

РК №1 по ПиК ЯП

Вариант А.

«Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список всех связанных сотрудников и отделов, отсортированный по отделам, сортировка по сотрудникам произвольная.

«Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список отделов с суммарной зарплатой сотрудников в каждом отделе, отсортированный по суммарной зарплате.

«Отдел» и «Сотрудник» связаны соотношением многие-ко-многим. Выведите список всех отделов, у которых в названии присутствует слово «отдел», и список работающих в них сотрудников.

Вариант 30

30	Факультет	Университет
----	-----------	-------------

Текст программы:

```
# используется для сортировки
from operator import itemgetter

class Faculty:
    """Факультет"""
    def __init__(self, id, name, salary, university_id):
        self.id = id
        self.name = name
        self.salary = salary
        self.university_id = university_id

class University:
    """Университет"""
    def __init__(self, id, name):
        self.id = id
        self.name = name
```

```

class FacultyInUniversity:
    """
    'Факультеты университета' для реализации
    СВЯЗИ МНОГИЕ-КО-МНОГИМ
    """
    def __init__(self, university_id, faculty_id):
        self.university_id = university_id
        self.faculty_id = faculty_id

# Университеты
universities = [
    University(1, 'МГУ'),
    University(2, 'Университет СПбГУ'),
    University(3, 'НГУ'),
]

# Факультеты
faculties = [
    Faculty(1, 'Физика', 50000, 1),
    Faculty(2, 'Информатика', 60000, 2),
    Faculty(3, 'Химия', 55000, 1),
    Faculty(4, 'Математика', 70000, 3),
    Faculty(5, 'Биология', 55000, 2),
]

faculty_in_university = [
    FacultyInUniversity(1, 1),
    FacultyInUniversity(2, 2),
    FacultyInUniversity(3, 1),
    FacultyInUniversity(4, 3),
    FacultyInUniversity(5, 2),
]

```

```

def main():
    """Основная функция"""

    # Соединение данных один-ко-многим

    one_to_many = [(f.name, f.salary, u.name) for u in universities for f in
faculties if f.university_id == u.id]

    # Соединение данных многие-ко-многим

    many_to_many_temp = [(u.name, fu.university_id, fu.faculty_id) for u in
universities for fu in faculty_in_university if u.id == fu.university_id]

    many_to_many = [(f.name, f.salary, university_name) for university_name,
university_id, faculty_id in many_to_many_temp for f in faculties if f.id ==
faculty_id]

    print('Задание A1')
    res_11 = sorted(one_to_many, key=itemgetter(2))
    print(res_11)

    print('\nЗадание A2')
    res_12_unsorted = []
    # Перебираем все университеты
    for u in universities:
        # Список факультетов университета
        u_faculties = list(filter(lambda i: i[2] == u.name, one_to_many))
        # Если университет не пустой
        if len(u_faculties) > 0:
            # Зарплаты факультетов университета
            u_salaries = [salary for _, salary, _ in u_faculties]
            # Суммарная зарплата факультетов университета
            u_salaries_sum = sum(u_salaries)
            res_12_unsorted.append((u.name, u_salaries_sum))

    # Сортировка по суммарной зарплате
    res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)
    print(res_12)

```

```

print('\nЗадание A3')
res_13 = {}
# Перебираем все университеты
for u in universities:
    if 'университет' in u.name.lower():
        # Список факультетов университета
        u_faculties = list(filter(lambda i: i[2] == u.name, many_to_many))
        # Только названия факультетов
        u_faculties_names = [x for x, _, _ in u_faculties]
        # Добавляем результат в словарь
        # ключ - университет, значение - список факультетов
        res_13[u.name] = u_faculties_names

print(res_13)

if __name__ == '__main__':
    main()

```

Результат выполнения:

Задание A1

```

[('Физика', 50000, 'МГУ'), ('Химия', 55000, 'МГУ'), ('Математика', 70000,
'НГУ'), ('Информатика', 60000, 'Университет СПбГУ'), ('Биология', 55000,
'Университет СПбГУ')]

```

Задание A2

```

[('Университет СПбГУ', 115000), ('МГУ', 105000), ('НГУ', 70000)]

```

Задание A3

```

{'Университет СПбГУ': ['Информатика']}

```