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**

