",2,"endoscopy"),
    ("Willy", 101, "2017-03-02 13:45:57",3,"urine test"),
    ("Kashey", 129, "2017-02-17 15:10:21",1,"blood test"),
    ("Kashey", 129, "2017-02-17 15:10:21",2,"urine test"),
    ("Ash", 103, "2018-04-12 14:34:32",1,"biopsy test"),
    ("Caju", 130, "2018-02-23 17:15:20",1,"right leg
       radiography"),
    ("Caju", 130, "2018-02-23 17:15:20",2,"ultrasonography"),
    ("Caju", 130, "2018-02-23 17:15:20",3,"stool test"),
    ("Benji", 104, "2017-09-16 14:01:45",1,"blood test"),
    ("Benji", 104, "2017-09-16 14:01:45",2,"urine test"),
    ("Star", 106, "2017-07-05 10:01:53",1,"blood test"),
    ("Star", 106, "2017-07-05 10:01:53",2,"urine test"),
    ("Cloud", 106, "2018-07-11 12:12:49",1,"blood test"),
    ("Cloud", 106, "2018-07-11 12:12:49",2,"urine test");
```

```
insert into performed (name,VAT_owner,date_timestamp,num,
   VAT_assistant)
values
   ("Bella",101,"2018-02-17 13:12:20",1,128),
   ("Lucy",102,"2018-02-20 09:20:18",1,127),
   ("Vitria", 129,"2018-08-07 13:20:04",1,127),
   ("Kika", 101,"2017-03-02 10:14:19", 2, 125),
   ("Willy", 101, "2017-03-02 13:45:57",1,125),
   ("Willy", 101, "2017-03-02 13:45:57",2,126),
   ("Willy", 101, "2017-03-02 13:45:57",3,127),
   ("Kashey", 129, "2017-02-17 15:10:21",1,128),
   ("Kashey", 129, "2017-02-17 15:10:21",2,130),
   ("Ash", 103, "2018-04-12 14:34:32",1,125),
   ("Caju", 130, "2018-02-23 17:15:20",1,128),
   ("Caju", 130, "2018-02-23 17:15:20",2,125),
   ("Benji", 104, "2017-09-16 14:01:45",1,125),
   ("Benji", 104, "2017-09-16 14:01:45",2,126),
   ("Star", 106, "2017-07-05 10:01:53",1,127),
   ("Star", 106, "2017-07-05 10:01:53",2,128),
   ("Cloud", 106, "2018-07-11 12:12:49",1,125),
   ("Cloud", 106, "2018-07-11 12:12:49",2,130);


insert into radiography (name,VAT_owner,date_timestamp,num,file
   )
values
   ("Bella",101,"2018-02-17 13:12:20",1,"dir1"),
   ("Lucy",102,"2018-02-20 09:20:18",1,"dir2"),
   ("Kika", 101,"2017-03-02 10:14:19", 2, "dir3"),
   ("Vitria", 129,"2018-08-07 13:20:04",1,"dir4"),
   ("Caju", 130, "2018-02-23 17:15:20",1,"dir5");


insert into test_procedure(name,VAT_owner,date_timestamp,num,
   type)
values
   ("Lucy",102,"2018-02-20 09:20:18",2,"blood test"),
   ("Deutsch", 109, "2018-05-04 16:12:04",1,"blood test"),
   ("Kika", 101, "2017-03-02 10:14:19",1,"blood test"),
   ("Leo", 117, "2017-04-12 14:47:10", 1, "blood test"),
   ("Willy", 101, "2017-03-02 13:45:57",3,"urine test"),
   ("Kashey", 129, "2017-02-17 15:10:21",1,"blood teste"),
   ("Kashey", 129, "2017-02-17 15:10:21",2,"urine test"),
   ("Benji", 104, "2017-09-16 14:01:45",1,"blood test"),
   ("Benji", 104, "2017-09-16 14:01:45",2,"urine test"),
   ("Star", 106, "2017-07-05 10:01:53",1,"blood test"),
   ("Star", 106, "2017-07-05 10:01:53",2,"urine test"),
   ("Cloud", 106, "2018-07-11 12:12:49",1,"blood test"),
   ("Cloud", 106, "2018-07-11 12:12:49",2,"urine test");


insert into indicator (name,reference_value,unit,description)
values
   ("Glucose",102,"miligrams","65-120 for canine species,
      70-120 for feline species. mg/dl of blood; for blood
      tests"),
   ("BUN",15,"miligrams","6-24 for canine species, 17-30 for
      feline species. mg/dl of blood; for blood tests"),
   ("creatinine",0.9,"miligrams","0.4-1.4 for canine species,
      0.6-1.6 for feline species. mg/dl of blood; for blood
      tests"),
   ("total protein",6.2,"grams","5.2-7.2 for canine species,
      5.3-7.2 for feline species. g/dl of blood; for blood
      tests"),
   ("cholesterol",120,"miligrams","110-314 for canine species,
       90-150 for feline species. mg/dl of blood; for blood
      tests"),
   ("hemoglobin",13,"grams","12.1-20.3 for canine species,
      9.3-15.9 for feline species. mg/dl of blood; for blood
      tests"),
   ("microalbumin",2.5,"miligrams","< 2.5 mg/dl of urine; for
      urine tests"),
   ("urine pH",6,null,"5.5-7; for urine tests"),
   ("blood pH",7.36, null,"7.32-7.34 for canine species,
      7.24-7.40 for feline species; for blood tests"),
   ("WBC",2, "/hpf","white blood cells found in urine; for
      urine tests; /hpf means per high power field");


insert into prescription (code,name,VAT_owner,date_timestamp,
   name_med, lab, dosage, regime)
values
   (12, "Deutsch", 109,"2018-05-04 16:12:04","Doxilax", "
      Montfield Lab", "400mg", "to be taken every 6h"),
   (4, "Deutsch", 109,"2018-05-04 16:12:04","Papilem", "
      Montfield Lab", "200mg", "to be taken every 8h,
      alternative days"),
   (2, "Kika", 101, "2017-03-02 10:14:19","Loxifell", "
      Manhattan Lab&Co", "100mg", "to be taken whenever
      necessary."),
   (2, "Kika", 101, "2017-03-02 10:14:19","Catnuciti", "
      Manhattan Lab&Co", "100ml", "to be taken whenever
      necessary."),
   (2, "Kika", 101, "2017-03-02 10:14:19", "Catxim", "
      Manhattan Lab&Co", "50ml", "to be taken whenever
      necessary."),
   (15,"Vitria", 129,"2018-08-07 13:20:04","Catxim", "
      Manhattan Lab&Co", "50ml", "to be taken whenever
      necessary."),
   (3,"Kashey", 129, "2017-02-17 15:10:21","Catxim", "
      Manhattan Lab&Co", "50ml","to be taken every 12h"),
   (1,"Willy", 101, "2017-03-02 13:45:57","Catnuciti", "
      Manhattan Lab&Co", "100ml","to be yaken whenever
      necessary");

insert into produced_indicator(name,VAT_owner,date_timestamp,
   num,indicator_name,value)
values
   ("Deutsch", 109, "2018-05-04 16:12:04",1,"creatinine",1.2),
   ("Willy", 101, "2017-03-02 13:45:57",3,"total protein",5.8)
      ,
   ("Kashey", 129, "2017-02-17 15:10:21",1,"creatinine",0.8),
   ("Kashey", 129, "2017-02-17 15:10:21",2,"WBC",3),
   ("Benji", 104, "2017-09-16 14:01:45",1,"Glucose",105),
   ("Benji", 104, "2017-09-16 14:01:45",2,"urine pH",6.5),
   ("Star", 106, "2017-07-05 10:01:53",1,"cholesterol",125),
   ("Star", 106, "2017-07-05 10:01:53",1,"microalbumin",3),
   ("Cloud", 106, "2018-07-11 12:12:49",1,"creatinine",1.8),
   ("Cloud", 106, "2018-07-11 12:12:49",1,"WBC",2.5);
```

3 The queries

3.1 Query 1

```
SELECT DISTINCT
    A.name AS animal_name,
    PO.name AS owner_name,
    A.species_name,
    A.age
FROM
    animal A,
    person PO,
    person PV,
    consult C
WHERE
    C.VAT_vet = PV.VAT AND PV.name = "John Smith" AND C.name =
       A.name AND C.VAT_owner = A.VAT AND C.VAT_owner = PO.VAT
```

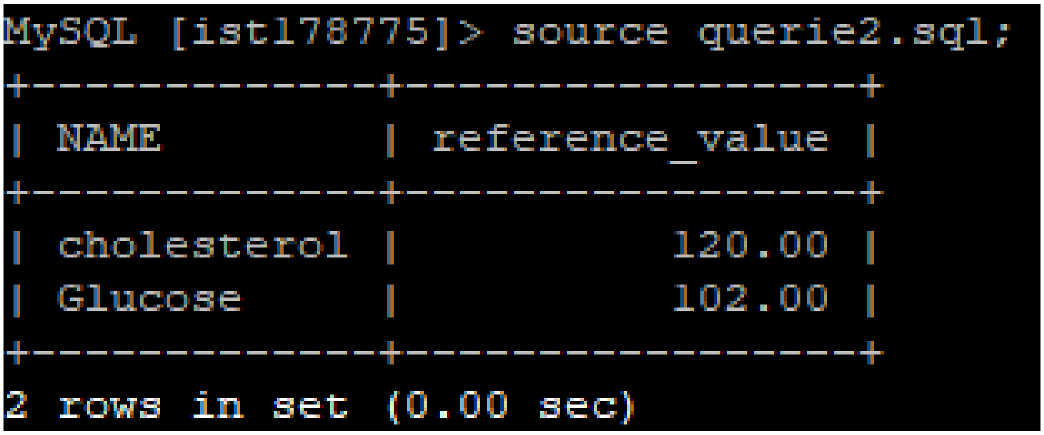


```
MySQL [ist178775]> source querie1.sql
+-------------+----------------+------------------------+------+
| animal_name | owner_name     | species_name           | age  |
+-------------+----------------+------------------------+------+
| Bochita     | John Smith     | chihuahua              |    8 |
| Caju        | Dexter         | beagle                 |    6 |
| Deutsch     | Derek Shepherd | bird                   |   16 |
| Eva         | Alex Karev     | red-eared slider turtle |    5 |
| Kashey      | John Smith     | chihuahua              |    7 |
| Kika        | Meredith Grey  | bulldog                |    3 |
| Leo         | Teddy Altman   | persian                |    4 |
| Willy       | Meredith Grey  | chihuahua              |   10 |
+-------------+----------------+------------------------+------+
8 rows in set (0.00 sec)
```

Figure 1: Result of query 1

3.2 Query 2

```
SELECT
    name,
    reference_value
FROM
    indicator
WHERE
    reference_value > 100 AND unit = "miligrams"
ORDER BY
    reference_value
DESC
```

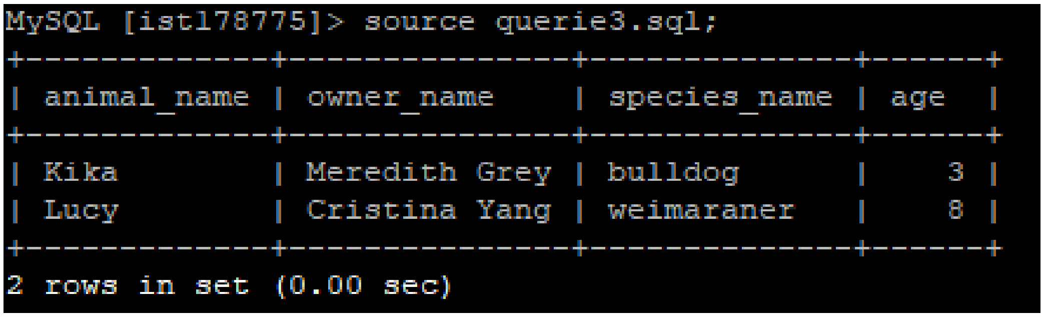


```
MySQL [ist178775]> source querie2.sql;
+-------------+-----------------+
| NAME        | reference_value |
+-------------+-----------------+
| cholesterol |          120.00 |
| Glucose     |          102.00 |
+-------------+-----------------+
2 rows in set (0.00 sec)
```

Figure 2: Result of query 2

3.3 Query 3

```
SELECT
    C.name AS animal_name,
    P.name AS owner_name,
    A.species_name,
    A.age
FROM
    consult C,
    person P,
    animal A
WHERE
    C.name = A.name AND C.VAT_owner = P.VAT AND C.weight > 30
        AND(
        C.o LIKE '%obese%' OR C.o LIKE '%obesity%'
    )
```

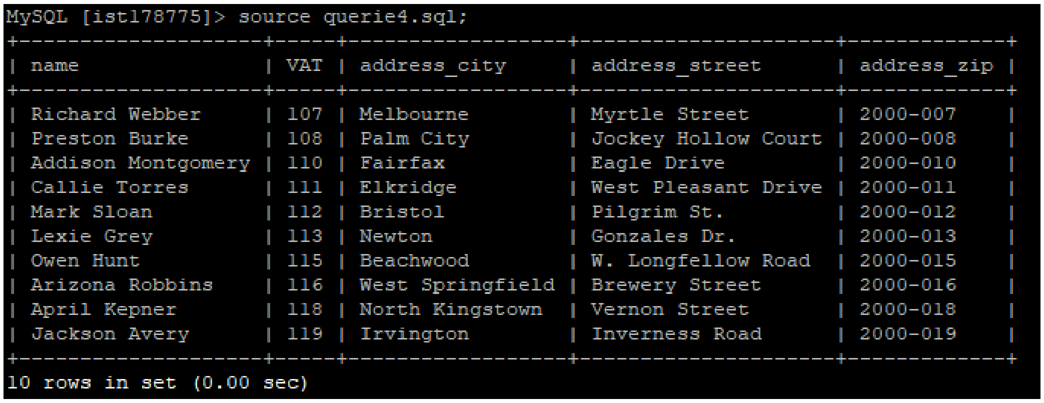


```
MySQL [ist178775]> source querie3.sql;
+-------------+---------------+--------------+------+
| animal_name | owner_name    | species_name | age  |
+-------------+---------------+--------------+------+
| Kika        | Meredith Grey | bulldog      |    3 |
| Lucy        | Cristina Yang | weimaraner   |    8 |
+-------------+---------------+--------------+------+
2 rows in set (0.00 sec)
```

Figure 3: Result of query 3

3.4 Query 4

```
SELECT
    P.name,
    P.VAT,
    P.address_city,
    P.address_street,
    P.address_zip
FROM
    person P,
    client Cl
WHERE
    P.VAT = Cl.VAT AND Cl.VAT NOT IN(
SELECT
    VAT
FROM
    animal
)
```

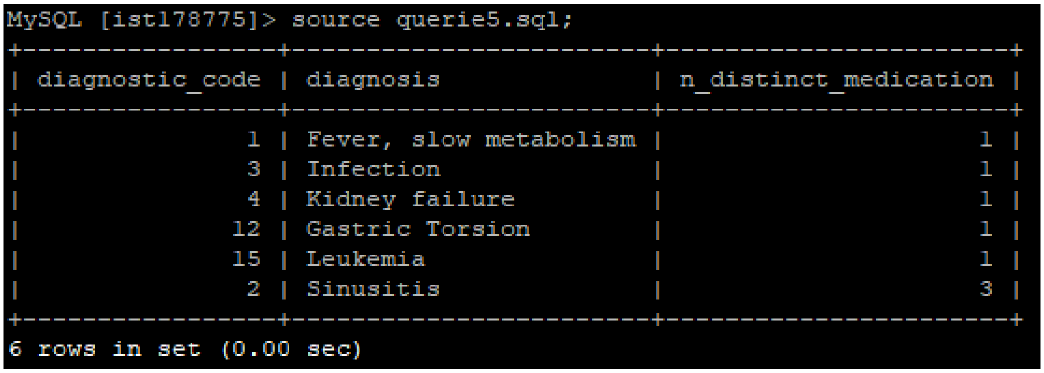


```
MySQL [ist178775]> source querie4.sql;
+--------------------+-----+------------------+---------------------+-------------+
| name               | VAT | address_city     | address_street      | address_zip |
+--------------------+-----+------------------+---------------------+-------------+
| Richard Webber     | 107 | Melbourne        | Myrtle Street       | 2000-007    |
| Preston Burke      | 108 | Palm City        | Jockey Hollow Court | 2000-008    |
| Addison Montgomery | 110 | Fairfax          | Eagle Drive         | 2000-010    |
| Callie Torres      | 111 | Elkridge         | West Pleasant Drive | 2000-011    |
| Mark Sloan         | 112 | Bristol          | Pilgrim St.         | 2000-012    |
| Lexie Grey         | 113 | Newton           | Gonzales Dr.        | 2000-013    |
| Owen Hunt          | 115 | Beachwood        | W. Longfellow Road  | 2000-015    |
| Arizona Robbins    | 116 | West Springfield | Brewery Street      | 2000-016    |
| April Kepner       | 118 | North Kingstown  | Vernon Street       | 2000-018    |
| Jackson Avery      | 119 | Irvington        | Inverness Road      | 2000-019    |
+--------------------+-----+------------------+---------------------+-------------+
10 rows in set (0.00 sec)
```

Figure 4: Result of query 4

3.5 Query 5

```
SELECT
    D.code AS diagnostic_code,
    D.name AS diagnosis,
    COUNT(DISTINCT P.name_med) AS n_distinct_medication
FROM
    prescription P,
    diagnosis_code D
WHERE
    P.code = D.code
GROUP BY
    P.code
ORDER BY
    COUNT(DISTINCT P.name_med) ASC
```



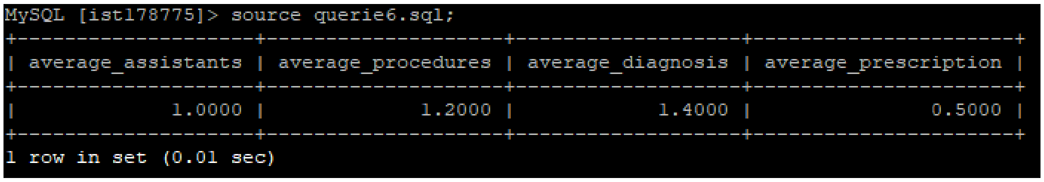
```
MySQL [ist178775]> source querie5.sql;
+-----------------+-----------------------+----------------------+
| diagnostic_code | diagnosis             | n_distinct_medication |
+-----------------+-----------------------+----------------------+
|               1 | Fever, slow metabolism |                    1 |
|               3 | Infection             |                    1 |
|               4 | Kidney failure        |                    1 |
|              12 | Gastric Torsion       |                    1 |
|              15 | Leukemia              |                    1 |
|               2 | Sinusitis             |                    3 |
+-----------------+-----------------------+----------------------+
6 rows in set (0.00 sec)
```

Figure 5: Result of query 5

3.6 Query 6

```
SELECT
    AVG(av.count_participation) average_assistants,
    AVG(av.count_procedures) average_procedures,
    AVG(av.count_diagnosis_code) average_diagnosis,
    AVG(av.count_prescription_name) average_prescription
FROM
    (
    SELECT
        t.consult_name av_consult_name,
        t.consult_time av_consult_time,
        COUNT(
            DISTINCT t.procedures_num,
            t.procedures_name,
            t.procedures_date_timestamp
        ) count_procedures,
        COUNT(
            DISTINCT t.diagnosis_code,
            t.diagnosis_name,
            t.diagnosis_VAT_owner,
            t.diagnosis_date_timestamp
        ) count_diagnosis_code,
        COUNT(
            DISTINCT prescription_med,
            pre_lab,
            pre_dosage
        ) count_prescription_name,
        COUNT(
            DISTINCT t.participation_name,
            t.participation_date_timestamp,
            t.participation_VAT_owner,
            t.participation_VAT_assistant
        ) count_participation
    FROM
        (
        SELECT
            c.name consult_name,
            pro.num procedures_num,
            pro.name procedures_name,
            pro.date_timestamp procedures_date_timestamp,
            cd.code diagnosis_code,
            cd.name diagnosis_name,
            cd.VAT_owner diagnosis_VAT_owner,
            cd.date_timestamp diagnosis_date_timestamp,
            pre.name_med prescription_med,
            pre.lab pre_lab,
            pre.dosage pre_dosage,
            c.date_timestamp consult_time,
            par.name participation_name,
            par.VAT_owner participation_VAT_owner,
            par.date_timestamp participation_date_timestamp,
            par.VAT_assistant participation_VAT_assistant
        FROM
            consult c
        LEFT JOIN procedures pro ON
            c.name = pro.name AND c.VAT_owner = pro.VAT_owner
               AND c.date_timestamp = pro.date_timestamp
        LEFT JOIN consult_diagnosis cd ON
            c.name = cd.name AND c.VAT_owner = cd.VAT_owner AND
               c.date_timestamp = cd.date_timestamp
        LEFT JOIN prescription pre ON
```

```
           c.name = pre.name AND c.VAT_owner = pre.VAT_owner
              AND c.date_timestamp = pre.date_timestamp
       LEFT JOIN participation par ON
           c.name = par.name AND c.VAT_owner = par.VAT_owner
              AND c.date_timestamp = par.date_timestamp
       WHERE
           YEAR(c.date_timestamp) = "2017"
    ) t
GROUP BY
    t.consult_name,
    t.consult_time
) av
```

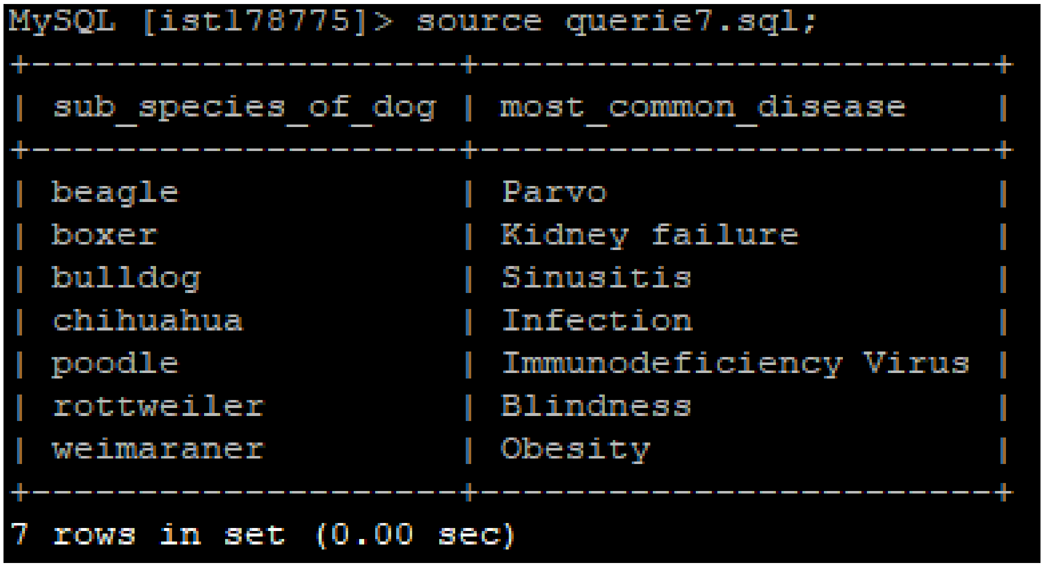


```
MySQL [ist178775]> source querie6.sql;
+--------------------+--------------------+-------------------+----------------------+
| average_assistants | average_procedures | average_diagnosis | average_prescription |
+--------------------+--------------------+-------------------+----------------------+
|             1.0000 |             1.2000 |            1.4000 |               0.5000 |
+--------------------+--------------------+-------------------+----------------------+
1 row in set (0.01 sec)
```

Figure 6: Result of query 6

3.7 Query 7

```
SELECT
    T.SN AS sub_species_of_dog,
    T.most_common_disease AS most_common_disease
FROM
    (
    SELECT
        A.species_name AS SN,
        D.code AS most_common_code,
        COUNT(D.code) AS occurrence,
        D.name as most_common_disease
    FROM
        diagnosis_code D,
        generalization_species GS,
        consult_diagnosis CD,
        animal A
    WHERE
        CD.name = A.name AND CD.code = D.code AND A.
           species_name = GS.name1 AND GS.name2 LIKE '%dog%'
    GROUP BY
        A.species_name,
        D.code
) T
WHERE NOT EXISTS
    (
    SELECT
        occurrence
    FROM
        (
        SELECT
            A.species_name AS SN,
            D.code AS most_common_code,
            COUNT(D.code) AS occurrence,
            D.name AS most_common_disease
        FROM
            diagnosis_code D,
            generalization_species GS,
            consult_diagnosis CD,
            animal A
        WHERE
            CD.name = A.name AND CD.code = D.code AND A.
               species_name = GS.name1 AND GS.name2 LIKE '%dog%'
        GROUP BY
            A.species_name,
            D.code
    ) T2
WHERE
    T2.occurrence > T.occurrence AND T2.SN = T.SN
)
GROUP BY
    T.SN
```

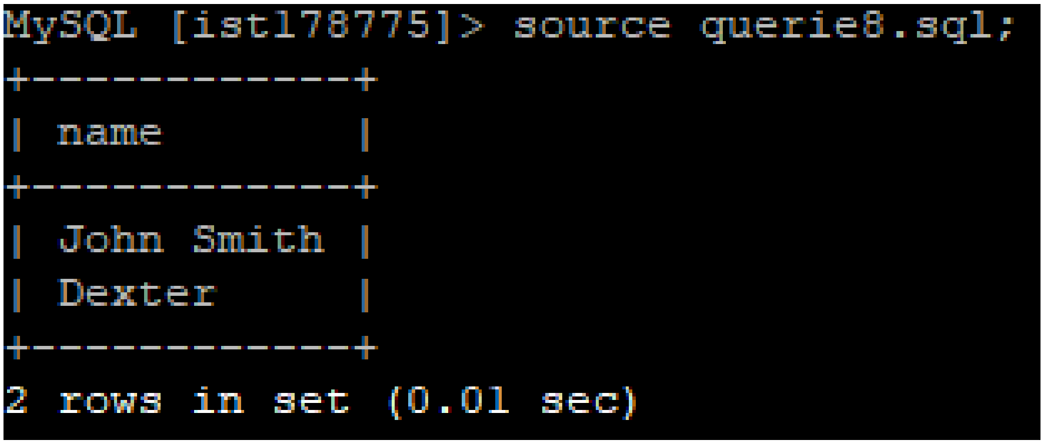


```
MySQL [ist178775]> source querie7.sql;
+--------------------+------------------------+
| sub_species_of_dog | most_common_disease    |
+--------------------+------------------------+
| beagle             | Parvo                  |
| boxer              | Kidney failure         |
| bulldog            | Sinusitis              |
| chihuahua          | Infection              |
| poodle             | Immunodeficiency Virus |
| rottweiler         | Blindness              |
| weimaraner         | Obesity                |
+--------------------+------------------------+
7 rows in set (0.00 sec)
```

Figure 7: Result of query 7

3.8 Query 8

```
SELECT DISTINCT
    P.name
FROM
    person P,
    client Cl,
    assistant A,
    veterinary V
WHERE
    (
        Cl.VAT = A.VAT OR Cl.VAT = V.VAT
    ) AND Cl.VAT = P.VAT
```



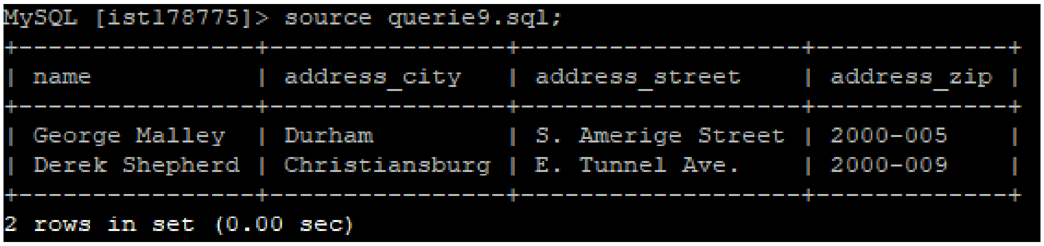
```
MySQL [ist178775]> source querie8.sql;
+-------------+
| name        |
+-------------+
| John Smith  |
| Dexter      |
+-------------+
2 rows in set (0.01 sec)
```

Figure 8: Result of query 8

3.9 Query 9

```
SELECT
    P.name,
    P.address_city,
    P.address_street,
    P.address_zip
FROM
```

```
    person P,
    client Cl,
    animal A
WHERE
    A.VAT = Cl.VAT AND Cl.VAT = P.vAT AND P.name NOT IN(
    SELECT
        P.name
    FROM
        animal A
    WHERE
        A.VAT = Cl.VAT AND Cl.VAT = P.VAT AND A.name NOT IN(
        SELECT
            A.name
        FROM
            animal A,
            person P,
            client Cl
        WHERE
            A.VAT = Cl.VAT AND Cl.VAT = P.VAT AND A.species_name
                LIKE "%bird%"
    )
)
```



```
MySQL [ist178775]> source querie9.sql;
+----------------+----------------+-------------------+-------------+
| name           | address_city   | address_street    | address_zip |
+----------------+----------------+-------------------+-------------+
| George Malley  | Durham         | S. Amerige Street | 2000-005    |
| Derek Shepherd | Christiansburg | E. Tunnel Ave.    | 2000-009    |
+----------------+----------------+-------------------+-------------+
2 rows in set (0.00 sec)
```

Figure 9: Result of query 9

4 Indexes

4.1 Indexes for query 1

4.1.1 Justification

An hash type index acts like a "dictionary" in the sense that the access is done directly. The search space is reduced because, given a "dictionary key", the search will be performed in the corresponding bucket (for which it points to) instead on the entire file records.

In our case, the "dictionary" key is "John Smith" and using an hash function we can access directly the corresponding bucket. The bucket contains all the records in which the name John Smith appears.

In summary, we have decided to use this type of index because it's preferable when queries involve direct comparison (=) and the performance is improved because the search space is smaller. If we have opted to use B+-Trees, the search space would be much bigger because it would require to transverse the entire Tree from root to leaf nodes and then search in the leaf nodes for the exact value.

For this query, the search is done in the *person* table, using *VAT* and *name* attributes. Since *VAT* is a primary key for this table, it's already indexed. Only the attribute *name* needs to be indexed to optimize the search of this query since it's done using *person.name*='John Smith'. Due to the fact that we search using a direct comparison (=), indexing is more efficient using hash rather than B+-Trees. The search on the other attributes of consult is also already indexed because the attributes correspond to primary keys.

4.1.2 SQL implementation instruction

```
CREATE INDEX person_name
    ON
    person(name)
    USING
    HASH;
```

4.2 Indexes for query 2

4.2.1 Justification

In this query the search is performed in the *reference value* and *unit* of indicators.

We have decided to use hash to index the *indicator unit*, for the reason explained above (direct comparison =).

Regarding the reference value, since the comparison involves an interval $[100, \infty[$, B^+-Tree index is preferable (when compared with hash indexing) because it has a more efficient performance in $>$,$<$, between or like operations.

Hash indexing does not perform well in these situations because it requires the search to jump between buckets to find all the values that belong to that interval, which may not be stored in the same bucket. This doesn't happen in the $B^+ - Tree$ because it's balanced and ordered. On the worst case scenario, hash indexing can search the entire database and return the results not ordered.

4.2.2 SQL implementation instructions

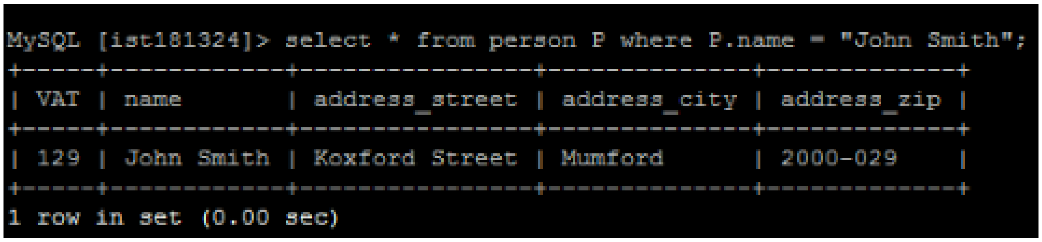
```
CREATE INDEX indicator_reference_value
    ON
    indicator(reference_value)
    USING
    BTREE;
```

```
CREATE INDEX indicator_unit
    ON
    indicator(unit)
    USING
    HASH;
```

5 Modifications (Update/deletion instructions)

5.1 Modification 1

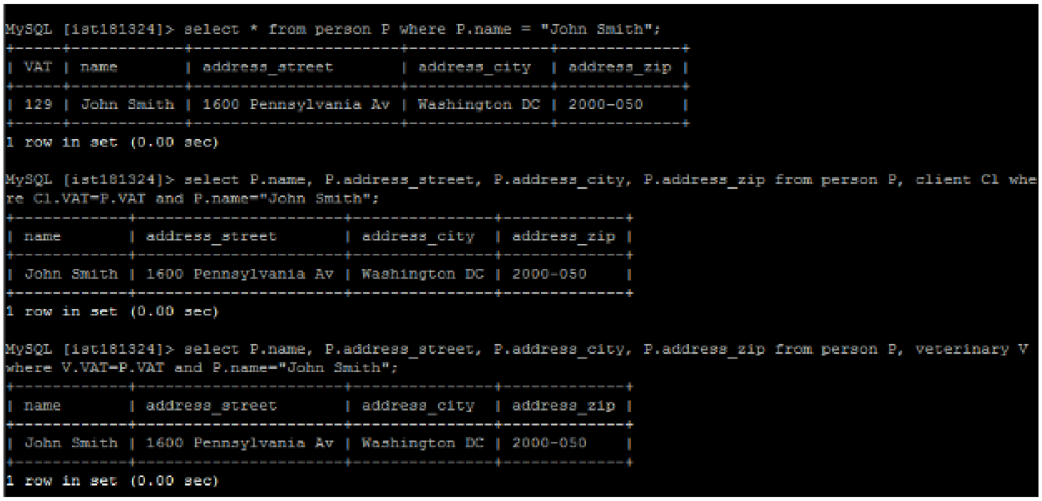
```
UPDATE
    person p,
    client
SET
    p.address_street = "1600 Pennsylvania Av",
    p.address_city = "Washington DC",
    p.address_zip = "2000-050"
WHERE
    p.name = "John Smith" AND p.VAT = client.VAT
```



```
MySQL [ist181324]> select * from person P where P.name = "John Smith";
+-----+------------+----------------+--------------+-------------+
| VAT | name       | address_street | address_city | address_zip |
+-----+------------+----------------+--------------+-------------+
| 129 | John Smith | Koxford Street | Mumford      | 2000-029    |
+-----+------------+----------------+--------------+-------------+
1 row in set (0.00 sec)
```

Figure 10: Before modification 1

Note: Because John Smith is both client and veterinary, it updates in both tables



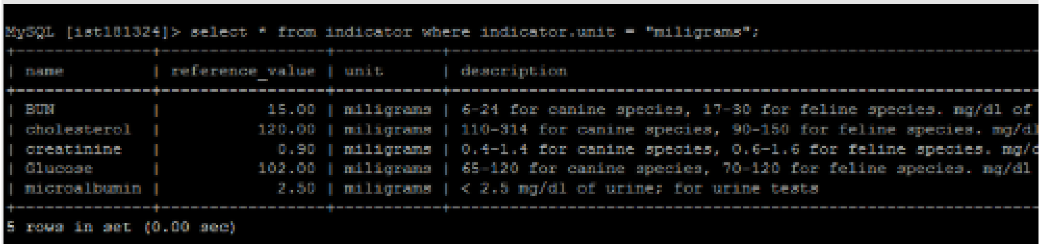
```
MySQL [ist181324]> select * from person P where P.name = "John Smith";
+-----+------------+----------------------+---------------+-------------+
| VAT | name       | address_street       | address_city  | address_zip |
+-----+------------+----------------------+---------------+-------------+
| 129 | John Smith | 1600 Pennsylvania Av | Washington DC | 2000-050    |
+-----+------------+----------------------+---------------+-------------+
1 row in set (0.00 sec)

MySQL [ist181324]> select P.name, P.address_street, P.address_city, P.address_zip from person P, client Cl whe
re Cl.VAT=P.VAT and P.name="John Smith";
+------------+----------------------+---------------+-------------+
| name       | address_street       | address_city  | address_zip |
+------------+----------------------+---------------+-------------+
| John Smith | 1600 Pennsylvania Av | Washington DC | 2000-050    |
+------------+----------------------+---------------+-------------+
1 row in set (0.00 sec)

MySQL [ist181324]> select P.name, P.address_street, P.address_city, P.address_zip from person P, veterinary V
where V.VAT=P.VAT and P.name="John Smith";
+------------+----------------------+---------------+-------------+
| name       | address_street       | address_city  | address_zip |
+------------+----------------------+---------------+-------------+
| John Smith | 1600 Pennsylvania Av | Washington DC | 2000-050    |
+------------+----------------------+---------------+-------------+
1 row in set (0.00 sec)
```

Figure 11: After modification 1

5.2 Modification 2

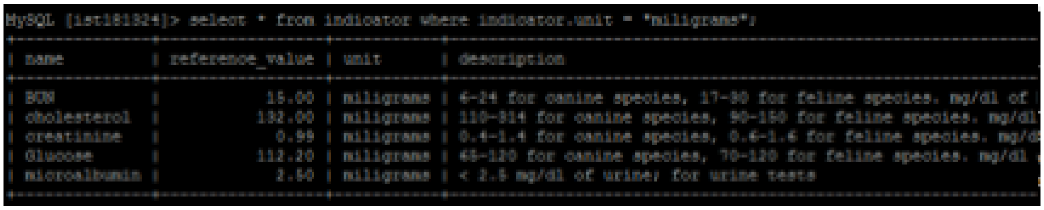
```
UPDATE
    indicator I,
    produced_indicator P,
    test_procedure T
SET
    I.reference_value = I.reference_value*1.1
WHERE
    I.unit = "miligrams" AND I.name = P.indicator_name AND P.
       name = T.name and P.VAT_owner = T.VAT_owner AND P.
       date_timestamp = T.date_timestamp AND P.num = T.num AND
       T.type LIKE '%blood%'
```

Note: From the indicators shown above, only *Cholesterol*, *creatinine* and *Glucose* are indicators used in blood tests, the rest are used in *urine tests*, that's the reason why only those are updated and the rest stays unchanged



```
MySQL [ist181324]> select * from indicator where indicator.unit = "miligrams";
+---------------+-----------------+-----------+--------------------------------------------------------------
| name          | reference_value | unit      | description
+---------------+-----------------+-----------+--------------------------------------------------------------
| BUN           |           15.00 | miligrams | 6-24 for canine species, 17-30 for feline species. mg/dl of
| cholesterol   |          120.00 | miligrams | 110-314 for canine species, 90-150 for feline species. mg/dl
| creatinine    |            0.90 | miligrams | 0.4-1.4 for canine species, 0.6-1.6 for feline species. mg/c
| Glucose       |          102.00 | miligrams | 65-120 for canine species, 70-120 for feline species. mg/dl
| microalbumin  |            2.50 | miligrams | < 2.5 mg/dl of urine; for urine tests
+---------------+-----------------+-----------+--------------------------------------------------------------
5 rows in set (0.00 sec)
```

Figure 12: Before modification 2

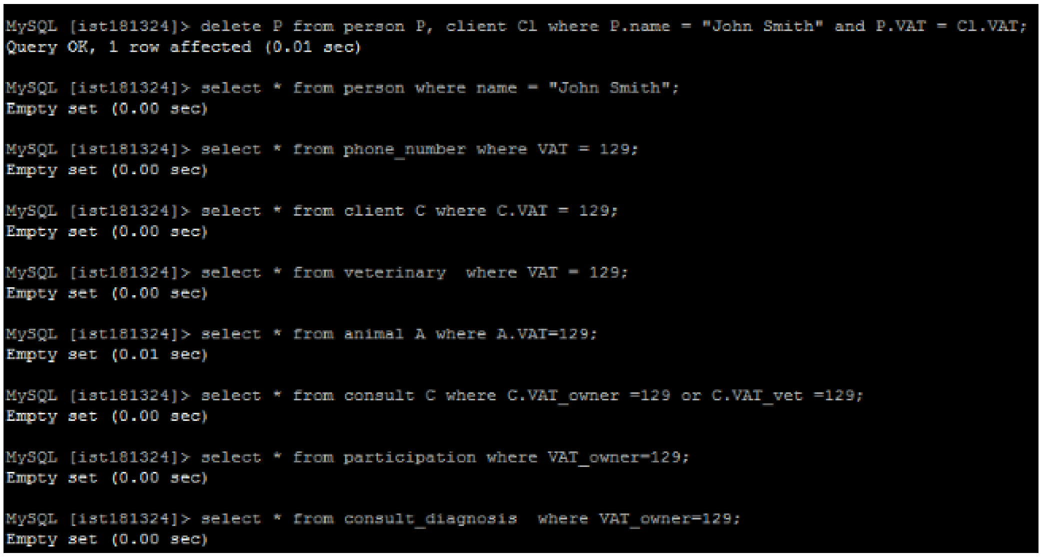


```
MySQL [ist181324]> select * from indicator where indicator.unit = "miligrams";
+---------------+-----------------+-----------+--------------------------------------------------------------
| name          | reference_value | unit      | description
+---------------+-----------------+-----------+--------------------------------------------------------------
| BUN           |           15.00 | miligrams | 6-24 for canine species, 17-30 for feline species. mg/dl of
| cholesterol   |          132.00 | miligrams | 110-314 for canine species, 90-150 for feline species. mg/dl
| creatinine    |            0.99 | miligrams | 0.4-1.4 for canine species, 0.6-1.6 for feline species. mg/d
| Glucose       |          112.20 | miligrams | 65-120 for canine species, 70-120 for feline species. mg/dl
| microalbumin  |            2.50 | miligrams | < 2.5 mg/dl of urine; for urine tests
+---------------+-----------------+-----------+--------------------------------------------------------------
```

Figure 13: After modification 2

5.3 Modification 3

```
delete P from person P, client Cl where P.name = "John Smith"
   and P.VAT = Cl.VAT
```



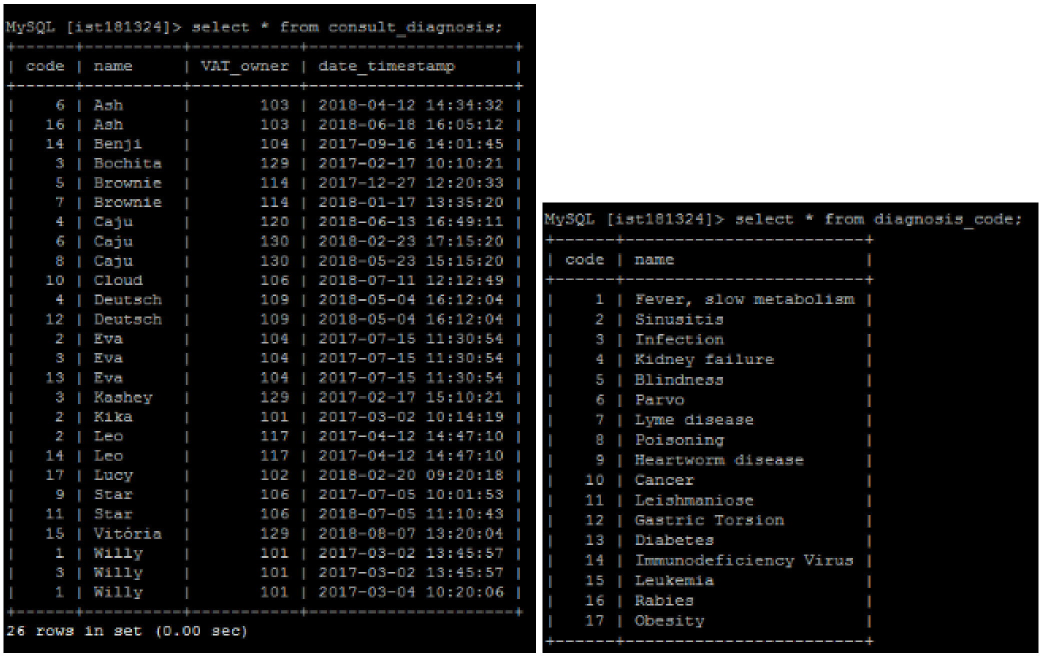
```
MySQL [ist181324]> delete P from person P, client Cl where P.name = "John Smith" and P.VAT = Cl.VAT;
Query OK, 1 row affected (0.01 sec)

MySQL [ist181324]> select * from person where name = "John Smith";
Empty set (0.00 sec)

MySQL [ist181324]> select * from phone_number where VAT = 129;
Empty set (0.00 sec)

MySQL [ist181324]> select * from client C where C.VAT = 129;
Empty set (0.00 sec)

MySQL [ist181324]> select * from veterinary  where VAT = 129;
Empty set (0.00 sec)

MySQL [ist181324]> select * from animal A where A.VAT=129;
Empty set (0.01 sec)

MySQL [ist181324]> select * from consult C where C.VAT_owner =129 or C.VAT_vet =129;
Empty set (0.00 sec)

MySQL [ist181324]> select * from participation where VAT_owner=129;
Empty set (0.00 sec)

MySQL [ist181324]> select * from consult_diagnosis  where VAT_owner=129;
Empty set (0.00 sec)
```

Figure 14: Modification 3

Note: We used on delete cascade command in every create table with a foreign key including a *VAT* number. The *VAT* number of customer John Smith (both client and doctor) is 129.

5.4 Modification 4

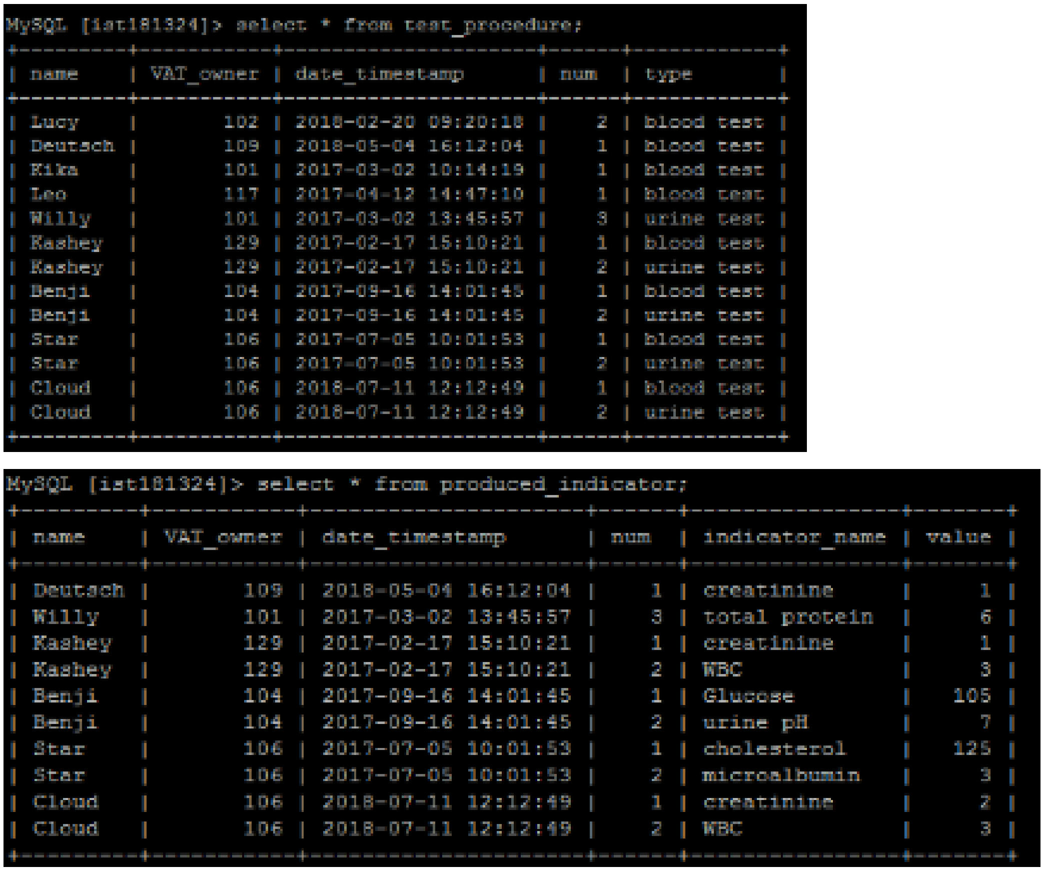
```
insert into diagnosis_code values ((select D.code from
   diagnosis_code D having D.code >= all (select D.code from
   diagnosis_code D))+1, "end-stage renal disease");

UPDATE
    consult_diagnosis CD1,
    consult_diagnosis CD2,
    diagnosis_code D1,
    diagnosis_code D2,
    indicator I,
    produced_indicator P,
    test_procedure T
SET
    CD1.code = D2.code
WHERE
    CD1.code = D1.code AND D1.name = "Kidney failure" AND D2.
       name = "end-stage renal disease" AND P.name = T.name AND
        P.VAT_owner = T.VAT_owner AND P.date_timestamp = T.
       date_timestamp AND P.num = T.num and T.type = "blood
       test" AND P.indicator_name = "creatinine" AND P.value >
       1.0 AND CD1.name = T.name AND CD1.VAT_owner = T.
       VAT_owner AND CD1.date_timestamp = T.date_timestamp
```



```
MySQL [ist181324]> select * from consult_diagnosis;
+------+----------+-----------+---------------------+
| code | name     | VAT_owner | date_timestamp      |
+------+----------+-----------+---------------------+
|    6 | Ash      |       103 | 2018-04-12 14:34:32 |
|   16 | Ash      |       103 | 2018-06-18 16:05:12 |
|   14 | Benji    |       104 | 2017-09-16 14:01:45 |
|    3 | Bochita  |       129 | 2017-02-17 10:10:21 |
|    5 | Brownie  |       114 | 2017-12-27 12:20:33 |
|    7 | Brownie  |       114 | 2018-01-17 13:35:20 |
|    4 | Caju     |       120 | 2018-06-13 16:49:11 |
|    6 | Caju     |       130 | 2018-02-23 17:15:20 |
|    8 | Caju     |       130 | 2018-05-23 15:15:20 |
|   10 | Cloud    |       106 | 2018-07-11 12:12:49 |
|    4 | Deutsch  |       109 | 2018-05-04 16:12:04 |
|   12 | Deutsch  |       109 | 2018-05-04 16:12:04 |
|    2 | Eva      |       104 | 2017-07-15 11:30:54 |
|    3 | Eva      |       104 | 2017-07-15 11:30:54 |
|   13 | Eva      |       104 | 2017-07-15 11:30:54 |
|    3 | Kashey   |       129 | 2017-02-17 15:10:21 |
|    2 | Kika     |       101 | 2017-03-02 10:14:19 |
|    2 | Leo      |       117 | 2017-04-12 14:47:10 |
|   14 | Leo      |       117 | 2017-04-12 14:47:10 |
|   17 | Lucy     |       102 | 2018-02-20 09:20:18 |
|    9 | Star     |       106 | 2017-07-05 10:01:53 |
|   11 | Star     |       106 | 2018-07-05 11:10:43 |
|   15 | Vitória  |       129 | 2018-08-07 13:20:04 |
|    1 | Willy    |       101 | 2017-03-02 13:45:57 |
|    3 | Willy    |       101 | 2017-03-02 13:45:57 |
|    1 | Willy    |       101 | 2017-03-04 10:20:06 |
+------+----------+-----------+---------------------+
26 rows in set (0.00 sec)
```

```
MySQL [ist181324]> select * from diagnosis_code;
+------+------------------------+
| code | name                   |
+------+------------------------+
|    1 | Fever, slow metabolism |
|    2 | Sinusitis              |
|    3 | Infection              |
|    4 | Kidney failure         |
|    5 | Blindness              |
|    6 | Parvo                  |
|    7 | Lyme disease           |
|    8 | Poisoning              |
|    9 | Heartworm disease      |
|   10 | Cancer                 |
|   11 | Leishmaniose           |
|   12 | Gastric Torsion        |
|   13 | Diabetes               |
|   14 | Immunodeficiency Virus |
|   15 | Leukemia               |
|   16 | Rabies                 |
|   17 | Obesity                |
+------+------------------------+
```

Figure 15: Before modification 4

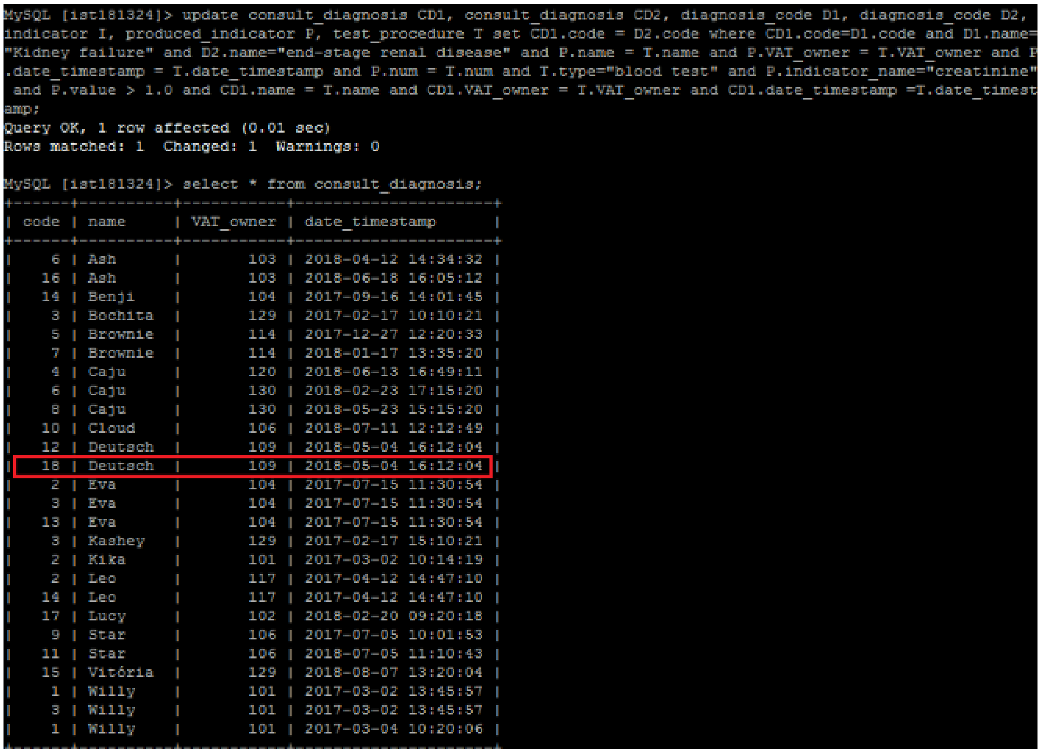


```
MySQL [ist181324]> select * from test_procedure;
+---------+-----------+---------------------+------+------------+
| name    | VAT_owner | date_timestamp      | num  | type       |
+---------+-----------+---------------------+------+------------+
| Lucy    |       102 | 2018-02-20 09:20:18 |    2 | blood test |
| Deutsch |       109 | 2018-05-04 16:12:04 |    1 | blood test |
| Kika    |       101 | 2017-03-02 10:14:19 |    1 | blood test |
| Leo     |       117 | 2017-04-12 14:47:10 |    1 | blood test |
| Willy   |       101 | 2017-03-02 13:45:57 |    3 | urine test |
| Kashey  |       129 | 2017-02-17 15:10:21 |    1 | blood test |
| Kashey  |       129 | 2017-02-17 15:10:21 |    2 | urine test |
| Benji   |       104 | 2017-09-16 14:01:45 |    1 | blood test |
| Benji   |       104 | 2017-09-16 14:01:45 |    2 | urine test |
| Star    |       106 | 2017-07-05 10:01:53 |    1 | blood test |
| Star    |       106 | 2017-07-05 10:01:53 |    2 | urine test |
| Cloud   |       106 | 2018-07-11 12:12:49 |    1 | blood test |
| Cloud   |       106 | 2018-07-11 12:12:49 |    2 | urine test |
+---------+-----------+---------------------+------+------------+
```

```
MySQL [ist181324]> select * from produced_indicator;
+---------+-----------+---------------------+------+----------------+-------+
| name    | VAT_owner | date_timestamp      | num  | indicator_name | value |
+---------+-----------+---------------------+------+----------------+-------+
| Deutsch |       109 | 2018-05-04 16:12:04 |    1 | creatinine     |     1 |
| Willy   |       101 | 2017-03-02 13:45:57 |    3 | total protein  |     6 |
| Kashey  |       129 | 2017-02-17 15:10:21 |    1 | creatinine     |     1 |
| Kashey  |       129 | 2017-02-17 15:10:21 |    2 | WBC            |     3 |
| Benji   |       104 | 2017-09-16 14:01:45 |    1 | Glucose        |   105 |
| Benji   |       104 | 2017-09-16 14:01:45 |    2 | urine pH       |     7 |
| Star    |       106 | 2017-07-05 10:01:53 |    1 | cholesterol    |   125 |
| Star    |       106 | 2017-07-05 10:01:53 |    2 | microalbumin   |     3 |
| Cloud   |       106 | 2018-07-11 12:12:49 |    1 | creatinine     |     2 |
| Cloud   |       106 | 2018-07-11 12:12:49 |    2 | WBC            |     3 |
+---------+-----------+---------------------+------+----------------+-------+
```

Figure 16: Before modification 4

Note: Among all the animals submitted to tests only *Deutsch*, *Kashey* and *Cloud* had tests which produced *creatinine* indicators. Between them, *Cloud* had a *urine test* so he was "discarded". Kashey had indeed a *blood test* but her *creatinine* level was below 1.0, thus not suffering any change. On the other hand, *Deutsch* had a blood test which produced a *creatinine* indicator with a value over 1.0. Furthermore, his diagnosis was replaced from *kidney failure* (code 4) to *end-stage renal disease* (code 18).



```
MySQL [ist181324]> update consult_diagnosis CD1, consult_diagnosis CD2, diagnosis_code D1, diagnosis_code D2,
indicator I, produced_indicator P, test_procedure T set CD1.code = D2.code where CD1.code=D1.code and D1.name=
"Kidney failure" and D2.name="end-stage renal disease" and P.name = T.name and P.VAT_owner = T.VAT_owner and P
.date_timestamp = T.date_timestamp and P.num = T.num and T.type="blood test" and P.indicator_name="creatinine"
 and P.value > 1.0 and CD1.name = T.name and CD1.VAT_owner = T.VAT_owner and CD1.date_timestamp =T.date_timest
amp;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MySQL [ist181324]> select * from consult_diagnosis;
+------+----------+-----------+---------------------+
| code | name     | VAT_owner | date_timestamp      |
+------+----------+-----------+---------------------+
|    6 | Ash      |       103 | 2018-04-12 14:34:32 |
|   16 | Ash      |       103 | 2018-06-18 16:05:12 |
|   14 | Benji    |       104 | 2017-09-16 14:01:45 |
|    3 | Bochita  |       129 | 2017-02-17 10:10:21 |
|    5 | Brownie  |       114 | 2017-12-27 12:20:33 |
|    7 | Brownie  |       114 | 2018-01-17 13:35:20 |
|    4 | Caju     |       120 | 2018-06-13 16:49:11 |
|    6 | Caju     |       130 | 2018-02-23 17:15:20 |
|    8 | Caju     |       130 | 2018-05-23 15:15:20 |
|   10 | Cloud    |       106 | 2018-07-11 12:12:49 |
|   12 | Deutsch  |       109 | 2018-05-04 16:12:04 |
|   18 | Deutsch  |       109 | 2018-05-04 16:12:04 |
|    2 | Eva      |       104 | 2017-07-15 11:30:54 |
|    3 | Eva      |       104 | 2017-07-15 11:30:54 |
|   13 | Eva      |       104 | 2017-07-15 11:30:54 |
|    3 | Kashey   |       129 | 2017-02-17 15:10:21 |
|    2 | Kika     |       101 | 2017-03-02 10:14:19 |
|    2 | Leo      |       117 | 2017-04-12 14:47:10 |
|   14 | Leo      |       117 | 2017-04-12 14:47:10 |
|   17 | Lucy     |       102 | 2018-02-20 09:20:18 |
|    9 | Star     |       106 | 2017-07-05 10:01:53 |
|   11 | Star     |       106 | 2018-07-05 11:10:43 |
|   15 | Vitória  |       129 | 2018-08-07 13:20:04 |
|    1 | Willy    |       101 | 2017-03-02 13:45:57 |
|    3 | Willy    |       101 | 2017-03-02 13:45:57 |
|    1 | Willy    |       101 | 2017-03-04 10:20:06 |
+------+----------+-----------+---------------------+
```

Figure 17: After modification 4

6 Creating views

6.1 View 1

```
CREATE OR REPLACE VIEW dim_date(
   date_timestamp,
   DAY,
   MONTH,
   YEAR
) AS SELECT
   C.date_timestamp,
   DAY(C.date_timestamp),
   MONTH(C.date_timestamp),
   YEAR(C.date_timestamp)
FROM
```

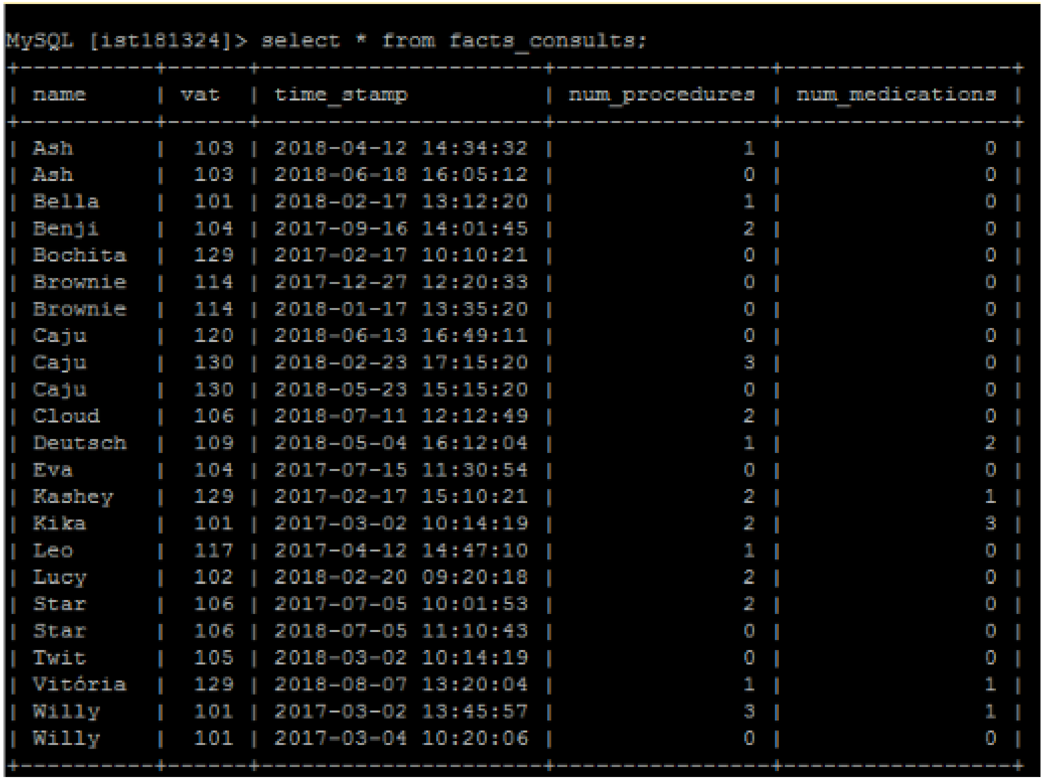
```
    consult C
```

6.2 View 2

```
CREATE OR REPLACE VIEW dim_animal(
    animal_name,
    animal_vat,
    species,
    age
) AS SELECT
    A.name,
    A.VAT,
    A.species_name,
    A.age
FROM
    animal A
```

6.3 View 3

```
CREATE OR REPLACE VIEW facts_consults(
    NAME,
    vat,
    time_stamp,
    num_procedures,
    num_medications
) AS SELECT
    da.animal_name,
    da.animal_vat,
    dd.date_timestamp,
    COUNT(
        DISTINCT da.animal_name,
        da.animal_vat,
        dd.date_timestamp,
        pro.num
    ),
    COUNT(
        DISTINCT da.animal_name,
        da.animal_vat,
        dd.date_timestamp,
        pre.name_med,
        pre.lab
    )
FROM
    consult c
LEFT JOIN dim_date dd ON
    dd.date_timestamp = c.date_timestamp
LEFT JOIN dim_animal da ON
    da.animal_name = c.name AND da.animal_vat = c.VAT_owner
LEFT JOIN procedures pro ON
    da.animal_name = pro.name AND pro.date_timestamp = dd.
       date_timestamp AND da.animal_vat = pro.VAT_owner
LEFT JOIN prescription pre ON
    da.animal_name = pre.name AND dd.date_timestamp = pre.
       date_timestamp AND da.animal_vat = pre.VAT_owner
GROUP BY
    da.animal_name,
    da.animal_vat,
    dd.date_timestamp
```



```
MySQL [ist181324]> select * from facts_consults;
+----------+-----+---------------------+----------------+-----------------+
| name     | vat | time_stamp          | num_procedures | num_medications |
+----------+-----+---------------------+----------------+-----------------+
| Ash      | 103 | 2018-04-12 14:34:32 |              1 |               0 |
| Ash      | 103 | 2018-06-18 16:05:12 |              0 |               0 |
| Bella    | 101 | 2018-02-17 13:12:20 |              1 |               0 |
| Benji    | 104 | 2017-09-16 14:01:45 |              2 |               0 |
| Bochita  | 129 | 2017-02-17 10:10:21 |              0 |               0 |
| Brownie  | 114 | 2017-12-27 12:20:33 |              0 |               0 |
| Brownie  | 114 | 2018-01-17 13:35:20 |              0 |               0 |
| Caju     | 120 | 2018-06-13 16:49:11 |              0 |               0 |
| Caju     | 130 | 2018-02-23 17:15:20 |              3 |               0 |
| Caju     | 130 | 2018-05-23 15:15:20 |              0 |               0 |
| Cloud    | 106 | 2018-07-11 12:12:49 |              2 |               0 |
| Deutsch  | 109 | 2018-05-04 16:12:04 |              1 |               2 |
| Eva      | 104 | 2017-07-15 11:30:54 |              0 |               0 |
| Kashey   | 129 | 2017-02-17 15:10:21 |              2 |               1 |
| Kika     | 101 | 2017-03-02 10:14:19 |              2 |               3 |
| Leo      | 117 | 2017-04-12 14:47:10 |              1 |               0 |
| Lucy     | 102 | 2018-02-20 09:20:18 |              2 |               0 |
| Star     | 106 | 2017-07-05 10:01:53 |              2 |               0 |
| Star     | 106 | 2018-07-05 11:10:43 |              0 |               0 |
| Twit     | 105 | 2018-03-02 10:14:19 |              0 |               0 |
| Vitória  | 129 | 2018-08-07 13:20:04 |              1 |               1 |
| Willy    | 101 | 2017-03-02 13:45:57 |              3 |               1 |
| Willy    | 101 | 2017-03-04 10:20:06 |              0 |               0 |
+----------+-----+---------------------+----------------+-----------------+
```

Figure 18: Select view ”facts_consult”