

Код программы

```
from pprint import pp

class Producer:
    """ Производитель. """

    def __init__(self, ident: int, name: str):
        self.id = ident
        self.name = name

class Part:
    """ Деталь. """

    def __init__(self, ident: int, name: str, price: int,
prod_id: int):
        self.id = ident
        self.name = name
        self.price = price
        self.prod_id = prod_id

class ProdPart:
    """ СВЯЗЬ МНОГИЕ-КО-МНОГИМ для деталей производителя. """

    def __init__(self, prod_id: int, part_id: int):
        self.prod_id = prod_id
        self.part_id = part_id

# Производители
PRODS = [
    Producer(1, 'ВАГОНМАШ'),
    Producer(2, 'Завод Драйв'),
    Producer(3, 'Партариум'),
    Producer(4, 'Фабрикатор'),
    Producer(5, 'Катлокси'),
    Producer(6, 'Завод.рф'),
]

# Детали
PARTS = [
    Part(1, 'Штифт', 100, 1),
    Part(2, 'Гайка', 50, 2),
    Part(3, 'Мост', 300, 3),
    Part(4, 'Вал', 700, 4),
    Part(5, 'Болт', 80, 5),
    Part(6, 'Уголок', 150, 6),
    Part(7, 'Двутавр', 500, 1),
    Part(8, 'Сетка', 400, 2),
    Part(9, 'Колесо', 1000, 3),
```

```

    Part(10, 'Скоба', 20, 4),
    Part(11, 'Панель', 600, 5),
    Part(12, 'Шайба', 50, 6),
]

# Производитель-Деталь
PROD_PARTS = [
    ProdPart(1, 1),
    ProdPart(2, 2),
    ProdPart(3, 3),
    ProdPart(4, 4),
    ProdPart(5, 5),
    ProdPart(6, 6),
    ProdPart(1, 7),
    ProdPart(2, 8),
    ProdPart(3, 9),
    ProdPart(4, 10),
    ProdPart(5, 11),
    ProdPart(6, 12),
]

# Слово для поиска в задании 1
TASK1_WORD = 'Завод'
# Буква для поиска в задании 3
TASK3_LETTER = 'Ш'

# Функция поиска среднего для задания 2
def task2_average(a: list[int]) -> float:
    return sum(a)/len(a)

def main():
    """ Основная функция. """
    # Соединение данных один ко многим
    one_to_many = [
        (prod.name, part.name, part.price)
        for prod in PRODS
        for part in PARTS
        if part.prod_id == prod.id
    ]

    # Соединение многие-ко-многим
    many_to_many_temp = [
        (prod.name, prod_part.prod_id, prod_part.part_id)
        for prod in PRODS
        for prod_part in PROD_PARTS
        if prod.id == prod_part.prod_id
    ]

    many_to_many = [
        (part.name, prod_name)
        for prod_name, prod_id, part_id in many_to_many_temp

```

```

        for part in PARTS
        if part.id == part_id
    ]

    print('Задание 1:')
    result1 = [x for x in one_to_many if TASK1_WORD in x[0]]
    pp(result1)

    print('Задание 2:')
    avgs = {
        name: 0
        for name in set([x[0] for x in one_to_many])
    }
    for x in one_to_many:
        avgs[x[0]] = task2_average(
            [y[2] for y in one_to_many if y[0] == x[0]]
        )
    result2 = sorted(avgs.items(), key=lambda item: item[1])
    pp(result2)

    print('Задание 3')
    result3 = [x for x in many_to_many if x[0][0] ==
TASK3_LETTER]
    pp(result3)

if __name__ == '__main__':
    main()

```

Результат выполнения

Задание 1:

```

[('Завод Драйв', 'Гайка', 50),
 ('Завод Драйв', 'Сетка', 400),
 ('Завод.рф', 'Уголок', 150),
 ('Завод.рф', 'Шайба', 50)]

```

Задание 2:

```

[('Завод.рф', 100.0),
 ('Завод Драйв', 225.0),
 ('ВАГОНМАШ', 300.0),
 ('Катлокси', 340.0),
 ('Фабрикатор', 360.0),
 ('Партариум', 650.0)]

```

Задание 3

[('Штифт', 'ВАГОНМАШ'), ('Шайба', 'Завод.рф')]