

# Лабораторная работа 13

---

## Задание

Создать базу знаний «Собственники», дополнив (и минимально изменив) базу знаний, хранящую знания (лаб. 13):

- «Телефонный справочник»: Фамилия, №тел, Адрес – структура (Город, Улица, №дома, №кв),
- «Автомобили»: Фамилия\_владельца, Марка, Цвет, Стоимость, и др.,
- «Вкладчики банков»: Фамилия, Банк, счет, сумма, др., знаниями о дополнительной собственности владельца. Преобразовать знания об автомобиле к форме знаний о собственности.

Вид собственности (кроме автомобиля):

- Строение, стоимость и другие его характеристики;
- Участок, стоимость и другие его характеристики;
- Водный\_транспорт, стоимость и другие его характеристики.

Описать и использовать вариантный домен: Собственность. Владелец может иметь, но только один объект каждого вида собственности (это касается и автомобиля), или не иметь некоторых видов собственности.

Используя конъюнктивное правило и разные формы задания одного вопроса (пояснять для какого №задания – какой вопрос), обеспечить возможность поиска:

1. Названий всех объектов собственности заданного субъекта,
2. Названий и стоимости всех объектов собственности заданного субъекта,
3. \* Разработать правило, позволяющее найти суммарную стоимость всех объектов собственности заданного субъекта.

Для 2-го пункт и одной фамилии составить таблицу, отражающую конкретный порядок работы системы, с объяснениями порядка работы и особенностей использования доменов (указать конкретные T1 и T2 и полную подстановку на каждом шаге)

## Код программы

```
include "lab13.inc"

domains
    lastname = symbol.
    phone = symbol.
    city, street = symbol.
    house_num, apart_num = integer.
    address_struct = address(city, street, house_num, apart_num).

    label, name, color = symbol.
    price = integer.
```

```

bank = symbol.
score, sum = integer.

size = integer.

property =
    car(name, color, price);
    building(name, price, size);
    plane(name, price, size).

predicates
    phonebook(lastname, phone, address_struct).
    bank_dep(lastname, bank, score, sum).
    own(lastname, property)

    property_names(lastname, name).
    property_prices(lastname, name, price).

    if_has_property(lastname, name, price).
    sum(lastname, price).

clauses

    phonebook("Ololo", "+78956783456", address("Moscow", "Izmaylovskay", 73,
226)).
    phonebook("Kek", "+76256785619", address("Saint Peterburg", "Neva", 45, 15)).
    phonebook("Cheburek", "+78956667819", address("Moscow", "Baumanskay", 3, 89)).
    phonebook("Cheburek", "+78958906578", address("Moscow", "Baumanskay", 3, 89)).

    own("Cheburek", car("Cadillac", "black", 18000000)).
    own("Cheburek", car("Nissan", "purple", 5000000)).
    own("Kek", car("Nissan", "black", 5500000)).
    own("Ololo", car("Porsche", "black", 20000000)).
    own("Cheburek", building("house", 18444000, 300000)).
    own("Cheburek", plane("ship", 18567000, 300000)).
    own("Kek", plane("korablik", 20000000, 300000)).

    bank_dep("Ololo", "sberbank", 50000, 76500).
    bank_dep("Ololo", "psb", 10000, 35000).
    bank_dep("Cheburek", "sberbank", 50000, 74520).
    bank_dep("Kek", "sberbank", 35000, 80456).

    if_has_property(Lastname, car, Price) :-
        own(Lastname, car(_, _, Price)), !.
    if_has_property(Lastname, building, Price) :-
        own(Lastname, building(_, Price, _)), !.
    if_has_property(Lastname, plane, Price) :-
        own(Lastname, plane(_, Price, _)), !.
    if_has_property(_, _, 0).

    sum(Lastname, Total_sum) :-
        if_has_property(Lastname, car, Building_price),
        if_has_property(Lastname, building, Car_price),
        if_has_property(Lastname, plane, Plane_price),

```

```
Total_sum = Building_price + Car_price + Plane_price.

property_names(Lastname, Name) :- own(Lastname, car(Name, _, _)).
property_names(Lastname, Name) :- own(Lastname, building(Name, _, _)).
property_names(Lastname, Name) :- own(Lastname, plane(Name, _, _)).

property_prices(Lastname, Name, Price) :-
    own(Lastname, car(Name, _, Price)).
property_prices(Lastname, Name, Price) :-
    own(Lastname, building(Name, Price, _)).
property_prices(Lastname, Name, Price) :-
    own(Lastname, plane(Name, Price, _)).

goal

%property_names("Cheburek", Name).
%Name=Cadillac
%Name=Nissan
%Name=house
%Name=ship
%4 Solutions
%property_prices("Kek", Name, Price).
%Name=Nissan, Price=5500000
%Name=korablik, Price=20000000
%2 Solutions
sum("Cheburek", Total_sum).
%Total_sum=55011000
%1 Solution
```

Словесное описание порядка поиска ответа на вопрос

№ шага	Сравниваемые термы; подстановка, если есть	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	Сравнение: property_prices("Kek", Name, Price) = phonebook("Ololo", "+78956783456", address("Moscow", "Izmaylovskay", 73, 226)). Унификация неуспешна (несовпадение функторов)	Прямой ход, переход к следующему предложению
2-18	...	...
19	Сравнение: property_prices("Kek", Name, Price) = property_prices(Lastname, Name, Price). Унификация успешна. Подстановка: {Lastname="Kek", Name=Name, Price=Price}	Новое состояние резольвенты: own("Kek", car(Name, _, Price))

№ шага	Сравниваемые термы; подстановка, если есть	Дальнейшие действия: прямой ход или откат (к чему приводит?)
20	Сравнение: <code>own("Kek", car(Name, _, Price)) = phonebook("Ololo", "+78956783456", address("Moscow", "Izmaylovskaya", 73, 226))</code> . Унификация неуспешна (несовпадение функторов)	Прямой ход, переход к следующему предложению
21-25	...	...
26	Сравнение: <code>own("Kek", car(Name, _, Price)) = own("Kek", car("Nissan", "black", 5500000))</code> . Унификация успешна. Подстановка: <code>{Lastname="Kek", Name="Nissan", Price=5500000}</code>	Новое состояние резольвенты: <i>пустая</i> <b>Вывод: Lastname="Kek", Label="Nissan", Price=5500000.</b> Откат, следующее предложение, подстановка: <code>{Lastname="Kek", Name=Name, Price=Price}</code>
27	Сравнение: <code>own("Kek", car(Name, _, Price)) = own("Ololo", car("Porsche", "black", 20000000))</code> . Унификация неуспешна (несовпадение термов)	Прямой ход, переход к следующему предложению
27-39	...	...
40	Сравнение: <code>own("Kek", car(Name, _, Price)) = property_prices(Lastname, Name, Price) :- own(Lastname, plane(Name, Price, _))</code> . Унификация неуспешна (несовпадение функторов)	Откат, достижение конца БЗ, переход к следующему предложению относительно шага 19. Новая подстановка: <code>{Lastname="Kek", Name=Name, Price=Price}</code>
41	Сравнение: <code>property_prices("Kek", Name, Price) = property_prices(Lastname, Name, Price) :- own(Lastname, building(Name, Price, _))</code> . Унификация успешна. Подстановка: <code>{Lastname="Kek", Name=Name, Price=Price}</code>	Новое состояние резольвенты: <code>own("Kek", building(Name, Price, _))</code> .
42	Сравнение: <code>own("Kek", building(Name, Price, _)) = phonebook("Ololo", "+78956783456", address("Moscow", "Izmaylovskaya", 73, 226))</code> . Унификация неуспешна (несовпадение функторов)	Прямой ход, переход к следующему предложению
43-61	...	...
62	Сравнение: <code>own("Kek", building(Name, Price, _)) = property_prices(Lastname, Name, Price)</code> . Унификация неуспешна (несовпадение функторов)	Откат, достижение конца БЗ, переход к следующему предложению относительно шага 41. Новая подстановка: <code>{Lastname="Kek", Name=Name, Price=Price}</code>

№ шага	Сравниваемые термы; подстановка, если есть	Дальнейшие действия: прямой ход или откат (к чему приводит?)
63	Сравнение: <code>property_prices("Kek", Name, Price)</code> = <code>property_prices(Lastname, Name, Price)</code> . Унификация успешна. Подстановка: { <code>Lastname="Kek", Name=Name, Price=Price</code> }	Новое состояние резольвенты: <code>own("Kek", plane(Name, Price, _))</code> .
64	Сравнение: <code>own("Kek", plane(Name, Price, _))</code> = <code>phonebook("Ololo", "+78956783456", address("Moscow", "Izmaylovskaya", 73, 226))</code> . Унификация неуспешна (несовпадение функторов)	Прямой ход, переход к следующему предложению
65- 73	...	...
74	Сравнение: <code>own("Kek", plane(Name, Price, _))</code> = <code>own("Kek", plane("korablik", 20000000, 300000))</code> . Унификация успешна. Подстановка: { <code>Lastname="Kek", Name="korablik", Price=20000000</code> }	Новое состояние резольвенты: <i>пустая</i> <b>Вывод: <code>Lastname="Kek", Label="korablik", Price=20000000</code></b> . Откат, следующее предложение, подстановка: { <code>Lastname="Kek", Name=Name, Price=Price</code> }
75	Сравнение: <code>own("Kek", plane(Name, Price, _))</code> = <code>bank_dep("Ololo", "sberbank", 50000, 76500)</code> . Унификация неуспешна (несовпадение функторов)	Прямой ход, переход к следующему предложению
76- 83	...	...
84	Сравнение: <code>own("Kek", plane(Name, Price, _))</code> = <code>property_prices(Lastname, Name, Price)</code> . Унификация неуспешна (несовпадение функторов)	Достижение конца БЗ, резольвента пуста, завершение работы.