

Текст программы

```
from operator import itemgetter

class prep:
    """класс преподавателя"""
    def __init__(self, id, fio, sal, course_id):
        self.id = id
        self.fio = fio
        self.sal = sal
        self.course_id = course_id

class course:
    """класс учебного курса"""
    def __init__(self, id, name):
        self.id = id
        self.name = name

class prep_course:
    """Класс для реализации многие ко многим"""
    def __init__(self, prep_id, course_id):
        self.prep_id = prep_id
        self.course_id = course_id

preps = [prep(1, 'Иванов Иван Иванович', 120, 3),
          prep(2, 'Антонов Петр Петрович', 100, 1),
          prep(3, 'Келдыш Елизавета Петровна', 110, 1),
          prep(4, 'Масленников Константин Юрьевич', 5, 2),
          prep(5, 'Афанасьев Геннадий Иванович', 6, 2)]

courses = [course(1, 'Курс Математический анализ'),
            course(2, 'АСОИУ'),
            course(3, 'Курс Физика'),

            course(11, 'Математический анализ (дополнительный)'),
            course(22, 'Модели данных (дополнительный)'),
            course(33, 'Физика (дополнительный)')]

courses_preps = [prep_course(1, 3),
                  prep_course(2, 1),
                  prep_course(3, 1),
                  prep_course(4, 2),
                  prep_course(5, 2),

                  prep_course(1, 11),
                  prep_course(2, 22),
                  prep_course(4, 33)]

def main():
    one_to_many = [(p.fio, p.sal, c.name)
                   for c in courses
                   for p in preps
                   if p.course_id == c.id]

    many_to_many_temp = [(c.name, cp.course_id, cp.prep_id)
                         for c in courses
                         for cp in courses_preps
                         if c.id == cp.course_id]

    many_to_many = [(p.fio, p.sal, c_name)
                    for c_name, c_id, pr_id in many_to_many_temp]
```

```

        for p in preps
        if p.id == pr_id]

answer_1 = {}
for c in courses:
    if 'Курс' in c.name:
        c_preps = list(filter(lambda i: i[2] == c.name, one_to_many))
        only_fio = [x for x, _, _ in c_preps]
        answer_1[c.name] = only_fio

print('Решение E1:\n', answer_1)

answer_2 = []
for c in courses:
    c_preps = list(filter(lambda i: i[2] == c.name, one_to_many))
    if len(c_preps) > 0:
        c_sal = [sal for _, sal, _ in c_preps]
        sr_sal = round(sum(c_sal) / len(c_preps), 2)
        answer_2.append((c.name, sr_sal))
print('Решение E2:\n', sorted(answer_2, key= itemgetter(1)))

answer_3 = {}
for p in preps:
    if p.fio[0] == 'А':
        p_courses = list(filter(lambda i: i[0] == p.fio, many_to_many))
        only_course = [x for _, _, x in p_courses]
        answer_3[p.fio] = only_course

print('Решение E3:\n', answer_3)

if __name__ == '__main__':
    main()

```

Результат выполнения программы

Решение E1:

{'Курс Математический анализ': ['Антонов Петр Петрович', 'Келдыш Елизавета Петровна'],
'Курс Физика': ['Иванов Иван Иванович']}

Решение E2:

[('АСОИУ', 5.5), ('Курс Математический анализ', 105.0), ('Курс Физика', 120.0)]

Решение E3:

{'Антонов Петр Петрович': ['Курс Математический анализ', 'Модели данных
(дополнительный)], 'Афанасьев Геннадий Иванович': ['АСОИУ']}

Process finished with exit code 0