SIBD PROJECT

ASSIGNMENT 2: IMPLEMENTING THE DATABASE

COURSE: INFORMATION SYSTEMS AND DATABASES
FACULTY: BRUNO MARTINS

MIGUEL RODRIGUES Nº76176
HENRIQUE PIEDADE Nº75546
RAFAEL VILLAREJO Nº91712
GROUP 63

9/11/2018

1 Database Creation

1.1 Creation Script

```
create table person
  (VAT varchar(15),
   name  varchar(150) not null,
   address_street varchar(150) not null,
   address_city varchar(50) not null,
   address_zip varchar (20) not null,
   primary key(VAT));

create table phone_number
  (phone varchar(20),
  VAT varchar(15),
  primary key(VAT,phone),
  foreign key(VAT) references person(VAT)
      on delete cascade on update cascade);

create table client
  (VAT varchar(15),
  primary key(VAT),
  foreign key(VAT) references person(VAT)
      on delete cascade on update cascade);

create table veterinary
  (VAT varchar(15),
  specialization varchar(30) not null,
  bio varchar(500) not null,
  primary key(VAT),
  foreign key(VAT) references person(VAT)
      on delete cascade on update cascade);

create table assistant
  (VAT varchar(15),
  primary key(VAT),
  foreign key(VAT) references person(VAT)
      on delete cascade on update cascade);

create table species
  (name varchar(50),
  description varchar(500) not null,
  primary key(name));

create table generalization_species
  (name1 varchar(50),
  name2 varchar(50),
  primary key(name1),
  foreign key(name1) references
      species(name) on delete cascade on
      update cascade,
  foreign key(name2) references
      species(name) on delete cascade on
      update cascade);

create table animal
  (name varchar(30),
  VAT varchar(15),
  species_name varchar(50),
  gender varchar(10) not null,
  colour varchar(25) not null,
  birth_year year not null,
  age int not null,
  primary key(name, VAT),
  foreign key(VAT) references client(VAT)
      on delete cascade on update cascade,
  foreign key(species_name) references
      species(name) on delete cascade on
      update cascade);

create table consult
  (name varchar(30),
  VAT_owner varchar(15),
  date_timestamp timestamp,
  s varchar(200),
  o varchar(200),
  a varchar(200),
  p varchar(200),
  VAT_client varchar(15),
  VAT_vet varchar(15),
  weight float not null check (weight >0),
  primary key(name, VAT_owner,
      date_timestamp),
  foreign key(name, VAT_owner) references
      animal(name,VAT) on delete cascade on
      update cascade,
  foreign key(VAT_client) references
      client(VAT) on delete cascade on
      update cascade,
  foreign key(VAT_vet) references
      veterinary(VAT) on delete cascade on
      update cascade);

create table participation
  (name varchar(30),
  VAT_owner varchar(15),
  date_timestamp timestamp,
  VAT_assistant varchar(15),
  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 on
      update cascade,
  foreign key(VAT_assistant) references
      assistant(VAT) on delete cascade on
      update cascade);
```

```
create table diagnosis_code
  (code varchar(15),
  name varchar(50) not null,
  primary key(code));

create table consult_diagnosis
  (code varchar(15),
  name varchar(30),
  VAT_owner varchar(15),
  date_timestamp timestamp,
  primary
      key(code,name,VAT_owner,date_timestamp),
  foreign key(code) references
      diagnosis_code(code) on delete
      cascade on update cascade,
  foreign
      key(name,VAT_owner,date_timestamp)
      references
      consult(name,VAT_owner,date_timestamp)
      on delete cascade on update cascade);

create table medication
  (name varchar(30),
  lab varchar(50),
  dosage varchar(15),
  primary key(name, lab, dosage));

create table prescription
  (code varchar(15),
  name varchar(30),
  VAT_owner varchar(15),
  date_timestamp timestamp,
  name_med varchar(30),
  lab varchar(50),
  dosage varchar(15),
  regime varchar(50) not null,
  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) on delete cascade on update
      cascade);

create table indicator
  (name varchar(20),
  reference_value float,
  units varchar(15) not null,
  description varchar(200) not null,
  primary key(name));

create table _procedure
  (name varchar(30),
  VAT_owner varchar(15),
  date_timestamp timestamp,
  num int not null,
  description varchar(300),
  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 on update cascade);

create table performed
  (name varchar(30),
  VAT_owner varchar(15),
  date_timestamp timestamp,
  num int not null,
  VAT_assistant varchar(15),
  primary key(name, VAT_owner,
      date_timestamp, num, VAT_assistant),
  foreign key(name, VAT_owner,
      date_timestamp, num) references
      _procedure(name,
      VAT_owner,date_timestamp, num) on
      delete cascade on update cascade,
  foreign key(VAT_assistant) references
      assistant(VAT) on delete cascade on
      update cascade);

create table radiography
  (name varchar(30),
  VAT_owner varchar(15),
  date_timestamp timestamp,
  num int not null,
  file varchar(100) not null,
  primary key(name, VAT_owner,
      date_timestamp, num),
  foreign key(name, VAT_owner,
      date_timestamp, num) references
      _procedure(name, VAT_owner,
      date_timestamp, num) on delete
      cascade on update cascade);

create table test_procedure
  (name varchar(30),
  VAT_owner varchar(15),
  date_timestamp timestamp,
  num int not null,
  type varchar(25) not null check (type in
      ('blood','urine')),
  primary key(name,
      VAT_owner,date_timestamp, num),
  foreign key(name,
      VAT_owner,date_timestamp, num)
      references _procedure(name,
      VAT_owner, date_timestamp, num) on
      delete cascade on update cascade);
```

```
create table produced_indicator
  (name varchar(30),
  VAT_owner varchar(15),
  date_timestamp timestamp,
  num int not null,
  indicator_name varchar(20),
  value float,
  primary key(name, VAT_owner, date_timestamp, num, indicator_name),
  foreign key(name, VAT_owner, date_timestamp, num) references test_procedure(name,
      VAT_owner,date_timestamp,num) on delete cascade on update cascade,
  foreign key(indicator_name) references indicator(name) on delete cascade on update cascade);
```

Only two of the constraints present in the assignment are done in this table creation (weight > 0 and test type being "blood" or "urine"). The rest are done using triggers, which are not yet used. The types chosen are ones that adequate to the needed information, without having more information. The same applies for having attributes that may be NULL or not. All foreign keys are to be updated on deletion or on updating, so that our database stays coherent when either a deletion or an update occur.

1.2 Population Script

In order to complete the different requested queries, instructions, indexes, and views, the following population was added to the database

```
insert into person(VAT,name,address_street,address_city,address_zip)
values('12345','John Smith', 'Street1', 'City1','1234-123'),
  ('54321','John Smith', 'Street 150','Citty2','1212-211'),
  ('12346','Johana Smith', 'Street2', 'City2','1234-991'),
  ('11111','John Johnes', 'Street2', 'City2','1234-981'),
  ('10010','Peter Albert', 'Fish Street', 'Old City','1441-191'),
  ('10011','Albertina Peterson', 'Good Street', 'Old City','1440-300'),
  ('19999','Edna Jules','Old Street', 'New City','0101-101'),
  ('10101','Jack Potter', 'Big Street', 'New City','3332-223');

insert into phone_number(VAT,phone)
values('12345', '911231231'),
  ('12346', '555-702-222'),
  ('12346', '555-702-224'),
  ('11111', '959595958'),
  ('10010', '718080800'),
  ('10101','+351900099098');

insert into client(VAT)
values('12346'),
  ('10011'),
  ('54321'),
  ('19999'),
  ('11111');

insert into assistant(VAT)
values('10101'),
  ('10010');
```

```
insert into veterinary(VAT,specialization, bio)
values('12345','Bird ophthalmology','Single, nice, good with birds, bat with cats. Graduated one
    year ago.'),
  ('10011','Dog therapy','Experient doctor, graduated in 1994, 2 kids, 2 cats, 3 dogs, no
      husband.');

insert into species(name, description)
values('mammal','any vertebrate of the class Mammalia, having the body more or less covered with
    hair, nourishing the young with milk from the mammary glands, and, with the exception of the
    egg-laying monotremes, giving birth to live young.'),
  ('reptile', 'any cold-blooded vertebrate of the class Reptilia, comprising the turtles, snakes,
      lizards, crocodilians, amphisbaenians, tuatara, and various extinct members including the
      dinosaurs.'),
  ('bird',' any warm-blooded vertebrate of the class Aves, having a body covered with feathers,
      forelimbs modified into wings, scaly legs, a beak, and no teeth, and bearing young in a
      hard-shelled egg.'),
  ('fish','any of various cold-blooded, aquatic vertebrates, having gills, commonly fins, and
      typically an elongated body covered with scales.'),
  ('dog','any carnivore of the dog family Canidae, having prominent canine teeth and, in the wild
      state, a long and slender muzzle, a deep-chested muscular body, a bushy tail, and large,
      erect ears'),
  ('cat','any of several carnivores of the family Felidae, as the lion, tiger, leopard or jaguar,
      etc'),
  ('snake','any of numerous limbless, scaly, elongate reptiles of the suborder Serpentes,
      comprising venomous and nonvenomous species inhabiting tropical and temperate areas.'),
  ('goldfish','a small, usually yellow or orange fish, Carassius auratus, of the carp family,
      native to China, bred in many varieties and often kept in fishbowls and pools.'),
  ('redbird','a bird that is red'),
  ('husky', 'looks like a wolf'),
  ('labrador', 'fun dog'),
  ('bluebird','a bird that is blue'),
  ('betta fish',' elegant tropical freshwater fish wiht different birght colors that come to the
      surface often to suck air from outside the water.');

insert into generalization_species(name1,name2)
  values('dog', 'mammal'),
  ('cat','mammal'),
  ('snake','reptile'),
  ('goldfish','fish'),
  ('betta fish','fish'),
  ('bluebird','bird'),
  ('labrador', 'dog'),
  ('husky', 'dog'),
  ('redbird','bird');

insert into animal(name, VAT,species_name, colour, gender, birth_year, age)
  values('Quim','12346','bluebird','blue','male','2013','5'),
  ('Joa','12346','labrador','black','female','2015','3'),
  ('Mari','10011','cat','grey','female','2008','10'),
  ('Garfield','10011','cat','orange','male','2000','18'),
  ('Bella','19999','bluebird','blue','female','2012','6'),
  ('Joa','10011','husky','blackish','female','2016','2'),
  ('Ruca','54321','labrador','golden','male','2009','9'),
  ('Ed','19999','redbird','red','male','2015','3');
```

```
insert into consult(name,VAT_owner,date_timestamp,s,o,a,p,VAT_client, VAT_vet,weight)
  values('Quim','12346','2014-10-09 18:19:20','looks good', 'has
      wings','NULL','NULL','12346','12345','1.2'),
  ('Bella','19999','2013-11-11 11:11:11','NULL', 'NULL','NULL','treat her','19999','12345','1.1'),
  ('Joa','12346','2015-01-09 10:10:20','NULL','NULL','has food
      posioning','NULL','12346','10011','20.73'),
  ('Joa','12346','2017-01-09 10:10:20','NULL','getting obese','has food
      posioning','NULL','12346','12345','31'),
  ('Joa','10011','2017-01-01 09:10:20','NULL','NULL','NULL','NULL','10011','12345','15'),
  ('Garfield','10011','2001-03-03 10-03-30','NULL','too fat','NULL','NULL','10011','12345','32'),
  ('Garfield','10011','2002-03-03 10-03-30','NULL','in danger of
      obesity','NULL','NULL','12346','12345','29'),
  ('Ruca','54321','2010-04-11 10:03:21','NULL','NULL','NULL','NULL','54321','10011','25'),
  ('Ruca','54321','2012-04-11 10:03:21','NULL','NULL','NULL','NULL','54321','10011','28'),
  ('Ruca','54321','2017-04-11 10:03:21','NULL','NULL','NULL','NULL','54321','10011','35');

insert into indicator(name, reference_value, units, description)
  values('indicator1','75','milligrams',"Blood"),
  ('indicator2','101','milligrams', 'Water'),
  ('indicator3','200','milligrams', 'CH34HCNSH'),
  ('indicator4','20','millilitres', 'very useful'),
  ('creatinine level','1.0','milligrams', 'renal stuff');

insert into participation(name, VAT_owner, date_timestamp, VAT_assistant)
  values('Quim','12346','2014-10-09 18:19:20','10101'),
  ('Bella','19999','2013-11-11 11:11:11','10101'),
  ('Joa','12346','2017-01-09 10:10:20','10010');

insert into diagnosis_code(code,name)
  values('C-01','diabetes'),
  ('C-02','kidney failure'),
  ('C-03','depression'),
  ('C-14','amnesia'),
  ('C-35','parvovirus');

insert into consult_diagnosis(code, name, VAT_owner,date_timestamp)
  values('C-01','Ruca','54321','2010-04-11 10:03:21'),
  ('C-01','Joa','12346','2017-01-09 10:10:20'),
  ('C-14','Joa','10011','2017-01-01 09:10:20'),
  ('C-14','Ruca','54321','2017-04-11 10:03:21'),
  ('C-35','Ruca','54321','2017-04-11 10:03:21'),
  ('C-02','Ruca','54321','2017-04-11 10:03:21'),
  ('C-02','Bella','19999', '2013-11-11 11:11:11');

insert into medication(name,lab,dosage)
  values('med1','lab2','50mg'),
  ('med2','lab2','100mg'),
  ('med2','lab2','400mg'),
  ('med25','lab90','100mg'),
  ('med1','lab90','200mg'),
  ('med12','lab1','250mg'),
  ('med43','lab0','40mg');
```

```
insert into prescription(code,name,VAT_owner,date_timestamp,name_med,lab,dosage,regime)
  values('C-01','Ruca','54321','2010-04-11 10:03:21','med2','lab2','400mg', "1x day"),
  ('C-01','Ruca','54321','2010-04-11 10:03:21','med2','lab2','100mg', "1x day"),
  ('C-02','Ruca','54321','2017-04-11 10:03:21','med12','lab1','250mg', "10x day"),
  ('C-14','Ruca','54321','2017-04-11 10:03:21','med43','lab0','40mg','3x day'),
  ('C-02','Bella','19999', '2013-11-11 11:11:11','med12','lab1','250mg','1x day 1x night'),
  ('C-02','Bella','19999', '2013-11-11 11:11:11','med1','lab90','200mg','1x night'),
  ('C-02','Bella','19999', '2013-11-11 11:11:11','med2','lab2','400mg','1x day');

insert into _procedure(name, VAT_owner,date_timestamp,num,description)
  values('Ruca','54321',' 2012-04-11 10:03:21',150,'test'),
  ('Ruca','54321','2017-04-11 10:03:21',200,'test2'),
  ('Garfield','10011','2002-03-03 10-03-30',35,'testting'),
  ('Joa','12346','2017-01-09 10:10:20',450,'all good'),
  ('Joa','12346','2017-01-09 10:10:20',451, 'not so good');

insert into performed(name, VAT_owner,date_timestamp,num, VAT_assistant)
  values('Joa','12346','2017-01-09 10:10:20',451,'10101'),
  ('Joa','12346','2017-01-09 10:10:20',450,'10101'),
  ('Garfield','10011','2002-03-03 10-03-30',35,'10101'),
  ('Ruca','54321','2017-04-11 10:03:21',200,'10010');

insert into radiography(name, VAT_owner,date_timestamp,num,file)
  values('Joa','12346','2017-01-09 10:10:20',451,'path/path');

insert into test_procedure(name,VAT_owner,date_timestamp,num,type)
  values('Joa','12346','2017-01-09 10:10:20',450,'blood'),
  ('Garfield','10011','2002-03-03 10-03-30',35,'urine'),
  ('Ruca','54321','2017-04-11 10:03:21',200,'blood');

insert into produced_indicator(name,VAT_owner,date_timestamp,num,indicator_name,value)
  values('Joa','12346','2017-01-09 10:10:20',450,'indicator1', '80'),
  ('Ruca','54321','2017-04-11 10:03:21',200,'creatinine level', '1.1'),
  ('Garfield','10011','2002-03-03 10-03-30',35,'indicator4','25');
```

2 Database Queries

```
--1

select c.name as animal_name, p.name as
    owner_name, a.species_name, a.age
from consult c, animal a, person p
where p.VAT = c.VAT_owner
and a.VAT = p.VAT
and a.name =c.name
and c.VAT_vet in(
  select VAT
  from person
  where person.name = 'John Smith')
group by c.name, p.name;
```

```
--2

select i.name, i.reference_value
from indicator i
where i.units = 'milligrams'
and i.reference_value >100
order by i.reference_value desc;
```

```
--3
select a.name as animal_name, p.name as
    person_name, a.species_name, a.age
from animal a, person p
where p.VAT =a.VAT
and exists(
  select c.name, c.VAT_owner
  from consult c
  where c.o
  like '%obese%'
  and c.name = a.name
  and c.VAT_owner = a.VAT
  union
  select c2.name, c2.VAT_owner
  from consult c2
  where c2.o
  like '%obesity%'
  and c2.name = a.name
  and c2.VAT_owner=a.VAT)
and exists(
  select weight
  from consult c3
  where c3.weight >30
  and c3.name =a.name
  and c3.VAT_owner = a.VAT
  and c3.date_timestamp >=all(
    select date_timestamp
    from consult c4
    where c3.name =c4.name
    and c3.VAT_owner=c4.VAT_owner));
```

```
MySQL [ist176176]> select c.name as animal_name, p.name as owner_name,
    ->  a.species_name, a.age
    -> from consult c, animal a, person p
    -> where p.VAT = c.VAT_owner
    -> and a.VAT = p.VAT
    -> and a.name =c.name
    -> and c.VAT_vet in(
    ->   select VAT
    ->   from person
    ->   where person.name = 'John Smith')
    ->   group by c.name, p.name;
+-------------+--------------------+--------------+-----+
| animal_name | owner_name         | species_name | age |
+-------------+--------------------+--------------+-----+
| Bella       | Edna Jules         | bluebird     |   6 |
| Garfield    | Albertina Peterson | cat          |  18 |
| Joa         | Albertina Peterson | husky        |   2 |
| Joa         | Johana Smith       | labrador     |   3 |
| Quim        | Johana Smith       | bluebird     |   5 |
+-------------+--------------------+--------------+-----+
5 rows in set (0.00 sec)
```

Figure 1: Query 1.

```
MySQL [ist176176]> select i.name, i.reference_value
    -> from indicator i
    -> where i.units = 'milligrams'
    -> and i.reference_value >100
    -> order by i.reference_value desc;
+------------+-----------------+
| name       | reference_value |
+------------+-----------------+
| indicator3 |             200 |
| indicator2 |             101 |
+------------+-----------------+
2 rows in set (0.01 sec)
```

Figure 2: Query 2.

```
MySQL [ist176176]> select a.name as animal_name, p.name as person_name, a.species_name, a.age
    -> from animal a, person p
    -> where p.VAT =a.VAT
    -> and exists(
    ->   select c.name, c.VAT_owner
    ->   from consult c
    ->   where c.o
    ->   like '%obese%'
    ->   and c.name = a.name
    ->   and c.VAT_owner = a.VAT
    ->   union
    ->   select c2.name, c2.VAT_owner
    ->   from consult c2
    ->   where c2.o
    ->   like '%obesity%'
    ->   and c2.name = a.name
    ->   and c2.VAT_owner=a.VAT)
    -> and exists(
    ->   select weight
    ->   from consult c3
    ->   where c3.weight >30
    ->   and c3.name =a.name
    ->   and c3.VAT_owner = a.VAT
    ->   and c3.date_timestamp >=all(
    ->     select date_timestamp
    ->     from consult c4
    ->     where c3.name =c4.name
    ->     and c3.VAT_owner=c4.VAT_owner));
+-------------+--------------+--------------+-----+
| animal_name | person_name  | species_name | age |
+-------------+--------------+--------------+-----+
| Joa         | Johana Smith | labrador     |   3 |
+-------------+--------------+--------------+-----+
1 row in set (0.00 sec)
```

Figure 3: Query 3.

```sql
--4

select p.name, p.VAT, p.address_street,
    p.address_city
from person p
where p.VAT in (
  select VAT
  from client)
and p.VAT not in (
  select a.VAT
  from animal a);
```

```
MySQL [ist176176]> select p.name, p.VAT, p.address_street, p.address_city
    -> from person p
    -> where p.VAT in (
    ->   select VAT
    ->   from client)
    -> and p.VAT not in (
    ->   select a.VAT
    ->   from animal a);
+------------+-------+----------------+--------------+
| name       | VAT   | address_street | address_city |
+------------+-------+----------------+--------------+
| John Johnes | 11111 | Street2        | City2        |
+------------+-------+----------------+--------------+
1 row in set (0.00 sec)
```

Figure 4: Query 4.

```sql
--5
select p.code, dc.name, count( distinct
    p.name_med, p.dosage,p.lab) as count
from prescription p, diagnosis_code dc
where dc.code=p.code
group by dc.code
order by count asc;
```

```
MySQL [ist176176]> select p.code, dc.name,
    -> count( distinct p.name_med, p.dosage,p.lab) as count
    -> from prescription p, diagnosis_code dc
    -> where dc.code=p.code
    -> group by dc.code
    -> order by count asc;
+------+----------------+-------+
| code | name           | count |
+------+----------------+-------+
| C-14 | amnesia        |     1 |
| C-01 | diabetes       |     2 |
| C-02 | kidney failure |     3 |
+------+----------------+-------+
3 rows in set (0.00 sec)
```

Figure 5: Query 5.

```sql
--6
select NA.a/NC.c as avg_assistants,
    NP.p/NC.c as avg_procedures, ND.d/NC.c
    as avg_diagnosis, NPP.p/NC.c as
    avg_aprescriptions
from
(select count(*) as c from consult where
    year(date_timestamp)=2017) as NC,
(select count(*) as a from participation
    where year(date_timestamp)=2017) as NA,
(select count(*) as p from _procedure where
    year(date_timestamp)=2017) as NP,
(select count(*) as d from
    consult_diagnosis where
    year(date_timestamp)=2017) as ND,
(select count(*) as p from prescription
    where year(date_timestamp)=2017) as NPP;
```

```
MySQL [ist176176]> select NA.a/NC.c as avg_assistants,
    -> NP.p/NC.c as avg_procedures, ND.d/NC.c as avg_diagnosis,
    -> NPP.p/NC.c as avg_aprescriptions
    -> from
    -> (select count(*) as c from consult where year(date_timestamp)=2017) as NC,
    -> (select count(*) as a from participation where year(date_timestamp)=2017) as NA,
    -> (select count(*) as p from _procedure where year(date_timestamp)=2017) as NP,
    -> (select count(*) as d from consult_diagnosis where year(date_timestamp)=2017) as ND,
    -> (select count(*) as p from prescription where year(date_timestamp)=2017) as NPP;
+---------------+---------------+---------------+-------------------+
| avg_assistants | avg_procedures | avg_diagnosis | avg_aprescriptions |
+---------------+---------------+---------------+-------------------+
|         0.3333 |         1.0000 |         1.6667 |            0.3333 |
+---------------+---------------+---------------+-------------------+
1 row in set (0.00 sec)
```

Figure 6: Query 6.

```sql
--7

    select a.species_name, cd.code, dc.name
    from consult_diagnosis cd inner join
        animal a
    on cd.name = a.name and cd.VAT_owner =
        a.VAT,
    diagnosis_code dc
    where a.species_name in (
      select gs.name1
      from generalization_species gs
      where gs.name2= 'dog')
    and dc.code = cd.code
    group by a.species_name
    order by count(*) desc;
```

```
MySQL [ist176176]>     select a.species_name, cd.code, dc.name
    ->     from consult_diagnosis cd inner join animal a
    ->     on cd.name = a.name and cd.VAT_owner = a.VAT,
    ->     diagnosis_code dc
    ->     where a.species_name in (
    ->       select gs.name1
    ->       from generalization_species gs
    ->       where gs.name2= 'dog')
    ->     and dc.code = cd.code
    ->     group by a.species_name
    ->     order by count(*) desc;
+--------------+------+----------+
| species_name | code | name     |
+--------------+------+----------+
| labrador     | C-01 | diabetes |
| husky        | C-14 | amnesia  |
+--------------+------+----------+
2 rows in set (0.00 sec)
```

Figure 7: Query 7.

```
--8

select p.name
from person p, client c
where p.VAT = c.VAT
and c.VAT in(
   select VAT
   from veterinary
   union
   select VAT
   from assistant);
```

```
MySQL [ist176176]> select p.name
    -> from person p, client c
    -> where p.VAT = c.VAT
    -> and c.VAT in(
    ->    select VAT
    ->    from veterinary
    ->    union
    ->    select VAT
    ->    from assistant);
+--------------------+
| name               |
+--------------------+
| Albertina Peterson |
+--------------------+
1 row in set (0.00 sec)
```

Figure 8: Query 8.

```
--9

  select distinct p.name, p.address_street,
      p.address_city
  from person p, client inner join animal
    on client.VAT = animal.VAT
  where animal.VAT not in (
     select VAT
     from animal
     where species_name not like '%bird%')
  and p.VAT =client.VAT;
```

```
MySQL [ist176176]>  select distinct p.name, p.address_street, p.address_city
    ->   from person p, client inner join animal
    ->     on client.VAT = animal.VAT
    ->   where animal.VAT not in (
    ->      select VAT
    ->      from animal
    ->      where species_name not like '%bird%')
    ->   and p.VAT =client.VAT;
+-----------+----------------+--------------+
| name      | address_street | address_city |
+-----------+----------------+--------------+
| Edna Jules | Old Street     | New City     |
+-----------+----------------+--------------+
1 row in set (0.00 sec)
```

Figure 9: Query 9.

3 Database Indexes

In order to optimize the search for the first case, the following code in order to use the person's name to search. Although this index was used, it didn't produce better results on the query. If the query was formulated in a different way it could produce improvements by also indexing the VAT in the consult, as an example (but it also didn't on this query).

```
create index person_idx on person(name);
```

For the second query, one must include the units and the reference_value, so that the search for a value over 1.0 in milligrams is easier to find for the database, and as this is the only query, there's no need for more than that. This index reduced the number of rows needed to search (seen using the EXPLAIN function) from 5 to 2.

```
create index indicator_idx on indicator(units,reference_value);
```

4 Database Instructions

```
--1
update person
set address_street = 'New JSmith client street'
where person.name = 'John Smith'
and person.VAT in(select VAT from client);

--2
update indicator i
set i.reference_value = i.reference_value * 1.1
where i.name in (
  select distinct produced_indicator.indicator_name
  from produced_indicator, test_procedure
  where test_procedure.num = produced_indicator.num
  and test_procedure.type='blood');

--3
delete from person
where name ='John Smith'
and person.VAT in(
  select VAT
  from client);

--4
insert into diagnosis_code(code,name)
    values('C-150','end_stage renal disease');
update consult_diagnosis cd
set cd.code = 'C-150',
  cd.name=cd.name,
  cd.VAT_owner =cd.VAT_owner,
  cd.date_timestamp = cd.date_timestamp
  where cd.code = 'C-02'
  and (cd.name, cd.VAT_owner, cd.date_timestamp) in(
    select p.name, p.VAT_owner, p.date_timestamp
    from produced_indicator p
    where p.indicator_name = 'creatinine level'
    and p.value > 1.0);
```

5 Database Views

```
create view dim_date as
  select distinct date_timestamp, year(date_timestamp) as year, month(date_timestamp) as month,
      day(date_timestamp) as day
  from consult;

create view dim_animal as
  select name ,VAT,species_name, age
  from animal;


create view facts_consults as select d2.name, d2.vat, d1.date_timestamp,
  (select count(*)
  from _procedure as p
  where d2.name = p.name
  and d2.vat = p.VAT_owner
  and d1.date_timestamp = p.date_timestamp
  group by p.name, p.VAT_owner, p.date_timestamp) as nr_procedures,
  (select count(*)
  from prescription as p
  where d2.name = p.name
  and d2.vat = p.VAT_owner
  and d1.date_timestamp = p.date_timestamp
  group by p.name, p.VAT_owner, p.date_timestamp) as nr_medications
  from dim_animal as d2, dim_date as d1, consult c
  where d2.name = c.name
  and c.VAT_owner = d2.vat
  and c.date_timestamp = d1.date_timestamp
  group by c.name, c.VAT_owner, c.date_timestamp;
```