

Выполнил: Бекетов Р.А.

Группа: ИУ5-32Б

Рубежный контроль по БКИТ №1

Вариант запросов: Б.

Вариант предметной области: 3.

Текст программы

Модуль *driver.py*

```
class Driver():

    def __init__(self, id_val, surname_val, salary_val, idOffice_val):
        self.id = id_val
        self.surname = surname_val
        self.salary = salary_val
        self.idOffice = idOffice_val
```

Модуль *fleet.py*

```
class Fleet():

    def __init__(self, id_val, name_val):
        self.id = id_val
        self.name = name_val
```

Модуль *fleet_driver.py*

```
class FleetDriver():

    def __init__(self, idOffice_val, idEmpl_val):
        self.idOffice = idOffice_val
        self.idEmpl = idEmpl_val
```

Модуль *main.py*

```
from entities.driver import Driver
from entities.fleet import Fleet
from entities.fleet_driver import FleetDriver

from operator import itemgetter

# Вариант Б, ПО №3
# ПО: водитель, автопарк

# Автопарки
nameOfFletts = ["Yandex", "Uber", "FreeTaxi", "BlaBlaCar", "notYandex", "notUber",
               "notFreeTaxi", "notBalBlaCar"]
fleets = [Fleet(index, name) for index, name in enumerate(nameOfFletts)]
```

```

# Водители
drivers = [
    Driver(0, "Иванов", 100, 1),
    Driver(1, "Артемьев", 5000, 0),
    Driver(2, "Павлов", 300, 3),
    Driver(3, "Кузьмин", 890, 0),
    Driver(4, "Арсеньев", 551, 3),
    Driver(5, "Петров", 696, 0),
    Driver(6, "Сидоров", 124, 2),
    Driver(7, "Васькин", 5, 2)
]

# АвтопаркВодитель
flDr = [
    FleetDriver(1, 0),
    FleetDriver(0, 1),
    FleetDriver(3, 2),
    FleetDriver(0, 3),
    FleetDriver(3, 4),
    FleetDriver(0, 5),
    FleetDriver(2, 6),
    FleetDriver(2, 7),

    FleetDriver(5, 0),
    FleetDriver(4, 1),
    FleetDriver(7, 2),
    FleetDriver(4, 3),
    FleetDriver(7, 4),
    FleetDriver(4, 5),
    FleetDriver(6, 6),
    FleetDriver(6, 7)
]

def main():
    # Соединение данных один-ко-многим
    oneToMany = [ (d.surname, d.salary, f.name)
        for f in fleets
        for d in drivers
        if d.idOffice == f.id
    ]

    # Соединение данных многие-ко-многим
    manyToMany_temp = [ (f.id, f.name, fd.idEmpl)
        for f in fleets
        for fd in flDr
        if f.id == fd.idOffice
    ]

    manyToMany = [ (d.surname, d.salary, fName)
        for d in drivers

```

```

        for fId, fName, dId in manyToMany_temp
            if d.id == dId
    ]

# Задание №1
taskFirst = sorted(oneToMany, key = itemgetter(0))
print("----Задание №1----")
print("№ | Фамилия | ЗП | Компания ")
print("-----")
for num, i in enumerate(taskFirst):
    print("{} | {} | {} | {}".format(num+1, i[0], i[1], i[2]))

# Задание №2
taskSecond = []
for i in range(4):
    driversInFleet = list(filter(lambda j: j[2] == fleets[i].name, oneToMany))
    taskSecond.append((fleets[i].name, len(driversInFleet)))
taskSecond = sorted(taskSecond, key = itemgetter(1), reverse=True)

print("\n----Задание №2----")
print("№ | Компания | Количество сотрудников ")
print("-----")
for num, i in enumerate(taskSecond):
    print("{} | {} | {}".format(num+1, i[0], i[1]))

# Задание №3
taskThird = []
for i in manyToMany:
    if (i[0][-2:] == "ов"):
        taskThird.append(i)

print("\n----Задание №3----")
print("№ | Фамилия | Компания ")
print("-----")
for num, i in enumerate(taskThird):
    print("{} | {} | {}".format(num+1, i[0], i[2]))

if __name__ == "__main__":
    main()

```

Результаты выполнения

- `(.venv) romanbeketov@MacBook-Air-Roman src % python3 main.py`

---Задание №1---

№	Фамилия	ЗП	Компания
---	---------	----	----------

1	Арсеньев	551	BlaBlaCar
2	Артемов	5000	Yandex
3	Васькин	5	FreeTaxi
4	Иванов	100	Uber
5	Кузьмин	890	Yandex
6	Павлов	300	BlaBlaCar
7	Петров	696	Yandex
8	Сидоров	124	FreeTaxi

---Задание №2---

№	Компания	Количество сотрудников
---	----------	------------------------

1	Yandex	3
2	FreeTaxi	2
3	BlaBlaCar	2
4	Uber	1

---Задание №3---

№	Фамилия	Компания
---	---------	----------

1	Иванов	Uber
2	Иванов	notUber
3	Павлов	BlaBlaCar
4	Павлов	notBlaBlaCar
5	Петров	Yandex
6	Петров	notYandex
7	Сидоров	FreeTaxi
8	Сидоров	notFreeTaxi