

Попов Андрей  
ИУ5-33Б Вар№3

```
class Driver:  
    def __init__(self, driver_id, surname, salary, fleet_id):  
        self.driver_id = driver_id  
        self.surname = surname  
        self.salary = salary  
        self.fleet_id = fleet_id  
  
class Fleet:  
    def __init__(self, fleet_id, name):  
        self.fleet_id = fleet_id  
        self.name = name  
  
class DriverFleet:  
    def __init__(self, driver_id, fleet_id):  
        self.driver_id = driver_id  
        self.fleet_id = fleet_id  
  
fleets = [  
    Fleet(1, "Центральный автопарк"),  
    Fleet(2, "Южный автопарк"),  
    Fleet(3, "Северный автопарк")  
]  
  
drivers = [  
    Driver(1, "Александров", 80000, 1),  
    Driver(2, "Белов", 75000, 1),  
    Driver(3, "Агапов", 85000, 2),  
    Driver(4, "Васильев", 70000, 3),  
    Driver(5, "Артемьев", 90000, 2)  
]  
  
driver_fleets = [  
    DriverFleet(1, 1),  
    DriverFleet(2, 1),  
    DriverFleet(3, 2),  
    DriverFleet(3, 3), # many-to-many  
    DriverFleet(5, 2),  
    DriverFleet(5, 1)  
]  
  
print("==== Запрос 1: Водители с фамилией на 'A' и их автопарки ===")  
drivers_A = [(d.surname, f.name)  
            for d in drivers  
            if d.surname.startswith('A')  
            for f in fleets  
            if f.fleet_id == d.fleet_id]  
for driver_fleet in drivers_A:
```

```
print(f"Водитель: {driver_fleet[0]}, Автопарк: {driver_fleet[1]}")  
  
print("\n==== Запрос 2: Автопарки с минимальной зарплатой водителей ===")  
min_salary_by_fleet = {}  
for f in fleets:  
    salaries = [d.salary for d in drivers if d.fleet_id == f.fleet_id]  
    if salaries:  
        min_salary_by_fleet[f.name] = min(salaries)  
  
sorted_by_salary = sorted(min_salary_by_fleet.items(), key=lambda x: x[1])  
for fleet, min_salary in sorted_by_salary:  
    print(f"Автопарк: {fleet}, Мин. зарплата: {min_salary}")  
  
print("\n==== Запрос 3: Все связи многие-ко-многим ===")  
relationships = [(d.surname, f.name)  
                 for rel in driver_fleets  
                 for d in drivers if d.driver_id == rel.driver_id  
                 for f in fleets if f.fleet_id == rel.fleet_id]  
sorted_relationships = sorted(set(relationships))  
for driver, fleet in sorted_relationships:  
    print(f"Водитель: {driver}, Автопарк: {fleet}")
```

==== Запрос 1: Водители с фамилией на 'А' и их автопарки ===

Водитель: Александров, Автопарк: Центральный автопарк

Водитель: Агапов, Автопарк: Южный автопарк

Водитель: Артемьев, Автопарк: Южный автопарк

==== Запрос 2: Автопарки с минимальной зарплатой водителей ===

Автопарк: Северный автопарк, Мин. зарплата: 70000

Автопарк: Центральный автопарк, Мин. зарплата: 75000

Автопарк: Южный автопарк, Мин. зарплата: 85000

==== Запрос 3: Все связи многие-ко-многим ===

Водитель: Агапов, Автопарк: Северный автопарк

Водитель: Агапов, Автопарк: Южный автопарк

Водитель: Александров, Автопарк: Центральный автопарк

Водитель: Артемьев, Автопарк: Центральный автопарк

Водитель: Артемьев, Автопарк: Южный автопарк

Водитель: Белов, Автопарк: Центральный автопарк